OPTIMAL DISTRIBUTED MALWARE DEFENSE IN MOBILE NETWORKS WITH HETEROGENEOUS DEVICES

Mrs. D. Anantha Nayaki*1, Ms. M. Chitra*2, Mrs. K. K. Kavitha*3

*1M.C.A., M.Phil., Asst.Professor/Computer Science, Selvamm College of Arts & Science, Namakkal, India
*2Research Scholar, Dept. of CS, Selvamm College of Arts & Science, Namakkal, India
*3MCA., MPhil., SET., HOD/Computer Science, Selvamm College of Arts & Science, Namakkal, India

Abstract—Mobile data access is suffering the curse of the computationally enhanced increase of smart phones, which overloads the traditional cellular network. This paper tries to offload the cellular network traffic through Delay Tolerant Networks (DTN) formed by the short-range communication technologies in these smart phones (e.g., WiFi, Bluetooth). In this paper, we investigate the problem of how to optimally distribute the content-based signatures of malware, which helps to detect the corresponding malware and disable further propagation, to minimize the number of infected nodes. We model the defense system with realistic assumptions addressing all the above challenges that have not been addressed in previous analytical work. Based on the framework of optimizing the system welfare utility, which is the weighted summation of individual utility depending on the final number of infected nodes through the signature allocation, we propose an encounter-based distributed algorithm based on Metropolis sampler. Through theoretical analysis and simulations with both synthetic and realistic mobility traces, we show that the distributed algorithm achieves the optimal solution, and performs efficiently in realistic environments.

Keywords: Optimal control, fluid models, delay tolerant networks, threshold policies

I. INTRODUCTION

Delay Tolerant Networks (DTNs) have drawn considerable research interest recently due to their capability to deliver messages in frequently partitioned networks. The key for message delivery is the underlying mobility of nodes. “Store, carry and forward” kind of protocols are one of the natural routing options in DTNs. Mobile nodes rarely possess a priori information on the encounter pattern, thus an intuitive solution is to disseminate multiple copies of the message in the network, increasing the probability that at least one of them will reach the destination node within a given time window, generally known as epidemic-style forwarding. As in the spread of contagious diseases, epidemic forwarding passes the message to any node it encounters which does not have the message (uninfected node). Finally, the destination receives the message when it meets an infected node. A downside of this scheme is that the network is overloaded with messages. In this paper, we refer to a more efficient variant of the plain epidemic routing, namely the two hops routing protocol. The source transmits copies of its message to all mobiles it encounters, but the latter relay the message only if they meet the destination. Efficient energy consumption is one of the major concerns in battery operated mobile nodes due to limited energy budget in DTNs. Besides, significant energy is also consumed in node discovery process due to periodic beaconing. Thus, we face a natural tradeoff: the higher the number of nodes involved, higher is the message delivery probability, while expending more energy. Moreover, nodes may even die due to energy shortage before actually participating in forwarding. This clearly points towards a need of controlled activation of nodes, making nodes available only when needed. Nodes can be made active later, e.g., using wake-up timers. Previous research in the context of sensor networks has discussed the benefit of optimal activation times of deployed sensor nodes. To the best of our knowledge, this is the first study of activation methodology to enhance energy saving policy in context of DTNs.

Our goal here is to obtain jointly optimal transmission and activation control policies that maximize the probability of successful delivery of the message by sometime, given the total energy budget and a bound on the activation rate of the relay nodes. We leverage fluid approximations of the system dynamics, and use tools from optimal control theory to obtain a closed-form optimal policy. As we will see later, this turns out to be a two-dimensional threshold type policy. We validate the model and the results using extensive simulations.
II. LITERATURE SURVEY

Control of forwarding schemes has been addressed in the DTNs literature before. Its follow-up, the authors optimize network performance by designing message relays. The authors consider buffer constraints and derive buffer scheduling policies in order to minimize the delivery time. In, we have provided a general framework for the optimal control of a broad class of monotone relay strategies. The more recent paper employs stochastic approximation to avoid the explicit estimation of network parameters. Optimal activation of nodes in redundantly deployed sensor networks has been studied before. A threshold based activation policy was shown to perform close to the optimal policy for dynamic node activation. Scheduling controlling the activity nodes to exploit energy harvest in features has been studied. Compared with the existing literature, this paper makes the following original contributions: i) Explicit accounting for the maximum allowed energy expenditure, the delivery probability within a given deadline, and the activation of relays; ii) A formulation rooted in optimization, which entails joint optimization of the activation control and the transmission control in order to maximize the time-constrained delivery probability. This is a non-standard dynamic optimization problem formulated with coupled controls. Once solved, interesting properties of the optimal solution and the special role of the control on the relay activation have emerged.

III. EXISTING SYSTEM

Consider a mobile network where a portion of the nodes are infected by malware. Our research problem is to deploy an efficient defense system to help infected nodes to recover and prevent healthy nodes from further infection. Typically, we should disseminate the content-based signatures of known malware to as many nodes as possible. Consequently, distributing these signatures into the whole network while avoiding unnecessary redundancy is our optimization goal.

Disadvantages of existing system:

We cannot rely on centralized algorithms to distribute the signatures because the service infrastructure is not always available.

IV. PROPOSED SYSTEM

In the network, there are different types of handsets and each malware only targets handsets with a specific OS. In the defense system, we use some special nodes named helper to distribute the signatures into the network. Generally speaking, the deployed helpers can be stationary base stations or access points. However, since mobile nodes are more efficient to disseminate content and information in the network, we focus on the case of mobile helpers. Consequently, there is limitation in storage on each mobile device for deploying the defense system. Although currently most smartphones have gigabytes of storage, users usually will not allocate all of them for the usage of malware defense. Our goal is to minimize the malware infected nodes in the system by appropriately allocating the limited storage with the consideration of different types of malware.

Advantages of proposed system:

To minimize the malware infected nodes in the system by appropriately allocating the limited storage.

V. SYSTEM DESCRIPTION

System Model:

Mobile malware that spreads in the mobile networks typically exploits both the MMS and opportunistic contacts to propagate from one device to another. In the network, there are different types of handsets and each malware only targets handsets with a specific OS. In the defense system, we use some special nodes named helper to distribute the signatures into the network. Generally speaking, the deployed helpers can be stationary base stations or access points. However, since mobile nodes are more efficient to disseminate content and information in the network, we focus on the case of mobile helpers. Consequently, there is limitation in storage on each mobile device for deploying the defense system. Although currently most smartphones have gigabytes of storage, users usually will not allocate all of them for the usage of malware defense. Our goal is to minimize the malware infected nodes in the system by appropriately allocating the limited storage with the consideration of different types of malware.

Notations and Malware Spreading Model

We consider a system of N heterogeneous wireless nodes belonging to K types (e.g., type of OS), which can be infected by K types of malware, denoted by set IK. In the defense system, we assume that there are S helpers, denoted by set SS, storing the signatures to help other nodes with detecting the malware. Furthermore, let us denote the maximum number of signatures that can be stored at helpers, and uk denote the number of helpers for malware k. To describe how the signatures of malware are stored in the helpers, we define xs,k as the indicator
whether helper s has the signature to detect malware k. Since different types of malware will infect different classes of nodes, we let \( v_k \) denote the maximum number of nodes that malware k can infect, and let \( v_0 k \) denote the number of infected nodes at the starting time.

VI. IMPLEMENTATION AND ALGORITHM

The Greedy Algorithm

Now, we give a greedy algorithm described in Algorithm 1 for the formulated problem. The obtained result by Algorithm 1 is the optimal solution, which is proved by Theorem 1. The algorithm repeatedly chooses signatures to store in the helpers: in each step, it tries to select one signature that brings the maximum system utility for a helper that still has enough storage. Therefore, our algorithm is likely to allocate more helpers to store the signatures of malware whose corresponding malware defending utilities are larger than others, which is achieved by using the heterogeneous features in terms of devices and malware.

**Algorithm 1. The Greedy Algorithm to Maximize the System Welfare**

1. Set \( x_{s,k} = 0, u_k = 0, \Delta F_k = 0 (k \in K, s \in S) \);
2. Initialize set \( R = \{1, 2, \ldots, K\} \) and \( \text{sum} = 0 \);
3. for Every malware \( k \) that \( k \in R \) do
4. \( \Delta F_k \leftarrow u_k (F_k (u_k + 1) - F_k (u_k)) \);
5. end for
6. while \( \text{sum} \leq \sum_{s \in S} A_s + \text{sum} \) and \( R \neq \emptyset \) do
7. Select \( i = \arg \max \{ \Delta F_k | k \in K \} \);
8. Select \( l = \arg \max \{A_s - \sum_{k \in R} x_{s,k} | x_{s,k} = 0, s \in S \} \);
9. Set \( x_{i,l} = 1 \);
10. Update \( u_k \leftarrow u_k + 1 \), \text{sum} \leftarrow \text{sum} + 1 \);
11. Update \( \Delta F_l \leftarrow u_l (F_l (u_l + 1) - F_l (u_l)) \);
12. if \( u_k \geq S \) then
13. \( R \leftarrow R \{ i \} \);
14. end if
15. end while

DISTRIBUTED ALGORITHM USING METROPOLIS SAMPLER

Now, we design a distributed algorithm for the signature distribution problem. The designed algorithm is based on a simulated annealing technique called Metropolis sampler. In the following sections, we first describe the basic notions and the framework of Metropolis sampler (details are available in [8]), then design the distributed algorithm based on simulated annealing with the Metropolis sampler, and finally prove that the proposed algorithm converges to the optimal performance.

**Encounter-Based Distributed Algorithm**

Based on the introduction of Gibbs distribution and Metropolis sampler, we now design the distributed algorithm for signature dissemination. We consider every encounter between any two helpers as one step of configuration changing in the algorithm. When nodes i and j meet, each one adjusts its current configuration according to the others. More specifically, one node, says i, randomly chooses a signature in its own buffer, and randomly chooses another one that is not in its buffer but in the buffer of node j to replace the chosen signature, which comes to a tentative configuration. After obtaining the replacement probability depending on the current and tentative configuration, node i decides whether to replace it or not. We assume the current configuration of the system is \( x \), and the configuration of the node i is \( x_i \), and node i. Therefore, the intuition behind our distributed algorithm is that each helper keeps on selecting the signature that gives a higher contribution to the system utility according to the local information. When the nodes encounter again and again, the algorithm approaches to the global optimal solution.

VII. PERFORMANCE EVALUATION

**Centralized Greedy Algorithm**

In this section, we present numerical results with the goal of demonstrating that our greedy algorithm for the signature distribution, denoted OPT, achieves the optimal solution and yields significant enhancement on the system welfare compared with prior heuristic algorithms. Related to the heuristic algorithms, we consider 1) Important First (IF), which uses as many helpers as possible to store the signature of the most popular malware, 2) Uniform Random (UR), where each helper randomly selects the target signatures to store, and 3) Proportional Allocation (PA), which is a heuristic policy that assigns signatures with the uniform distribution proportional to the market sharing and the weights of different malware. To simulate a more realistic scenario, we model the malware in the system according to the market share of different handset OS of 2009. In the simulation, we change the malware killing rate and spreading rate, and consider a system with nodes that can be infected by five different types of malware, which are RIM targeted malware 36 percent;
Android targeted 28 percent; iPhone 21 percent; Windows Mobile 10 percent, and others 5 percent. We set N = 500 and have 100 helpers with uniform random storage size from one to five signatures to deploy in the antimalware software. In the experimental setup, the number of initial infected nodes is set to be 10 percent of all nodes. Related to the utility function and weighting factors, we set Gk(L,k) = L(k) - L = 2 and w = 12; I=4; I=8; I=16; I=16 to differentiate the system contributions of different malware defending effects by considering the factor that usually the malware spreading in the largest market sharing OS would result in the most serious damage.

![Performance of different algorithms](image)

**Fig. 2.** Performance of different algorithms for the malware defense system with (a) variable malware recovering rate; and (b) variable malware spreading rate.

### VIII. CONCLUSION AND FUTURE WORK

In this paper, we investigate the problem of optimal signature distribution to defend mobile networks against the propagation of both proximity and MMS-based malware. We introduce a distributed algorithm that closely approaches the optimal system performance of a centralized solution. Through both theoretical analysis and simulations, we demonstrate the efficiency of our defense scheme in reducing the amount of infected nodes in the system. At the same time, a number of open questions remain unanswered. For example, the malicious nodes may inject some dummy signatures targeting no malware into the network and induce denial-of-service attacks to the defense system. Therefore, security and authentication mechanisms should be considered. From the aspect of malware, since some sophisticated malware that can bypass the signature detection would emerge with the development of the defense system, new defense mechanisms will be required. At the same time, our work considers the case of OS targeting malware. Although most of the current existing malware is OS targeted, cross-OS malware will emerge and propagate in the near future. How to efficiently deploy the defense system with the consideration of cross-OS malware is another important problem. We are continuing to cover these topics in the future work.

### REFERENCES


