Proceedings of the 2014 Winter Simulation Conference
A. Tolk, S. Y. Diallo, I. O. Ryzhov, L. Yilmaz, S. Buckley, and J. A. Miller, eds.

REASONING ABOUT MOBILE MALWARE USING HIGH PERFORMANCE COMPUTING BASED POPULATION SCALE MODELS

Karthik Channakeshava

Ericsson
250 Holger Way
San Jose, CA 95134, USA

Keith Bisset
Madhav V. Marathe
Anil Kumar S. Vullikanti

Network Dynamics and Simulation Science Laboratory
Virginia Bioinformatics Institute
1880 Pratt Drive, MC 0477 Virginia Tech
Blacksburg, VA 24061, USA

ABSTRACT

We present a high performance computing (HPC) based modeling approach to reason about mobile malware. The ubiquity of smart phones and devices and the use of local protocols for disseminating information over such devices has raised new security challenges. The HPC approach to study mobile malware propagation problem involves: (*i*) a realistic and detailed representation of mobile devices, their time varying location, their usage patterns and the urban environment within which they are operated, leading to dynamic interaction networks over which malware can spread – these networks are large, heterogeneous and time varying; and (*ii*) a high performance computing based simulation environment that can study diffusion of malware over such networks. We use EpiCure, an individual based high performance simulation tool for malware modeling, that scales to networks spanning urban regions with over 10M individuals. We find that malware dynamics in realistic networks are very different from those in random waypoint (RWP) mobility models. Next, we study the impact of some of the worm model parameters and properties associated with population mobility and social contact networks; we use detailed statistical analysis to identify the significant parameters and their interaction. Finally, we use EpiCure to study SMS/MMS based malware and hybrid malware that spread using both proximity based Bluetooth networks and infrastructure based cellular networks.

1 INTRODUCTION

Advances in computing and communication technologies have made mobile communication devices extremely powerful, blurring the distinction between today's PCs and mobile phones. Most of these devices are capable of communicating over multiple physical layer technologies, interfacing with disparate networks. Furthermore users are becoming increasingly reliant on their mobile devices for their daily activities. *Internet of Things* (IoT) have further spurred this area, wherein mobile devices, powerful sensing and computing systems interact with *things*, forming a rapidly evolving eco-system. This has also resulted in the recent surge of mobile malware — viruses, worms, spam and other malicious software, which used to be a minor nuisance, have now become a major threat (Ferrie et al. 2004; Symantec 2005) due to their potential to affect the safety and health of the society at large. In addition to exploiting software and hardware loopholes, these malware programs exploit characteristics of the human social contact networks to achieve rapid spread. Developing methods to control their spread is regarded as a fundamental security challenge.

The diffusion of malware shares a number of similarities with the spread of infectious diseases. *Internet* or *wireless epidemiology* (Ma, Voelker, and Savage 2005; Kleinberg 2007) acknowledges this close relationship and aims to use mathematical and biological principles developed by epidemiologists to study

malware in the cyber-world. However, malware on short-range wireless communication networks bridges the phenomena of human epidemics and worms on the Internet, giving rise to significant differences in how it spreads (Kleinberg 2007). A number of mathematical and simulation based approaches have been proposed to study epidemiological problems in wireless networks, e.g., Rhodes and Nekovee (2008), Yan and Eidenbenz (2006), Mickens and Noble (2007), Su et al. (2006), Yan et al. (2007). These approaches either study the problem using compartmental mean field models or use simple mathematical models of mobility so as to derive rigorous analytical solutions. Many of the underlying assumptions often do not hold for real-world problems motivating a HPC-based modeling approach. A number of recent papers have taken steps in this direction; see Yan et al. (2007), Su et al. (2006), Channakeshava et al. (2011), Wang et al. (2009), Husted and Myers (2011) and the references therein. A recent survey on this topic by Peng, Yu, and Yang (2014) provides a recent comprehensive overview of the literature.

Summary of results. Here, building on our earlier work (Channakeshava et al. 2011, Channakeshava et al. 2009, Beckman et al. 2013), we describe a high performance agent-based computing based methodology to study malware propagation at an urban population scale. The methodology comprises of the following components: (*i*) A set of generative models using diverse commercial and open source data sets combined with social and behavioral theories to obtain a synthetic representation of the dynamic smartphone interaction networks; the network is spatially explicit and captures a chosen urban context. The models can be further refined to extend to IoT; see Beckman et al. (2013), Barrett et al. (2009), Channakeshava et al. (2011); (*ii*) a HPC-based simulation system that allows us to study the diffusion of malware, subject to some given initial conditions; the system can be used to represent complex counter-measures for monitoring and controlling the spread. Using this environment, we show the following:

(*i*) *Realistic representation of the dynamic smartphone network impacts malware diffusion and the efficacy of countermeasures.* As observed by Yan et al. (2007), we find that mobility has a significant impact on the dynamics. However, their results are based on various RWP mobility model. We find significant differences in the structure of the dynamic network as well as malware dynamics when using realistic mobility patterns derived in Channakeshava et al. (2011). Specifically, the rate of spread seems to be slower in activity based models, which can be accounted for by the lower temporal degrees and clustering coefficient, compared to RWP. The results are described in Section 3.

(*ii*) *Marketshare has important implication on malware dynamics.* We study the impact of some of the key properties associated with the malware model parameters (such as idle time and probability of infection), and network parameters (such as density and market share). It is quite challenging to understand how these different parameters interact. We perform a 3-factor analysis of variance (ANOVA) on the simulation response to determine the interactions between these parameters; this identifies unexpected interactions between the idle time parameter and the market share. Overall, we find that the market share and location density have a very significant impact on the speed of malware spread. The results are described in Section 4.

(*iii*) *Malware that uses SMS and Bluetooth networks can spread faster and exhibits natural jumps in the spatial diffusion process.* We compare the dynamics of Bluetooth based malware with one that propagates using the SMS/MMS messages (referred to as *sms-only*), and a hybrid malware that spreads on both proximity Bluetooth links as well as the cellular infrastructure (referred to as *hybrid*). Such diverse models can be easily implemented within the EpiCure framework, illustrating its flexibility. We find that the dynamics of the *sms-only* malware, under some conditions, are very similar to the Bluetooth malware in nature. In contrast, the dynamics of *hybrid* malware are very different from *sms-only* malware. Assuming that the entire set of devices are susceptible to a hybrid malware, we find that the malware spreads extremely aggressively infecting almost the entire susceptible population within a few hours. The results are described in Section 5.

Organization. We discuss the activity based models and the EpiCure framework in Section 2. The comparison with RWP models is discussed in Section 3. We present the analysis of sensitivity and the hybrid malware in Sections 4 and 5, respectively.

2 PRELIMINARIES

We briefly recall our earlier work on developing synthetic populations, dynamic synthetic mobile networks and HPC-based modeling environment; see Eubank et al. (2004), Barrett et al. (2009), Channakeshava et al. (2011), Channakeshava et al. (2009) for details.

2.1 In Context Synthetic Mobility and Socio-Communication Network Framework

Real data for large scale mobility is not easily available, at least in the public domain, because of privacy and proprietary issues. A common approach to deal with lack of realistic data is to use simplified stochastic models, which match aggregate properties, e.g., degree distribution. However, as argued by Li et al. (2004), and many other researchers subsequently, there are significant limitations to such simplified models. Some researchers have used proprietary datasets, e.g., Wang et al. (2009), which give snapshots of call records from some wireless providers. However, it is not clear how to extend such results to incorporate user behavior, which is dynamic, and has a significant impact on mobility.

Table 1: Bluetooth worm model terminology.

Terms	**Definition**
T_{idle}	Idle time between worm infection cycles. No inquiry requests are sent out by the infected node during this time.
p_{to}	Probability of the Bluetooth inquiry request timing out without a response from any neighboring device.
p_{inf}	Probability of infecting a susceptible neighboring device that responds to an inquiry and other Bluetooth connection establishment steps.
N_{resp}	Maximum number of responses that an inquiry request gathers during a single infection cycle.
T_{to}	Time taken for an inquiry request to timeout before going for the next infection cycle. No responses are received from any neighboring device.
d	Location density used to construct the with in building mobile device proximity network.
m	Market share of the mobile devices that are assumed to be susceptible to the malware under study.

In light of these issues, we use a synthetic mobility and contact network model constructed using a first-principles based approach (Eubank et al. 2004; Barrett et al. 2009)— this involves detailed activity modeling in large urban regions, and will be referred to as ABM in the rest of the paper. This approach integrates over a dozen public and commercial datasets, and involves the following steps (see Barrett et al. 2009; Channakeshava et al. 2011; Channakeshava et al. 2009 for more details): (*i*) Create a synthetic urban population using several databases from commercial and public sources, while preserving their privacy and maintaining statistical indistinguishability; (*ii*) Use activity templates of individuals to create the activity-based mobility models. This generates the social network for individuals using the US census, survey data and time-use surveys; (*iii*) Assign detailed route plans to individuals based on the locations where activities are performed and the road network that connects the locations; (*iv*) Construct detailed movement patterns using a cellular automata based micro-simulation for individuals over the transportation infrastructure; and (*v*) Construct the Bluetooth proximity network using a sub-location model. Each activity location in the study is assigned an area based on occupancy and is divided into grid cells of 10 m side, which correspond to the sub-locations. Each device arriving at an activity location is assigned a random sub-location, and devices belonging to the same sub-location are considered to be in range.

Although a rigorous comparison of realistic mobility models is outside the scope of this paper, we contrast ABM with the UDel mobility models Kim, Sridhara, and Bohacek (2009) (or UDelModels) used in Husted and Myers (2011) and Yan et al. (2009). The UDelModels are based on data from the Bureau of Labor Statistics, and Urban Planning Research. Mobility data generated from UDelModels is realistic but the scale and demographic aspects of these models are limited to a few city blocks with close to 10000. The worker meeting research data make mobility of devices within buildings more realistic in UDelModels.

Further, simulations for durations longer than 4 hours are computationally intractable (Husted and Myers 2011).

2.2 HPC Framework for Malware Simulation: EpiCure

Malware dynamics over realistic wireless communication networks cannot be completely understood analytically. This motivates the need for computer simulations for explicitly generating and analyzing the dynamics; see Yan and Eidenbenz (2006), Mickens and Noble (2007). Here we describe a HPC-based modeling environment EpiCure that builds on our earlier work, a system named EpiNet (Channakeshava et al. 2009). NS-2 and Qualnet are other examples of similar efforts. EpiNet (Channakeshava et al. 2009) is a parallel simulation tool that scales to networks with about 20,000 nodes. The main optimizations in EpiNet include approximation of the mobility events and a probabilistic timed transition system (PTTS) model for the worm spread. This PTTS model abstracts many of the protocol level details, and instead uses some of the key parameters in the worm model.

EpiCure (Channakeshava et al. 2011) builds on EpiNet, and is designed specifically to work on commodity cluster architectures. It runs extremely fast for realistic instances that involve: (*i*) large time-varying networks consisting of millions of heterogeneous individuals with time varying interaction neighborhoods, (*ii*) dynamic interactions between the propagation of the malware, individual behavior, and exogenous interventions, and (*iii*) large number of replicated runs necessary for statistically sound estimates of malware spread dynamics. EpiCure runs several orders of magnitude faster than other comparable simulation tools. Further, it is very flexible so that analysts can easily represent a range of interventions leading to improved human productivity and ease of use. EpiCure gives a 100X improvement over EpiNet (more, compared to other approaches). EpiCure further simplifies the worm model and uses a hybrid MPI-Threads[1] implementation on multi-core architectures, in order to achieve this speedup.

1. **Network:** Synthetic population from the city of Chicago and the New River Valley, VA. Activities and activity durations are rounded to 5 minute intervals ($300\,s$).
2. **Number of initial infections:** 1%, 5%, 10%
3. T_{idle}: $20\,s$; T_{to}: $12.80\,s$, N_{resp}: 4
4. **Infection seed time:** 8 AM (at the beginning of simulation)
5. **Seeds:** 5 (for each combination of input parameters)
6. **Simulation duration:** 8 hours from 8 AM to 4 PM
7. **Studies:**
   (a) *Influence of worm parameters:* T_{idle}, p_{to}
   (b) *Influence of network parameters:* Market share (m), Location Density (d)
8. **Average runtime:** ≈ 2 hours

Figure 1: Experimental setting and parameters studied.

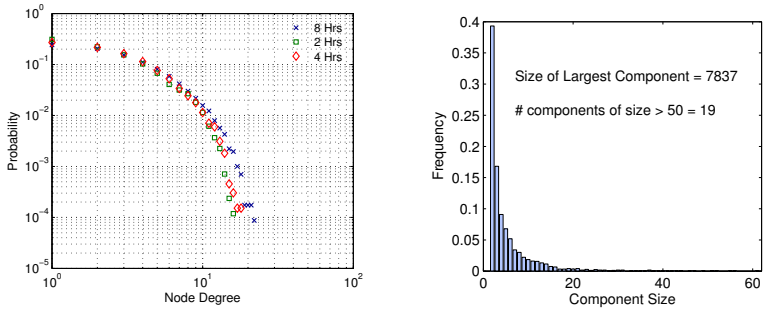
2.3 Datasets, Assumptions and Experimental Design

Figure 1 summarizes our experimental setting. Table 1 summarizes the Bluetooth worm parameters we use in the rest of the paper. We use synthetic Bluetooth networks for two regions in the US, constructed using the methods described in Section 2.1: (a) Chicago downtown with about 30K devices/people and 2200

[1]Message passing interface (or MPI) is a standard that defines the syntax and semantics of a core of library routines useful to writing portable parallel programs.

activity locations, (b) two networks for the New River Valley, VA, with 77,659 and 126,800 devices/people, and 284,941 and 1,507,328 links, respectively. The activities are extracted for a duration of 8 hours. Figure 2 shows some properties of the resulting proximity network for the Chicago model; notice the non-Poisson nature of these distributions.

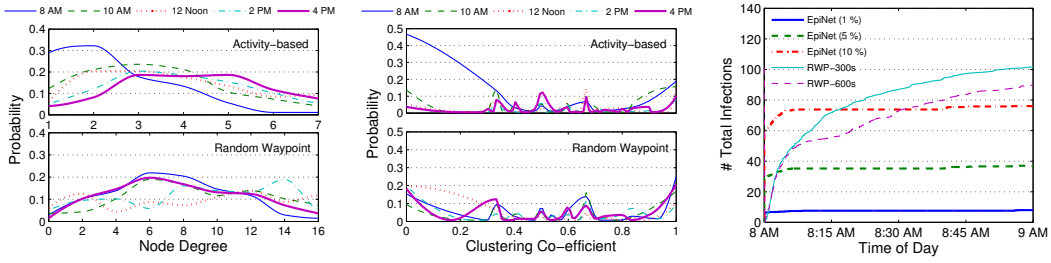
We study the dynamics of malware spread using the following measures: (*i*) the cumulative number of infections, as a function of time of day; (*ii*) a function $T(q,.)$, that denotes the time taken for a q-fraction of devices to get infected. Here, $(.)$ indicates the variable that is being altered in the study. For example, when we study the spread by varying the idle time of the worm T_{idle}, then $(.)$ represents T_{idle}; (*iii*) p_{to}, which is the probability that an inquiry request does not discover any neighboring Bluetooth devices. This parameter is interesting as it can help us understand the effect of configuring a Bluetooth device as non-discoverable; (*iv*) the location density (d), and market share (m).



(a) (b)

Figure 2: Network characteristics obtained as a result of the activity-based mobility models and the *sub-location modeling*. We use a union graph for determining these measures from individual networks constructed every 300 *s*. **(a):** Degree distribution of the network; **(b):** Histogram of the component sizes in the network.

3 IMPORTANCE OF REALISTIC SOCIO-COMMUNICATION NETWORKS



(a) (b) (c)

Figure 3: **(a):** Comparison of degree distribution between activity-based (top) and RWP models (bottom) at different snap shots during the 8 hour duration of simulation; **(b):** Clustering coefficient distributions for activity-based (top) and RWP (bottom) for the same time instants.; **(c):** Comparison between the infection spread with RWP and activity-based mobility models within a single location. RWP shows much faster rate of propagation due to its property of having a high density of devices around the center.

The RWP uses several parameters for generating mobility—number of nodes, location area, minimum and maximum speeds and pause times, e.g., Yan et al. (2007); these are selected to best match the activity-based model (see Channakeshava et al. 2011; Channakeshava et al. 2009 for details). Figure 3 shows differences in some of the characteristics in the networks resulting from these two models. Specifically, the

Table 2: Simulation parameters for RWP (using NS-2) and activity-based mobility models (using EpiCure, EpiNet).

Parameters	**RWP Model**	**Activity-based Model**
Number of Locations	1	1
Node Number	109	91–147
Node Velocity	0.5–1.5 m/s	–
Pause Time	300 s, 600 s	–
Node arrival & departure	No arrivals/departures	Every 300 s nodes arrive or depart
Initially infected	1 infected device	1%, 5% & 10%

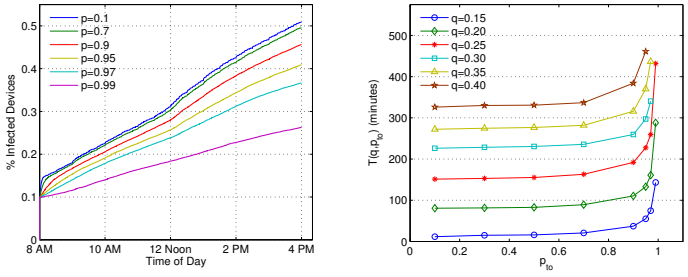
temporal degree and clustering coefficient is higher in the RWP model, which accounts for the differences in malware dynamics, as will be discussed later.

We now study the worm spread dynamics in the two models. Making comparisons between activity-based and RWP models is not straight-forward as they depend on different parameters. The instantaneous occupancy of the location (computed for every 300 s interval) for the activity model ranges from 91–147. Since arrivals and departures are rounded to 300 s interval the occupancy does not change in the interval we consider. For RWP we consider 109 devices, which is the average occupancy for the duration of the simulation. Since the number of nodes differs in the two models, we cannot perform a direct one-to-one comparison. We varied the parameters of the initial infection sizes in case of ABM (1%, 5% and 10% devices initially infected) and the pause time (300 s and 600 s) and speeds ranging from 0.5 m/s and 1.5 m/s in case of RWP. The simulations were conducted using EpiNet/EpiCure and NS-2 for ABM and RWP, respectively. We simulate the scenario for an hour (between 8 AM and 9 AM) and observe the number of nodes infected by 9 AM. For ABM, we randomly select 1%, 5% and 10% of the location's total occupancy as the initial infection size, and consider a single randomly infected device for RWP. Figure 3c shows a comparison between the infection growth in the two models (averaged over 5 seeds). For the RWP model, the infections surge initially and infect almost all of the devices present in the location within the first hour. This can be attributed to the higher degree and clustering, as seen in Figure 3. In fact, we observe a higher rate of infection spread for the case when pause times are lower (300 s). In contrast, the initial surge of infections quickly saturates in ABM, and remains much lower.

4 SENSITIVITY ANALYSIS

We now study the sensitivity of different input parameters on malware dynamics, and the interactions between them. We find the market share has a significant impact.

4.1 Sensitivity to Bluetooth Worm Parameters



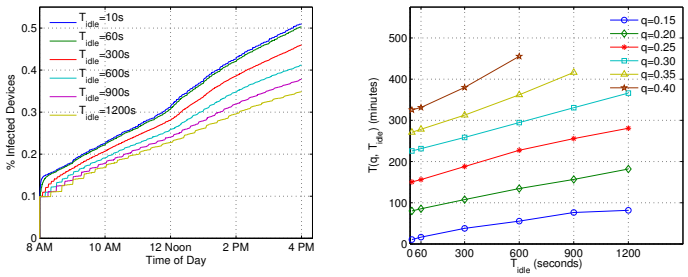
(a) (b)

Figure 4: Infection spread dynamics with variation in p_{to}. **(a):** Infection spread with varying p_{to} starting with 10% initially infected devices; **(b):** $T(q, p_{to})$ for the varying p_{to} values in the same scenario.

We now study the sensitivity of EpiCure to some of the parameters in the worm spread model discussed in Section 2.2. Specifically, we consider (i) the timeout probability, p_{to}, which is defined as the probability that an inquiry request timeout occurs without a single inquiry response, and (ii) the idle time, T_{idle}. Surprisingly, as shown in Figure 4, there is very limited effect of p_{to} on the total number of infections. Until p_{to} becomes 0.9—meaning that the infected devices' timeout occurs 90% of inquiry requests—the infection still spreads to a large number of devices (about 45%). Beyond this, the successful completion of inquiry with a response reduces and the infection does not seem to take off. Only about 20% of the devices become infected starting with 10% initially infected devices. Clearly, this shows that disabling the *discoverable* mode in Bluetooth devices can cause a large slowdown in the spread of the malware. Figure 4b shows the plot of $T(q, p_{to})$, i.e, the time taken to infect q-percentage of devices when varying p_{to}.

Next, we study the sensitivity of the dynamics to the idle time, T_{idle}, in Figure 5. Figure 5a shows the cumulative percentage of infected devices as T_{idle} is increased. Intuitively, it would seem that an intelligent worm can adapt its idle time to maximize the spread. However, we find that the spread obtained with 20s idle time denotes the upper limit of the worm propagation. Reducing the idle time to 10 s does not result in an increase in the total infected devices. As expected, the idle time does affect the initial speed of the spread as the devices wait longer in each infection cycle. Nevertheless, once a certain number of devices become infected, eventually the speed increases to infect more devices. Figure 5b shows the $T(q, T_{idle})$ values for the variation in T_{idle}. For most values of idle time at least 35% of devices become infected starting with 10% initially infected devices.

Overall, we find that the worm spreads much slower than the observations of (Yan and Eidenbenz 2009, Wang et al. 2009). Additionally, we find that a smaller fraction of devices is infected, e.g., at most 50% of devices become infected during 8 hours starting with 10% initially infected devices. This is clear evidence of the specific mobility model underlying our data-set, which mixes much slower than uniform mobility, and is much more heterogeneous.



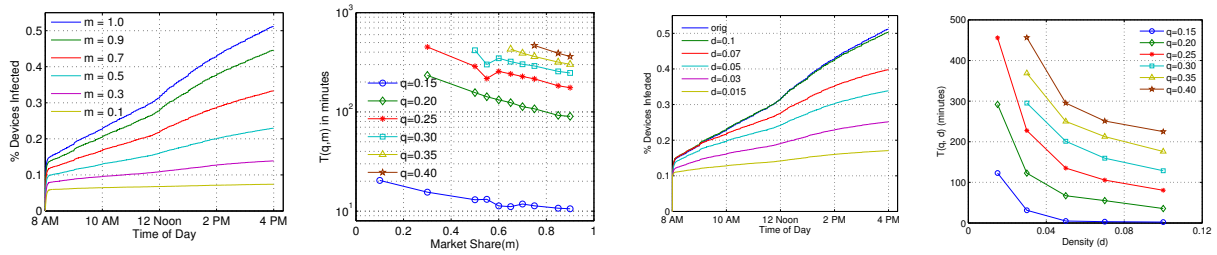
(a) (b)

Figure 5: Infection spread dynamics with variation in T_{idle}. **(a):** Infection spread with varying T_{idle} starting with 10% initially infected devices; **(b):** $T(q, T_{idle})$ for the varying T_{idle} values in the same scenario.

4.2 Sensitivity to Network Parameters

We now study the effects of the ABM network structure on the worm dynamics. Specifically, we consider: (*i*) the market share, denoted by m, which represents the number of devices that have similar characteristics to the infected device and are susceptible to the same malware, and (*ii*) location density, denoted by d, which denotes the density of devices within a location.

Effect of market share. The market share is a simple parameter that allows us control over the susceptible fraction, and here we study its impact on the worm dynamics. We consider a range of m values from 10% to 90%. Figure 6a shows that the speed of the worm is directly proportional to m, as expected, and both the speed and the final outbreak size are reduced with m. However, we find that the $T(q, m)$ function (Figure 6b) shows very different characteristics, when compared with the results of Wang et al. (2009);



(a) (b) (c) (d)

Figure 6: **(a):** Infection spread with varying market share of the susceptible devices; **(b):** $T(q,m)$ for different values of q and varying market shares of the susceptible devices. Infection spread dynamics with variation in location density (d). **(c):** Infection spread with varying d; **(d):** $T(q,d)$ changes with varying density of locations.

in particular, we find a steeper variation in $T(q,m)$, and this suggests different effects arising out of our detailed mobility model, since Wang et al. (2009) assume a uniform distribution of susceptible devices within a cell. We also observe a threshold effect— the time taken to infect at least 20% of the devices, starting with 10% initially infected, changes drastically for all values of market share.

Effect of location density. Next, we control the network structure by altering the density within the locations in our dataset. Clearly, a higher density implies higher degree, and would suggest faster spread (analogous to the impact of higher speed in Yan and Eidenbenz 2009). This is indeed borne out in Figure 6c, where we observe a significant variation in the number of infections with d. A surprising feature is that there seems to be some kind of threshold effect, and the dynamics do not change until the density has been altered quite a bit (by a factor of 10).

4.3 Characterizing Interactions between Parameters

So far we have considered the worm and network model parameters in isolation. Next, we study how these parameters couple together and change the response, which in this case is the final % of devices infected. We setup a *balanced factorial experiment design* for the following independent factors (variables):

- Idle time between infection cycles, denoted by T_{idle},
- probability of infection, p_{inf} and,
- market share of the devices, m.

We identify 5 levels for each parameter, and measure the *response variable*, the final % devices infected for 10 replicates (see Table 3). We use the analysis of variance (ANOVA) technique to understand their interactions.

We find that these factors interact very strongly with each other, especially, T_{idle} and m. 3-factor ANOVA is based on the following model:

$$y_{ijkl} = \mu + \alpha_j + \beta_k + \gamma_l + (\alpha\beta)_{jk} + (\alpha\gamma)_{jl} + (\beta\gamma)_{kl} + (\alpha\beta\gamma)_{jkl} + \varepsilon_{ijkl}$$

where,

1. y_{ijkl} is the measurement of the response variable, in this case, it is the percentage of devices infected at the end of the simulation (or the % Final Infections)
2. α_j, β_k and γ_l are the effects of the T_{idle}, p_{inf} and m on the outcome.
3. i is the number of replicates in the experiments and is 10 in this case as we repeat the experiment 10 times for different seed values.

Table 4 shows the results for 3-factor ANOVA to determine the interaction between T_{idle}, p_{inf} and m. Column 1 shows the model number for each of the interaction models indicated in column 2. The parameters that interact are listed in column 3. Columns 4-6 show the interaction study on the response

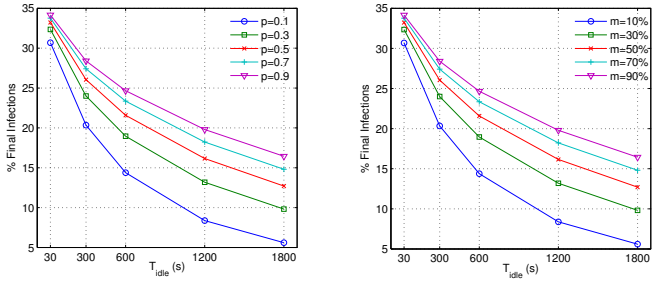
Table 3: Levels or values for the variables we have set in the ANOVA study. There are 5 levels for each factor and 10 replicates for each combination of these parameters.

Factors	**Levels**
T_{idle}	30s, 5min, 10min, 20min, 30min
p_{inf}	0.1, 0.3, 0.5, 0.7, 0.9
m	10%, 30%, 50%, 70%, 90%

Table 4: 3-factor balanced ANOVA to determine the interaction between idle time T_{idle}, probability of infection p_{inf} and market share m. T, P and M represent T_{idle}, p_{inf} and m, respectively. There is a high level of interaction between all the parameters and T_{idle} and m interact the most (Model 3).

No.	**Interactions**	**Source**	**Sum of Squares (SS)**	**Degrees of Freedom (df)**	**F test**
1	All 1-way	[T][P][M]	12053	1237	3791.923*
2	2-way	[TP][TM]	4669	1205	4063.270*
3	2-way	[TP][PM]	7514	1205	6880.722*
4	2-way	[TM][PM]	1002	1205	431.778*
5	All 2-way	[TP][TM][PM]	566	1189	120.075*
6	All 3-way	[TPM]	71	1125	-

variable, the final % of devices infected. The F-statistic shows the interaction and the level of interaction from among the factors. From the table, we see that there is significant interaction between the 3 factors. The interaction in the 3-way automatically indicates this. Among the factors T_{idle} and m interact the most (model 3 in row 4). The the two way interaction that is dropped from model 5 is `[TM]` and so it shows the interaction between the 2 factors. Figure 7 shows the interaction of T_{idle} with p_{inf} and m for the NRV1 network. The x-axis shows the variations in T_{idle} and the y-axis shows the percentage of devices finally infected at the end of the day.



(a) (b)

Figure 7: Interaction of T_{idle} with p_{inf} and m for the NRV1 network plotted as a function of the percentages of devices infected at the end of the day (denoted as "% Final Infections"). **(a):** Shows the interaction between T_{idle} and p_{inf} for a low market share ($m = 10\%$); **(b):** Plots the interaction between T_{idle} and market share, m for a low probability of infection ($p_{inf} = 0.1$).

5 HYBRID SMS/MMS MALWARE MODELING

In the final section, we study the dynamics of the SMS/MMS malware spread and differentiate it with the proximity based Bluetooth malware. We consider two simple and contrasting scenarios: (*i*) a purely SMS/MMS malware that propagates using the SMS/MMS message, referred to as *sms-only*, and (*ii*) hybrid malware that jump from device to device over proximity Bluetooth link as well as a link over the cellular

infrastructure through SMS/MMS messages, referred to as *hybrid*. We find that the dynamics of the *sms-only* malware, under some conditions, are very similar to the Bluetooth malware in nature. In contrast, the dynamics of *hybrid* malware are substantially different from *sms-only* malware. They spread really fast and infect almost the entire susceptible population within a few hours. The basic result is consistent with earlier results and intuitive.

5.1 Implementation of SMS/MMS Malware

We use the EpiCure framework to implement the SMS/MMS malware models. While Bluetooth worms only spread to devices which come within range of the infected device, SMS malware is implemented in the framework as stealthy malware that intermittently sends SMS messages to the people in the address book of an infected device. These malware can be of differing kinds and can select the victim number at random or select the user based on call statistics. A more frequently called individual is targeted before an infrequent one and so on. We evaluate the two approaches and find that in a fast spreading SMS malware the randomness or preferred selection strategy results in similar dynamics.

5.2 Comparing the Propagation Dynamics

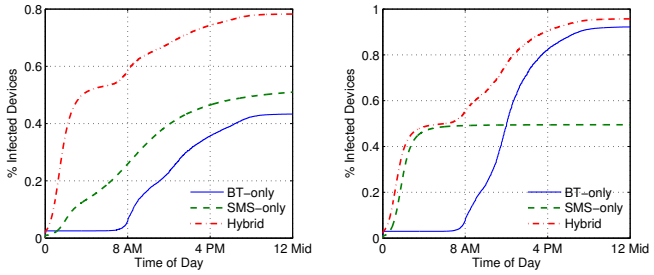


Figure 8: Comparison of the dynamics of Bluetooth, SMS and hybrid malware in the (8a) NRV1 and (8b) NRV2 networks.

Figure 8 shows the propagation of the three kinds of malware on the NRV1 and NRV2 networks (discussed in Section 2.3). The NRV2 network is better connected with more links than the other networks. As expected the hybrid malware propagates faster than either the Bluetooth malware or the SMS malware alone (Figure 8a and Figure 8b). There is an interesting dynamics for the SMS malware in the NRV2 network as shown in Figure 8b. We observe that the Bluetooth malware propagates slowly initially and still achieves a significant growth beyond 12 Noon and infects almost the same % of devices as the hybrid malware. This is an interesting behavior and is due to higher # social links, as discussed in Section 2.3. The social links indicates the contacts in the SMS network (or the social network of the individuals). This is an interesting case where we see that the proximity malware spreads much faster than the SMS malware. Accurately modeling the dynamic proximity network formed due to realistic human mobility in an urban region is crucial to obtain such results.

6 CONCLUSIONS

We described a high performance computing oriented agent-based modeling framework (EpiCure) to study malware dynamics at urban population scale. Building on our earlier work described in Channakeshava et al. (2011), we carry out detailed analysis of malware dynamics on realistic networks. The role of developing realistic human mobility models is critical in inferring such networks. The structure of the network is influenced by the built urban infrastructure as well as daily human activities. Using EpiCure we

studied two important problems: (*i*) effect of market share on malware dynamics; this questions was first studied explicitly in Wang et al. (2009); and (*ii*) dynamics of malware in hybrid systems where a malware uses both cellular and Bluetooth infrastructure (Wang et al. 2009). The role of individual based models was highlighted in these two studies. As smartphones continue to become ubiquitous and IoT becomes ubiquitous, detection, control and response to malware will become all the more important (see, e.g., Heer et al. 2011).

Environments such as EpiCure allow analysts to study emerging threats such as Stuxnet. Furthermore, they can be used in understanding the role fo social media and wireless devices in organizing large-scale emergent human activities e.g. demonstrations, evacuations after a disaster event, etc.

ACKNOWLEDGMENTS

The authors have been partially supported by the following grants: DTRA Grant HDTRA1-11-1-0016, DTRA CNIMS Contract HDTRA1-11-D-0016-0010, NSF Career CNS 0845700, NSF ICES CCF-1216000, NSF NETSE Grant CNS-1011769 and DOE DE-SC0003957.

REFERENCES

Barrett, C., D. Beckman, M. Khan, V. S. A. Kumar, M. Marathe, P. Stretz, T. Dutta, and B. Lewis. 2009. "Generation and analysis of large synthetic social contact networks". In *Proceedings of the 2009 Winter Simulation Conference*, edited by M. Rosetti, R. Hill, B. Johansson, A. Dunkin, and R. Ingalls, 1003–1014. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Beckman, R., K. Channakeshava, F. Huang, J. Kim, A. Marathe, M. Marathe, G. Pei, S. Saha, and A. K. S. Vullikanti. 2013. "Integrated multi-network modeling environment for spectrum management". *IEEE Journal on Selected Areas in Communications* 31 (6): 1158–1168.

Channakeshava, K., K. Bisset, V. S. A. Kumar, M. Marathe, and S. Yardi. 2011. "High performance scalable and expressive modeling environment to study mobile malware in large dynamic networks". In *Proceedings of the 25th IEEE Symposium on International Parallel & Distributed Processing*, 770–781.

Channakeshava, K., D. Chafekar, K. Bisset, V. S. A. Kumar, and M. Marathe. 2009. "EpiNet: A Simulation Framework to Study the Spread of Malware in Wireless Networks". In *Proceedings of the 2nd International Conference on Simulation Tools and Techniques*, Article 6.

Eubank, S., H. Guclu, V. S. A. Kumar, M. V. Marathe, A. Srinivasan, Z. Toroczkai, and N. Wang. 2004. "Modeling Disease Outbreaks in Realistic Urban Social Networks". *Nature* 429 (6988): 180–184.

Ferrie, P., P. Szor, R. Stanev, and M. Mouritzen. 2004. "Security Responses: Symbos.cabir". Technical report, Symantec Corporation.

Heer, T., O. Garcia-Morchon, R. Hummen, S. L. Keoh, S. S. Kumar, and K. Wehrle. 2011. "Security Challenges in the IP-based Internet of Things". *Wireless Personal Communications* 61(3):527–542.

Husted, N., and S. Myers. 2011. "Why mobile-to-mobile wireless malware won't cause a storm". In *Proceedings of the 4th USENIX conference on Large-scale Exploits and Emergent Threats*. USENIX Association.

Kim, J., V. Sridhara, and S. Bohacek. 2009. "Realistic Mobility Simulation of Urban Mesh Networks". *Ad Hoc Networks* 7 (2): 411–430.

Kleinberg, J. 2007. "Computing: The Wireless Epidemic". *Nature* 449 (7160): 287–288.

Li, L., D. Alderson, W. Willinger, and J. Doyle. 2004. "A first-principles approach to understanding the internet's router-level topology". Volume 34, 3–14: ACM.

Ma, J., G. Voelker, and S. Savage. 2005. "Self-stopping Worms". In *Proceedings of the 2005 ACM Workshop on Rapid Malcode*, 12–21.

Mickens, J. W., and B. D. Noble. 2007. "Analytical Models for Epidemics in Mobile Networks". In *Proceedings of the 2007 IEEE International Conference on Wireless and Mobile Computing, Networking and Communication*, 77–84.

Peng, S., S. Yu, and A. Yang. 2014. "Smartphone Malware and Its Propagation Modeling: A Survey". *IEEE Communications Surveys Tutorials* 16 (2): 925–941.

Rhodes, C., and M. Nekovee. 2008. "The Opportunistic Transmission of Wireless Worms between Mobile Devices". *Physica A: Statistical Mechanics and its Applications* 387 (27): 6837–6844.

Su, J., K. K. W. Chan, A. G. Miklas, K. Po, A. Akhavan, S. Saroiu, E. de Lara, and A. Goel. 2006. "A Preliminary Investigation of Worm Infections in a Bluetooth Environment". In *Proceedings of the 4th ACM Workshop on Recurring Malcode*, 9–16.

Symantec 2005, September. "SymbOS.Cardtrp.A". http://www.symantec.com/security_response/writeup.jsp?docid=2005-092215-2634-99.

Wang, P., M. C. Gonzalez, C. A. Hidalgo, and A.-L. Barabasi. 2009. "Understanding the Spreading Patterns of Mobile Phone Viruses". *Science* 324 (5930): 1071–1076.

Yan, G., L. Cuellar, S. Eidenbenz, and N. Hengartner. 2009. "Blue-Watchdog: Detecting Bluetooth worm propagation in public areas". In *Proceedings of the 2009 IEEE/IFIP International Conference on Dependable Systems Networks*, 317–326.

Yan, G., and S. Eidenbenz. 2006. "Bluetooth Worms: Models, Dynamics, and Defense Implications". In *Proceedings of the 22nd Annual Computer Security Applications Conference*, 245–256.

Yan, G., and S. Eidenbenz. 2009. "Modeling Propagation Dynamics of Bluetooth Worms". *IEEE Transactions on Mobile Computing* 8 (3): 353–368.

Yan, G., H. D. Flores, L. Cuellar, N. Hengartner, S. Eidenbenz, and V. Vu. 2007. "Bluetooth Worm Propagation: Mobility Pattern Matters!". In *Proceedings of the 2nd ACM Symposium on Information, Computer and Communications Security*, 32–44.

AUTHOR BIOGRAPHIES

KARTHIK CHANNAKESHAVA (kchannak@gmail.com) is a software engineer at Ericsson, San Jose, CA. He received his Ph.D. in Computer Engineering from Virginia Tech, Blacksburg, VA.

KEITH BISSET (kbisset@vbi.vt.edu) is a senior scientist in NDSSL, Virginia Tech. He holds a Ph.D. in Computer Science from New Mexico State University, New Mexico.

MADHAV M. MARATHE (mmarathe@vbi.vt.edu) is a Professor in Dept. of Computer Science and Director of NDSSL at Virginia Tech. He received a Ph.D. in Computer Science from the University at Albany, SUNY.

ANIL KUMAR S. VULLIKANTI (akumar@vbi.vt.edu) is an Associate Professor in Dept. of Computer Science and NDSSL at Virginia Tech. He holds a Ph.D. in Computer Science from IISc, India.