Coupled Multi-Agent Simulations for Mobile Security & Privacy Research

Benjamin Henne, Christian Szongott, Matthew Smith
Distributed Computing & Security Group
Leibniz Universität Hannover
Hannover, Germany
Email: {henne, szongott, smith}@dcsec.uni-hannover.de

***Abstract*—The complexity of mobile device digital ecosystems is rapidly increasing. Not only the absolute number of mobile devices, but also their diversity is on an ongoing rise. Devices are equipped with multiple short-range communication technologies that are used by different applications, which can become entry points for security or privacy threats. These threats need to be addressed in an early stage, while the mobile ecosystem is still relatively small and flexible. In prior work we presented the advantages of simulation approaches in the field of security and privacy research within the mobile domain. In this paper, we extend the Mobile Security & Privacy Simulator, which allows us to model the ecosystem of mobile devices and its surrounding environment with more details to achieve more realistic results. For this purpose we build up additional sub-simulations for discrete real world sub-scenarios like indoor places and connect them with the main map-based simulation. By this coupling we are able to better fit a simulation to the real world using tailored models for agents as well as for their environment depending on each sub-scenario's characteristics.**

keywords-security; privacy; mobile; coupled simulation

I. Introduction

The ecosystem of mobile networked devices is rapidly evolving and growing since the birth of the smartphone and modern tablets. Personal and business devices with mobile broadband network, GPS-support and short-range communication interfaces such as Bluetooth or NFC, and device-to-device ad-hoc networks become the central digital hub in our lives. People use mobile devices for traditional applications such as working on local documents, browsing the web, reading and writing emails, or doing their online banking. But they are also used for modern applications like participating in social networks, consuming and publishing social media, using context-dependent and in particular location-aware services, or accessing any kind of data in the cloud. Additionally modern device-to-device communication such as contact-less payment allows for direct data exchange in the mobile ecosystem without using any provider network.

At the end of 2011 the number of apps for mobile devices rose beyond one million (iOS plus Android). These apps allow users to do nearly anything with their mobile devices. However, the immense number of apps renders the protection of the security and privacy of mobile devices nearly impossible. Neither the users themselves nor the app distributors are able to test each new app for such threats accurately enough. While the diversity of apps rises, the number of the mobile operating systems drops due to the rising market share of only a few big players, dominated by Apple's iOS and Google's Android.

The complex ecosystem of mobile context-enabled devices brings along a multitude of security and privacy implications. Those implications range from security issues that arise from malicious software trying to steal data to privacy issues that emerge from wrong or careless use of context information such as public geo-referencing of social media. Although the number of devices in this ecosystem has already grown to a considerable height and applications are very diverse, we are still at the very beginning of a new era of mobile computing. The upcoming era of everybody's ubiquitous access to contextualized data as well as the pure mass of the big data [1] involves lots of security and privacy issues. It is vital that suitable new usage concepts and countermeasures are in place before the mobile population grows to a size where threats are hard to be dammed up.

Research in the field of security and privacy of digital ecosystems is fraught with complications for a number of reasons: Real world privacy issues are hard to study without invading the privacy of people involved in a study. Thus it is doubly hard to evaluate countermeasures if real world problems cannot be studied in detail. Due to these reasons research in the field of privacy such as obfuscation [2] or k-anonymity [3] often is theoretical in nature. One main problem with most of these theoretical works is that they assume users to be mostly similar in matters of their privacy-related behavior. However, privacy is a very personal issue with a wide variety of perceptions, requirements and feelings that cannot be captured by any unifying model.

Studying security-related aspects of digital ecosystems also has severe limitations due to ethical and legal issues. For instance it is not possible to study the effectiveness of a countermeasure against a self-propagating virus since releasing the virus to test the countermeasure is not possible. Although laboratory setups can be implemented to test small-scale infections, they cannot be used to prove any assumption about the capability to stop an epidemic spread of the malware.

To explore these kind of scenarios that suffer from the problems mentioned above, we built an agent-based Mobile Security & Privacy (MoSP) simulator. The MoSP simulator [4] enables us to evaluate questions about threats to personal privacy, mobile malware spreading or the effectiveness of

countermeasures using simulation without getting in conflict with law or ethics. In prior work, the MoSP simulator was primarily used to investigate mobile environments based on real world road maps. If people left the road network and went into buildings, the simulator had to switch back to mathematical models for indoor scenarios. In this paper we present a coupled-simulation extension of the MoSP simulator to include further simulations for indoor sub-scenarios. To achieve this, multiple synchronized simulations (one map simulation, multiple indoor simulations) are coupled and thus allow for a more realistic and versatile modeling of the mobile ecosystem. The rest of this paper is organized as follows: Section II describes the current version of the MoSP road map level simulator. In Section III we present our extension of the Siafu context simulator for indoor simulation. The coupling of both is described in Section IV. In Section V we show a mobile security example scenario for the coupled MoSP simulation. Related work is summed up in Section VI. Section VII concludes the paper and portrays possible future work.

II. Mobile Security & Privacy Simulation

The Mobile Security & Privacy (MoSP) simulator[1] has been built for the agent-based simulation of privacy and security scenarios on a road map level. The simulator is built upon SimPy [5], a process-based discrete-event simulation language where each person is simulated by a process that is responsible for the agent's movement and main logic.

People move in a 2-dimensional Euclidean space and most of the movement occurs along roads. Coordinates exact down to the centimeter-scale. This allows even physical contact or NFC interaction of people to be simulated. Unlike most other simulators roads are also modeled with width that is extracted from existing geo data to be able to simulate walk-by and related movements. These parameters have a measurable impact on security and privacy evaluations that rely on peoples' distance. Geo data like the road network, street widths or points of interest is taken from the OpenStreetMap project.

Information about the world (road names or types, locations and metadata of points of interest, or areas) can be retrieved from the geo model by any agent. The geo model can be queried for other agents in a certain proximity, which can be used for radio range as well as visual contact evaluations.

The agents' movement can be random or routed - stopovers and destinations are determined separately by each agent's logic. The main logic that steers an agent's actions and interactions can be implemented in different ways. Using a deterministic finite state machine is common, but other models like the belief-desire-intention model can be used as well.

A person can have multiple assigned actions, which represent the interactions of the person or an electronic device it is carrying along during the simulated time. Depending on its type, an action can be implemented in different ways as shown in Figure 1: First, it can be implemented in the person's logic and hence is only executed each time the person is "thinking". Second, an action (for instance an independently acting mobile device) can be implemented as a separate action process that is bound to the person. Third, a person's action can be triggered by another person, for instance if two persons are meeting.

For each node of the road network graph a world object can be defined. People passing such a node can interact with it. There are two types of world objects: Passive objects waiting for an interacting agent (a toll booth in which an agent actively pays a fee) and active objects that can also act autonomously, like for instance a camera taking CCTV footage of passersby.

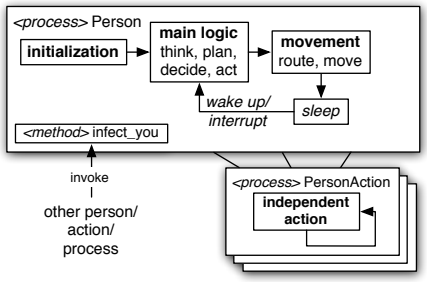

Fig. 1. MoSP simulator: a Person and its Action implementations

III. Coupled simulations

Security and privacy simulations of mobile ecosystems are characterized by lots of people, which are moving, communicating and interacting with their surrounding and each other. To gain best results when simulating such an ecosystem of mobile devices, the agents' logic as well as their surrounding has to be modeled as detailed and realistic as possible.

A single simulation typically considers only one topology for agents' relationships and interactions. Such a simulation is capable of modeling a specific environment with a particular degree of detail. The boundaries of digital ecosystems are however fluid and new environments, devices and scenarios merge into the overall system. We found no single simulator that is capable of simulating the depth and breadth needed to study security and privacy threats in its entirety. The demand for realistic simulations in this context comes to a point, where existing topologies/models are not capable to incorporate enough details into the simulated world. To bridge this gap, we propose to couple different kinds of simulators [6] each having a specific model for a dedicated sub-scenario.

The MoSP simulator for instance can be used to investigate problems on a road map level. Simulated people are able to walk along roads and leave them at specific locations for some time, relaxing or taking photos, which can affect the privacy of near persons. The model for such a MoSP simulation is a road map. Thus the degree of detail is also that of a road map. If we want to enable people to enter buildings we collide with the limits of the geographic model. The limited road map model is not capable of describing the interior of buildings in a detail (walls, furniture, hotspots) that is sufficient to serve as the basis for a person's indoor movement and actions.

[1] Online: http://bhenne.github.com/MoSP/

Up to now the only possibility to simulate indoor parts of a scenario was to comprise mathematical models for such locations into the simulation. But there are some disadvantages of such mathematical models. As exposed in [4], simulation has to be preferred to any theoretical model. In the following we present how indoor parts of a complex scenario can be modeled and researched along with the existing road-level simulations by creating a sub-scenario simulation for each indoor place that is connected to the road-level simulation.

For modeling indoor scenarios we extended the *Siafu* context simulator [7]. Siafu originally has been designed to evaluate mobile context-aware applications and services in a 2d-plane. To simulate buildings detailed enough for mobile security and privacy evaluations, we added the ability to simulate multiple stories by extending Siafu's world to 2.5-dimensions with a height attribute. The vertical aspect of multiple-story buildings is one of the main differences between Siafu and the MoSP road level simulation. Inside buildings people can be located at different height levels and infections or privacy breaches can occur over neighboring stories. Simulated agents move around within one story and change stories using stairs or elevators that have been implemented as well.

In contrast to the geo data based MoSP simulator, the world model of Siafu is defined by a set of images. An alpha map defines pixels that agents cannot walk on, like walls, tables or other obstacles. Additional images define locations of places (like seats, stairs or hotspots). The distribution of context variables (like WiFi or GPS reception) can be modeled by overlay images with colors representing local values for each pixel. An exemplary indoor map is shown in Figure 3. In the simulation agents move step by step in the Moore neighborhood of the pixel-based square lattice. Path finding is done based on gradient maps. To be conforming to our MoSP simulation requirement of exact positioning, we extended the world model to further use the eight-cells-surrounding for navigation, but added a more exact sub-pixel-size positioning for the agents. The corresponding pixel-to-meter mapping and the size of a story is defined within the configuration.

The centerpiece for the agents' behavior is the agent model. Here the behavior and actions of the agents is defined. An agent can act on his own like taking a drink in a cafe, sitting there while checking emails on his iPad. Additionally, agents can interact with each other based on the vicinity of other agents like sharing photos via ad-hoc communication, or places where agents receive discount coupons from a Bluetooth hotspot. This is where MoSP comes into play. For wireless communication we extended the vicinity-detection to also consider agents in nearby stories. In our 2.5-dimensional model a signal's radio reception in other stories is calculated by leaving the horizontal coordinates unchanged and calculating a reduced radio radius depending on the height and absorbability of the intermediate ceiling.

IV. Coupling of Simulations

For coupling a single road simulation with subordinate sub-scenario simulations the most obvious way is a MoSP road simulation having active locations, which are represented by indoor simulations that are controlled directly by the main road simulation. People interacting with such a location are sent to and received back from the indoor sub-simulation.

The transfer of agents between both simulations has to be made after all processes (people, actions and locations) ended their current simulation step, because of possible ongoing agents' interaction. Thus, the transfer cannot not be made by a location itself. It only appends the transfer to a queue that is processed between simulation steps. The actual transfer process is responsible for the message exchange and thereby for the transfer of agents and the timing of all simulations.

For the coupling of all simulations we use a multi-server model where each simulation implements an external control as a server and a central control server controls the simulations and dispatches messages between them. The control server always knows the actual or at least the last state of every controlled simulation and can react to connection and timing problems. Connecting different simulators does not imply that each simulator has to implement a common protocol. The central control server is the only one that has to know all control commands. It has to translate messages between simulations with different message formats. This eases the integration of new simulators. To simplify the usage of our coupled simulation framework, the control server only needs to know where to find the simulations that are to be coupled (which IPs/ports to connect). Before starting the coupled simulation it queries each simulation for all other necessary information. By this, the configuration for each location is made at the specific location and the usage of sub-simulations becomes as modular as possible.

Figure 2 shows the protocol scheme for controlling a coupled simulation. At the very beginning of a simulation the control server queries every simulation for its identity. If later one simulation wants to send data to another, it only needs to know the location name like "Mona's coffee dreams" or the connection id instead of the IP address for that indoor simulation. The control and dispatch process operates as follows: First, all simulations are commanded to *step 1* step (a simulated second) forward. The control server waits until all simulations stop again and notify this by a *step_done* message. A simulation that needs to send data to another can push data with a variant called *step_done_push*. By always using push messages, the optional *get* command including more communication overhead could become unnecessary. The *get* command is used to collect messages from simulations. When all data is collected, the control server processes it. It may translate data to other formats based on the simulation type or simply put each record in the destination's queue. After this process, the control server sends the corresponding data (*put*) to the destination simulations. Finally it starts again with a new *step 1* message. In the current version of the command and dispatch protocol messages can only contain information about agents that change simulations and log messages that are centrally stored by the main simulation. In a future version other data like world properties could be exchanged as well.

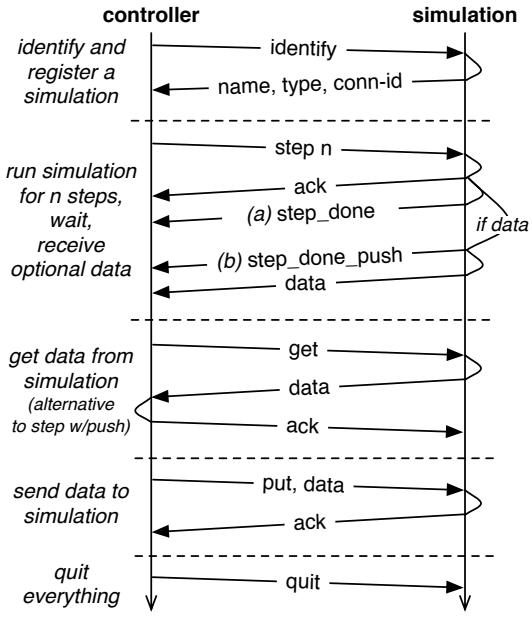

Fig. 2. Control and Dispatch Protocol for coupled Simulations

A. Scenario setup

The setup of a coupled MoSP simulation is fairly easy. Road map nodes that have to be connected to a sub-scenario simulation are marked with a special tag in the OSM geo data. These locations are selected as sub-scenario locations in the major road map simulation: For each corresponding node a world object is generated that handles the transfer of persons to the sub-scenario simulation via the control server. Indoor simulations are generated based on the tagged geo data and simulation templates by an automated configuration tool.

V. Coupled Simulation Example

The complexity of digital ecosystems with regard to mobile devices is rapidly increasing. While devices hitherto communicate using local networks like public WiFi hotspots or corporate networks more and more devices also communicate in an ad-hoc fashion without the need of any network infrastructure to be present. With the growing number of communication technologies, the number of possible entry points for security threats also rises. Using a local backend infrastructure like a WiFi network, devices can communicate in the whole coverage area of that network. In the case of ad-hoc communication, the distance between each two devices is relevant if a device-to-device infection is possible. Thereby one device infects another and both move and spread the infection to other devices later. In such a case, the question in place is how and when the spread of such a mobile infection becomes epidemic. It is necessary to identify which are the crucial parameters that drive the spread of an infection to become epidemic.

Besides the technical implementation of an epidemic spread, the parameters in question are the technical transmission of the infection (communication protocol and initiation), the transmission duration and range, and eventually an incubation time that has to elapse before another device can be infected.

In the following we evaluate the correlation of the transmission range and the infection count based on our coupled simulation. In the evaluated scenario an attacker tries to infect mobile devices with a device-to-device infection. We assume the transmission time of the infection to be 15 seconds. Choosing a cafe as the primary attack location allows the use of short-range transmissions of the generic infect, since people stay closely together for a long time compared to moving people outside of buildings. People that get infected inside the cafe will subsequently carry out the infection.

The simulated scenario takes place in the district Linden-Nord of the city Hannover. Using the MoSP simulator, we simulate the outdoor activity. People move within the district ($v = 1.2\frac{m}{s}$) and stop off for some time t (30s $\leq t \leq$ 20min). Sometimes they decide to walk to a public place to relax or meet friends, and stay there (2min $\leq t \leq$ 2h). With the same probability they decide to visit a cafe for some time (5min $\leq t \leq$ 2h). The district can be simulated as a closed system or open system. In a closed system people do not enter or leave the simulated area during the simulation time. In an open system agents have a set probability of leaving the simulation and new agents can enter as well. In the following we use a closed system simulation with a duration of 8 hours. The total number of inhabitants of the simulated area is about 16,000 people (demography: $< 18 : 13\%, > 59 : 16\%, 18{-}59 : 71\%$). Based on demography, we assume that 80% of the inhabitants own a mobile phone, whereof 50% are vulnerable mobile devices i.e. smartphones with a suitable vulnerability. Based on estimation a quarter of all people are on the road. Thus, the simulated population for our scenario involves 1,600 people. For the simulation we defined 60 partly overlapping public places of different sizes on the map. The cafes in this scenario are simulated as sub-scenario simulations using the extended Siafu simulator. We selected 6 popular cafes on the map and simulate each with 80 seats and an approximate size of $2{\times}480\text{m}^2$. Figure 3 shows the first floor of such a cafe with the seats marked in red and entrance marked in green. The screenshot shows 16 agents, 11 of which have devices that are not infected (blue dots) and 5 with devices that are infected with the mobile malware (red triangles).

When people are transferred from the road level simulation to the indoor simulation they select a free seat, walk to it and stay there. We assume that they use their mobile device on average every 15 minutes. When devices are actively used by their owners they can get infected. As described in Section III, devices can be infected across stories. Outside the cafes people use their device less often (about once an hour).

Figure 4 shows the infection numbers for different infection radii in the described scenario. The four different graphs show the infections at the four different locations agents can be. Agents walking on the road and agents waiting between walks are infected with the same ratio. Walking people can only get infected by other people walking in the same direction with

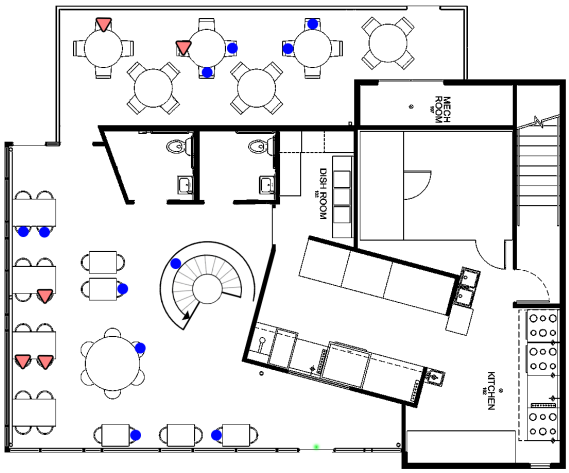

Fig. 3. Screenshot of an indoor simulation showing infected (red triangles) and uninfected (blue circles) agents on the first floor of a cafe environment

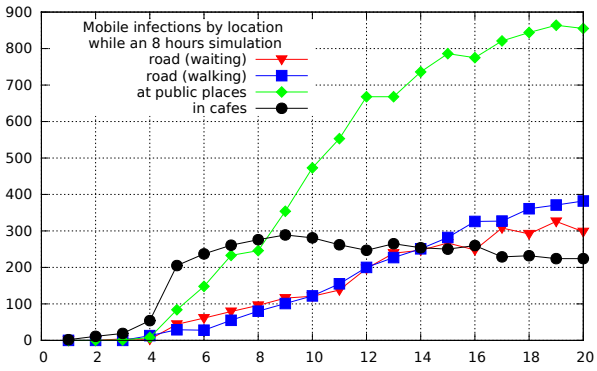

Fig. 4. Mobile Infections per Location depending on Infection radius (1,600 people within 8 hours)

TABLE I
LOW-RANGE MOBILE INFECTIONS PER LOCATION

radius *(in m)*	indoor cafe	public place	road (walk)	road (wait)
1	2 (0)	0	0	0
2	11 (0)	0	0	0
3	19 (0)	1	0	4
4	54 (0)	8	13	3
5	205 (38)	84	29	44
6	237 (71)	148	28	61
7	261 (101)	233	55	79

Cafe numbers in braces are infections across stories.

a distance less than the infection radius. Infections between people walking in different directions need an infection radius of at least 36m ($2 \cdot 1.2\frac{m}{s} \cdot 15s + r$, r = orthogonal road offset $\leq \frac{1}{2}$ road width). Drive-by attacks to people waiting on the road are only possible at infection radii greater than 18m if applying the 15 seconds infection duration. For these reasons infections on the road are lower compared to other ones.

The number of infections at public places is the highest of all. Looking at the details of this infection type, there are different reasons for the high amount: The chosen public places all are located besides a road or in a pedestrian area. By this at least one eighth of people infected at a public place have been infected by people waiting or walking in the road. About a quarter of the public places have a size (offset from center) less or equal 12m and the size of nearly all does not exceed 15m. Public places partly border to each other or even overlap. Thus about one eighth of the infections are transmitted between different public places.

While infections outside start approximately after one and a quarter hours when infected people leave the cafe, infections in cafes start right after the simulation begins. Infections outside are propagated slowly for radii lower than 6m. As shown in Table I, the infection already spreads as of radii between one and two meters at the indoor simulations compared to the outdoor one. This is the distance of each two people at the same table or some neighboring tables (cf. Figure 3). Infections inside the cafe propagate from table to table for short distances, while higher distances include infections across tables and stories. The infection number in all cafes converges against about 300 infections for radii greater than 8m. This is caused by the cafe size of about 19m $\times$ 24m with an effective infection area of about 15m $\times$ 20m. Increasing the infection radius in a cafe to more than 8m does not change the amount of infections, because the whole cafe area is already covered by the 8m in the mean and additionally more than one person is rapidly infected inside the cafe. Compared to public places, the current sub-scenarios are closed systems. Infections cannot occur across the border of road and indoor simulations. Thus, neither drive-by or "stand-outside" infections happen inside the cafe nor the other way around.

By modeling the cafe location in detail using the sub-scenario simulation we get a much more realistic behavior and placement of people. Thus, simulation results are more realistic than using one global model approach. Compared to the simulated public places where people are placed by adding a location-dependent random offset to the center coordinates of the place, the sub-scenario modeling allows us to take details into account that are not covered by the road-level geo model.

VI. RELATED WORK

In [8] Mascetti et al. present the impact of user movement in the evaluation of privacy-preserving techniques for location-based services. They compared experimental results of evaluations with mostly random movement on the one hand and generated movement data created with the Siafu simulator [7] on the other. They found out that there is a great need for specific scenarios and movement since random movement leads to significantly different and unrealistic results.

The spreading of malware via different communication technologies has been evaluated in multiple works. Wang et al. investigate in [9] a mobile virus outbreak via MMS and Bluetooth based on anonymized users' mobile phone billing records and coordinates of closest mobile phone towers, while Su et al. use a trace-driven simulation in [10] to evaluate the propagation of Bluetooth worms. While users' locations of

the first are too coarse-grained, traces of the second base on a MIT reality mining project of 100 students that limitedly represent everybody's daily routines and mobility. Akriditis et al. present in [11] a simulated study of WiFi-based malware in metropolitan area networks.

In [12] Carianha et al. present an improved mix zones approach to preserve location privacy in VANETs. For their evaluation they use the VeinS framework, which is based on a bidirectional-coupled simulation model. VeinS integrates the network simulator OMNET++ and the road traffic micro-simulation tool SUMO. It incorporates different kinds of simulation (communication and movement) instead of coupling similar simulations with different environment details.

UDel Models is a simulation suite for simulating MANETs and urban mesh networks with the focus on realistic mobility and propagation. The 3d mobility model includes pedestrian movement in multi-story buildings and on outdoor sidewalks. In [13] Kim et al. present a layered mobility model that consists of activities and sub-tasks based on time use and management research studies, and node dynamics as grouping and speed-distance relationships. The mobility model is used to generate traces, which can be used with other simulators.

MAGS [14] is a multi-agent geo-simulator that can be used to simulate people in a 2d or 3d virtual environment. While MAGS originally simulates people outside, there is also a Mall_MAGS prototype [15], which simulates shopping people in a mall. Both systems do not incorporate mobile devices, security and privacy. Additionally, there exists no coupling of both systems up to now.

Further related work to the use of simulation for mobile security and privacy research has been summed up in [4].

VII. Conclusion and Future Work

In this paper we presented a coupled simulation for security and privacy research in mobile ecosystems. The coupling of simulations allows the different aspects of the environment of the digital ecosystem to be modeled separately but studied as a whole. We coupled an outdoor simulation and indoor simulations populated by agents using mobile smart devices. The agents can have different usage behaviors dependent both on their current activity and their current location. Modeling the capabilities and usage patterns of different smart devices allows us to study the qualities of different threats. As an example we showed the epidemic qualities of different types of mobile malware, dependent on the transfer mechanism. The range and the usage scenario have significant effects on which environments become the infection hotbeds and how the malware propagates. This can prove valuable knowledge when designing countermeasures against emerging threats to mobile device ecosystems.

There are several areas for future work. Firstly the level of detail and realism of the simulation can be further enhanced, by modeling more types of locations, in particular the introduction of work areas in combination with a daily routine of commuters. To simplify simulation definition of indoor spaces an abstract coupled simulation definition based on OSM maps, images and a visual agent modeling as used by tools like Repast Simphony[2] could be used. Based on this definition, code and configurations for the different simulators could then be generated automatically. Also a more flexible interaction channel between the simulators is desirable. This would for instance allow someone waiting outside a cafe to be infected by a device within the cafe.

The second area of future work is the actual study of different security and privacy threats that arise in mobile device ecosystems by the use of MoSP simulation. The integration of GPS receivers and high definition cameras into virtually all new devices, the growing use of device-to-device communication, location-based services and NFC payment systems all offer interesting research questions which should be studied before full real world deployment has been achieved.

References

[1] M. Smith, B. Henne, C. Szongott, and G. von Voigt, "Big data privacy issues in public social media," in *2012 Digital Ecosystems and Technologies Conference (DEST), to appear*, 2012.

[2] C. a. Ardagna, M. Cremonini, S. De Capitani di Vimercati, and P. Samarati, "An Obfuscation-Based Approach for Protecting Location Privacy," *IEEE Transactions on Dependable and Secure Computing*, vol. 8, no. 1, pp. 13–27, June 2009.

[3] B. Gedik and L. Liu, "Protecting Location Privacy with Personalized k-Anonymity: Architecture and Algorithms," *IEEE Transactions on Mobile Computing*, vol. 7, no. 1, pp. 1–18, Jan. 2008.

[4] B. Henne, C. Szongott, and M. Smith, "Towards a mobile security & privacy simulator," in *2011 IEEE Conference on Open Systems (ICOS)*, sept. 2011, pp. 95 –100.

[5] (2012, Feb.) SimPy: Simulation in Python. [Online]. Available: http://simpy.sourceforge.net/

[6] M. Smith, S. Hanemann, and B. Freisleben, "Coupled simulation/emulation for cross-layer enabled mobile wireless computing," in *Proceedings of the Second International Conference on Embedded Software and Systems*, 2005, pp. 375–385.

[7] M. Martin and P. Nurmi, "A Generic Large Scale Simulator for Ubiquitous Computing," in *2006 3rd Annual International Conference on Mobile and Ubiquitous Systems - Workshops*, Jul. 2006, pp. 1–3.

[8] S. Mascetti, D. Freni, C. Bettini, X. S. Wang, S. Jajodia, and U. D. Milano, "On the Impact of User Movement Simulations in the Evaluation of LBS Privacy-Preserving Techniques," in *Proceedings of the 1st International Workshop on Privacy in Location-Based Applications, Malaga, Spain, October 9, 2008*, 2008.

[9] P. Wang, M. C. González, C. A. Hidalgo, and A.-L. Barabási, "Understanding the spreading patterns of mobile phone viruses." *Science (New York, N.Y.)*, vol. 324, no. 5930, pp. 1071–6, May 2009.

[10] J. Su, K. K. W. Chan, A. G. Miklas, K. Po, A. Akhavan, S. Saroiu, E. de Lara, and A. Goel, "A preliminary investigation of worm infections in a bluetooth environment," in *Proceedings of the 4th ACM workshop on Recurring malcode*, ser. WORM '06, 2006, pp. 9–16.

[11] P. Akritidis, W. Y. Chin, V. T. Lam, S. Sidiroglou, and K. G. Anagnostakis, "Proximity breeds danger: emerging threats in metro-area wireless networks," in *Proceedings of 16th USENIX Security Symposium on USENIX Security Symposium*, ser. SS'07, 2007, pp. 22:1–22:16.

[12] A. Carianha, L. Barreto, and G. Lima, "Improving location privacy in mix-zones for vanets," in *Performance Computing and Communications Conference, 2011 IEEE 30th International*, nov. 2011, pp. 1–6.

[13] J. Kim, V. Sridhara, and S. Bohacek, "Realistic mobility simulation of urban mesh networks," *Ad Hoc Netw.*, vol. 7, pp. 411–430, Mar. 2009.

[14] B. Moulin, W. Chaker, J. Perron, P. Pelletier, J. Hogan, and E. Gbei, "Mags project: Multi-agent geosimulation and crowd simulation." in *COSIT*, ser. LNCS, vol. 2825, 2003, pp. 151–168.

[15] W. Ali and B. Moulin, "2d-3d multiagent geosimulation with knowledge-based agents of customers' shopping behavior in a shopping mall," in *Spatial Information Theory*, 2005, vol. 3693, pp. 445–458.

[2] Online: http://repast.sourceforge.net/