Hindawi Publishing Corporation
Abstract and Applied Analysis
Volume 2014, Article ID 623436, 10 pages
http://dx.doi.org/10.1155/2014/623436

Research Article

Agent-Based Model to Study and Quantify the Evolution Dynamics of Android Malware Infection

Juan Alegre-Sanahuja,[1] Javier Camacho,[1] Juan Carlos Cortés López,[1] Francisco-José Santonja,[2] and Rafael Jacinto Villanueva Micó[1]

[1] *Instituto Universitario de Matemática Multidisciplinar, Universitat Politècnica de València, 46022 Valencia, Spain*
[2] *Departamento de Estadstica e Investigación Operativa, Universitat de València, 46100 Valencia, Spain*

Correspondence should be addressed to Juan Carlos Cortés López; jccortes@imm.upv.es

Received 2 August 2014; Accepted 22 September 2014; Published 28 October 2014

Academic Editor: Jinde Cao

In the last years the number of malware Apps that the users download to their devices has risen. In this paper, we propose an agent-based model to quantify the Android malware infection evolution, modeling the behavior of the users and the different markets where the users may download Apps. The model predicts the number of infected smartphones depending on the type of malware. Additionally, we will estimate the cost that the users should afford when the malware is in their devices. We will be able to analyze which part is more critical: the users, giving indiscriminate permissions to the Apps or not protecting their devices with antivirus software, or the Android platform, due to the vulnerabilities of the Android devices that permit their rooted. We focus on the community of Valencia, Spain, although the obtained results can be extrapolated to other places where the number of Android smartphones remains fairly stable.

1. Introduction

The security in devices connected to the Internet is an issue that has long been concerned, from governments and companies to individual users. However, this threat seems not being perceived by the smartphone users taking into account the potential risky behavior of them and the sensitive data and pictures the users store in their devices. Moreover, the risk increases with the new companies policies that permit the employees the use of their own smartphones in the work accessing to company sensitive data and applications (bring your own device BYOD).

Different types of malware have already been documented [1, 2] and it may be a threat that must be studied to quantify the users' potential risk. Here, we will focus on Android platform because most of the smartphones use Android OS [3].

During the year 2011 appeared the first study on the characterization of viruses on mobile OS Android [1, 2]. This study categorizes the types and families of viruses found, depending on the type of installation, activation, effects on the infected device, the user management of the permissions, and so forth, showing the diversity of different virus families and the ineffectiveness of the traditional antivirus methods on mobile devices.

Also, there are several works that have approached the analysis and detection of malware on the Android platform [4–6]. The common objective of these works is to propose new methods of virus detection on mobile devices from a dynamic point of view, that is, to detect at runtime anomalous or unwanted behavior of the device (system calls, network access, and memory or file modifications). In contrast, static and classic antivirus methods are based on repositories of previously known viruses that do not protect the user in case of the spread of an unknown new virus type. However, dynamic detection of viruses is unsuitable for mobile devices for their CPU and memory consumption. The two approaches, static and dynamic methods, have their own advantages and disadvantages, and both may be bypassed and unable to avoid the spread of new viruses.

1.1. State of the Art. In the literature, there are several approaches to the mathematical modeling for the spread of viruses on mobile devices. In [7] the authors describe a framework and the main guidelines to design reliable agent-based malware models considering infections via SMS/MMS, Bluetooth RF, IM, P2P, and email. In [8–10] the authors propose approaches based on mathematical epidemic techniques where the malware infection follows similar dynamics to the infectious diseases.

Also, there are models based on the physical architecture of the mobile and wireless networks [12] or based on the mobility of the users, but they do not consider the interconnectivity based on the exchange of applications [9].

To the best of our knowledge, we do not know any paper showing quantification, prediction and/or simulation about how the users install malware Apps. However, literature about the application of machine learning techniques to detect malware Apps in the markets can be found [6]. Nevertheless, any of the above approaches do not take into account the infection model based on an App-market ecosystem, like smartphones environment does.

1.2. Our Proposal. Likely, the model guidelines suggested in [7] are the most suited to the current scenario. In that contribution, an agent-based model of malware dynamics covering all of the possible infection models except the App-market ecosystem model is proposed. The integration of the App-market ecosystem is the key aspect that we will consider in this paper.

As we indicated previously, researchers and companies characterized mobile malware and proposed alternative methods to prevent, detect, and avoid mobile malware. Also, different companies publish periodically mobile malware reports with estimations and statistics. However, in the literature there is a lack of studies that quantify the effects of the malware infection in the Android platform in order to show realistic data to know the extent of the threat as our model does [13]. Our model (*App-Model*) complements the agent-based malware modeling suggested in [7] introducing a new infection process based on applications downloaded from the App-market. In Figure 1 we can see a rough description of the items which we deal with to build the App-Model.

The App-Model will quantify the Android malware infection evolution (to know the real threat for the users), the number of potential infected smartphones (to estimate the population of smartphones affected by malware), and the type of malware that affects these infected smartphones in the community of Valencia, Spain [14]. The results can be exported to other regions where the number of Android smartphone users is fairly stable.

We must say that other approaches as machine learning or data mining techniques could be used to study the evolution of the malware infection; however these techniques do not take into account the behavior of the actors (markets, Apps, and clients). The knowledge of their behavior and how they interact allows simulating new scenarios where the behavior may be different and predicting the evolution of the malware infection considering these changes.

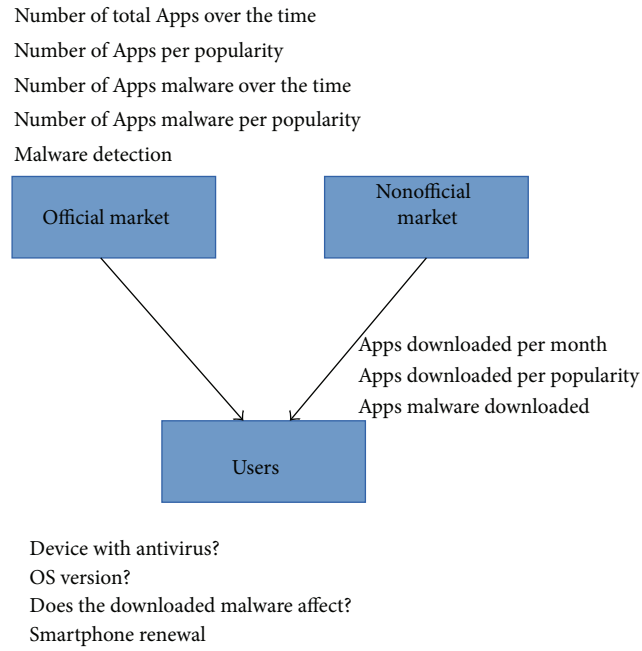

Figure 1: General structure of the agent-based model. Issues we are going to take into account in the modelling process.

Additionally, note that with the results of the model we will be able to analyse the critical part of the smartphones business model related to malware; that is, we will find out which part is more critical: the users, giving indiscriminate permissions to the Apps or not protecting their devices with mobile antivirus software, or the Android platform, due to the vulnerabilities of the Android devices that permit their rooted. Furthermore, we will be able to estimate the cost that the users should afford in case that they have in their devices malware that causes financial charges.

The paper is organized as follows. In Section 2 we present the agents of the model: Apps, markets, users, habits, and so forth. In Section 3 we describe how the agent-based model evolves over the time. Section 4 is devoted to carry out simulations, present results, and discuss them. Conclusions are drawn in Section 5.

2. Material and Methods

To conduct our study, we set the time period in a month. The starting time-point ($t = 0$) is Jul 2011. This has been chosen because in Jul 2011 none or only very few smartphones could have been infected.

The agent-based approach allows the analysis of service interactions among the agents and fits perfectly the relation between mobile device users and App-markets.

Then, in this model, two domains including their agents will be considered: the markets, where the agents are the Apps that belong to different markets, and the users, where the agents are the mobile devices (or clients) that belong to every user. The study of the behavior of the agents has been studied in reference [15].

The App has the attributes *malware* and *type* that indicate whether an App is malware and its type, respectively. Given that the effect over the client produced by a malware App can be one or more of the malicious payload described, we consider that if a malware App carries more than one payload, the type of the malware App belongs to the most upper level payload, according to financial charge, privilege escalation, remote control, and information collection [15].

Whether the *client* is infected or not, the OS version, if the device has or not software protection, and the kind of infection are the attributes of the client. The *privilege escalation* malware affects the client depending on the OS version [15].

Additionally, we consider that *clients* download a certain number of Apps every month, determined by *download* method, select the downloaded App by the method *selection*, and determine if the downloaded App infects the client or not with the *infection* method. More details related to download process are the following.

(i) *Download method*: we admit that the number of Apps downloaded by a user in a month follows a Poisson distribution:

$$f(k,\lambda) = \frac{e^{-\lambda}\lambda^{k}}{k!}, \quad k = 0,1,2,\ldots, \tag{1}$$

where k is the number of downloaded Apps and $\lambda > 0$ is the average number of Apps downloaded every month in every smartphone.

(ii) *Selection method*: knowing k from *download method*, this method selects randomly k Apps from the markets. The selection will depend on the popularity and the number of downloads.

(iii) *Infection method*: with the k selected Apps, we take the ones that are malware, and this method determines if the App affects the *client* or not, depending on the App attributes (*malware* and *type*) and the *Client* attributes (OS version and antivirus).

Now, we are going to summarize, until the end of this section, the main results given in [15] that we will use throughout this paper.

2.1. Official Market. Let us describe the main features of the official market also known as Google Play [16].

(i) Considering that the data of the new Apps entering every month in the official market show a linear trend, it can be modeled, in the mean square sense, by the function

$$f_{\mathrm{OM}}(t) = 225\,970 + 20\,740.1t, \quad t = 0,1,2,\ldots, \tag{2}$$

where t is the number of months since July 2011. More details are in [15].

(ii) Analogously, the number of malware Apps in the official market is modeled by the function

$$f_{\mathrm{OM}m}(t) = 64.9 + 35.9t, \quad t = 0,1,2,\ldots, \tag{3}$$

where t is the number of months since July 2011 which corresponds to $t = 0$. More details are in [15].

(iii) Distribution of Apps according to their popularity: Apps are classified according to their popularity. The initial distribution of the 221 875 Apps in July 2011 is given in Table 1.

(iv) Distribution of malware Apps according to their popularity: the malware Apps initially distributed by popularity (July 2011) can be seen in Table 2. Repackaging [2], that is, take a popular App, introduce some malware code, and upload it again to the market, is considered.

(v) Malware detection: the admitted effectiveness of the App scanning service of Android market is around 40% [15].

(vi) Distribution of malware Apps according to their type: the distribution of malware Apps in the official market according to [2] is shown in Table 3.

2.2. Nonofficial Market. Let us describe the markets other than Google Play [17].

(i) Considering that the data of the new Apps entering every month in the nonofficial market also show a linear trend, it can be modeled, in the mean square sense, by the function

$$f_{\mathrm{NOM}}(t) = 1.65 f_{\mathrm{OM}}(t), \quad t = 0,1,2,\ldots, \tag{4}$$

where t is the number of months since July 2011 which corresponds to $t = 0$. Details can be found in [15].

(ii) Similarly, the new malware Apps entering every month in the nonofficial market are given by the function

$$f_{\mathrm{NOM}m}(t) = 529.9 + 225.9t, \quad t = 0,1,2,\ldots, \tag{5}$$

describing the evolution of Apps in the nonofficial market, where t is the number of months since July 2011 which corresponds to $t = 0$.

(iii) Distribution of Apps according to their popularity: we classify the Apps depending on their popularity in the nonofficial market as it is shown in Table 4.

(iv) Distribution of malware Apps according to their popularity: we classify the Apps depending on their popularity in the nonofficial market as it is shown in Table 5.

(v) Malware detection: there is not antivirus service in the nonofficial market.

(vi) Distribution of malware Apps according to their type: the distribution of malware Apps, according to [2], in the nonofficial market by their type is shown in Table 6.

Table 1: Initial distribution of Apps by popularity in July 2011.

Popularity	None	2.5	2.5–3.0	3.0–3.5	3.5–4.0	4.0–4.5	>4.5
Number of Apps	114 789	5 985	6 053	13 512	21 599	31 493	28 444
%Apps	51.74%	2.70%	2.73%	6.09%	9.73%	14.19%	12.82%

Table 2: Initial distribution of malware Apps by popularity in July 2011 taking into account repackaging.

Popularity	None	2.5	2.5–3.0	3.0–3.5	3.5–4.0	4.0–4.5	>4.5
Number of Apps	9	0	0	1	2	39	35

Table 3: Distribution of malware Apps in the official market according to their type.

Type	%	Type	%
Financial charge	14.22%	Remote control	43.49%
Privilege escalation	14.22%	Information collection	28.07%

2.3. Users. Let us describe the main features regarding the users behavior.

(i) Number of users: in the community of Valencia there were 1 176 954 Android smartphones. We are going to consider this value constant over the time.

(ii) Smartphone renewal: a user changes his/her smartphone every 11.5 months, in average [18].

(iii) Users with antivirus installed in their devices: we assume that the number of users with antivirus installed in their devices is 33% [19]. However, this figure is under discussion; therefore, we will simulate taking values in $[0, 0.66]$, the maximum unbiased likely interval. Moreover, the admitted effectiveness of these antivirus software is between 20.2% and 79.6% [2].

(iv) Average number of Apps downloaded: every user downloads an average of 6.2 Apps per month [15].

(v) OS version evolution and infection by *privilege escalation* malware: the evolution of the OS versions installed on the smartphones is shown in Table 7.

Table 8 shows the percentage of devices that can be affected by the most common Android privilege escalation vulnerabilities [20].

(vi) App downloads by popularity: let us define c_1, c_2, c_3, and c_4 with $0 \leq c_1 < c_2 < c_3 < c_4$, the average number of downloads of Apps in the community of Valencia with less than 500 downloads, between 500 and 5000 downloads, between 5000 and 50000 downloads, and more than 50000 downloads all over the world, respectively. Denoting by d_1, d_2, d_3, and d_4 the probabilities a user in the community of Valencia downloads an App which number of downloads in all the world are less than 500, between 500 and 5000, between 5000 and 50000 or more than 50000 downloads, respectively, then

$$d_i = \frac{c_i}{C}, \quad i = 1, 2, 3, 4, \tag{6}$$

where $C = c_1 + c_2 + c_3 + c_4$ and c_i, $i = 1, 2, 3, 4$, should satisfy that

$$c_1 = 26.4885 - 0.439884c_2 - 0.24981c_3 - 0.0978913c_4, \tag{7}$$

where

$$0 \leq c_1 < c_2 < c_3 < c_4 < 270.59. \tag{8}$$

More details are in [15].

(vii) When can a user be infected by a malware App? A downloaded malware App infects the client if one of the following conditions are met.

(a) The downloaded malware App is of the type privilege escalation, the OS is vulnerable, and there is not any installed antivirus.

(b) The downloaded malware App is of the type remote control, financial charge, or information collection and there is not any installed antivirus.

(c) The downloaded malware App is of the type privilege escalation, the OS is vulnerable, and the installed antivirus does not detect the malware (the antivirus is not effective).

(d) The downloaded malware App is of the type remote control, financial charge, or information collection and the installed antivirus does not detect the malware (the antivirus is not effective).

(viii) Probability that a user detects that his/her smartphone is infected and fixes it: we also mentioned that the average replacement cycle of smartphones is 11.5 months [18]. Anyway, the user detects and repairs infections caused by financial charge malware when he/she receives the mobile bill.

3. The App-Model Evolution Rules

The users and the markets have their own rules that define the initialization point and the evolution for the agents sets. The evolution rules for the client agents simulate the behavior of the users, establishing how many Apps are downloaded monthly by a client, how the App selection method by the client based on the App's popularity is, if the downloaded App infects the device, and how long a user changes his/her device.

Table 4: Initial distribution of Apps by popularity in July 2011 in the nonofficial market. The percentages are the same as in the official market, Table 1.

Popularity	None	2.5	2.5–3.0	3.0–3.5	3.5–4.0	4.0–4.5	>4.5
Number of Apps	189 894	9 900	10 014	22 353	35 730	52 098	47 055
%Apps	51.74%	2.70%	2.73%	6.09%	9.73%	14.19%	12.82%

Table 5: Initial distribution of malware Apps by popularity in July 2011 in the nonofficial market. Repackaging is also considered.

Popularity	None	2.5	2.5–3.0	3.0–3.5	3.5–4.0	4.0–4.5	>4.5
Number of Apps	48	3	3	6	9	219	198

Table 6: Distribution of malware Apps in the nonofficial market according to their type.

Type	%	Type	%
Financial charge	50.10%	Remote control	10.06%
Privilege escalation	35.58%	Information collection	4.26%

The evolution rules for the App-markets establish the number of new Apps in every market each month, how the markets control the new submitted Apps (Google Play uses *Bouncer* which scans submitted Apps looking for malware), how the markets distributes the Apps by popularity, and so forth.

Then, using the considerations introduced so far, we are going to describe the evolution rules of the model. Recall that the time period is a month and the starting point of the model $t = 0$ corresponds to July 2011.

First, we sample percentages d_1, d_2, d_3, and d_4 as described in (6), (7), and (8). Then for every month t.

(i) State the official market:

(a) determine the number of Apps in this market in month t according to (2);

(b) distribute them according to their popularity following the percentage values in Table 1;

(c) determine the number of malware Apps in this market in month t according to (3);

(d) distribute them according to their popularity following the percentage values in Table 2;

(e) malware detection: 40% of malware is detected and removed.

(ii) State the nonofficial market:

(a) determine the number of Apps in this market in month t according to (4);

(b) distribute them according to their popularity following the percentage values in Table 4;

(c) determine the number of malware Apps in this market in month t according to (5);

(d) distribute them according their popularity following the percentage values in Table 5.

(iii) User behavior. For every user:

(a) Download method: take a random value u between 0 and 1 and obtain the maximum value of k such that $\sum_{j=1}^{k} f(j, \lambda) \leq u$ (see expression (1)).

(b) Selection method: select k Apps from each market with a probability of 50%, in such a way that their popularity is rated according to the probabilities d_1, d_2, d_3, and d_4, and malware or not with probability $f_{\mathrm{OM}m}(t)/f_{\mathrm{OM}}(t)$ for the official market and $f_{\mathrm{NOM}m}(t)/f_{\mathrm{NOM}}(t)$ for the nonofficial market.

(c) Infection method: if any of the downloaded Apps are malware, for each malware App, one has the following.

(1) If it has been downloaded from the official market, determine its type with probabilities given in Table 3. Then, it infects the smartphone depending on the OS installed (Table 7) if there is antivirus and its effectiveness.

(2) If it has been downloaded from the nonofficial market, determine its type with probabilities given in Table 6. Then, it infects the smartphone depending on the OS installed (Table 7) if there is antivirus and its effectiveness.

(d) Check if the user detects if the smartphone is infected and fix it only in case the malware is financial charge and the repair is done at the end of the month.

(e) Check if the user changes his/her smartphone (every 11.5 months in average).

The algorithmic evolution of the App-Model described above is drawn as the flowchart shown in Figure 2. The left side of the figure represents the evolution of the clients and the right side the evolution of the Apps that evolve in parallel. The start point represents the initial month of the model ($t = 0$), where the model creates the clients and sets their attributes. After this, and for every step ($t = i$), the model begins its evolution and all of the clients (left side of the figure) run their methods in the showed order and change, if needed, their attributes. Also, for every step ($t = i$),

Table 7: Distribution of OS versions in Android smartphones from July 2011 until February 2013 [11].

Version	Affected	July 2011	October 2011	February 2012	June 2012	October 2012	February 2013
1.5	Cupcake	1.40%	0.90%	0.40%	0.20%	0.10%	0.00%
1.6	Donut	2.20%	1.40%	0.80%	0.50%	0.30%	0.20%
2.1	Eclair	17.50%	10.70%	6.60%	4.70%	3.10%	1.90%
2.2	Froyo	59.40%	40.70%	25.30%	17.30%	12.00%	7.50%
2.3	Gingerbread	18.60%	44.40%	62.00%	64.00%	54.20%	44.10%
3.0	Honeycomb	0.90%	1.90%	3.30%	2.40%	1.80%	1.20%
4.0	Ice-cream	0.00%	0.00%	1.60%	10.90%	25.80%	28.60%
4.1	Jelly	0.00%	0.00%	0.00%	0.00%	2.70%	16.50%

Table 8: Percentage of devices that can be affected by the most common privilege escalation vulnerabilities, depending on the Android OS version.

Version	Name	Affected	Version	Name	Affected
1.5	Cupcake	100%	2.3	Gingerbread	100%
1.6	Donut	100%	3.0	Honeycomb	0.00%
2.1	Eclair	96.70%	4.0	Ice-cream	31.00%
2.2	Froyo	98.80%	4.1	Jelly	0.00%

the model establishes the markets that are changing every month, sets the Apps attributes, and groups them depending on the number of downloads (right side of the figure). After this, and for every step, the number of Apps of the markets is recalculated according their evolution curve. All this process runs in parallel, but on every step the selection method of the clients can be executed only after the Apps are grouped. After the last step of evolution of the model ($t = T$), the end point of the simulation is reached.

4. Results and Discussion

Once the model has been built and the evolution rules are stated, there are some model parameters unknown but satisfying some restrictions:

(i) Apps download percentages per popularity d_1, d_2, d_3, d_4, and d_5, satisfying (6), (7) and (8),

(ii) the percentage of smartphones with antivirus, denoted by A, is in $[0, 0.66]$ [19], and

(iii) the effectiveness of the antivirus protection, denoted by E, is in $[0.202, 0.796]$ [2].

Now, in first place, we are going to see if the model output depends on the number of smartphone users. If it is, we will have to simulate the behavior of 1176 954 users. Otherwise, we will be able to reduce the number of users in order to run the simulation very much quicker.

Secondly, we will simulate a large amount of runs in order to estimate the number of the monthly infections by malware Apps.

4.1. Model Evolution Depending on the Number of Users. In this first experiment, we take fixed values of d_1, d_2, d_3, d_4, d_5, A, and E and we run simulations for 1000, 5000, 7000, 10000, 15000, 20000, 30000, 40000, 50000, 65000, 80000, 100000, 120000, and 150000 users during $t = 1, \ldots, 15$ months. Then, in Table 9 we can see the comparison of percentage of cumulative (aggregated) and residual (new ones) infected users for month $t = 15$. Few differences can be noted. Therefore, we do not need to simulate the 1176 954 Android smartphones in the community of Valencia to obtain reliable and accurate results. After some tests, we decided to consider 50000 users.

4.2. Estimations. Thus, in order to compute reliable estimations based on 95% confidence intervals (CI 95%), we use the technique called latin hypercube sampling (LHS) [21] to select sets of parameters to be substituted into the model. Latin hypercube sampling (a type of stratified Monte Carlo sampling) is an efficient method for achieving equitable samples of all input parameters simultaneously. Moreover, the random selection of the sets of parameters done by LHS will allow us to study the model sensitivity by the CI 95%.

In our case, taking 50000 smartphone users, starting in July 2011 and finishing in December 2014 ($t = 0, 1, 2, \ldots, 41$ months), and following the evolution rules, LHS was used to generate 100000 different values of each input parameter d_1, d_2, d_3, d_4, d_5, A, and E sampled as follows.

(1) Sample values $0 \leq c_1 < c_2 < c_3 < c_4 < 270.59$ such that $c_1 = 26.4885 - 0.439884c_2 - 0.24981c_3 - 0.0978913c_4$, and calculate $d_i = c_i/C$, $i = 1, 2, 3, 4$.

(2) Sample a value of A uniformly in the interval $[0, 0.66]$.

(3) Sample a value of E uniformly in the interval $[0.202, 0.796]$.

We used these samples to run 100000 evaluations of the model obtaining 100000 model outputs (infected smartphones) for each month $t = 0, 1, 2, \ldots, 41$. Then, for each month we take the 100000 model outputs and calculate

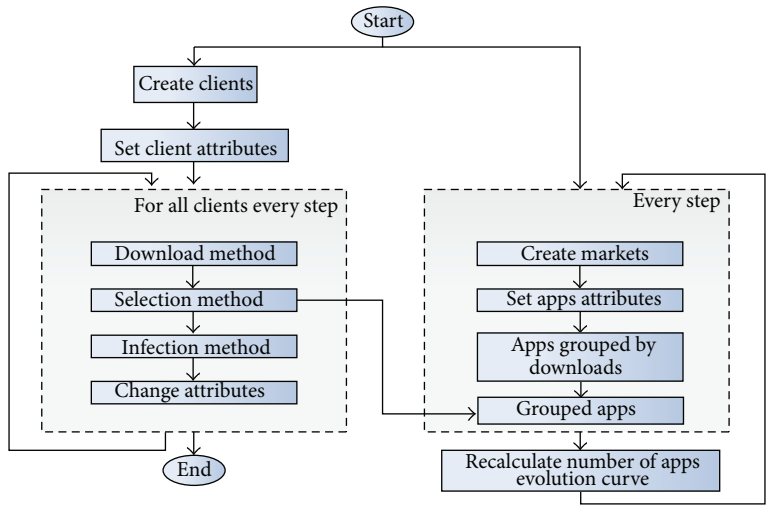

Figure 2: App-Model flowchart. In this figure we describe the evolution process of the model from $t = 0$ (start point) to $t = T$ (end point), showing, for every time instant, the creation of the agents, the assignation of attributes, the order of performance of the methods, and their interaction.

Table 9: Comparison of percentage of cumulative and residual infected users for month $t = 15$. The results are very similar.

Number of users	% of accumulated infected	% of residual infected
1000	5.10%	0.70%
5000	5.52%	0.68%
7000	5.29%	0.59%
10000	5.23%	0.63%
15000	4.77%	0.53%
20000	4.95%	0.48%
30000	4.88%	0.51%
40000	5.19%	0.52%
50000	5.13%	0.55%
65000	5.12%	0.53%
80000	5.06%	0.55%
100000	5.15%	0.56%
120000	4.89%	0.56%
150000	5.01%	0.55%

the mean and the 95% confidence intervals taking into account the empirical 2.5% and 97.5% percentiles.

In Figure 3 we can see the evolution of the cumulative infections since July 2011 until December 2014 with a 95% confidence interval. In Table 10 we can see the numerical values of the mean and CI 95% of the cumulative infections in the community of Valencia in July 2013, July 2014, and December 2014.

In Figure 4 we can see the evolution of the new (residual) infected smartphones every month with a 95% confidence interval since July 2011 until December 2014. It can be seen that since October 2012, there is a certain stabilization in the number of new infected smartphones. In Table 11 we can see the numerical values of the mean and CI 95% of the residual infections in the community of Valencia in July 2013, July 2014, and December 2014.

Table 10: Mean and CI 95% of the accumulated infected smartphones in July 2013, July 2014, and December 2014 in the community of Valencia and the corresponding percentages predicted by the model.

	Mean	CI 95%
July	86163	[57388, 110540]
2013	7.32%	[4.88%, 9.39%]
July	139623	[93191, 178450]
2014	11.86%	[7.92%, 15.16%]
December	162788	[108680, 207756]
2014	13.83%	[9.23%, 17.65%]

Table 11: Mean and CI 95% of the residual infected smartphones in July 2013, July 2014, and December 2014 in the community of Valencia and the corresponding percentages predicted by the model.

	Mean	CI 95%
July	3818	[2448, 5108]
2013	0.32%	[0.21%, 0.43%]
July	4037	[2613, 5367]
2014	0.34%	[0.22%, 0.46%]
December	4105	[2660, 5461]
2014	0.35%	[0.23%, 0.46%]

Finally, in Figure 5, we show the mean and the 95% confidence interval of cumulative infected smartphones by privilege escalation (PE) and financial charge (FC) malware. Comparing Figure 5 to Figure 3 we can see that financial charge malware infects a half of the smartphones according to [2, 22]. In Table 12 we can see the numerical values of the mean and CI 95% of the cumulative infections in the community of Valencia in July 2013, July 2014, and December 2014.

Table 12: Mean and CI 95% of the accumulated infected smartphones by privilege escalation and financial charge in July 2013, July 2014, and December 2014 in the community of Valencia and the corresponding percentages predicted by the model. These figures can give us an idea about the amount of money that the financial charge malware moves every month.

	Privilege escalation		Financial charge	
	Mean	CI 95%	Mean	CI 95%
July	24857	[16360, 32743]	39824	[26340, 51857]
2013	2.11%	[1.39%, 2.78%]	3.38%	[2.24%, 4.41%]
July	38481	[25493, 50185]	66861	[44418, 86341]
2014	3.27%	[2.17%, 4.26%]	5.68%	[3.77%, 7.34%]
December	44414	[29447, 57718]	78643	[52304, 101289]
2014	3.77%	[2.50%, 4.90%]	6.68%	[4.44%, 8.61%]

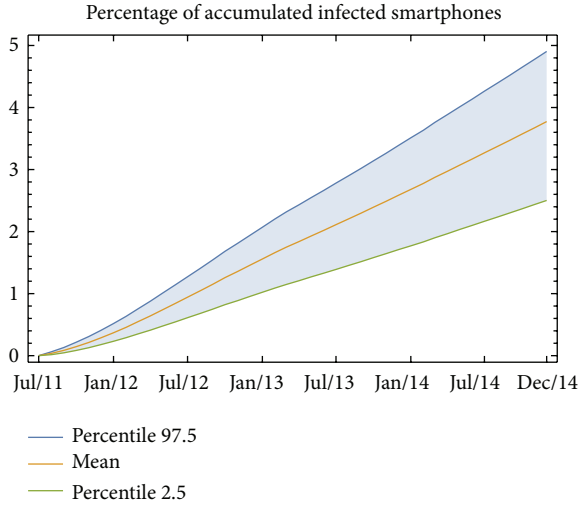

Figure 3: Model evolution of the cumulative smartphone infections every month since July 2011 until December 2014. The line in the middle is the mean and those up and down correspond to 95% confidence interval.

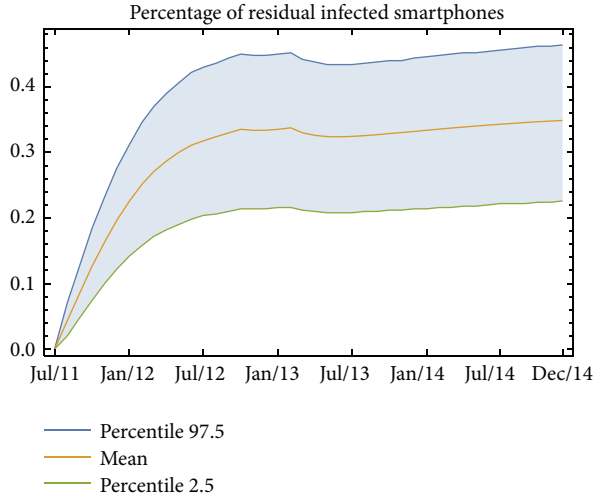

Figure 4: Evolution of new smartphone infections every month since July 2011 until December 2014. The line in the middle is the mean and those up and down correspond to 95% confidence interval. Nowadays, there is a stabilization in the number of new infected smartphones.

4.3. Model Validation. In [13], Patterson talks about Google's Android security chief Adrian Ludwig who gave a talk at the Virus Bulletin conference in Berlin. In this talk, Ludwig said that the problem Google wants to solve is that most of independent security researchers do not have access to a platform such as Google to measure how many times a malware App has been installed. Also, he mentioned that security researchers are very good at finding and fixing malware, but in the absence of reliable data that indicate how frequently a malware App has been installed the threat level can become exaggerated. Reports that reach publication are often extremely exaggerated. To emphasize this point, Ludwig revealed in his analysis that some of the most publicized recent malware discoveries are installed in less than one per million installations. Additionally, he reported that based on the data from tracking over one and a half billion App installs Google obtained convincing evidence that the rate of *potentially harmful Apps* installed is stable at about 1 200 per million App installs or about 0.12%.

Furthermore, the official reports as F-Secure Report (mobile threat report September 2013), Trend Micro Report (Trend Labs Security report 3Q 2013), or Secure List Report (mobile malware evolution February 2014) do not only show the number of devices affected by installed malware Apps but also show the number of Apps detected as malware.

As a consequence, comparing the figures given by the proposed model to the real ones is not going to be an easy task because of lack of real data. In fact, to our knowledge, the only data about potentially harmful Apps installed are the one mentioned above: stable and about 0.12%.

Then, taking into account that the conference was held on October 3rd, 2013 [13], we may compare this data with prediction of the model for new smartphone infections in September 2013: stable and mean 0.33% with CI 95% [0.21%,0.44%].

Hence, our model predicts a stable situation of harmful Apps installed, as Google says, and a little bit higher number of infected smartphones than Google. This slight difference may be due to the development of the techniques for detecting malware during the period of time considered in our simulation, resulting in increased effectiveness of antivirus

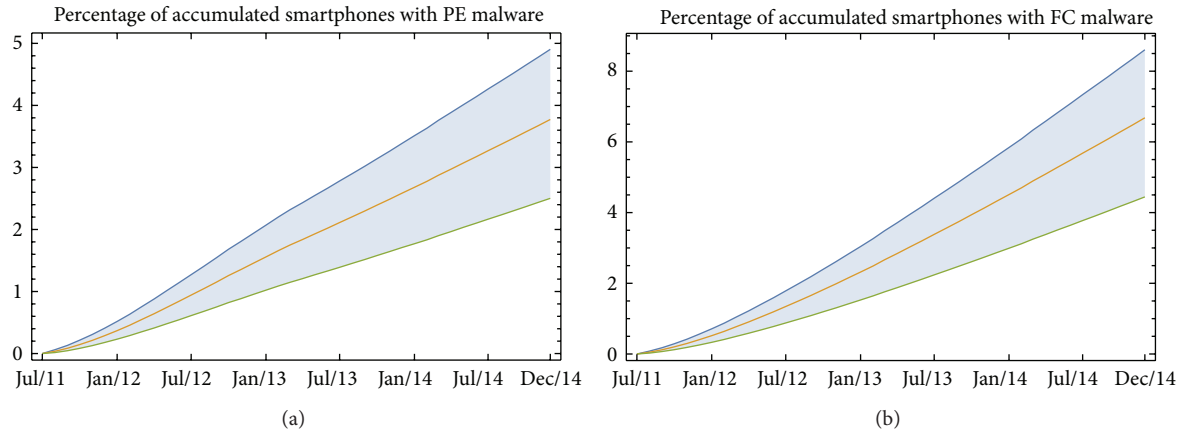

Figure 5: Evolution of the cumulative smartphone infections due to privilege escalation (PE) on the left and financial charge (FC) on the right, malware every month since July 2011 until December 2014. The line in the middle is the mean and those up and down correspond to the 95% confidence interval.

software than that used in the initial parameters of our simulation in terms of the effectiveness of antivirus software and therefore reducing the number of malware installed in the Google analysis. Taking into account this regard, we consider that our model provides valid results in terms of estimation of number of infected smartphones and in terms of stable evolution of the infections.

5. Conclusion

In this paper we present an agent-based model to quantify the Android malware infection evolution. Some model outputs are compared to data given by Google and the results are fairly similar, stable, and a little bit higher for the model predictions.

Considering the parameters of our model and our simulations, the obtained results show that, given a specific population of devices with Android OS, one has the following.

(i) A mean of 0.3% of devices are infected every month by some kind of malware. This number is stable over time from October 2012 onwards, considering the growing curve for the total Apps and malware Apps.

(ii) Taking into account cumulative values from July 2011 to December 2014, we predict that the infections will be around a mean of 13.83% over the total number of devices considered.

(iii) From this 13.83%, around the half of the total (48%) will be infections by financial charge malware type, and around a third (27%) will be infections by privilege escalation malware type. The remainder (25%) will be infections by remote control and information collection malware type.

(iv) Thus, the infections by financial charge, remote control, and information collection malware type are due to the users because they give indiscriminate permissions to the Apps and do not protect properly their mobile with antivirus software. Therefore, we show that two-thirds of the infections are caused by these two factors, showing that the most critical part for the malware infections at smartphones is the users habits and the ineffectiveness of the traditional antivirus software, not due to the OS vulnerabilities.

(v) Quantifying and monetizing the financial charge malware incidence: we can consider that, from the 0.3% new infected devices during a month, the half part is infected by financial charge and that every infection causes a monthly overrun of 30 euros (we have some examples of mobile bills such that their owners suffered an infection of financial charge malware and the amount of these bills are around 30 euros) in every device. Considering that the total population of Android devices in Spain is 10 853 813, the number of infected devices by financial charge malware type during a month is 16 280 (i.e. 0.15%) and the financial charge caused by this kind of malware during a month will be 488 400 euros.

With our model, we show realistic data that can be considered in order to quantify the real threat for the users and the number of potential infected smartphones. With these results, we consider that preventive strategies against mobile malware should be developed mainly focusing on new malware detection approaches before being downloaded by the users, because, as we shown, the users decisions and the ineffectiveness of the traditional antivirus software approach are the critical part for the infections.

Moreover, with the presented model, despite the increasing of Apps, we could see that the number of new infected smartphones achieved stable figures, and then it is not expected a significant change in the current stable trend.

One of the most interesting features of the presented model is that if some of the parameters vary because of changes in the behavior of the actors (markets, Apps, and clients) we only have to tune the corresponding model parameters and perform the simulations to predict the evolution of the infected smartphones for the new scenario.

Finally, we want to point out that this model and simulations can be extrapolated to other regions where the number of Android smartphones is fairly stable over the time.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this article.

Acknowledgment

This work has been partially supported by the Ministerio de Economía y Competitividad Grant MTM2013-41765-P.

References

[1] http://www.malgenomeproject.org/.

[2] Y. Zhou and X. Jiang, "Dissecting android malware: characterization and evolution," in *Proceedings of the 33rd IEEE Symposium on Security and Privacy*, pp. 95–109, San Francisco, Calif, USA, May 2012.

[3] "Gartner Smart Phone Marketshare 2013 Q2," http://www.gartner.com/newsroom/id/2573415.

[4] F. di Cerbo, A. Girardello, F. Michahelles, and S. Voronkova, "Detection of malicious applications on android OS," in *Computational Forensics*, S. Sako, K. Franke, and S. Saitoh, Eds., vol. 6540 of *Lecture Notes in Computer Science*, pp. 138–149, Springer, Berlin, Germany, 2011.

[5] T. Isohara, K. Takemori, and A. Kubota, "Kernel-based behavior analysis for android malware detection," in *Proceedings of the 7th International Conference on Computational Intelligence and Security (CIS '11)*, pp. 1011–1015, Hainan, China, December 2011.

[6] A. Shabtai, U. Kanonov, Y. Elovici, C. Glezer, and Y. Weiss, ""Andromaly": a behavioral malware detection framework for android devices," *Journal of Intelligent Information Systems*, vol. 38, no. 1, pp. 161–190, 2012.

[7] A. Bose and K. G. Shin, "Agent-based modeling of malware dynamics in heterogeneous environments," *Security and Communication Networks*, vol. 6, no. 12, pp. 1576–1589, 2013.

[8] J. W. Mickens and B. D. Noble, "Modeling epidemic spreading in mobile environments," in *Proceedings of the 4th ACM workshop on Wireless security (WiSe '05)*, pp. 77–86, Cologne, Germany, 2005.

[9] K. Ramachandran and B. Sikdar, "Modeling malware propagation in networks of smart cell phones with spatial dynamics," in *Proceedings of the 26th IEEE International Conference on Computer Communications (INFOCOM '07)*, pp. 2516–2520, Anchorage, Alaska, USA, May 2007.

[10] P. Wang, M. C. González, C. A. Hidalgo, and A.-L. Barabasi, "Understanding the spreading patterns of mobile phone viruses," *Science*, vol. 324, no. 5930, pp. 1071–1076, 2009.

[11] "Android (operating system)," http://en.wikipedia.org/wiki/Android_(operating_system).

[12] C. Fleizach, M. Liljenstam, P. Johansson, G. M. Voelker, and A. Méhes, "Can you infect me now?: Malware propagation in mobile phone networks," in *Proceedings of the ACM Workshop on Recurring Malcode (WORM '07)*, pp. 61–68, Alexandria, Va, USA, November 2007.

[13] S. M. Patterson, Contrary to what you've heard, Android is almost impenetrable to malware, 2013, http://qz.com/131436/contrary-to-what-youve-heard-android-is-almost-impenetrable-to-malware/.

[14] http://en.wikipedia.org/wiki/Valencian_Community.

[15] J. Alegre, J.-C. Cortés, F.-J. Santonja, and R.-J. Villanueva, "Quantifying the behaviour of the actors in the spread of Android malware infection," in *Mathematical Modeling in Social Sciences and Engineering*, Chapter 10, pp. 101–112, Nova Science Publishers, New York, NY, USA, 2013.

[16] https://play.google.com/store.

[17] http://www.getjar.com/.

[18] T. T. Ahonen and A. Moore, *Smartphone Penetration Rates by Country! We Have Good Data (Finally)*, 2011, http://communities-dominate.blogs.com/brands/2011/12/smartphone-penetration-rates-by-country-we-have-good-data-finally.html.

[19] A. Mylonas, A. Kastania, and D. Gritzalis, "Delegate the smartphone user? Security awareness in smartphone platforms," *Computers and Security*, vol. 34, pp. 47–66, 2013.

[20] *Webinar: X-Ray Results- Mobile Device Vulnerabilities*, Duo Security, 2012, http://www.duosecurity.com/.

[21] A. Hoare, D. G. Regan, and D. P. Wilson, "Sampling and sensitivity analyses tools (SaSAT) for computational modelling," *Theoretical Biology and Medical Modelling*, vol. 5, article 4, 2008.

[22] Department of Homeland Security and Federal Bureau of Investigation, *(U//FOUO) DHS-FBI Bulletin: Threats to Mobile Devices Using the Android Operating System*, 2013, http://publicintelligence.net/dhs-fbi-android-threats/.