Available online at www.sciencedirect.com

SciVerse ScienceDirect

journal homepage: www.elsevier.com/locate/cose

Computers & Security

Delegate the smartphone user? Security awareness in smartphone platforms

*Alexios Mylonas, Anastasia Kastania, Dimitris Gritzalis**

Information Security and Critical Infrastructure Protection Research Laboratory, Department of Informatics, Athens University of Economics and Business (AUEB), 76 Patission Ave., Athens GR-10434, Greece

ARTICLE INFO

Article history:
Received 14 April 2012
Received in revised form
4 October 2012
Accepted 20 November 2012

Keywords:
Application markets
Security awareness
Security survey
Smartphone platforms
Smartphone security

ABSTRACT

Smartphone users increasingly download and install third-party applications from official application repositories. Attackers may use this centralized application delivery architecture as a security and privacy attack vector. This risk increases since application vetting mechanisms are often not in place and the user is delegated to authorize which functionality and protected resources are accessible by third-party applications. In this paper, we mount a survey to explore the security awareness of smartphone users who download applications from official application repositories (e.g. Google Play, Apple's App Store, etc.). The survey findings suggest a security complacency, as the majority of users trust the app repository, security controls are not enabled or not added, and users disregard security during application selection and installation. As a response to this security complacency we built a prediction model to identify users who trust the app repository. The model is assessed, evaluated and proved to be statistically significant and efficient.

1. Introduction

Unlike PC software, smartphone applications (or 'apps') adopt centralized distribution architectures and are usually available to users from app repositories or app marketplaces. These app repositories may either be official (i.e. maintained by the smartphone platform, e.g. Apple's App Store, Microsoft's App Hub), or not (e.g. Amazon Appstore for Android). The security models of smartphone platforms provide different options with respect to the permitted source of applications (Barrera and Van Oorschot, 2011; Mylonas et al., 2011a). In addition, the strictness of app vetting controls in an app repository[1] ranges from relaxed app submission in community-based app repositories, such as Google play, to strictly controlled repositories that follow the 'walled garden' model, such as Apple's App Store (Barrera and Van Oorschot, 2011; Mylonas et al., 2011a, 2011b). Regardless of how strict and centralized the security model of a platform may be, it always leaves some choice to the user. Again, this delegation can be simply authorizing access to some protected resources, or may give user the choice to infer if an application may impair her security and privacy.

Meanwhile, the rate of downloads for smartphone applications from app repositories is on the rise (Baghdassarian and Milanesi, 2010). This popularity of smartphone applications has drawn the attention of attackers, who try to use the app repository as a security and privacy attack vector. In this context, an increasing number of malicious applications have already been discovered in app repositories (Felt et al., 2011b; Zhou et al., 2012b).

This is one of the reasons that smartphones have also drawn the security literature's attention. The security literature that focuses on smartphone applications has elaborated on malicious application identification (Egele et al., 2011; Enck

* *Corresponding author*. Tel.: +30 2108203505; fax: +30 2105810116.
E-mail addresses: amylonas@aueb.gr (A. Mylonas), ank@aueb.gr (A. Kastania), dgrit@aueb.gr (D. Gritzalis).

[1] Unless stated otherwise, in the rest of the paper the term "app repository" refers to an official app repository.

0167-4048/$ – see front matter © 2012 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.cose.2012.11.004

et al., 2009,2010, 2011; Nauman et al., 2010; Zhou et al., 2012a, 2012b; Zhou and Jiang, 2012). Automated scanners have been proposed to aid advanced users deduce whether an application requests permissions that can impair the security and\or privacy of users (Enck et al., 2010; Felt et al., 2011a; Hornyack et al., 2011). Nonetheless, it is unclear whether the burden of making security decisions is reasonable for normal users. Studies have shown that normal users are not able to make such decisions, nor are able to use security controls adequately (Furnell, 2005, 2007; Furnell et al., 2006; Whitten and Tygar, 1999).

In this paper, we examine the security awareness of smartphone users who install applications from official app repositories. We conducted a survey with the aim to answer the following main research questions:

Q1: Do smartphone users enable security controls on their devices?
Q2: Do users consider security while choosing and downloading applications?
Q3: Do smartphone users trust applications from the official app repository?

The survey scope includes only users who download applications from the official app repositories of the current popular smartphone platforms, i.e. Android, BlackBerry, iOS, Symbian and Windows Phone. Our survey results indicate a clear lack of smartphone users' security awareness. Contrarily to Q1 and Q2, the security unaware users of Q3 cannot be identified with the use of software, e.g. smartphone agents, Mobile Device Management (MDM) (Redman et al., 2011), etc. For this reason, we propose and evaluate the effectiveness of a prediction model that identifies users who trust applications from the app repository.

The rest of the paper is organized as follows. The next section presents related work. Section 3 provides the reader with the methodology of the survey. In Sections 4 and 5 the findings from the summary of the sample responses and the essential statistical analysis are presented, respectively. In Section 6 the prediction model is described and its effectiveness is evaluated. Finally, Section 7 includes the survey's limitations, whereas Section 8 includes a discussion of the results and conclusions.

2. Related work

Even thought smartphones are well studied in security literature, the relevant research work on the security awareness of smartphone users is currently rather limited and mainly focuses on Android. Chia et al. (2012) studied risk signalling concerning the privacy intrusiveness of Android applications in two application repositories, i.e. Android market[2] and AppBrain.com. Their results suggest that the number of dangerous permissions that an application requests is positively correlated with its popularity. Even though users understand the notion of application popularity, the fact that an application is popular does not imply that it respects the users' privacy. Moreover, their results indicate that the current risk signals employed by an app repository (e.g. developer's website, application reputation) become ineffective over time, as users tend to click through them. Our findings also indicate that users tend to ignore the reputation and the reviews of an application, as well as the security and agreement messages revealed during application installation from app repositories.

Similarly to our user survey, smartphone users were found to ignore security messages during application installation in (Felt et al., 2012; Kelley et al., 2012). Moreover, they were unable to comprehend the permissions and the risks that are related with them (Felt et al., 2012; Kelley et al., 2012). As a result, in both studies the Android security messages did not assist most users to make appropriate security decisions. Our results suggest that the majority of respondents ignore every aspect of security and privacy during application selection, as well as the app's reputation, reviews and security and agreement messages. Nonetheless, when explicitly asked, a minority of users in the survey conducted by Felt et al. (2012) reported that they have cancelled the installation of an application due to its permission requests. In our survey a minority of users was found to delve into security and agreement messages; they tend to be security and technology savvy.

Finally, in the user study conducted in (Kelley et al., 2012) users erroneously believed that applications undergo security analysis during their submission in the Android Market. In our study we also found such misconceptions about application testing in application markets. Moreover, most users were unaware of the existence of the application testing mechanism.

3. Methodology

To assess the security awareness of smartphone users, a survey was conducted from September to December 2011, in Athens, Greece. This section presents the survey methodology, as well as some details about the tests mounted so as to ensure the validity and statistical significance of the results.

3.1. *Data collection*

The survey responses were collected from random people on the street and from public transport means (train stations, underground, airports), via structured interviews (Flick, 1998). A questionnaire (see Appendix B) was used for the structured interviews. The duration of the interview completion was 5–8 min, on average. The discussion with the user aimed to ensure the validity of her responses, the comprehension of the questions, and the comprehension of technical terms.

Questions 5 and 7 were used to filter out the non-smartphone users and the smartphone users who did not install third-party applications in their devices. We excluded these two user groups because the survey focused on smartphone users who do download applications from app repositories. Questions 2 and 8 were the only free text questions.

[2] Android Market is the smartphone app repository Google maintained before merging and rebranding it with other digital content services in Google Play (http://googleblog.blogspot.com/2012/03/introducing-google-play-all-your.html, March 2012).

3.2. Sample demographics and background

Reports on Internet penetration in the EU member states (Communications Committee, 2011) show that Greece had one of the highest increases in fixed Internet connection penetration in 2011. On the other hand, mobile Internet penetration was only 27.1%, considerably lower than the EU average (34.6%). However, in January 2011[3] the subscriptions for access to mobile Internet via cell phones increased by 140%, comparing to the first half of 2010. This indicates that smartphone adoption in Greece is on a significant rise.

The sample's population includes 458 smartphone users, aged ($min_{age} = 15$, $max_{age} = 52$). Among them, 81% were aged [15–30] (Appendix D contains the sample's age distribution) and 70.1% were male. The 56.3% reported that they were non-security savvy users (Q4). Regarding their IT skills, the sample reported that: (a). 10.3% had moderate IT skills, (b). 41.3% had good IT skills, and (c). 48.5% had excellent IT skills.

Amongst the smartphone users, the 61.6% of the sample were aware of the term *jailbreaking* and 81.4% were aware of smartphone malware existence. Also, 95.2% were concerned about their privacy, 75.8% stored personal data in their devices, and 35.8% stored business data. Finally, 30.1% of them reported that they had misplaced their device, at least once in the past.

The popularity of smartphone platforms in the sample is depicted in Table 1. The second column presents the proportion of users in each platform, along with the popularity rank amongst the platforms. The platform popularity that was discovered in the Greek sample is comparable to a recent Gartner report[4] concerning global smartphone popularity. Android, iOS and Symbian were the dominant smartphone platforms in both surveys, while in our sample Windows Phone had more popularity than Blackberry.

3.3. Sample considerations

To ensure the quality of the statistical analysis, the survey sample must satisfy the simple and randomness, reliability, and representativeness properties. This section summarizes the statistical tests conducted to ensure that these properties are fulfilled by the sample.

The collection methodology of the sample was *simple* and *random*, as the selection of a smartphone user u_i was not dependent on the selection of user u_{i-1}. We approached random users and asked whether they owned smartphones and whether they wished to participate in a smartphone survey about smartphone app usage. Then, we asked if they knew what an app repository (or app market) is, and if they had installed from it any third-party application. A printed questionnaire was given to the smartphone user only if she had installed at least one application in her device.

Sample *reliability* deals with the accuracy of measurement in a statistical analysis (Cronbach, 1951). Reliability analysis ensures that if the same measurement instrument (i.e. the questionnaire) is given to a different sample, then the results of the statistical analysis must be similar. In this context, the survey results may change only if the sample expresses a different opinion and not because of any confusion or misinterpretation. Hence, to ensure the reliability of the sample, we successfully tested it against the Cronbach alpha algorithm ($\alpha = 0.506$).

Table 1 – Platform popularity.

Operating system	Sample popularity	Gartner popularity
Android	38.4% (1)	50.9% (1)
BlackBerry	9.2% (5)	8.8% (4)
iOS	23.8% (2)	23.8% (2)
Symbian	16.6% (3)	11.7% (3)
Windows	12.0% (4)	1.9% (5)
Other	0% (6)	2.9% (6)

We tested the *representativeness* of our sample by performing x^2 goodness-of-fit tests. We tested the null hypothesis, i.e. whether the observed frequencies in our sample differ from the expected ones by assuming equal probability for every question. We rejected the null hypothesis for every response that we collected from our sample, hence the sample is representative for populations with similar ages [15–52].

Finally, we also a priori estimated the required sample size (Cohen, 1988), so as to conduct our statistical analysis tests with the x^2 distribution, with power level 95% and significance level 5%. As such, we estimated that the required sample size for the statistical analysis is 145 cases. Details regarding these tests are given in Appendix C.

4. Results from descriptive statistics

This section presents the findings that were obtained by descriptive statistics. The percentages refer to proportions of the sample population or to smartphone platforms users. In the latter, they refer to the proportions of a particular smartphone platform.[5] Contrarily to inferential statistics, descriptive statistics provide response summaries and, as a result, are bound to the Greek sample. Nonetheless, the survey summaries indicate the smartphone users' lack of security awareness. A discussion of the security findings discovered from descriptive statistics is included in the following sections.

4.1. User's trust in the app repository

A commonly adopted definition for *trust* is hardly available in the literature. In this work we adopt the definition of McKnight and Chervany (1996), where trust is "*the extent to which one party is willing to depend on somebody or something in a given situation with a feeling of relative security, even though negative consequences are possible*". In the context of smartphone application repositories, the negative consequences occur when the app repository is used as a privacy or security

[3] Observatory of the Greek Information Society (http://www.observatory.gr/).

[4] Gartner 4Q11 Market Share (%) (http://www.gartner.com/it/page.jsp?id=1924314).

[5] In this case, they are given to summarize the responses of smartphone users and not to compare the frequency of the answers among platforms, as the number of different platform users differs.

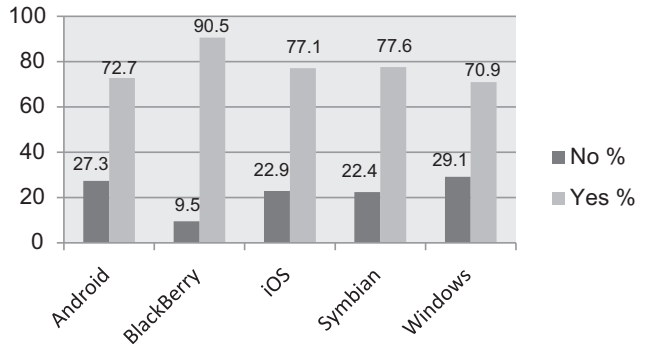

Fig. 1 – Trust in app repository among smartphone platforms.

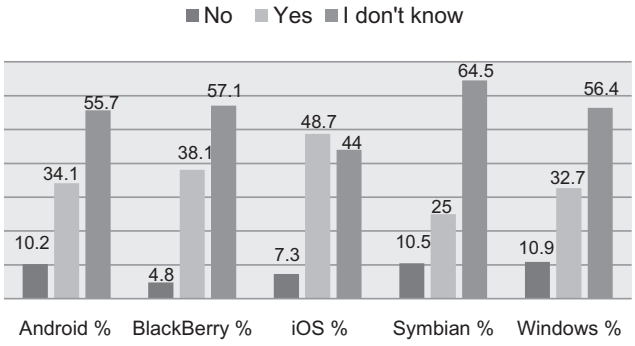

Fig. 2 – Application testing among smartphone platforms.

attack vector, i.e. malicious applications are submitted in the repository that impair the security and/or privacy of the users who download them. Some researchers consider that the concept of trust in McKnight's and Chervany's work refers to business trust, only, i.e. the social context of trust that a computer system performed the functions it was supposed to perform, and not on information security, which was not a major concern at that time. Some others believe that this definition is also applicable to information security.

Our results revealed that 76% of the survey participants believed that applications downloaded from the application repository are secure.[6] This is not in line with literature findings on smartphone security and malicious software; namely:

1. Security controls that application repositories often use, such as security analysis of apps (hereinafter 'application testing'), application 'kill switch' (i.e. remote application removal mechanism), etc., are not enabled in all official application repositories (i.e. Google Play, Blackberry App World, OVI store, Apple's App Store and Windows Marketplace) (Mylonas et al., 2011b).
2. Application testing has been proven inadequate to filter malicious applications by the smartphone security literature (Anderson et al., 2010; Egele et al., 2011; Enck et al., 2009, 2010, 2011; Hornyack et al., 2011; McDaniel and Enck, 2010; Nadji et al., 2011; Quirolgico et al., 2011), as well as by Cohen's analysis on malicious software detection (1989).
3. Application testing fails to satisfy the heterogeneity in each user's privacy requirements (McDaniel and Enck, 2010), suffers from time restrictions and the increasing rate of application submission in the repository (Anderson et al., 2010; McDaniel and Enck, 2010), and often requires manual analysis, which exposes the security testers to social engineering attacks (Felt et al., 2011b; Niemela).

Fig. 1 shows that approximately 3 out of 4 respondents in each smartphone platform do trust the app repository. BlackBerry users topped, in terms of trust in the app repository (90.5%), even though this app repository appears to have the least security controls in place (Mylonas et al., 2011b).

Finally, we note that this feeling of trust can be one of the available enablers for users that do not download apps in their devices (e.g. the users in Androulidakis and Kandus (2011b, 2012)) to start doing so.

4.2. *Misconception about application testing*

Another interesting finding is that users are unaware of the presence or absence of the mechanism that official app repositories use to test apps against malicious behaviour. More specifically, 54.6% of the sample's population was not aware whether any application testing takes place in the app repository. This finding indicates that users regard the app repository as trusted, regardless of the fact that they are not sure whether app testing takes place.

Application testing has gained considerable attention in the smartphone security literature. Application testing may be performed by using the resources of the application repository, e.g. as in Quirolgico et al. (2011); Gilbert et al. (2011); Zhou et al. (2012a, 2012b), or be crowdsourced as in Amini et al. (2012). In addition, application testing can be amended with a submission policy that aims to deter submission of malicious applications, by imposing momentary penalties to mischievous users (Mylonas et al., 2011b).

In practice, application testing does not take place during application submission in the Google Play[7] and in the BlackBerry App World (Mylonas et al., 2011b). This means that the two app repositories may be used more easily as an attack vector, in order to deliver malware to smartphone users. In the analysis presented in Fig. 2, the misconception of smartphone users with respect to application testing in the repository is augmented, since (a) some users (7–10%) faulty reported that application testing does not take place in the iOS, Symbian and Windows Phone platform, and (b) relatively more users (34–38%) faulty reported that application testing does take place in the Android and BlackBerry platform. From these two misconceptions, the latter is the most security critical, as users appear to consider that applications have been security tested.

As the distribution model of smartphone applications is centralized, this may pose an opportunity for malware mitigation. If the application repository employs an app testing mechanism, this mechanism can prevent malware spreading

[6] For brevity reasons, in the rest of the paper we refer to this as 'trust in the app repository'.

[7] Our survey took place before Bouncer, i.e. Google Play's app testing mechanism, was enabled in Google Play.

via its distribution mechanism. Nonetheless, the existing app testing mechanisms suffer from false negatives and false positives. In any case, even though their effectiveness is debatable, these mechanisms do add a level of defense against the malware threat; especially from the threat of non-sophisticated attackers (Mylonas et al., 2011a).

4.3. *Security and agreement messages*

Users tend to ignore repetitive warnings (Egelman et al., 2008; Motiee et al., 2010; Sunshine et al., 2009), especially when a warning appears while the user attempts to fulfil a task (Motiee et al., 2010). Moreover, users tend to pay less attention to consequent warnings, especially when these warnings resemble an End-User License Agreement (EULA) (Böhme and Köpsell, 2010).

The security models of smartphones prompt users with security messages when an application requests access to a protected resource during its installation, i.e. installation permission, or request for runtime access to a resource, etc. They also prompt users with agreement messages (e.g. licensing messages, term and conditions, etc.), concerning the privacy of their data. The security models are heterogeneous, ranging from open and community-based models to strictly controlled ones that follow the walled garden model (Barrera and Van Oorschot, 2011; Mylonas et al., 2011a, 2011b). The former delegate a user to make all authorization decisions for app access requests (e.g. as in Android). According to the latter, every application undergoes security analysis during submission and the user makes only a few authorization decisions (e.g. as in Apple's iOS). In both cases the security models assume that the user will scrutinize these messages, so as to make informed decisions. Otherwise, a user may be exposed to security and privacy attacks generated even by non-sophisticated attackers (Mylonas et al., 2011b).

Fig. 3 depicts the sample's responses with regard to attention paid by the respondents to security and agreement messages. Users were found to focus more on security messages rather than agreement messages. More specifically, the percentage of users who always scrutinize security messages is 38.6%, while only 10% thoroughly examine the agreement messages. Moreover, Fig. 3 suggests a clear asymmetry between the discovered attitude towards security messages and the security model expectations about user attitude towards them. This asymmetry becomes a serious vulnerability when the examination of security messages is a cornerstone in the security model. For instance, in a community-based security model, e.g. as in Android, which authorizes users to grant application access to

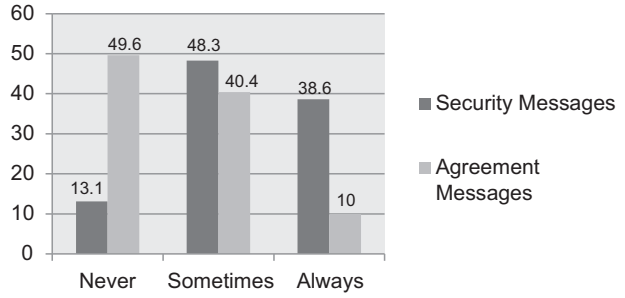

Fig. 3 – Security and agreement messages.

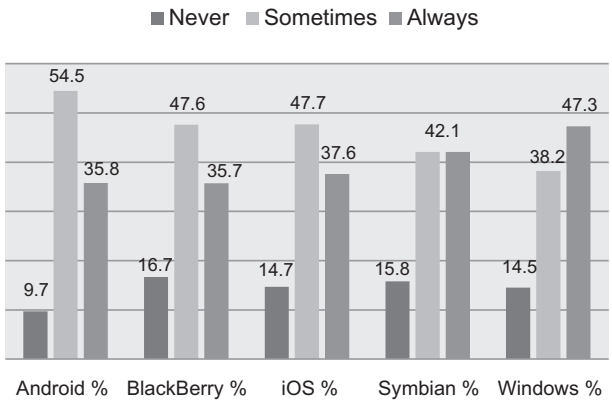

Fig. 4 – Per platform examination of security messages.

protected resources, ignorance of security messages leaves users exposed to malware.

In this context, Fig. 4 shows the user attitude towards security messages in each smartphone platform, where the majority of users only occasionally inspect security messages and the minority of them never inspect security messages. Also, Fig. 5 shows that the majority of users in each platform ignore agreement messages. This is orthogonal with the users' view to privacy, where the majority of users (95.2%) reported concern of their privacy.

4.4. *Pirated applications*

Pirated smartphone applications are a common malware spreading vector (Zhou et al., 2012a; Zhou and Jiang, 2012). Attackers often lure naive (smartphone) users into downloading malware that are masquerading as popular\attractive applications. These applications may enter unobstructed in application repositories that do not operate application testing. Even in application repositories adopting the *walled garden* model (Barrera and Van Oorschot, 2011), which adds a layer of protection against this threat, users often modify (crack) the security model of the device (e.g. in iPhone jailbreaking), so as to install free versions of applications (cracked) that do not reside in the official app repository. This security model modification circumvents the security mechanisms and may expose the user to malware.

The 60.7% of the Greek sample reported preference to pirated applications from original ones. This percentage is in accordance with a survey about PC software piracy that was

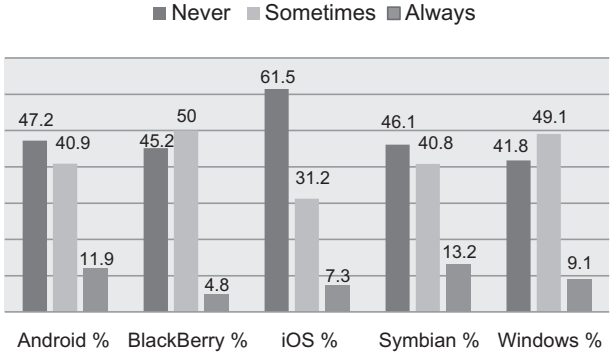

Fig. 5 – Per platform examination of agreement messages.

performed in (BSA, 2012), revealing a considerable piracy rate in Greece (61% while only 33% on average in the EU). As depicted in Fig. 6, preference to pirated applications in Android and Symbian were on average the greatest. Android and Symbian are the two of the three platforms where the security model allows apps to be installed from sources other than the official repository (Mylonas et al., 2011a). BlackBerry is the third platform allowing this flexibility, but as in iOS and Windows Phone that both follow the *walled garden* model, a half of its users reported preference in pirated applications. As a result, there is a significant likelihood that users get infected by a trojanized pirated application.

4.5. *Adoption of security controls*

Sample analysis findings suggest a poor adoption of security controls. This holds for physical controls (i.e. encryption, device password lock, remote data wipe, remote device locator), which pre-exist in most smartphones (hereinafter for brevity reasons we refer to this group of physical controls as 'pre-installed' controls), and for third-party security software that had to be installed by the user (question 17), e.g. anti-virus, anti-theft software, etc.

As far as third-party security software is concerned, 24.5% of the survey participants used smartphone security software (referred hereto as smartphone *secsoft*). On the contrary, 85.8% of the sample uses secsoft in their personal computer (PC). This is a clear indication of the asymmetry, with respect to security awareness of the available threats and security controls in the two platforms. This indication is augmented as: (a) 60% of the sample population reported that they have not searched the app repository for smartphone secsoft that is provided without cost, and (b) 27% were not aware that smartphone secsoft exists. One should note, however, that the heterogeneity of the smartphone security models often imposes restrictions on the availability of third-party security software. For instance, the inability to place hooks in the OS due to sandbox restrictions limits the antivirus availability or the security features that it can offer in smartphones. As a result, users often 'root' their devices, so as to install such security software. This often breaks the security offered by the platform and may also void their warranty.

Moreover, 34.3% of the respondents believed that smartphone secsoft is not essential. This attitude may stem from technological parameters, e.g. battery consumption, overall device performance (i.e. delays), and/or cognitive limitations (i.e. unawareness of existing threats and available controls), and/or psychological parameters, i.e. the trust in the application market itself. Nonetheless, smartphone secsoft poses an additional line of defence against malware threat, in particular against the threat of non-sophisticated attackers (Mylonas et al., 2011a).

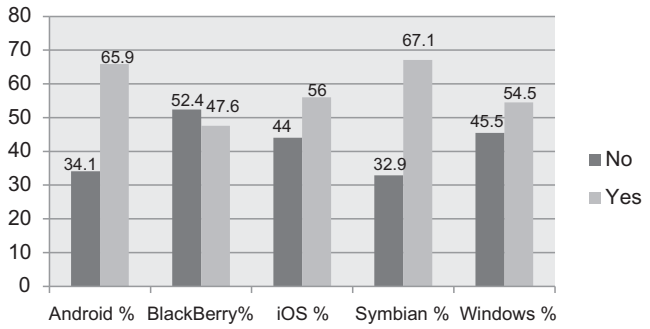

Fig. 6 – Preference to pirated application in each platform.

Fig. 7 presents the per-platform responses to the above mentioned questions. On average, Android users were more often found to enable smartphone secsoft, especially antivirus software. This may be due to the DroidDream incident that took place in February 2011, where 50 malicious applications were detected in the Google Play (Lookout). In addition, only 18.4% of Symbian participants used secsoft, even though the majority of them consider them essential. This poor adoption is alerting given that several malware variants target Symbian (Coursen, 2007). On the other hand, the notably poor adoption of secsoft that was found in iOS may be due to (a) the restrictions imposed by Apple's sandbox (i.e. users are not allowed to install third-party antivirus software in their device), or (b) the fact that iOS applications undergo security analysis during their submission. However, iOS users can opt for third-party anti-theft software (e.g. remote data wipe, remote device locator, etc.), which offer more functionality than the pre-installed software offered by Apple.

Users reported low percentages in searching for free smartphone secsoft, except in Android, where the percentage is 50.6%. Finally, on average, only 66.5% of the users in every platform considered smartphone secsoft essential.

As far as the pre-installed security controls are concerned, the survey demonstrates a poor adoption. Device password lock (64.4%) was, on average, the most adopted control, while the rest had the following adoption: encryption (22.7%), remote data wipe (15.1%), and remote device locator (23.1%). Another finding was that 27.9% of the survey respondents did not use any of them. This finding, as well as the poor adoption of these controls, suggests a serious risk of unauthorized access, due to the device's portability and small size. Also, the percentage of users who do not use these controls is significantly larger than the percentage of the participants who do not use secsoft in their devices (PC and smartphone, 7.2%).

Fig. 8 illustrates the adoption of the above controls in each smartphone platform. We note that (a) remote data wipe and remote device locator had the greatest adoption in iOS users, and (b) a misconception about encryption exists in iOS users. For the latter, when a user password protects her iPhone, she also enables encryption with a key derived from the provided PIN. In the survey, 72.6% of the iOS users who enabled password lock believed that they had encryption disabled.

4.6. *Application selection criteria*

The examination of the survey results, revealed eight application selection criteria, namely: (a) cost, (b) developer, (c) review, (d) reputation, (e) security/privacy, (f) usefulness, (g) usability, and (h) efficiency. Smartphone users identified these criteria directly, e.g. "*if the application is useful for me*", "*how expensive it is*" or indirectly, e.g. "*how many stars it has*", "*I read what other users think about it in the application page or forums*".

Survey findings indicate that smartphone users disregard security issues when it comes to application selection from the official app repository. A smartphone user who is concerned about her security and privacy is expected to spend time on the

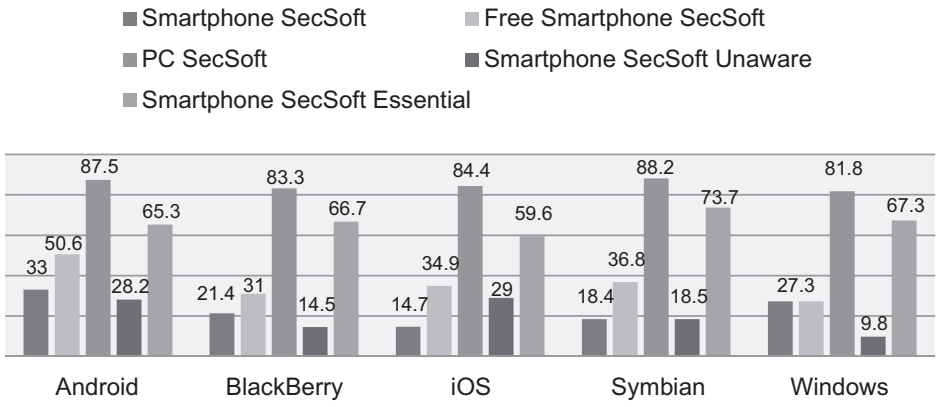

Fig. 7 – Third-party SecSoft adoption in smartphone platforms.

application's reviews, its reputation, and its functionality, in particular in permission-based security models (e.g. Android, Windows Phone) where access to sensitive functionality is user-granted. The survey revealed that only 10.5% of the respondents delve into reviews and only 8.7% consider reputation. More than the half of the sample population (58.5%) identified usefulness as an application selection criterion. Usability, efficiency, and cost had a lesser proportion of respondents. Finally, the application source (developer), as well as security or privacy issues (e.g. drop the installation of an app because a combination of requested permissions may impair privacy), were identified by a low proportion of respondents (3.7% and 3.5%, respectively) (Table 2).

5. Results based on inferential statistics

To further investigate the security awareness of smartphone users, we checked the correlation between the responses in the questionnaire. For doing so, we drafted the contingency tables of each pair in the categorical variables in the questionnaire (appendix A lists the variables and their notations). Then, for each pair of categorical variables a and b, we examined whether an association between them exists, by computing the appropriate x^2 distribution test of independence. More specifically, for each variable pair a, b we tested the null hypothesis (significance level a = 0.05), i.e. H_0: *variables a and b are independent* and H_1: *variables a and b are not independent*.

In the cases where: (a) the expected cell frequency for each cell in the contingency table was ≥5, and (b) a statistically significant relationship ($p < 0.05$) between the variables was discovered, we further investigated the association by computing the φ coefficient. This section discusses the analysis' findings. We group the findings into: (a) user beliefs and perceptions, (b) user practices, and (c) app selection criteria.

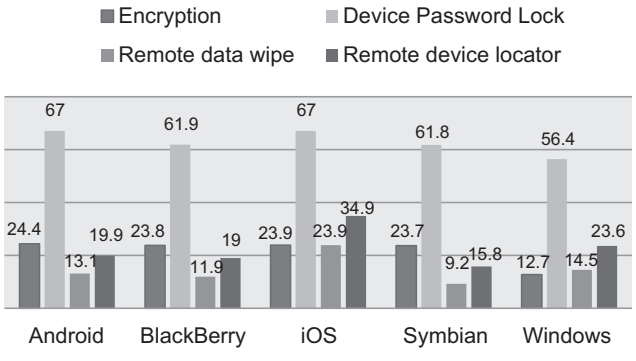

Fig. 8 – Adoption of pre-installed security controls in each platform.

5.1. Associations of user beliefs and perceptions

In the sequel we will describe the findings from inferential statistics, with respect to: (a) user trust in the app repository, (b) users' regard of smartphone secsoft being essential, (c) privacy concern and preference to pirated applications, (d) awareness of the smartphone secsoft existence, and (e) awareness of application testing in the app repository.

5.1.1. User trust in the app repository

Table 3 depicts the associations between trust in the app repository and the rest of the responses in the questionnaire. The findings suggest that users who trust the app repository are more likely to: (a) be non-security savvy, and (b) have either good or moderate IT skills. Moreover, users that are unaware of smartphone malware are more likely to trust the app repository. This is alerting because the app repository may be more easily used as a malware distribution point, as users tend to be unaware of the threat and incapable of discovering it themselves.

Table 2 – App selection criteria.

App criterion	% of respondents
Usefulness	58.5
Usability	15.7
Efficiency	15.1
Cost	13.1
Reviews	10.5
Reputation	8.7
Developer	3.7
Security/privacy	3.5

Table 3 – Associations of *TrustRep* variable.

Variable label	x^2	df	p	φ
Application testing in OAR (*No*)	11.410	1	0.001	−0.158
BlackBerry OS	5.323	1	0.021	0.108
IT expertise (*Moderate*)	5.137	1	0.023	0.106
IT expertise (*Good*)	11.697	1	0.001	0.160
IT expertise (*Excellent*)	22.518	1	<0.001	−0.222
No use of security software	18.161	1	<0.001	−0.199
Security messages (*Always*)	7.871	1	0.005	−0.131
Security messages (*Sometimes*)	8.196	1	0.004	0.134
Security savvy user	10.893	1	0.001	−0.154
Smartphone malware existence	16.448	1	<0.001	−0.190
Smartphone security software existence	4.673	1	0.031	−0.101
Store personal data	5.685	1	0.017	0.111

Survey findings indicate that smartphone secsoft is more likely not to be in place, as users tend to be unaware of it. In this context, as these users were also found more likely to store personal data, the risk of unauthorized data access via malware, which is spread in the app repository increases, especially in app repositories lacking app testing. Also, users who trust the app repository are less likely to scrutinize security messages. They were found more likely to occasionally inspect them. This is a serious vulnerability of smartphone security models, as they delegate users to permit access to protected resources via these messages. Also, BlackBerry users are more likely to trust the *BlackBerry App World*, even though several available security controls for app repositories are not enabled in it (Mylonas, 2011b).

Our findings suggest that users who do not trust the app repository are more likely users with excellent IT skills and security savvy. Also, they tend to be aware of smartphone malware, as well as smartphone secsoft. Finally, the users who do not use security software on any device (i.e. smartphone, PC) tend not to trust the application repository.

Table 4 – Associations of secsoft essential.

Variable label	x^2	df	p	φ
No use of security software	23.334	1	<0.001	−0.226
Privacy concern	8.841	1	0.003	0.139
Searched for free smartphone security software	9.997	1	0.002	0.148
Security or privacy	5.782	1	0.016	0.112
Security savvy user	10.803	1	0.001	0.154
Smartphone malware existence	12.322	1	<0.001	0.164
Use device password lock	8.435	1	0.004	0.136
Use remote locator	4.749	1	0.029	0.102
Use security software in PC	10.928	1	0.001	0.154
Use security software in smartphone	77.329	1	<0.001	0.411

Table 5 – Associations of privacy concerns.

Variable label	x^2	df	p	φ
Agreement messages (*Never*)	7.098	1	0.008	−0.124
Agreement messages (*Sometimes*)	4.735	1	0.030	0.102
Smartphone security software essential	8.841	1	0.003	0.139

5.1.2. *Users who consider smartphone secsoft essential*

Survey results (Table 4) indicate that those who consider smartphone secsoft essential are more likely to use security software in their PC. Nonetheless, the results do not suggest likewise for the adoption of smartphone secsoft, an attitude also found in (Androulidakis and Kandus, 2011b). Participants who use smartphone secsoft tend to regard them as essential and users who consider smartphone secsoft essential are less likely to respond that they do not use security software in any of their devices. The results revealed a positive correlation with awareness of smartphone malware and search for free smartphone security software in the app repository. Also, the users are more likely to be security savvy and concerned about their privacy. Nonetheless, the results also indicate that the users who do not regard secsoft as essential are more likely not to consider security issues during application selection. Thus, they are exposed to malicious applications, especially when app testing does not take place in the repository.

As far as the pre-installed security controls are concerned, a positive correlation was found between users who regard secsoft essential and the use of device password lock and remote device locator. On the other hand, a statistically significant positive correlation with encryption and remote data wipe was not found. As a result, unless a third-party secsoft is installed, these users are exposed to unauthorized access to their data (e.g. theft, data leakage, etc.).

5.1.3. *Privacy concern and preference to pirated applications*

Table 5 illustrates that privacy-concerned smartphone users are more likely to regard smartphone security software as essential. These users are more likely to respond that they occasionally inspect agreement messages and less likely to ignore them.

The results indicate that users who prefer pirated smartphone applications tend to ignore agreement messages. In specific, we found a negative correlation with the respondents who scrutinize agreement messages and a positive correlation with the ones who occasionally inspect them (Table 6).

Table 6 – Associations of prefer pirated apps.

Variable label	x^2	df	p	φ
Agreement messages (*Always*)	6.357	1	0.012	−0.118
Agreement messages (*Never*)	9.625	1	0.002	0.145

Table 7 – Associations of secsoft existence.

Variable label	x^2	df	p	φ
Android OS	7.480	1	0.006	0.128
Application testing in OAR (*Don't know*)	13.373	1	<0.001	−0.171
IT expertise (*Moderate*)	12.678	1	<0.001	−0.166
Smartphone malware existence	65.791	1	<0.001	0.379
Trust app repository	4.673	1	0.031	−0.101
Use device password lock	21.010	1	<0.001	0.214
Use remote locator	10.025	1	0.002	0.148
Use security software in smartphone	55.041	1	<0.001	0.347

5.1.4. *Smartphone security software existence*

Survey findings suggest that users with moderate IT skills are less likely to be aware of the existence of smartphone secsoft. Also, users who are unaware of smartphone secsoft are more likely to trust the app repository and tend to be unaware of smartphone malware. Therefore, the risk of malware is considerable and, in the case that these users download a malicious application from the app repository, they will be unable to find that they are infected. The results revealed that participants who use security software tend to be aware of their existence.

Moreover, the results suggest that Android users were more likely to be aware of the existence of smartphone secsoft. This may be due to the fact that a lot of smartphone malware target the Android platform (Felt et al., 2011b; Zhou et al., 2012b; Zhou and Jiang, 2012). The users who are aware of smartphone secsoft existence are more likely to have the device password protected. Also, they are more likely aware of smartphone malware and less likely unaware whether application testing takes place in the app repository. Finally, the results suggest that users who are unaware of the existence of smartphone secsoft are more likely to have the remote device locator disabled. Thus, in the case of temporal misplace or permanent device loss the device can hardly be recovered. A summary of the findings appears in Table 7.

5.1.5. *Application testing*

Survey findings (see Table 8) show that users who do not know if application testing takes place in the app repository are less likely users with excellent IT skills. Also, the users who respond that application testing takes place in the app repository are more likely to be users with excellent IT skills. Nonetheless, 42.3% of the Android users and 43.5% of the BlackBerry users who were technically savvy responded, somewhat erroneously, that apps undergo security analysis during their submission in the app repository.

Our findings suggest that users with good IT skills are more likely to respond that they do not know whether application testing takes place in the app repository. Nonetheless, users with good IT skills tend to trust the app repository, albeit they are not aware if application testing takes place in it. Moreover, users who do not use smartphone secsoft and are unaware of smartphone malware tend to be uncertain if app testing takes place in the app repository. This suggests that these users ignore endpoint security, even though they are uncertain if apps have been security analyzed in their submission. They are also ignorant about the malware threat.

The results revealed that users who are uncertain if application testing takes place in the app repository are less likely to scrutinize security messages. This suggests that ignorance of security messages is not a result of the trust in the efficiency of the app testing mechanism.

Finally, a negative correlation between iOS users and the users who respond that they do not know whether application testing occurs in the Apple's App Store was found, which was expected due to the walled garden approach employed by Apple. Nonetheless, a positive correlation with the users who answer that app testing happens in the Apple's App Store was not found.

5.2. *Associations for user practices*

In the sequel we refer to the results of the inferential tests, with respect to: (a) user adoption of smartphone secsoft, (b) lack of adoption of secsoft in any user device, (c) adoption of pre-installed security controls, (d) storage of personal and business data, and (e) examination of security/messages.

5.2.1. *Use of secsoft in smartphone*

The survey findings suggest that the smartphone users who use smartphone security software in their smartphone were also aware of their existence, considered them essential, have searched for free secsoft in the repository, and were aware of smartphone malware (see Table 9). Moreover, these users tend to respond that application testing takes place in the app repository. This suggests that users opt for both centralized security and endpoint security.

Our findings indicate that smartphone users are more likely to have encryption, device password lock, and the remote locator mechanism disabled when they do not use smartphone secsoft. This poor adoption of available security

Table 8 – Associations of app testing.

Variable a label	Variable b label	x^2	df	p	φ
Application testing in OAR (*Don't know*)	iOS	6.421	1	0.011	−0.118
Application testing in OAR (*Don't know*)	IT expertise (*Excellent*)	9.231	1	0.002	−0.142
Application testing in OAR (*Don't know*)	IT expertise (*Good*)	5.089	1	0.024	0.105
Application testing in OAR (*Don't know*)	Security messages (*Always*)	12.873	1	<0.001	−0.168
Application testing in OAR (*Don't know*)	Smartphone malware existence	12.426	1	<0.001	−0.165
Application testing in OAR (*Don't know*)	Use security software in smartphone	12.412	1	<0.001	−0.165
Application testing in OAR (*Yes*)	IT expertise (*Excellent*)	9.132	1	0.003	0.141

Table 9 – Associations of smartphone secsoft usage.

Variable label	x^2	df	p	φ
Android OS	11.180	1	0.001	0.156
Application testing in OAR (*Don't know*)	12.412	1	<0.001	−0.165
Application testing in OAR (*Yes*)	19.261	1	<0.001	0.205
iOS	7.399	1	0.007	−0.127
No use of security software	11.512	1	0.001	−0.159
Searched for free smartphone security software	57.785	1	<0.001	0.355
Smartphone malware existence	19.486	1	<0.001	0.206
Smartphone security software essential	77.329	1	<0.001	0.411
Smartphone security software existence	55.041	1	<0.001	0.347
Use device password lock	7.252	1	0.007	0.126
Use encryption	20.782	1	<0.001	0.213
Use remote locator	26.787	1	<0.001	0.242

controls leaves users exposed to several smartphone threats, such as the ones listed in Theoharidou et al. (2012). The risk of malware threat is severe, as these users were found unaware about smartphone malware. In addition, these users were more likely to be unaware if application testing takes place in the app repository. Therefore, this poor adoption of security controls seems not to stem from users' trust in the application testing effectiveness, but from users' unfamiliarity with the available smartphone threats.

Finally, the survey results indicate that users who adopt smartphone secsoft are more likely Android users and less likely iPhone users. Also, Android users were more likely to have searched the app repository for free secsoft ($x^2 = 12.708$, df = 1, $p \leq 0.001$, $\varphi = 0.171$). The former is true because, currently, more smartphone malware target Android (Felt et al., 2011b; Zhou et al., 2012b; Zhou and Jiang, 2012). For iOS users, the lack of adoption of smartphone secsoft may steam from Apple's walled garden approach.

5.2.2. *No adoption of security software*

The results that appear in Table 10 indicate that users who do not use security software on any device (i.e. either PC or smartphone) tend to believe that smartphone security software is not essential (and also tend not to use them). Also, these users are more likely not to trust the app repository. Nonetheless, any correlation with the rest users' security practices, especially regarding the adoption of pre-installed security controls and attitude towards security messages, was not found. Thus, a safe deduction about their level of awareness cannot be performed.

Table 10 – Associations of users who do not use secsoft in any device.

Variable label	x^2	df	p	φ
Smartphone security software essential	23.334	1	<0.001	−0.226
Trust app repository	18.161	1	<0.001	−0.199
Use security software in smartphone	11.512	1	0.001	−0.159

Table 11 – Associations of user with encryption enabled.

Variable label	x^2	df	p	φ
IT expertise (*Excellent*)	9.198	1	0.002	0.142
IT expertise (*Good*)	5.048	1	0.025	−0.105
Searched for free smartphone security software	26.123	1	<0.001	0.239
Security messages (*Always*)	37.705	1	<0.001	0.287
Security messages (*Never*)	6.352	1	0.012	−0.118
Security messages (*Sometimes*)	18.335	1	<0.001	−0.200
Smartphone malware existence	7.121	1	0.008	0.125
Use device password lock	17.609	1	<0.001	0.196
Use remote data wipe	6.749	1	0.009	0.121
Use remote locator	9.954	1	0.002	0.147
Use security software in smartphone	20.782	1	<0.001	0.213

5.2.3. *Associations of pre-installed controls*

Tables 11–14 summarize the statistically significant correlations of the answers regarding the 'pre-installed' security controls (i.e. encryption, device password lock, remote data wipe, remote device locator) with the rest of the answers in the questionnaire. The results suggest that the technical skills of the respondents tend to affect the adoption of security controls. In specific, users were found to enable encryption, remote data wipe, and remote device locator when they tend to be technically savvy. A similar correlation was not found in the adoption of device password lock. Our results indicate that users who do not password-protect their devices are more likely to be non-security savvy. This is a correlation which was not found in the other security controls. Also, the results revealed that participants are more likely to disable together the security controls: encryption, remote data wipe, and remote device locator. In this case, the risk of unauthorized physical access is severe (e.g. theft, data leakage, etc.).

Users having good IT skills tend not to encrypt their data. The respondents who encrypt their data are more likely aware of smartphone malware and use device password lock. The results suggest that these users are more likely to have searched the app repository for free smartphone secsoft. Moreover, the survey results indicate that participants who do not encrypt their data are more likely not to use smartphone secsoft, be unaware about smartphone malware, as well as not search for free smartphone

Table 12 – Associations of use device password lock.

Variable label	x^2	df	p	φ
Searched for free smartphone security software	27.105	1	<0.001	0.243
Security savvy	5.743	1	0.017	0.112
Smartphone malware existence	19.852	1	<0.001	0.208
Smartphone security software essential	8.435	1	0.004	0.136
Smartphone security software existence	21.010	1	<0.001	0.214
Use encryption	17.609	1	<0.001	0.196
Use remote data wipe	15.773	1	<0.001	0.186
Use remote locator	18.775	1	<0.001	0.202
Use security software in smartphone	7.252	1	0.007	0.126

Table 13 – Associations of remote data wipe control.

Variable label	x^2	df	p	φ
IT expertise (*Excellent*)	9.121	1	0.003	0.141
IT expertise (*Good*)	7.723	1	0.005	−0.130
Use device password lock	15.773	1	<0.001	0.186
Use encryption	6.749	1	0.009	0.121
Use remote locator	161.497	1	<0.001	0.594

secsoft in the app repository. Thus, the risk of a remote unauthorized access attack is serious. Also, users who either occasionally inspect security messages, or ignore them at all, are more likely to disable encryption. Thus, they are exposing their data to remote unauthorized access. In contrast, users who always inspect security messages are more likely to encrypt their data.

The survey results suggest that users who tend to disable device password lock are less likely to use smartphone secsoft and tend to disable remote data wipe, remote locator as well as encryption. Thus, they are exposed to physical unauthorized access attacks without any relevant control in place. The users who password-protect their device are more likely to be aware about the existence of smartphone malware and smartphone secsoft and consider the latter as essential. Nonetheless, our results do not indicate that they also tend to use smartphone secsoft. Also, users who do not password lock their device are more likely not to use smartphone secsoft and not have searched for free secsoft in the app repository. In this case, the device and its data are exposed to unauthorized access attacks.

Survey results show that users with good IT skills tend not to enable remote date wipe. Users who do not enable remote data wipe are less likely to enable device password lock and remote device locator. In this case, the protection against unauthorized access attacks depends on the strength of device password lock mechanism. In smartphones this mechanism may suffer, apart from the traditional attacks against device passwords (e.g. guessing, shoulder-surfing, etc.), from smudge attacks when graphical passwords are used (Aviv et al., 2010).

Participants with good IT skills are less likely to enable remote locator in their devices. Users who enable remote locator tend to be aware of the existence of smartphone malware and security software. Nonetheless, our results do not suggest that they tend to use smartphone secsoft in their device, even though they tend to consider them essential. On the contrary, our results revealed that participants are more likely not to adopt both remote device locator and smartphone secsoft. As far as the pre-installed security controls are concerned, users who enable the remote device locator control are more likely to have the device password control enabled, as well as the remote data wipe. As a result, again the protection against unauthorized access relies on the strength of the device lock mechanism.

Table 14 – Associations of remote locator.

Variable label	x^2	df	p	φ
IT expertise (*Excellent*)	10.505	1	0.001	0.151
IT expertise (*Good*)	6.983	1	0.008	−0.123
Smartphone malware existence	7.598	1	0.006	0.129
Smartphone security software essential	4.749	1	0.029	0.102
Smartphone security software existence	10.025	1	0.002	0.148
Use device password lock	18.775	1	<0.001	0.202
Use encryption	9.954	1	0.002	0.147
Use remote data wipe	161.497	1	<0.001	0.594
Use security software in smartphone	26.787	1	<0.001	0.242

5.2.4. *Storage of personal and business data*

The associations of users who store personal data in their device are summarized in Table 15. A significant association with the controls that thwart unauthorized access to data was not found. This may result in the unauthorized exposure of sensitive data when these controls are not enabled, as it was also found in (Androulidakis and Kandus, 2011a). This was also true with the users who have misplaced their device.

Survey results suggest that users who store business data in their devices are most likely to also store personal data. This indicates that the same device is used for personal and business purposes, and, as a result, the impact of unauthorized access to the data is greater. The results also reveal that users who store personal data are more likely to: (a) trust the app repository, and (b) occasionally inspect security messages. This is a notable finding, as the likelihood that a user installs from the app repository a privacy breaching application that gains unauthorized access to her data is serious. Finally, users who are unaware of smartphone malware are more likely to store personal data on their device, being exposed to unauthorized access to their data. Such an attitude was also found in (An and Kandus, 2011a), where users that stored personal data did not follow security practices, e.g. backups.

5.2.5. *Security and agreement messages*

Our findings reveal that users who always check an app's security messages are more likely to have excellent IT skills. On the other hand, users with moderate IT skills are more likely to respond that they sometimes inspect security messages. This is alerting, as users who are not technically savvy are not able to make the appropriate security decisions, which is not in accordance with the expectations of permission-based smartphone security models and, thus, shows a serious vulnerability.

Also, users who occasionally check security messages are more likely to be non-security savvy users, while those who scrutinize them tend to be security savvy. This suggests, in

Table 15 – Associations of personal data.

Variable label	x^2	df	p	φ
Security messages (*Always*)	5.118	1	0.024	−0.106
Security messages (*Sometimes*)	7.514	1	0.006	0.128
Smartphone malware existence	4.545	1	0.033	−0.100
Store business data	45.775	1	<0.001	0.316
Trust app repository	5.685	1	0.017	0.111

contrast to what permission driven security models (e.g. Android) assume, that users who are non-security savvy cannot make appropriate security decisions. Users who scrutinize security messages tend to be aware of smartphone malware and smartphone secsoft.

The analysis of the correlation between agreement messages and security messages revealed that: (a) users who always check security messages are less likely to respond that they ignore agreement messages, (b) users that scrutinize agreement messages are less likely to respond that they ignore security messages, (c) users who scrutinize agreement messages are more likely to always scrutinize security messages too. Our findings also revealed that users who never check security messages are more likely to never check agreement messages too. This suggests that some smartphone users totally disregard any prompt from applications. Thus, different methods to communicate security authorization are needed, such as visual notifications (Howell and Schechter, 2010), especially in the case of permission driven security models. Also, this user behaviour, where users click through security messages, is known to exist in other platforms, e.g. Windows User Account Control Dialogs (Motiee et al., 2010), in browser messages that warn about invalid SSL certificates and phishing scams (Egelman et al., 2008; Sunshine et al., 2009). In this context, it is surprising that some smartphone security models delegate users to make all authorization decisions for app access requests. Moreover, our results indicate that users who ignore security messages tend to be unaware whether apps undergo security analysis during their submission. The survey demonstrates that this security complacency does not stem from user trust in the efficiency of the app testing mechanism used by the app repository. These findings are depicted in Table 16.

5.3. *Application selection criteria*

Survey results suggest that users who inspect an app's reputation during application selection are more likely to be security savvy users. On the other hand, users who are non-security savvy are more likely not to examine the reputation of an app during application selection.

Our findings indicate that users who select usefulness as an app selection criterion are more likely to ignore the app's review. Security-wise, this is important as in app repositories where the security model is community driven, e.g. in Android, the reviews often give indications whether an app is malicious or not. In the same context, users are more likely to ignore the app's reputation. Reputation is important in any commodity market, even thought the literature has identified shortcomings of reputation systems (Marmol and Perez, 2009). In the context of app repositories, an application with good reputation does not necessary respect the user's privacy, but reputation often poses an additional line of defence against malevolent users. An app's reputation may be low because several users have found it to be malicious or permission-hungry, i.e. requesting more permissions than those that are expected from its functionality, suspiciously draining resources (e.g. CPU, battery). Finally, the users who consider security or privacy issues during application selection are more likely to be security savvy users and regard smartphone security software essential. The above are depicted in Table 17.

6. Trust repository prediction model

The survey findings revealed strong indications regarding the lack of security awareness in smartphone users. The users' belief that applications deriving from the official repository are secure is a serious vulnerability. The trust in the application repository is more likely to (a) be encountered in users who are not technology and security savvy, and (b) increase the users' security complacency, as they are more likely not to scrutinize security messages and not use security software. Moreover, this trust is not always justified by the availability of security controls in the app repository (e.g. app testing) or their effectiveness regarding the mitigation of malware. As a result, this security complacency of users regarding trust in the app repository may be exploited by attackers using the repository as a malware infection vector.

The users with this security complacency cannot be identified via software solutions (e.g. software a, MDM, etc.). Therefore, we provide a prediction model that can be used by an

Table 16 – Associations of security and agreement messages.

Variable a label	Variable b label	x^2	df	p	φ
Agreement messages (*always*)	Security messages (*Always*)	59.801	1	<0.001	0.361
Agreement messages (*always*)	Security messages (*Never*)	7.709	1	0.005	−0.130
Agreement messages (*never*)	Security messages (*Always*)	52.431	1	<0.001	−0.338
Agreement messages (*never*)	Security messages (*Never*)	57.022	1	<0.001	0.353
Security messages (*always*)	IT expertise (*Excellent*)	16.578	1	<0.001	0.190
Security messages (*always*)	IT expertise (*Good*)	4.632	1	0.031	−0.101
Security messages (*always*)	IT expertise (*Moderate*)	10.330	1	0.001	−0.150
Security messages (*always*)	Security savvy user	8.097	1	0.004	0.133
Security messages (*always*)	Smartphone malware existence	8.554	1	0.003	0.137
Security messages (*always*)	Smartphone security software existence	6.628	1	0.010	0.0120
Security messages (*never*)	Application testing in OAR (*Don't know*)	5.264	1	0.022	0.107
Security messages (*sometimes*)	IT expertise (*Excellent*)	9.100	1	0.003	−0.141
Security messages (*sometimes*)	IT expertise (*Moderate*)	8.250	1	0.004	0.134
Security messages (*sometimes*)	IT expertise (*Moderate*)	8.250	1	0.004	0.134
Security messages (*sometimes*)	Security savvy user	6.485	1	0.011	−0.119

Table 17 – Associations of application selection criteria.

Variable a label	Variable b label	x^2	df	p	φ
Reputation	Security savvy user	6.319	1	0.012	0.117
Security or privacy	Smartphone security software essential	5.782	1	0.016	0.112
Security or privacy	Security savvy user	12.949	1	<0.001	0.168
Usefulness	Reputation	12.219	1	<0.001	−0.163
Usefulness	Reviews	6.270	1	0.012	−0.117

organization to identify them and subsequently engage them in a relevant security training so as to raise their awareness (Thomson and von Solms, 1998). To build the prediction model we computed the logistic regression model for the categorical variable *TrustRep*, denoting user trust in the app repository. In order to select the independent coefficients (x_i) of the prediction model we applied stepwise Forward Wald (Harrell et al., 1984). For readability reasons the details about the estimation of the logistic regression model are given in Appendix E.

The prediction model is described by Equation (1). Table 18 presents details of the model's selected predictors, i.e. the predictor's condition, factor (b_i) and significance level (p). The model assigns to the binary variable *TrustRep* the value *True*, i.e. predicts that the given user trusts the app repository, when the probability $\exp(z)/(1+\exp(z)) \geq 0.5$.

$$z = 1.351 * x_1 + 1.092 * x_2 - 1.688 * x_3 + 1.523 * x_4 + 1.314 * x_5 - 0.475 * x_6 - 0.741 * x_7 \quad (1)$$

We statistically assessed the prediction model, so as to ensure its significance by performing the following tests regarding the model's: (a) statistical significance, (b) goodness-of-fit (i.e. whether the model adequately describes the data), and (c) the minimum sample size required for logistic regression. Our model was successfully tested against the above mentioned tests. For readability and space reasons the details of the model's statistical assessment is given in Appendix E.

6.1. *Model evaluation*

The experimental results regarding the effectiveness of our prediction model, in terms of the correct percentage of its predictions in the Greek sample, are depicted in Table 19. The model is prone to Type I errors (false positives), performing 25.5% correct predictions. On the contrary, Type II errors (false negative) are negligible, as the model correctly predicted 96% of the users who trust the app repository. We argue that a significant Type I error coupled with a negligible Type II error is a good security tradeoff, as far as smartphone awareness is concerned. This is so, as the Type I error would incur nuisance to users who do not trust the app repository. For instance, in an organization these users may be asked to undergo relevant security training in order to raise their awareness. On the other hand, Type II errors may lead to security incidents.

Table 18 – *TrustRep* prediction model coefficients.

x_i	x_i condition	b_i	p
x_1	IT = "Moderate"	1.351	0.007
x_2	IT = "Good"	1.092	0 < 0.001
x_3	BlackBerry = False	−1.688	0 < 0.001
x_4	NoUse = False	1.523	0 < 0.001
x_5	Security/Privacy = False	1.314	0.003
x_6	Pirated = False	−0.475	0.047
x_7	Personal = False	−0.741	0.005

Table 19 – Evaluation of the prediction model in the Greek sample (n = 458).

Predicted observed	No	Yes	Percentage correct
No	28	82	25.5%
Yes	14	334	96.0%
Overall percentage			79.0%

The prediction model was also tested against a sample different from the one that was used to estimate the prediction model. This would give an indication on whether our model, which was built via inferential statistics, suffered by the demographics of the Greek sample. In this context, the prediction model was tested against a sample (n = 102, $\min_{age}$ = 19, $\max_{age}$ = 59) comprising of smartphone users living in the United Kingdom ('UK sample'). The data collection methodology of this sample was the same as that of the Greek sample. Concerning the demographics of the UK sample, ~30% were security savvy users, 72.5% were male, 80.4% of the population was aged [19–30] and 68.6% stored personal data on their device. Regarding their technical skills ~28.% of the respondents were technically savvy and ~16% were non-technical savvy. Most of the respondents (78.4%) trusted the app repository; ~41% preferred pirated apps and only ~6% reported security or privacy issues as an app selection criterion. Finally, only 45% and 21.6% of the users scrutinized security and agreement messages, respectively. Examining the adoption of security controls revealed that: (a) 18.6% used smartphone secsoft, (b) ~6% did not use any software secsoft in any device, (c) ~61% password protected and 18.6% encrypted their device, and (d) 10.8% used remote wipe and 14.7% remote locator.

Table 20 depicts the performance (i.e. percentage of correct predictions) of our model in the UK sample. It turns out that the accuracy of the model with the UK sample is similar to the accuracy with the Greek one. Again, the model is more prone to Type I error, having a slight increase in the correct percentage of true negatives. Regarding true positives, the model had a 5% decrease in the percentage of correct predictions and 0.6% of the overall correct predictions percentage. This indicates that the statistically inferred prediction model is correct and not dependent on the demographics of the Greek sample. Due to the small size of the UK sample, further deductions or comparisons would be unsafe.

Table 20 – Evaluation of the prediction model in the UK sample (n = 102).

Predicted observed	No	Yes	Percentage correct
No	7	15	31.8%
Yes	7	73	91.3%
Overall percentage			78.4%

7. Limitations

One limitation of our study is that findings may be affected by the sample's demographics. It might be the case that significant differences exist in the security awareness of smartphone users between different demographics (e.g. different country and/or continent). Also, our results may be biased towards the male population, since the majority of the participants were male. However, in the sample we found only a few statistical significant differences in the responses between male and female respondents. Also, the majority of the participants were aged [15–30], which also introduces a kind of bias to our analysis. Nonetheless, as users in these ages tend to be the early adopters of technology, we consider that our analysis gives considerable insight about smartphone security awareness in general. This is also validated by the common security findings from our research and the existing relevant literature, which are comparable and suggest a clear security unawareness of smartphone users.

Our data collection relies on self-reported statistics, e.g. we asked users to classify their IT expertise and whether they had an information security background (Theoharidou and Gritzalis, 2007) (at minimum comprehension of the security notions of threat, risk, safeguards, attacks) – deriving either from academic sources (e.g. undergraduate/postgraduate information security courses) or industrial sources (e.g. certifications on information security). Although we decided to avoid a direct validation of these responses (e.g. with a background quiz or by inspecting the user's device), to keep the length of the instrument short and avoid the user's fatigue, our discussion with the user ensured the: (a) validity of her responses, (b) comprehension of the questions, and (c) comprehension of technical terms. Furthermore, during the discussion the researchers were cautious not to reveal directly that the final purpose of the survey was to measure the level of security and privacy awareness, as we expected that this would inflate their responses, e.g. as in Androlidakis and Kandus (2012). This was in fact the reason why in the beginning of the survey we claimed to do a survey about a smartphone app usage and not about smartphone security. Nonetheless, the findings showed that we successfully avoided such misleading behaviours, since the majority of the answers we collected are alerting in terms of the user's security awareness. Finally, we used Q5, Q7 and Q17–Q18 as control questions, in order to ensure that the researcher did not accidentally overlook any ambiguous responses from outliers during the interview. In the case that the control questions identified outliers during the sample's analysis, the relevant data were excluded from the analysis (leaving us with 458 cases).

Another limitation of the survey is that it is cross-platform (i.e. focuses on five different popular smartphone platforms), thus it may suffer by the heterogeneity of security software controls that are available in each platform. In this context, a security control may be provided by the platform as a service e.g. (remote wipe in iOS is provided with iCloud), whereas it may require the installation of a third-party application from the app repository in another platform (e.g. in Android). Moreover, a security control may not be supported, due to restrictions of the platform's security model, e.g. antivirus in iOS. Also, the same security control may be implemented differently in these platforms and, thus, the provided security, as well as the available attacks against it may differ significantly. For instance, Android offers device lock using graphical passwords, which – apart from the traditional attacks against passwords (e.g. guessing, shoulder-surfing, etc.) – may suffer from smudge attacks (Aviv et al., 2010). Furthermore, software exists in some platforms, which protects access (e.g. via encryption, passwords, etc.) to selected assets, such as corporate documents or multimedia files. This software may be available either as a standalone app in the app repository, or as part of a Mobile Device Management (MDM) solution. Nonetheless, our survey did not focus on the above mentioned details regarding security controls. It examined whether users adopt these security controls, without focussing on the implementation details or the software origin (i.e. third-party or offered by the platform), finding that the majority of participants do not use them.

8. Discussion and conclusions

The smartphone platforms' security models delegate users to make security decisions while downloading apps from official app repositories. This delegation is heterogeneous, ranging from simply authorizing access to some protected resources, to authorizing the user to deduce whether an app may impair her security and privacy. The survey findings (see Table 21) are not in-line with the expectation of smartphones' security models for a reasonable security aware user. In contrast, they suggest that users are not adequately prepared to make appropriate security decisions.

The findings of the survey analysis show that the majority of smartphone users believe that downloading applications from the app repository is risk-free. It is unclear why this security complacency exists, as the majority of smartphone users who trust the app repository we unaware whether application testing mechanism is enabled in it. Two factors may be decisive in misleading users. First, the fact that an app is distributed from an official app repository may mislead the users into believing that the app is secure or that security controls exist in the app repository. Second, smartphone users may be unable to realize that their device is not just a telephone. Users who use mobile telephony – and in particular those using fixed telephony – have been accustomed to use voice services in a context with limited threats. In contrast to feature phones and fixed landline devices, smartphones combine the capabilities of a feature phone and a handheld computer and, as a result, it is exposed to the union of threat targeting both devices (Theoharidou et al., 2012).

On the contrary with what popular permission-driven smartphone security models (e.g. Android, Symbian, Windows Phone, etc.) may expect, our findings suggest that users tend to ignore security messages that are prompted to them. Also, some users were found to be more likely to ignore all messages that an application displays. This is a vulnerability that violates the trust model of smartphone security models. This is since these models assume that smartphone users will thoroughly inspect these messages in order to make correct decisions that affect their privacy and security. It should be noted that this user behaviour, where users click through security messages, is known to exist in other

Table 21 – Main findings of the survey.

Q1: Do smartphone users enable security controls on their devices?
Finding 1: Smartphone secsoft is poorly adopted. (24.5% of sample)
Finding 2: Smartphone secsoft is considered not essential (34.3%)
Finding 3: Most users have not searched the repository for free smartphone secsoft (60%)
Finding 4: Few users are unaware of the smartphone secsoft existence (27%)
Finding 5: Poor adoption of 'pre-installed' security controls
Finding 5.1: Encryption (22.7%)
Finding 5.2: Remote data wipe (15.1%)
Finding 5.3: Remote device locator (23.1%)
Finding 5.4: No adoption of any pre-installed security control (27.9%)
Finding 6: Users tend to have disabled smartphone secsoft along with encryption, device pass lock and remote device locator
Q2: Do users consider security while choosing and downloading applications?
Finding 1: Only a few users scrutinize security messages (38.6%)
Finding 2: Only a few users scrutinize agreement messages (10%)
Finding 3: Users prefer pirated apps (60.7%)
Finding 4: Users disregard security during app selection
Finding 4.1: Only a few users read reviews (10.5%)
Finding 4.2: Only a few users inspect reputation (8.7%)
Finding 4.3: Only a few users consider security or privacy issues (3.5%)
Finding 5: Users who occasionally inspect security messages or ignore them at all are more likely to disable encryption
Finding 6: Users who always inspect security messages are more likely technically and security savvy users
Finding 7: Users who ignore security messages are more likely to also ignore agreement messages
Finding 8: Non-security savvy users are more likely not to inspect app's reputation
Finding 9: Users with usefulness as app selection criterion tend to ignore app reviews and reputation
Finding 10: Users who consider security or privacy issues during app selection are more likely security aware users
Q3: Do smartphone users trust the app repository?
Finding 1: Users believe that downloading apps from the repository is secure (approximately 3/4 users in each platform)
Finding 2: Misconceptions were found regarding app testing in the repository
Finding 2.1: Users were unaware if app testing takes place in the app repository (~half of the sample population)
Finding 2.2: Users expected app testing in repositories that did not use it
Finding 2.3: Users reported that app testing was not applied in 'walled gardened' repositories
Finding 3: Users were found exposed to malicious applications:
Finding 3.1: Users who trust app repository are more likely non security and IT savvy
Finding 3.2: Unaware users of smartphone malware more likely trust the app repository
Finding 3.3: Users who trust the repository tend to be unaware about smartphone secsoft and tend not use them
Finding 4: Users who trust app repository are less likely to always inspect security messages
Finding 5: Users with good IT skills more likely to trust app repository, regardless that they tend to be unaware whether app testing takes place in it
Other findings:
Finding 1: Users who store business data are more likely to store personal data
Finding 2: Users unaware about smartphone malware are more likely to store personal data
Finding 3: Users who store personal data are more likely to trust the app repository and occasionally inspect security messages

platforms, e.g. Windows UAC (Motiee et al., 2010), browser messages that warn about invalid SSL certificates and phishing scams (Egelman et al., 2008; Sunshine et al., 2009). Moreover, users are trained to click through interruptive messages, which appear while they are completing a task (Motiee et al., 2010), as well as tend to ignore consequent warnings, especially the ones which resemble EULAs. This may partially explain why users did not scrutinize security messages that smartphones prompt them. The importance of this vulnerability is increased due to permission overdeclaration of smartphone applications, i.e. developers tend to request access to more permissions than needed for the API they use (Felt et al., 2011a). This permission overdeclaration may be exploited by deputy attacks (Felt et al., 2011c; Grace et al., 2011), or by malicious advertisements (Shekhar et al., 2012) that share the permissions of an application.

Our analysis shows that only technically and security savvy users tend to inspect these messages. This is also important, as smartphone devices are becoming more widespread and thus more users who are not security or technical savvy tend to use them. Moreover, the findings show that user security or privacy (e.g. protection from surveillance, unauthorized access to personal data) is not a criterion during app selection from the official repository.

Smartphone users were found exposed to several security and privacy threats, as the majority of the available security controls were disabled. Moreover, the results suggest that the respondents' technical skills tend to affect the adoption of security controls. In specific users were found to enable encryption, remote data wipe and remote device locator when they tend to be technically savvy. A similar correlation was not found with the use of device password lock. This is expected since this control is available in feature phones and users have been, in general, trained to use simple authentication mechanisms. Nonetheless, our results indicate that users who do not password protect their devices are more likely to be non-security savvy – a correlation which was not found in the other security controls.

Our results indicate that users tend to use the same device both for business and personal purposes. Therefore, the impact

of unauthorized access to stored data is greater. Moreover, the majority of smartphone users did not install any third-party security software on their device. However, these users reported that they facilitate security software in their PCs. This denotes a clear asymmetry in the security awareness of users, between the PC and smartphone platform. Furthermore, a considerable number of users reported that smartphone security software is not essential. This attitude towards secsoft may stem from technological parameters, e.g. battery consumption, overall device performance (i.e. delays), false positives and/or cognitive limitations (i.e. unawareness of existing threats and available controls), and/or psychological parameters, i.e. the trust in the application market itself. However, smartphone security software adds an additional line of defence against malware and can protect non-security savvy users at least from the threat of non-sophisticated attackers (Mylonas et al., 2011a).

Furthermore, the users' belief that applications deriving from the official repository are secure seems to be a severe vulnerability. This is so, as trust in the application repository is more likely to: (a) be encountered in users who are not technology and security savvy, and (b) increase the users' security complacency, as they are more like not to scrutinize security messages and not use secsoft software. Moreover, this trust is not always justified by the availability of security controls in the app repository (e.g. app testing) or their effectiveness regarding the mitigation of malware. Hence, this user security complacency regarding trust in the app repository may be exploited by attackers who use the repository as a malware infection vector. The users who trust the application repository cannot be identified via software solutions (e.g. software agents, MDM). As such, we provide a prediction model that identifies them. The suggested model performs well enough, even if it is evaluated in a different sample than the one used for its design. As the latter sample includes smartphone users from the United Kingdom, this suggests that the proposed model is not affected by the demographics of the Greek sample. The prediction model can be used by an organization to identify users who have the above mentioned security complacency regarding the app repository and engage them in a relevant training awareness program. The prediction model is prone to Type I errors, which may cause frustration to a user who does not trust the app repository (e.g. she may be asked to undergo relevant security training). However, we argue that a significant Type I error coupled with a negligible Type II error is a good security trade-off, as far as smartphone awareness is concerned.

For future work we plan to extend our analysis by comparing the smartphone security awareness in selected user subgroups of our sample. We also plan to further explore and compare the security awareness of users with different demographics.

Acknowledgements

This research has been co-funded by the European Union (European Social Fund – ESF) and Greek national funds through the Operational Program "Education and Life Learning" of the National Strategic Reference Framework (NSRF) – Research Funding Program: Heracleitus II. Investing in knowledge society through the European Social Fund.

The authors would like to express their appreciation to the anonymous reviewers, as well as to Marianthi Theoharidou (AUEB), for their valuable comments and suggestions.

Appendix A

Table A.1 – Variables in the instrument model.

Variable	Variable label	Values	Question
IsAware	Security savvy user	No, yes	Q4
IT	IT expertise	Moderate, good, excellent	Q3
Android	Android OS device	No, yes	Q6
BlackBerry	BlackBerry OS device	No, yes	Q6
Symbian	Symbian OS device	No, yes	Q6
Windows	Windows OS device	No, yes	Q6
iPhone	iOS device	No, yes	Q6
TrustRep	Trust app repository	No, yes	Q9
AppTesting	Application testing in OAR	Don't know, no, yes	Q10
SecMessages	Security messages	Never, sometimes, always	Q11
AgreementMsgs	Agreement messages	Never, sometimes, always	Q12
Privacy	Privacy concern	No, yes	Q13
Jailbreaking	iPhone jailbreaking	No, yes	Q14
Pirated	Prefer pirated apps	No, yes	Q15
MalwareExistence	Smartphone malware existence	No, yes	Q16
Smartphone	Use security software in smartphone	No, yes	Q17
Pc	Use security software in PC	No, yes	Q17
NoUse	No use of security software	No, yes	Q17
SecSoftExistence	Smartphone security software existence	No, yes	Q18
SoftEssential	Smartphone security software essential	No, yes	Q19
Encryption	Use encryption	No, yes	Q20
PassLock	Use device password lock	No, yes	Q20
RemoteWipe	Use remote data wipe	No, yes	Q20
RemoteLoc	Use remote locator	No, yes	Q20
FreeSoft	Searched for free smartphone security software	No, yes	Q21
Personal	Store personal data	No, yes	Q22
Business	Store business data	No, yes	Q23
Misplaced	Device misplaced	No, yes	Q24
Cost	Cost	No, yes	Q8
Developer	Developer	No, yes	Q8
Reviews	Reviews	No, yes	Q8
Reputation	Reputation	No, yes	Q8
Security/Privacy	Security or privacy	No, yes	Q8
Usefulness	Usefulness	No, yes	Q8
Usability	Usability	No, yes	Q8
Efficiency	Efficiency	No, yes	Q8

Appendix B

Athens University of Economics and Business
Information Security and Critical Infrastructure Protection
Research Laboratory

This is a voluntary and anonymous questionnaire. **Please read the following questions and answer as honestly and responsibly as possible**.

Research coordinators: **Alexios Mylonas**, Ph.D. Researcher (amylonas@aueb.gr)
Dimitris Gritzalis, Professor (dgrit@aueb.gr)

1. Sex Male □ Female □

2. Age: _____

3. IT expertise: Excellent □ Good □ Moderate □

4. Have you completed an information security course (undergraduate or postgraduate) or relative certification (e.g. CISSP, CISA)? Yes □ No □

5. I am an owner of a: Smartphone □ PC/Laptop/Netbook □ Other:

6. Which is the operating system of your smartphone?
Android □ BlackBerry □ Symbian □ Windows Mobile □ iPhone iOS □ Other:

7. Do you install new applications on your smartphone? Yes □ No □

8. What are your main criteria in choosing which application to install on your smartphone?
_____

9. Do you consider the applications in the official application repository to be secure for installation on your device? Yes □ No □

10. Have the applications in the official application repository undergone security application testing before you download them on your smartphone?
Yes, they have □ No, they have not □ I do not know □

11. Do you pay attention to security messages appearing during the installation of a new application on your smartphone? Always □ Sometimes □ Never □

12. Do you pay attention to licensing agreements appearing during the installation of a new application on your smartphone? Always □ Sometimes □ Never □

13. Are you concerned about the privacy and protection of your personal data?
Yes □ No □

14. Do you know what the term *iPhone jailbreaking* means? Yes □ No □

15. Do you prefer the installation of a pirated application instead of buying the original application version? Yes □ No □

16. Are you aware of the existence of smartphone malicious software (virus, worm, trojan horse, etc.)? Yes □ No □

17. In which devices do you use security software (e.g. antivirus, firewall, etc.)?
Smartphone □ PC/Laptop/Netbook □ None □ Other:

18. Are you aware of the existence of smartphone security software (e.g. antivirus, firewall, etc.)? Yes □ No □

19. Do you consider smartphone security software essential? Yes □ No □

20. Which of the following smartphone protection mechanisms do you use?
File Encryption □ Device Password Lock □ Remote file deletion □ Device Locator □ None □ Other:

21. Have you ever searched the application repository for free smartphone security software?
Yes □ No □

22. Do you store personal data on your smartphone? Yes □ No □

23. Do you store business data on your smartphone? Yes □ No □

24. Have you ever misplaced your smartphone? Yes □ No □

Thank you for your time and effort.

Appendix C

Table C.1 – Test results for a priori sample size estimation.

Effect size w	Error probability (α)	Power (1-β error probability)	df	Required sample size
0.3	0.05	0.95	1	145

Appendix D

Table D.1 – Distribution of ages in the sample.

Range	% of users
[15–22]	43
[23–30]	38
[31–52]	19

Appendix E. Prediction model estimation and assessment

The logistic regression equation for *TrustRep* is (Hosmer and Lemeshow, 2000):

$$\text{logit}(p) = \log(p/1-p) = b_0 + b_1 * x_1 + \ldots + b_n * x_n \quad \text{(D.1)}$$

where p = Probability (*TrustRep* = 1) and b_i are the factors of a set of independent variables (x_i), which predict the *TrustRep*, given in log-odds units. *TrustRep* is the dependent binary category variable, denoting user trust in the app repository (*TrustRep* = 1) or absence of trust in the app repository (*TrustRep* = 0) respectively. Let:

$$z = b_0 + b_1 * x_1 + \ldots + b_n * x_n \quad \text{(D.2)}$$

then the probability that a user trusts the app repository can be given from equation (D.3).

$$p = \exp(z)/(1+\exp(z)) \quad \text{(D.3)}$$

The prediction model seeks to maximize the probability of equation (D.3). For this reason, we applied stepwise Forward Wald (FW) (Harrell et al., 1984), so as to discover the coefficients (independent variables (x_i)) of the prediction model. FW starts with an empty model, i.e. a model with no prediction coefficients, and iteratively adds coefficients that are statistically significant and maximize statistic scores. Iteration stops when no coefficient can be added.

Table E.1 presents the 7 independent coefficients x_i, selected by FW along with: (a) the condition for $x_i = 1$, (b) the condition enablement frequency in the sample ($n = 458$), (c) their factors (b_i), and (d) their significance level p (i.e. whether the predictor gives a significant contribution in prediction of the dependent variable). For instance, the coefficient x_1 has $b_1 = 1.351$, significance level $p = 0.007$, and, $x_1 = 1$ when the user's IT skills are "Moderate" – there are 47 smartphone users with moderate IT skills in the sample. FW did not give a statistical significant solution with a constant and as a result $b_0 = 0$. In this respect, equation (D.2) is:

$$z = 1.351 * x_1 + 1.092 * x_2 - 1.688 * x_3 + 1.523 * x_4 + 1.314 * x_5 - 0.475 * x_6 - 0.741 * x_7 \quad \text{(D.4)}$$

Table E.1 – *TrustRep* prediction model coefficients.

x_i	x_i condition	Condition frequency	b_i	p
x_1	IT = "Moderate"	47	1.351	0.007
x_2	IT = "Good"	189	1.092	0 < 0.001
x_3	BlackBerry = False	416	−1.688	0 < 0.001
x_4	NoUse = False	425	1.523	0 < 0.001
x_5	P&S = False	442	1.314	0.003
x_6	Pirated = False	180	−0.475	0.047
x_7	Personal = False	111	−0.741	0.005

Regarding the assessment of the model we conducted the essential statistic tests, so as to ensure: (a) the model's statistical significance, (b) the model's goodness-of-fit, and (c) the minimum sample size requirement for logistic regression. Initially, we tested the null hypothesis that the dependent variable, denoting trust in the app repository, and the seven independent variables x_i are unrelated, i.e. the selected predictors variables x_i do not contribute to the prediction of the dependent variable *TrustRep*. We denote with p the probability that the null hypothesis is true and reject it if $p < 0.05$.

Table E.2 describes the results of these tests. We reject the null hypothesis ($p < 0.05$), hence our model is statistically significant. The model has seven degrees of freedom, as each of its coefficients has one degree of freedom. Then the model was tested against the Hosmer–Lemeshow good-of-fit test (Hosmer and Lemeshow, 2000), so as to determine whether the model adequately 'fits', i.e. describes, the data. The significance level p in this test must be greater than 0.05, otherwise the model does not adequately fit the data. Also, we checked the ability of the model's independent variables to explain the variance of the dependent variable (Nagelkerke, 1991). For this test, the Nagelkerke R^2 value must be in the range [0,1]. Table E.3 shows that the model was successfully tested against these two tests. Finally, Harrell et al. (1984) argue that for a valid logistic regression model at minimum the same size must have 10 cases for each candidate predictor

Table E.2 – Model's statistical significance

x^2	df	p
193.272	7	<0.001

Table E.3 – Results of Hosmer–Lemeshow and Nagelkerke tests

Statistical test	Value
Hosmer and Lemeshow	0.828
Nagelkerke R^2	0.459

(variable) of the instrument. Our sample size meets this requirement, as there exist 36 candidate predictors ($n = 458$, i.e. $>10^{*}36 = 360$).

REFERENCES

Amini S, Lin J, Hong J, Lindqvist J, Zhang J. Towards scalable evaluation of mobile applications through crowdsourcing and automation. Technical Report CMU-CyLab-12-006. Carnegie Mellon University; 2012.

Anderson J, Bonneau J, Stajano F. Inglorious installers: security in the application marketplace. In: Proc. of the 9th workshop on the economics of information security (WEIS'10); 2010.

Androulidakis I, Kandus G. A survey on saving personal data in the mobile phone. In: Pernul G, et al., editors. Proc. of 6th international conference on availability, reliability and security (ARES-2011). Austria; 2011a. p. 633–8.

Androulidakis I, Kandus G. Mobile phone downloading among students: the status and its effect on security. In: Proc. of 10th international conference on mobile business (ICMB-2011). Italy; 2011b. p. 235–42.

Androulidakis I, Kandus G. Feeling secure vs. being secure, the mobile phone user case. In: Proc. of 7th international conference in global security, safety and sustainability (ICGS3-2011). Greece: Springer; 2012. p. 212–9. LNCS-99.

Aviv A, Gibson K, Mossop E, Blaze M, Smith J. Smudge attacks on smartphone touch screens. In: Proc. of the 4th USENIX conference on offensive technologies. USENIX Association; 2010. p. 1–7.

Baghdassarian S, Milanesi C. Forecast: mobile application stores, world-wide, 2008-14. Tec Report G00209676. Gartner; 2010.

Barrera D, Van Oorschot P. Secure software installation on smartphones. IEEE Security and Privacy 2011;9(3):42–8.

Böhme R, Köpsell S. Trained to accept?: a field experiment on consent dialogs. In: Proc. of the 28th international conference on human factors in computing systems (CHI '10). USA: ACM; 2010. p. 2403–6.

BSA. BSA global software piracy study: shadow market. Technical Report. 9th ed. Business Software Alliance; 2012.

Chia P, Yamamoto Y, Asokan N. Is this app safe? A large scale study on application permissions and risk signals. In: Proc. of the 21st international world wide web conference; 2012.

Cohen F. Computational aspects of computer viruses. Computers & Security 1989;8(4):297–8.

Cohen J. Statistical power analysis for the behavioral sciences. Lawrence Erlbaum; 1988.

Communications Committee. Broadband access in the EU: situation at 1 July 2011. Technical Report COCOM11-24; 2011.

Coursen S. The future of mobile malware. Network Security 2007; 2007(8):7–11.

Cronbach L. Coefficient alpha and the internal structure of tests. Psychometrika 1951;16:297–334.

Egele M, Kruegel C, Kirda E, Vigna G. Pios: detecting privacy leaks in iOS applications. In: Proc. of the net and distributed system security symposium (NDSS-2011); 2011.

Egelman S, Cranor L, Hong J. You've been warned: an empirical study of the effectiveness of web browser phishing warnings. In: Proc. of the 26th annual SIGCHI conference on human factors in computing systems (CHI '08). New York, USA: ACM; 2008. p. 1065–74.

Enck W, Ongtang M, McDaniel P. On lightweight mobile phone application certification. In: Proc. of the 16th ACM conference on computer and communications security (CCS '09). USA: ACM; 2009. p. 235–45.

Enck W, Gilbert P, Chun B, Cox L, Jung J, McDaniel P, et al. Taintdroid: an information-flow tracking system for realtime privacy monitoring on smartphones. In: Proc. of the 9th USENIX symposium on operating systems design and implementation (OSDI'10). USENIX Association; 2010. p. 1–6.

Enck W, Octeau D, McDaniel P, Chaudhuri S. A study of android application security. In: Proc. of the 20th USENIX conference on security. USA: USENIX Association; 2011.

Felt A, Chin E, Hanna S, Song D, Wagner D. Android permissions demystified. In: Proc. of the 18th ACM conference on computer and communications security (CCS '11). USA: ACM; 2011a. p. 627–38.

Felt A, Finifter M, Chin E, Hanna S, Wagner D. A survey of mobile malware in the wild. In: Proc. of the 1st ACM workshop on security and privacy in smartphones and mobile devices (SPSM '11). ACM; 2011b. p. 3–14.

Felt A, Wang H, Moshchuk A, Hanna S, Chin E, Greenwood K, Wagner D, et al. Permission re-delegation: attacks and defenses. In: Proc. of the 20th USENIX security symposium, USA; 2011c.

Felt A, Ha E, Egelman S, Haney A, Chin E, Wagner D. Android permissions: user attention, comprehension, and behavior. Technical Report UCB/EECS-2012-26; UC Berkeley; 2012.

Flick U. An introduction to qualitative research. Sage Publications; 1998.

Furnell S. Why users cannot use security. Computers & Security 2005;24(4):274–9.

Furnell S. Making security usable: are things improving? Computers & Security 2007;26(6):434–43.

Furnell S, Jusoh A, Katsabas D. The challenges of understanding and using security: a survey of end-users. Computers & Security 2006; 25(1):27–35.

Gilbert P, Chun BG, Cox LP, Jung J. Vision: automated security validation of mobile apps at app market. In: Proc. of the 2nd international workshop on mobile cloud computing and services (MCS '11). USA: ACM; 2011. p. 21–6.

Grace M, Zhou Y, Wang Z, Jiang X. Detecting capability leaks in android-based smartphones. Technical Report. North Carolina State University; 2011.

Harrell F, Lee K, Califf R, Pryor D, Rosati R. Regression modeling strategies for improved prognostic prediction. Statistics in Medicine 1984;3(2):143–52.

Hornyack P, Han S, Jung J, Schechter S, Wetherall D. These aren't the droids you're looking for: retrofitting android to protect data from imperious applications. In: Proc. of the 18th ACM conference on computer and communications security (CCS '11). USA: ACM; 2011. p. 639–52.

Hosmer D, Lemeshow S. Applied logistic regression, vol. 354. Wiley-Interscience; 2000.

Howell J, Schechter S. What you see is what they get. In: Proc. of the IEEE workshop on web 2.0 security and privacy; 2010.

Kelley P, Consolvo S, Cranor L, Jung J, Sadeh N, Wetherall D. A conundrum of permissions: installing applications on an android smartphone. In: Workshop on usable security (USEC-2012); 2012.

Lookout. Security alert: droiddream malware found in official android market, http://blog.mylookout.com/blog/2011/03/01/security-alert-malware-found-in-official-android-market-droiddream/; [accessed April 2012].

Marmol F, Perez G. Security threats scenarios in trust and reputation models for distributed systems. Computers & Security 2009;28(7): 545–56.

McDaniel P, Enck W. Not so great expectations: why application markets haven't failed security. IEEE Security and Privacy (SP) 2010;8(5):76–8.

McKnight DH, Chervany NL. The meanings of trust. Technical Report WP9604. University of Minnesota Information Systems Research Center; 1996.

Motiee S, Hawkey K, Beznosov K. Do windows users follow the principle of least privilege?: investigating user account control practices. In: Proc. of the 6th symposium on usable privacy and Security (SOUPS '10). USA: ACM; 2010. p. 1–13.

Mylonas A, Dritsas S, Tsoumas B, Gritzalis D. Smartphone security evaluation – the malware attack case. In: Samarati P, Lopez J, editors. Proc. of the international conference on security and cryptography (SECRYPT'11). Spain: SciTePress; 2011a. p. 25–36.
Mylonas A, Tsoumas B, Dritsas S, Gritzalis D. A secure smartphone applications roll-out scheme. In: Proc. of the 8th international conference on trust, privacy & security in digital business (TRUST-2011). France: Springer; 2011b. p. 49–61. LNCS-6863.
Nadji Y, Giffn J, Traynor P. Automated remote repair for mobile malware. In: Proc. of the 27th annual computer security applications conference (ACSAC'11). USA: ACM; 2011. p. 413–22.
Nagelkerke N. A note on a general definition of the coefficient of determination. Biometrika 1991;78(3):691–2.
Nauman M, Khan S, Zhang X. Apex: extending android permission model and enforcement with user-defined runtime constraints. In: Proc. of the 5th ACM symposium on information, computer and communications security (CCS'10). USA: ACM; 2010. p. 328–32.
Niemela J. Just because it's signed doesn't mean it isn't spying on you, http://www.f-secure.com/weblog/archives/00001190.html; [accessed April 2012].
Quirolgico S, Voas J, Kuhn R. Vetting mobile apps. IT Professional 2011;13:9–11.
Redman P, Girard J, Wallin L. Magic quadrant for mobile device management software. Technical Report G00211101. Gartner; 2011.
Shekhar S, Dietz M, Wallach D. AdSplit: separating smartphone advertising from applications. Technical Report; 2012.
Sunshine J, Egelman S, Almuhimedi H, Atri N, Cranor LF. Crying wolf: an empirical study of ssl warning effectiveness. In: Proc. of the 18th USENIX security symposium (SSYM '09). USA: USE Association; 2009. p. 399–416.
Theoharidou M, Gritzalis D. A Common Body of Knowledge for Information Security. IEEE Security & Privacy 2007;5(2):64–7.
Theoharidou M, Mylonas A, Gritzalis D. A risk assessment method for smartphones (AICT 376). In: Proc. of the 27th IFIP information security and privacy conference. Springer; 2012. p. 443–56.
Thomson M, von Solms R. Information security awareness: educating your users effectively. Information Management & Computer Security 1998;6(4):167–73.
Whitten A, Tygar J. Why johnny can't encrypt: a usability evaluation of PGP 5.0. In: Proc. of the 8th USENIX security symposium. USA; 1999.
Zhou W, Zhou Y, Jiang X, Ning P. Detecting repackaged smartphone applications in third-party android marketplaces. In: Proc. of 2nd ACM conference on data and application security and privacy (CODASPY-2012). ACM; 2012a. p. 317–26.
Zhou Y, Jiang X. Dissecting android malware: characterization and evolution. In: Proc. of the IEEE symposium on security and privacy (SP). IEEE; 2012. p. 95–109.
Zhou Y, Wang Z, Zhou W, Jiang X. Hey, you, get off of my market: detecting malicious apps in official and alternative android markets. In: Proc. of the 19th network and distributed system security symposium (NDSS '12). USA; 2012b;.

Alexios Mylonas is a Researcher and a Ph.D. candidate with the Information Security and Critical Infrastructure Protection Research Laboratory of the Dept. of Informatics, Athens University of Economics and Business (AUEB), Greece. He received his B.Sc. in Informatics (Athens Univ. of Economics and Business) and his M.Sc. in Information Security (Royal Holloway, Univ. of London). His current research interests include smartphone security, digital forensics, insider threat, and risk assessment.

Anastasia Kastania received a B.Sc. in Mathematics and a Ph.D. in Medical Informatics from the University of Athens, Greece. She is an Adjunct Lecturer with the Dept. of Informatics of Athens University of Economics and Business, Greece. She maintains a long lasting record as a researcher in national and EU-funded projects. Her current research interests include telemedicine and e-health, e-learning, bioinformatics, mathematical modelling and statistics, and web and reliability engineering.

Dimitris Gritzalis is a Professor of ICT Security and the Director of the Information Security and Critical Infrastructure Protection Research Laboratory, with the Dept. of Informatics of Athens University of Economics and Business (AUEB), Greece. He holds a B.Sc. (Mathematics, Univ. of Patras), a M.Sc. (Computer Science, City University of New York) and a Ph.D. (Critical Information Systems Security, Univ. of the Aegean). His current research interests focus on VoIP systems security, digital forensics and critical infrastructure protection. He has served as Associate Commissioner of the Greek Data Protection Com and as the President of the Greek Computer Society. Prof. Gritzalis is the Academic Editor of the Computers & Security Journal.