A RISK ASSESSMENT STUDY OF CHILDREN PRIVACY OF THE MOBILE APPs

Mazen Alghamdi [1], Ron Ruhl [2], Sergey Butakov [3], Dale Lindskog [4]

Master of Information Systems Security Management, Concordia University College of Alberta
Edmonton, Alberta, Canada

mazenalghamdi10@gmail.com, ron.ruhl, Dale.Lindskog, sergey.butakov @concordia.ab.ca

***Abstract*— While privacy issues in children's mobile applications are becoming increasingly sophisticated, the permissions in these applications are undergoing development without proper security implementation. Mobile applications with excessive privacy features can cause serious data leakages and other similar privacy issues, which can exert detrimental repercussions on children's security. In this paper, we have conducted a risk assessment for this security matter, where we identified the risk, applied the controls, determined the residual risk, and established a set of recommendations.**

***Keywords*— Children, Mobile Applications, Permissions, Privacy, Risk Assessment**

I. INTRODUCTION

In June, 2013, a study of the "Top Smartphone Platforms" showed that Android platform (Google) "remains on top as of April 2013"[25]. In fact, the Android Google Market offers many free apps that mobile users can install.

In 2011, Felt, A. et al.[26] stated that 93% of the Android's free apps have "dangerous" permissions[26]. For example, the SEND_SMS permission represents one of the "dangerous" permissions that can be managed by an application [26].

Ideally, an Android's platform informs the users about the set of permissions that are required for an installed app. However, an app can contain extra privileges and mobile users may install the app regardless of the risk.

Felt et al. conducted a study [27] with many related concerns, posing the question: "do users care about their phones' privacy and security?" They surveyed 308 Android mobile users, and only "17% paid attentions to permissions at install-time", and "42% were completely unaware of permissions"[27].

In the Android platform, shown in Figure 1 of Appendix A, a series of mandatory processes grant permission to an app. This architecture reveals that the interaction between the system and the app occurs between two virtual machines (VMs). The app VM includes the application process where APIs undergo invocation from the API library. Subsequently, a private interface RPC stub becomes invoked in order to request an operation from the system process in the other VM. In order to conduct an operation in the system process, a permission set of validations and checks occur based on the Android permission validation mechanism. For instance, an application invokes API to call ClipboardManager.getText. Afterwards, the RPC interface will act as a proxy and the API call will be relayed as IClipboard$Stub$Proxy to request the ClipboardService of the system process [27].

Other permissions, such as the WRITE_EXTERNAL_STORAGE and BLUETOOTH permissions, are "enforced by Unix groups, rather than the Android permission validation mechanism"[27]. In addition, the "signature" permissions related to the backup process and delete application packages are "granted only to applications that are signed with the device manufacturer's [27].

In some cases, an application's developer may use unnecessary permissions, which result in the applications requesting privileged access that is unrelated to the purpose of the developed application, such as the permissions ACCESS_GPS or ACCESS_LOCATION[27]. For example, a developer may use the ACCESS_MOCK_LOCATION during the test and then forget to remove it from the application. The permission ACCESS_MOCK_LOCATION can allow an application to "create mock location providers for testing"[21]; such permissions can comprise a technique in the privacy monitoring tools that aim to prevent data leakage of the location. The AndroidMainfest permission [21] includes a reasonable description of the permissions that can be used by privacy monitoring tool developers.

II. CONCERN FOR PRIVACY

Nowadays, most children use mobile devices such as Smartphone to download many applications that may cause serious invasions of the children's privacy. Several studies revealed the unfortunate fact that private data leaks from the children's devices as a result of mobile apps [6][5][4][14].

The Federal Trade Commission (FTC)[5] expresses concern about the criteria involved in collecting and sharing children's information, such as emails, addresses, and phone numbers[5]. This research also considers other criteria, such as device IDs, location, videos, and photos (Table 1).

This investigation addresses the aspects and challenges of protecting children's privacy in the mobile environment. The main focus in this research aims to develop testing criteria that have been endorsed by the Children's Online Privacy Protection Act (COPPA). Samples of the testing criteria are located in Table 1 of Section 6.

In this research, we have conducted a risk assessment, where we have identified the risk of data leakage by the apps, defined and applied the controls of privacy monitoring tools, determined the residual risk of data leakage after applying the monitoring tools, and provided a set of recommendations.

The risk assessment has been conducted based on the NIST special publication 800-30 Risk Assessment Model (Figure 2) and followed by a set of recommendations, some of which involve using the monitoring tools.

Some privacy monitoring tools claim to provide users with enhancements. Many of these applications can be considered as "a good little helper"[18] that only offer "a quick overview

of all the installed apps and their permissions"[18]. Other tools [19] possess more features, such as the ability to scan mobile apps for problems known as “dirty secrets” and send the results to the user's (parent) e-mail. These tools [19] assert that they can also "uninstall the unwanted app"[19]. Mobile users can pay for similar applications [20] and obtain additional enhancement features, such as "install and backup fixed apps"[20]. Anti-virus companies also provide some of these privacy monitoring tools to enhance and correct privacy issues. For instance, Trend Micro offers a free tool called "Privacy Scanner for Facebook" [21], which has been designed to increase privacy issues surrounding the use of Facebook. This tool scans Facebook settings, identifies any risks in the settings, recommends changes, and verifies the implementation of such changes [21]. The numbers and the names of the selected privacy monitoring tools are subject to change throughout the research.

In his discussion on "Conducting a Risk Assessment for Mobile Devices,” David Frei [28] listed the "available Industry Risk Assessment Models". The research in this paper was conducted on the basis of one of these models, the NIST SP 800-30. (Fig.2)

III. Related Work

Adam et al. [24] investigated aspects of COPPA in the website environment. Their research focused on some criteria

background information for policy parameters, Yong [17] consulted COPPA. As result, the research recommended "a policy tool in combination with a better interface in mobile platforms"[17]. Accordingly, the present investigation establishes the need for a mobile privacy risk assessment as an important step for creating an efficient policy tool.

While the three previous researches [24] [16] [17]e studied COPPA’s implications on websites, website advertisements and mobile advertising, the current research focuses on the COPPA’s relevance to mobile apps for the purposes of developing testing criteria to evaluate privacy monitoring apps. In fact, this research supports the study of COPPA and the analysis of the mobile privacy issues by providing a set of recommendations through an NIST risk assessment.

dictated problems with privacy issues, none of these investigations have conducted a risk assessment. For instance, an article in the Business Insider website [4] discusses similar experiments, conducted by WSJ, in which 101 apps were examined in different mobile operating systems. The outcome of this study showed that some apps transmitted "tons of data back to the maker or to ad networks, not necessarily with the user's consent, or knowledge"[4].

user. These researchers tested "100 randomly picked applications" and found that 18 apps possessed "dubious" permissions [14].

While [14] [5], and [4] conducted experiments with apps to uncover evidence of data leakage, these authors neglected to conduct a risk assessment. Thus, these authors [14] [5], and [4] identified the risk of data leakage without providing a risk assessment. In contrast, the current study identifies the risk of data leakage, applies the controls of privacy monitoring tools, determines the residual risk, and provides a set of recommendations.

from COPPA that a "web site operator" should consider. For instance, a "web site operator" should obtain "parental consent before collecting protected information from children"[24].

The current investigation uses a risk assessment methodology: NIST SP 800-30. In order to avoid conflict, the first phase of this research develops the criteria by extracting it from COPPA; Section 6 provides additional discussions of this methodology.

While Adam et al. [24] studied privacy aspects of COPPA, Xiaomei et al. [16] investigated COPPA’s compliance in the environment of children’s websites. In particular, these authors "examined advertisements placed on popular children’s websites"[16]. The study found that some children lacked the ability to distinguish ads from other content in the targeted websites. Consequently, the "majority of children’s websites" featured ads that attracted children. The study found that only 47% of children’s websites were compliant with COPPA "when they collected personal information from children"[16]. Similar to [24] and [16], our research examines privacy aspects impacting children; however, this investigation focuses on the mobile apps environment rather than that of websites.

While Xiaomei et al. [16] studied advertisements’ compliance with COPPA, Yong [17] focused on mobile advertising. Specifically, Yong’s research posed the question: "what is the current policy parameter in regards to the protection of personal data in complex mobile-based digital ecosystems?"[17]. In order to provide b

This research aims to develop testing criteria that have been endorsed by the Children's Online Privacy Protection Act (COPPA). Samples of the testing criteria can be found in Table 1 of Section 6.

A paper entitled "Privacy Enhancing Technology Guidelines and Testing Methodology" attested to the lack of an "established set of criteria for which users can seek to evaluate privacy enhancing technologies"[13]. Specifically, the paper discussed many concerns about developing criteria in order to test the privacy enhancing tools. This research studied COPPA in order to gather and utilize privacy criteria in the risk assessment.

While some studies have in

Enck et al. [12] tested "30 popular third-party Android applications", finding "68 instances of potential misuse of users’ private information across 20 applications"[12]

Similarly, Berry Hoekstra Damir[14] similarly stated that some mobile apps leak data that are related to the privacy of the mobile

The present research was conducted similarly to Theoharidou et al. [11], who created a "Risk Assessment Method for Smartphones" and requested future research to extend the review of the mobile threats "along with an analytical dictionary of permission combinations"[23]. This research conducted a comprehensive study of application permissions and other related issues that have a significant impact on children's mobile privacy.

Jonas and Ty [15] have proposed a different approach that modifies Android applications such as Whatsapp and Angry Birds prior to installation. These researchers incorporated many steps into their method, such as modifying the "application code to make sure it doesn't crash because of permission issues"[15].

However, these authors neglected to test reviewer claims about their approach, such as "the manual removal of permissions on several applications." In fact, other reviewers, such as gizmodo.com, commented on the Privacy Blocker App. Specifically, Gizmodo.com stated that this tool "can confuse malicious Android apps with Garbage Code" [15]. Thus, these authors neglected to conduct an investigation of the mobile privacy monitoring tools, showing that research studies have not yet provided evidences of these tools' effectiveness.

Siew Yong et al [1] presented a paper about the security issues of the "mobile Wi-Fi robot toys," which should represent a major concern for parents whose children use such a mobile device. This research implemented a scenario-based methodology and a set of recommendations, such as the idea that "parents should be able to create a list of friends that their children can correspond with"[1].

The present study utilizes such an approach with sets of scenarios and recommendations, which are detailed in the discussion section, Section VI.

IV. Methodology

The testing environment for this research included several children's apps and privacy monitoring tools installed in a mobile device, Galaxy Nexus S, with an Android OS (4.0.4). However, a SIM card was not used. Applications were downloaded from the Android "Play Store," which required a Google account.

In order to conduct the risk assessment, we developed the test criteria from a legal privacy perspective, the COPPA Act [3]. Specifically, we considered the COPPA criteria listed in Table 1.

The COPPA Act. Criteria
First & last name
A home or other physical address
Online contact information
A screen or user name that functions as online contact information
A telephone number
Identifier that can be used to recognize a user over time and across different Web sites or online services
A photograph, video, or audio file, where such file contains a child's image or voice
Geo-location (GPS)

Table 1: The COPPA Act. Criteria

During the related review, we found that NIST SP 800-30 can address the integration of the risk assessment using privacy constraints on Android mobile apps.

In the first step of the NIST SP 800-30 model, we examined the Android system characterization, which includes permission processes in the Android platform as discussed in introduction of this research (Fig 1).

The second step focused on identifying the threat, especially the data leakage by mobile apps. We sought to determine the way in which threat agencies such as malware developers can exploit the vulnerabilities inherent in mobile apps.

In the third step of the methodology, this research identified the vulnerability while installing several children's apps on the test phone.

	Application Name
A	BomberMan
B	Fruit Ninja
C	Street Fighter V
D	Jewel Pop Mania
E	Street Fighter Zero 2
F	Ultimate Mortal Kompact

Table 2: List of the Children's Applications

Subsequently, several monitoring tools were installed on the test device in order to identify controls for the fourth step of NIST SP 800-30. These tools were selected based on the rating average of the Google Play Store application and website [22].

We conducted a thorough study of the current controls in the form of monitoring apps and the nature of the vulnerabilities in terms of the privacy leakage caused by the children's apps. This investigation enabled us to determine the likelihood of this type of risk, as shown in Table 5. This part of the study incorporated the fifth and sixth steps of the NIST SP 800-30.

Based on the likelihood and impact of the risk, the seventh step in this research determined the level of risk. For example, the data leakage of geographic location and the leakage of text messages constitute different levels of risk; such text messages may include banking credential data or other sensitive information.

In the eighth step, the research established a set of recommendations to avoid any data leakage caused by children's mobile apps. These recommendations were included in the last step of the research. This step, risk assessment, represented the main contribution of this research.

The selected COPPA criteria were used in the risk assessment model in order to identify the vulnerabilities in the mobile privacy monitoring tools (step 3 NIST 800-30).

The research included several scenarios and preliminary hypotheses:

Scenario 1: This scenario features a lack of privacy monitoring tools. While the child is downloading apps, the apps are leaking private data from the mobile device. Many researchers have confirmed that some apps leak private data (Section III). However, rather than verifying these studies, this research focuses on selecting some of the suspected apps that are listed in the body of the research review section in order to use this information in the other two scenarios.

Scenario 2: Although privacy monitoring tools are established, the suspected apps are attempting to leak private data out of the mobile device. Based on the analysis of network traffic, the captured packet includes some private data, as shown by the COPPA criteria in Table 1.

Scenario 3: Despite the established privacy monitoring tools, the suspected apps are trying to leak private data from the mobile device. Based on the analysis of network traffic, the captured packet does not include any private data, as shown in the COPPA criteria in Table 1.

V. Experiment Results

Our experiments examined the AndroidMainfest.xml file of the targeted apps prior to and after installing the monitoring apps. We have also investigated the network traffic using Wireshark to verify the results. Consequently, we found that some children's apps used permissions that contained excessive privileges. For instance, these apps were using the "Android.permision.ACCESS_COARSE_LOCATION," which enables the app's owner to determine the location of the mobile device. In addition, some apps contained "Android.permision.WRITE_EXTERNAL_STORAGE," which provided access to the SDcard's data, including the photos and videos if these were present. In addition to these permissions, the children's apps used "Android.permision.READ_PHONE_STATE," in order to retrieve data about the mobile device, including the hardware configuration and additional information about the device components (Fig.3).

Children's Mobile Apps VS. Criteria	A	B	C	D	E	F
First & last name	✓	✓	✓	✓	✓	✓
A home or other physical address	✓	✓	✓	✓	✓	✓
Contact information	✓	X	✓	X	X	✓
A screen or user name that functions as online contact information	X	X	X	X	X	X
A telephone number	✓	✓	✓	✓	✓	✓
Identifier that can be used to recognize a user over time and across different Web sites or online services	✓	✓	✓	✓	✓	✓
A photograph, video, or audio file, where such file contains a child's image or voice	✓	✓	✓	✓	✓	✓
Geo-location (GPS)	✓	✓	✓	✓	✓	✓

Table 3: Children's Mobile Applications vs. Criteria

Subsequently, we applied the first monitoring tool, "Permission Manager," to the children's apps. We found that this monitoring app effectively removed such dangerous permissions (Fig.). In particular, this monitoring app used "comilion.protected.ACCESS_COARSE_LOCATION" to protect the location of the mobile device. The tool also utilized "comilion.protected.WRITE_EXTERNAL_STORAGE" to prevent access to the SD card's data and "comilion.protected.READ_PHONE_STATE" to block access to the mobile device information, including hardware configuration.

However, the second monitoring tool, "Advanced Permission" proved ineffective, as its application failed to modify the mainfest.xml file in order to protect the permissions.

The third monitoring tool, "Privacy Protector" demonstrated effectiveness in one case, which involved disabling the access to the mobile device location .

Mobile privacy monitoring tools VS. Criteria	Permission Manager	Advanced Permission	Privacy Protector
First & last name	✓	X	X
A home or other physical address	✓	X	X
Online contact information	✓	X	X
A screen or user name that functions as online contact information	✓	X	X
A telephone number	✓	X	X
Identifier that can be used to recognize a user over time and across different Web sites or online services	✓	X	X
A photograph, video, or audio file, where such file contains a child's image or voice	✓	X	X
Geo-location (GPS)	✓	X	✓

Table 4: Monitoring tools VS. Criteria

The following section discusses the experiment results and aligns them to the NIST framework, while the final section provides a list of recommendations.

VI. Discussions

The experiment results in the previous section demonstrated the existence of several vulnerabilities in children's apps, such as the data leakages of mobile numbers, names, locations, and hardware configurations. Such vulnerabilities can occur through many sources, including malware writers. The NIST framework has emphasized the importance of the vulnerability assessment in order to use it in the control analysis. A proper vulnerability assessment constitutes a necessary step for developing an effective monitoring tool. For instance, the second monitoring app has proven to lack effectiveness due a poor understanding of the vulnerability and disability for protecting the data leakage. Based on the NIST framework, we have developed a risk probability table in order to demonstrate the rating of each risk:

Likelihood	Description
High	The capability of data leakage by the kid's apps is significant, and the probability of confidential exploitation are sufficient without using a monitoring tool.
Medium	The capability of the data leakage by the kid's apps is medium, and the probability of confidential exploitation are sufficient without using a proper monitoring tool.
Low	The capability of data leakage by the kid's apps is limited, and controls are in place that effectively reduces the probability of vulnerability exploitation.

Table 5: The Likelihood of Risk

VII. RECOMMENDATIONS

The NIST framework discusses the risk impact analysis. In the case of this study, the main impact of such risk involves losing confidential data. Consequently, the results of this research recommend the use of proper monitoring tools with children's apps, which may otherwise negatively impact the children's confidentiality in the Android mobile environment.

	Impact		
Likelihood	NM	IM	SM
High	High	High	Low
Medium	High	High	Low
Low	Medium	Medium	Low

Table 6: Risk Level Matrix
*NM=No Monitoring tool in place.
*IM= Insufficient Monitoring tool in place.
*SM=Sufficient Monitoring tool in place.

In addition, this study advises parents to use a strong password for the assigned Google account in order to prevent children from installing any malicious apps. At the same time, however, we argue that any app with permissions for the user's sensitive information should request the user's age and reject the app installation in the case that children are younger than 12 years old.

We also recommend strengthening the wireless network to prevent any malicious activity in the network traffic, which may include confidential data. We propose that Android systems should require the app developers to implement a proper encryption in order to protect the data transmitted between the children's app and the app's owner. Finally, we recommend that parents monitor their children's mobile device and continually examine the downloaded apps for possible security threats.

VIII. CONCLUSION AND FUTURE WORK

The research has discovered the unfortunate reality that the permissions in children's apps are developed without proper security implementations. Accordingly, we collected the evidence by investigating AndroidMainfest.xml and the network traffic using Wireshark. We then conducted a risk assessment for this security matter, where we identified the risk, applied the controls in the form of permissions monitoring tools, and listed a set of recommendations. Based on this investigation, we perceive a future opportunity for a risk assessment to investigate the way in which malware activities can exploit the permission vulnerabilities that we found in this research. Since we discovered that some monitoring tools can demonstrate effectiveness in preventing data leakage, another promising research area involves investigating the effectiveness of the monitoring tools in an environment where malware targets mobile permissions.

IX. ACKNOWLEDGMENT

My deepest gratitude and appreciation goes to my advisor, Prof. Ron Ruhl, for his support and guidance throughout the research. His continued mentoring led me in the right direction. I would like to extend my appreciation to the committee members, Dr. Butakov, and Dr. Lindskog. Finally, I want to express sincere gratitude to my life partner, my wife, as without her encouragement, I would not have the opportunity to complete this research project.

X. REFERENCES

[1] Siew Yong; Lindskog, D.; Ruhl, R.; Zavarsky, P., "Risk Mitigation Strategies for Mobile Wi-Fi Robot Toys from Online Pedophiles," Privacy, security, risk and trust (passat), 2011 ieee third international conference on and 2011 ieee third international conference on social computing (socialcom) , vol., no., pp.1220,1223, 9-11 Oct. 2011

[2] Emre Yildirim. "Mobile Privacy: Is There An App For That? On smart mobile devices, apps and data protection", M.S. thesis, Institute for Information Law (IViR) University of Amsterdam, Amsterdam Available [Online] http://www.cs.uml.edu/~glchen/papers/privacy-tsp08.pdf

[3] [26] COPPA - Children's Online Privacy Protection Act, SEC. 1302. DEFINITIONS. http://www.coppa.org/coppa.htm

[4] PASCAL-EMMANUEL GOBRY, "Mobile Apps Breach Your Privacy!, Says WSJ" http://www.businessinsider.com/mobile-apps-breach-your-privacy-says-wsj-2010-12 DEC. 18, 2010,

[5] Federal Trade Commission (FTC) Staff Report , The. " Mobile Apps for Kids: Disclosures Still Not Making the Grade" Available [Online] http://www.ftc.gov/os/2012/12/121210mobilekidssappreport.pdf , December 2012

[6] SCOTT THURM and YUKARI IWATANI KANE , "WSJ, Your Apps Are Watching You", Available [Online] http://online.wsj.com/article/SB10001424052748704694004576020083703574602.html, December 17, 2010

[7] The Office of the Privacy Commissioner of Canada, Available [Online] http://www.priv.gc.ca/index_e.asp

[8] The Office of the Information and Privacy Commissioner of Alberta, http://www.oipc.ab.ca/pages/home/default.aspx

[9] Alastair R. Beresford, Andrew Rice, Nicholas Skehin, "MockDroid: trading privacy for application functionality on smartphones", Available [Online] http://www.cl.cam.ac.uk/~acr31/pubs/beresford-mockdroid.pdf

[10] The Open Web Application Security Project (OWASP) Top Ten risks [Online] http://owasptop10.googlecode.com/files/OWASP%20Top%2010%20-%202013.pdf

[11] Yajin Zhou, Xinwen Zhang, Xuxian Jiang, and Vincent W. Freeh, "Taming Information-Stealing Smartphone Applications (on Android", Department of Computer Science, NC State University Huawei America Research Center, Available [Online] http://www.cs.ncsu.edu/faculty/jiang/pubs/TRUST11.pdf

[12] William Enck, Peter Gilbert, Byung-Gon Chun, Landon P. Cox, Jaeyeon Jung, Patrick McDaniel, Anmol N. Sheth, "TaintDroid: An Information-Flow Tracking System for Realtime Privacy Monitoring on Smartphones", In: 9th USENIX Symposium on Operating Systems Design and Implementation. (2010), [Online] http://static.usenix.org/events/osdi10/tech/full_papers/Enck.pdf

[13] Tara M. Swaminatha, "Privacy Enhancing Technology Guidelines and Testing Methodology" Available [Online] http://www.w3.org/2001/01/qa-ws/pp/tara-swaminatha-cigital.html

[14] Berry Hoekstra Damir Musulin, "Privacy consequences of using Android apps", University of Amsterdam System and Network Engineering, https://www.os3.nl/_media/2010-2011/students/jochem_van_kerkwijk/ssn/report_priv_cons_using_android_apps.pdf

[15] Jonas Helfer, Ty Lin, "Giving the User Control over Android Permissions", December 15, 2012 [Online] http://css.csail.mit.edu/6.858/2012/projects/helfer-ty12.pdf

[16] Xiaomei Cai, Xiaoquan Zhao. "Online advertising on popular children's websites: Structural features and privacy issues. Department of Communication", George Mason University, VA, United States

[17] Yong Jin Park. "Mapping User-Privacy Environments in Mobile Based Platforms". University of Michigan, Howard University, [Online] http://papers.ssrn.com/sol3/papers.cfm?abstract_id=2207757

[18] Google Play Store, Privacy Monitor think Android https://play.google.com/store/apps/details?id=com.think_android.securitymonitor&hl=en

[19] Google Play Store, Privacy Inspector xeudoxus https://play.google.com/store/apps/details?id=com.xeudoxusprivacy.inspector

[20] Google Play Store, Privacy Blocker xeudoxus https://play.google.com/store/apps/details?id=com.xeudoxus.privacy.blocker&feature=more_from_developer

[21] Mainfest.permission, http://developer.android.com/reference/android/Manifest.permission.html

[22] Google Play Store application and website https://play.google.com/store

[23] Marianthi Theoharidou, Alexios Mylonas, Dimitris Gritzalis, A Risk Assessment Method for Smartphones, 27th IFIP TC 11 Information Security and Privacy Conference, SEC 2012, Heraklion, Crete, Greece, June 4-6, 2012. Proceedings, pp 443-456, 2012

[24] Barth, A.; Datta, A.; Mitchell, J.C.; Nissenbaum, H., "Privacy and contextual integrity: framework and applications," Security and Privacy, 2006 IEEE Symposium on , vol., no., pp.15 pp.,198, 21-24 May 2006 doi: 10.1109/SP.2006.32

[25] Andrew Martonik "Android US market share dips slightly, remains on top as of April 2013" Jun 04 2013, www.androidcentral.com/android-us-market-share-dips-slightly-remains-top-april-2013

[26] Adrienne Porter Felt, Erika Chin, Steve Hanna, Dawn Song, David Wagner "Android Permissions Demystified". CCS '11: Proceedings of the 18th ACM conference on Computer and communications security 2011.

[27] Adrienne Porter Felt, Elizabeth Hay , Serge Egelman, Ariel Haneyy , Erika Chin, David Wagner "Android Permissions: User Attention, Comprehension, and Behavior", Symposium on Usable Privacy and Security (SOUPS) 2012, July 11-13

[28] David Frei "Conducting a Risk Assessment for Mobile Devices" 2012, May 9 http://centva.issa.org/wp-content/uploads/2012/05/Risk-Assessment-Mobile-Devices.pdf

[29] Google Play Store, Permission Manager https://play.google.com/store/apps/details?id=com.gmail.permissionmanager&hl=en

[30] Google Play Store, Advanced Permission https://play.google.com/store/apps/details?id=com.gmail.heagoo.pmaster&hl=en

[31] Google Play Store, Privacy Protector https://play.google.com/store/apps/details?id=com.xeudoxus.privacy.blocker&feature=more_from_developer

XI. APPENDIX

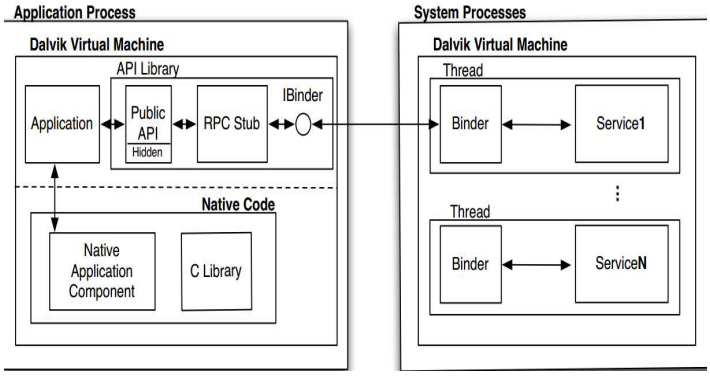


Fig 1. Permission process in the Android platform [27].

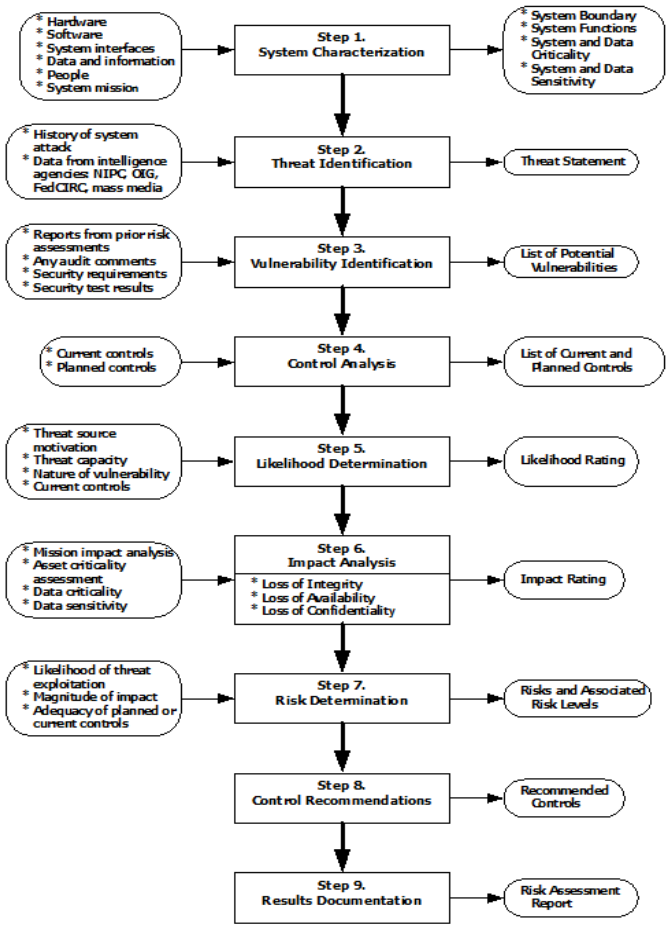


Fig 2. NIST special publication 800-30 Risk Assessment Model

The following figures show the date leakage of sensitive information from the children's apps while no monitoring tool in place.

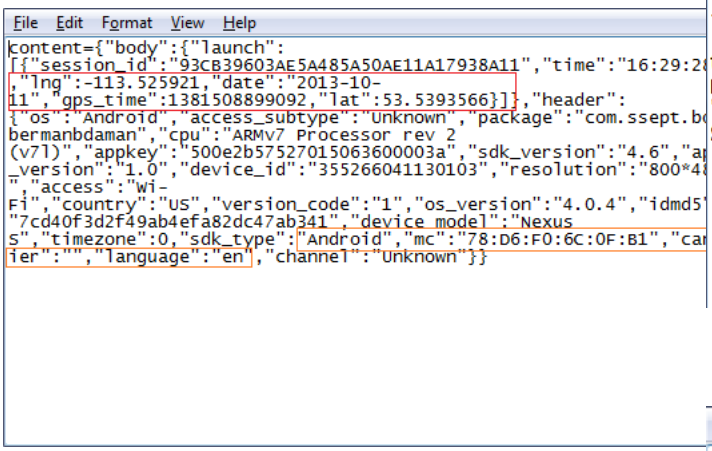


Fig 3. data leakage by BomberMan app.

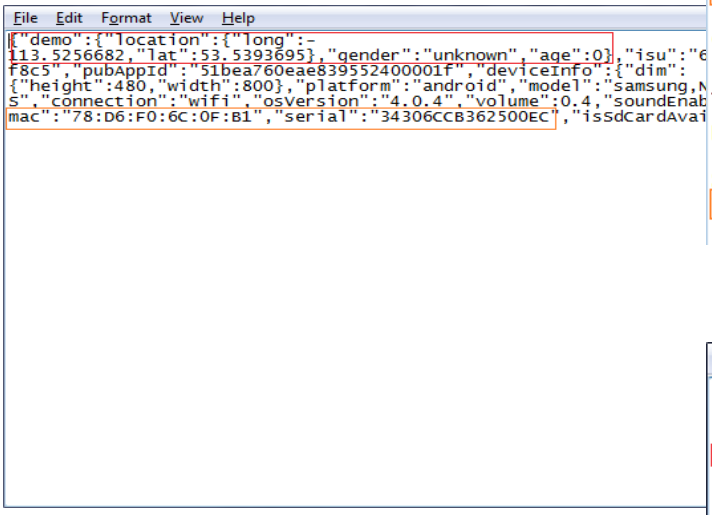


Fig 4: data leakage by Fruit Ninja app.

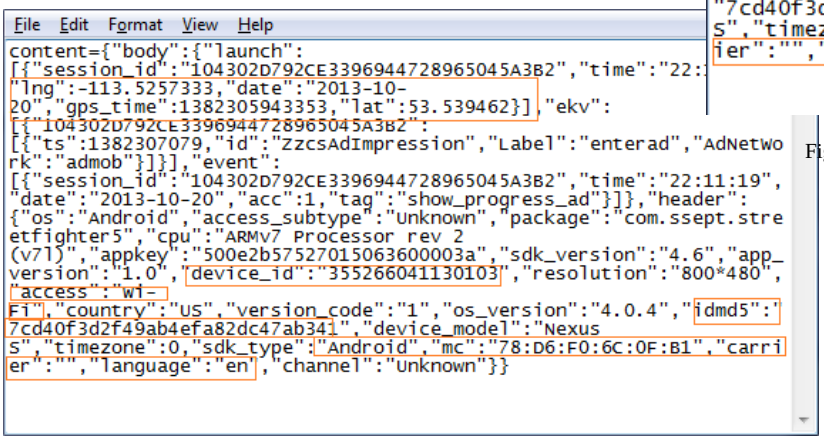


Fig 5: data leakage by Street Fighter V app.

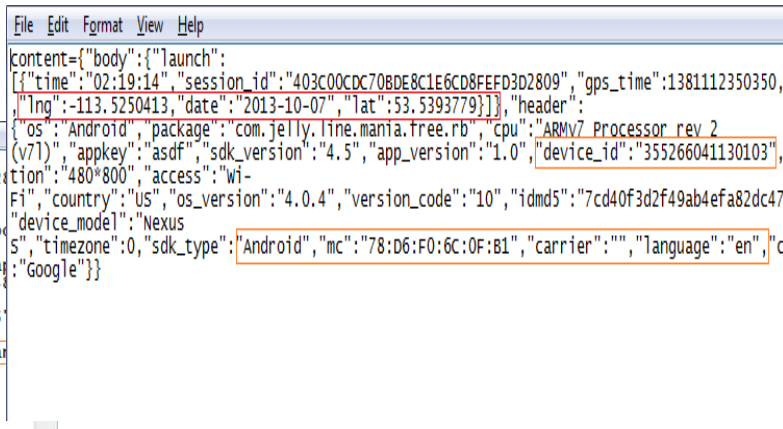


Fig 6:. data leakage by Jewel Pop Mania app.

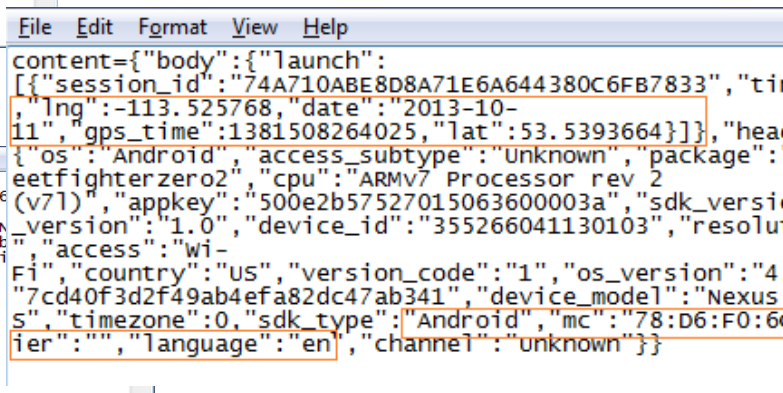


Fig 7: data leakage by Street Fighter Zero 2 app.

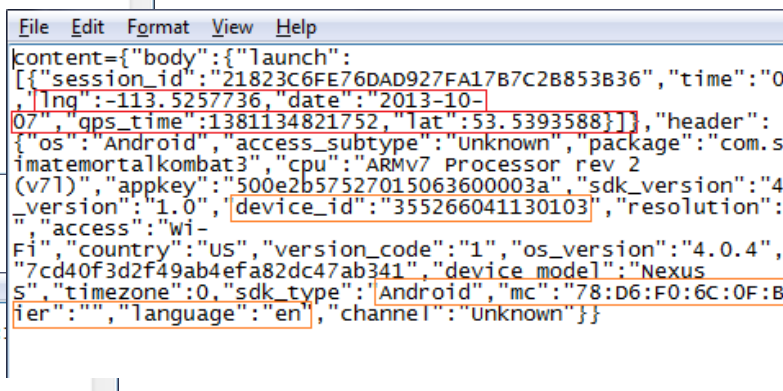


Fig 8: data leakage by Ultimate Mortal Compact app.

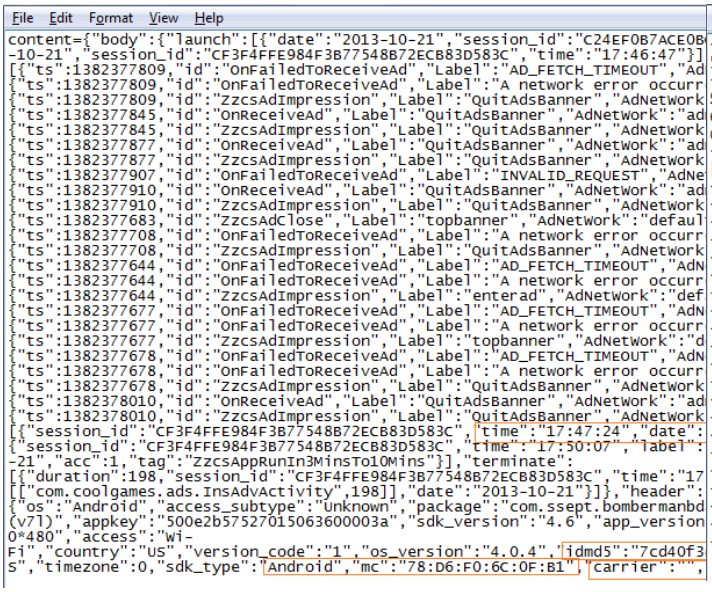


Fig 9: Privacy Protector tool preventing location of BomberMan app.

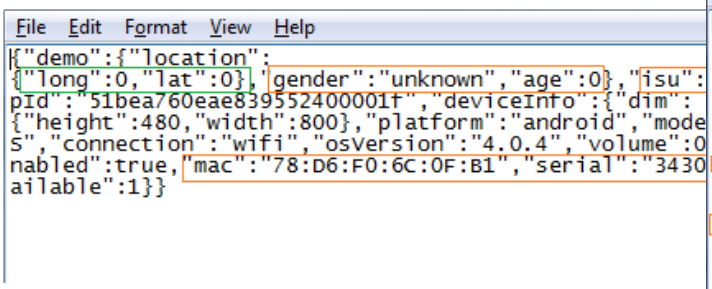


Fig 10: Privacy Protector tool preventing location of Fruit Ninja app.

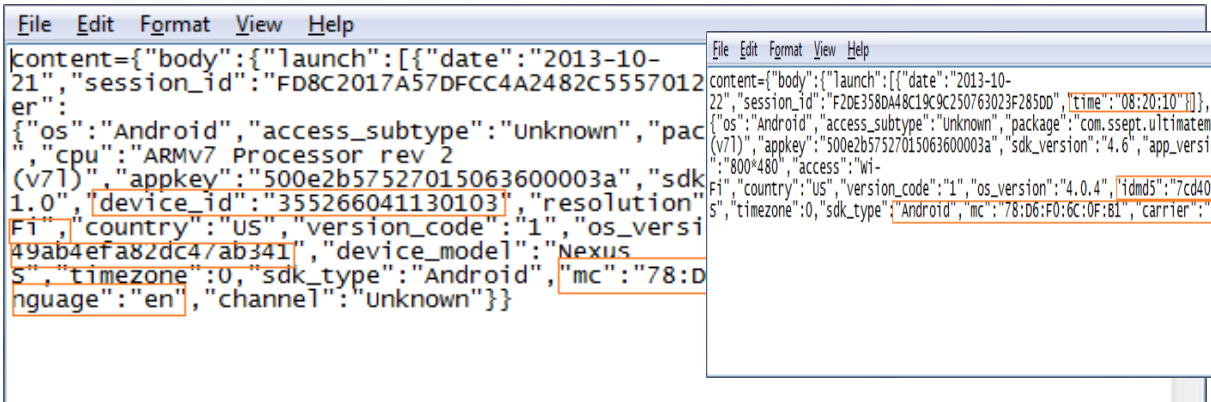


Fig 11: Privacy Protector tool preventing location of Street Fighter V app.

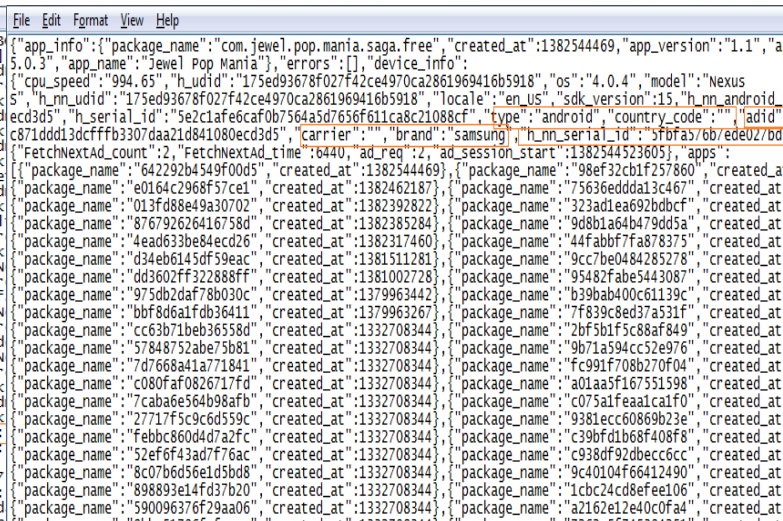


Fig 12: Privacy Protector tool preventing location of Jewel Pop Mania app.

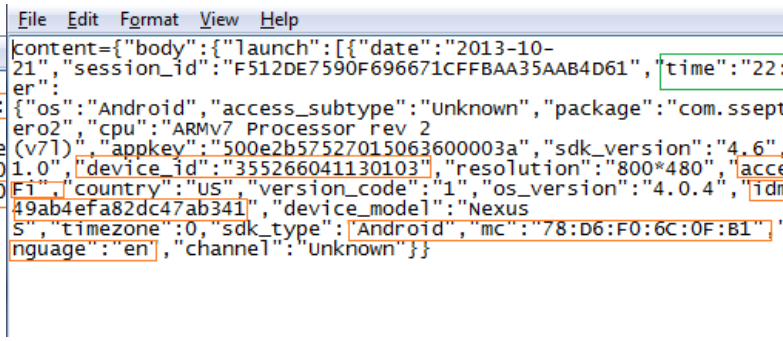


Fig 13: Privacy Protector tool preventing location of Street Fighter Zero 2 app.

Fig 14: Privacy Protector tool preventing location of Ultimate Mortal Compact app.

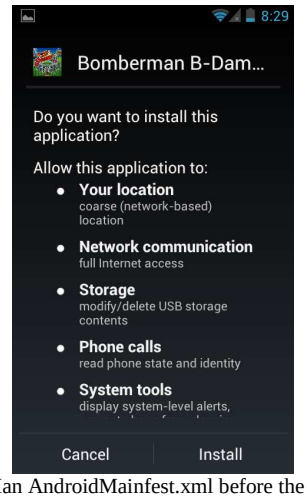


Fig 17: BomberMan AndroidMainfest.xml before the monitoring app in place ”Permission Manager”.

The following figures are show the children’s app permissions before and after applying the monitoring app in place”Permission Manager”.

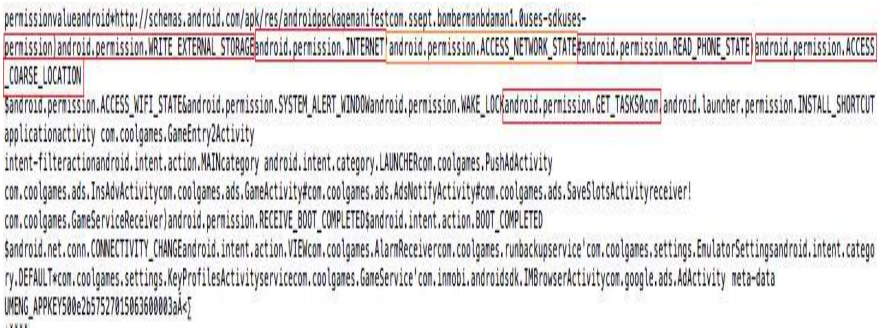


Fig 15: BomberMan app permissions before the monitoring app in place”Permission Manager”.

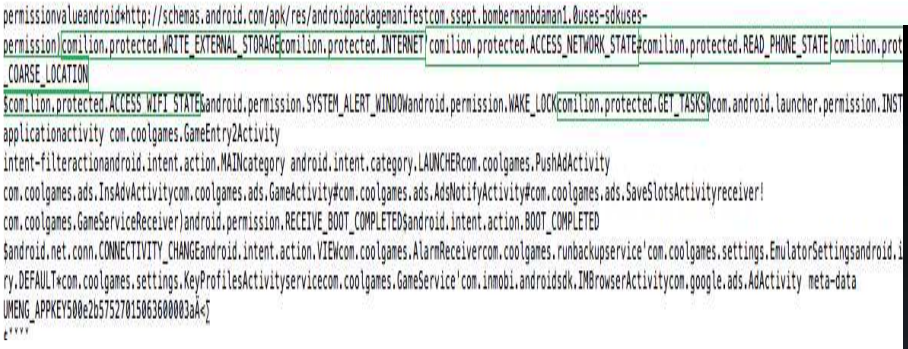


Fig 16: BomberMan app permissions when the monitoring app in place”Permission Manager”.

.

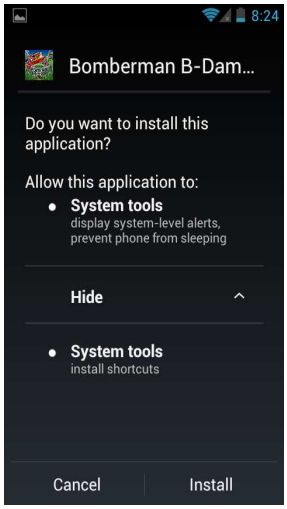


Fig 18: BomberMan AndroidMainfest.xml after the monitoring app in place”Permission Manager”.

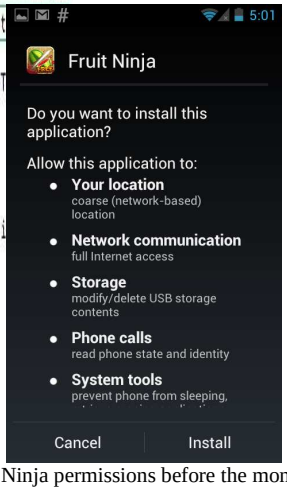


Fig19: . Fruit Ninja permissions before the monitoring app in place”Permission Manager”.

.

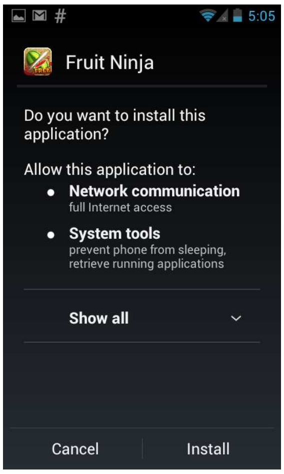
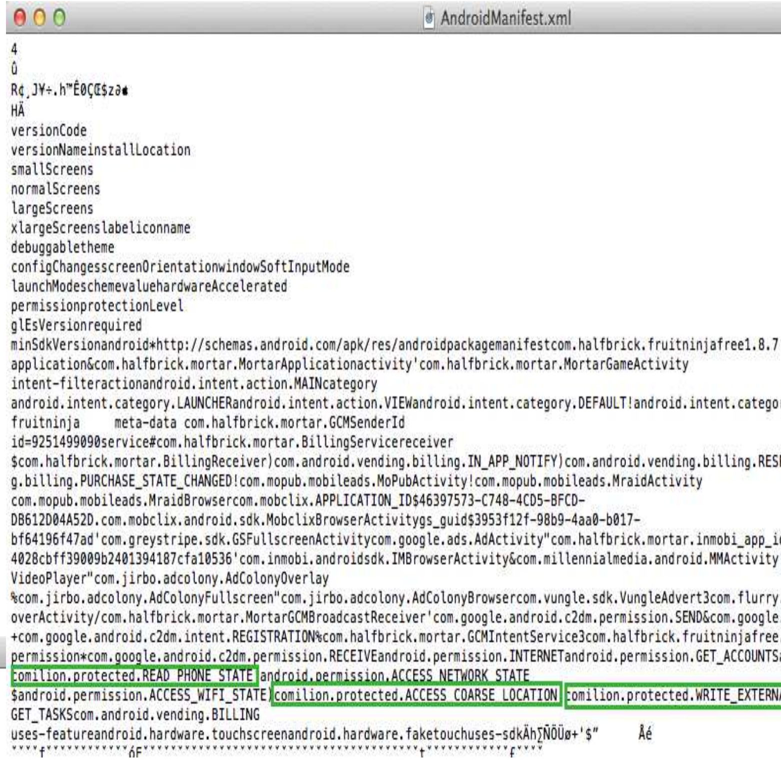


Fig 20: Fruit Ninja permissions when the monitoring app in place "Permission Manager".

```
AndroidManifest.xml
4
û
R¢_J¥÷.h™Ê0ÇŒ$zǝ
HÃ
versionCode
versionNameinstallLocation
smallScreens
normalScreens
largeScreens
xlargeScreenslabeliconname
debuggabletheme
configChangesscreenOrientationwindowSoftInputMode
launchModeschemevaluehardwareAccelerated
permissionprotectionLevel
glEsVersionrequired
minSdkVersionandroid*http://schemas.android.com/apk/res/androidpackagemanifestcom.halfbrick.fruitninjafree1.8.7
application&com.halfbrick.mortar.MortarApplicationactivity'com.halfbrick.mortar.MortarGameActivity
intent-filteractionandroid.intent.action.MAINcategory
android.intent.category.LAUNCHERandroid.intent.action.VIEWandroid.intent.category.DEFAULT!android.intent.category.BROWSABLEdata
fruitninja     meta-data com.halfbrick.mortar.GCMSenderId
id=9251499090service#com.halfbrick.mortar.BillingServicereceiver
$com.halfbrick.mortar.BillingReceiver)com.android.vending.billing.IN_APP_NOTIFY)com.android.vending.billing.RESPONSE_CODE2com.and
illing.PURCHASE_STATE_CHANGED!com.mopub.mobileads.MoPubActivity!com.mopub.mobileads.MraidActivity
com.mopub.mobileads.MraidBrowsercom.mobclix.APPLICATION_ID$46397573-C748-4CD5-BFCD-
DB612D04A52D.com.mobclix.android.sdk.MobclixBrowserActivitygs_guid$3953f12f-98b9-4aa0-b017-
bf64196f47ad'com.greystripe.sdk.GSFullscreenActivitycom.google.ads.AdActivity"com.halfbrick.mortar.inmobi_app_id
4028cbff39009b2401394187cfa10536'com.inmobi.androidsdk.IMBrowserActivity&com.millennialmedia.android.MMActivity'com.millennialmedi
eoPlayer"com.jirbo.adcolony.AdColonyOverlay
%com.jirbo.adcolony.AdColonyFullscreen"com.jirbo.adcolony.AdColonyBrowsercom.vungle.sdk.VungleAdvert3com.flurry.android.FlurryFul
rActivity/com.halfbrick.mortar.MortarGCMBroadcastReceiver'com.google.android.c2dm.permission.SEND&com.google.android.c2dm.intent.R
+com.google.android.c2dm.intent.REGISTRATION%com.halfbrick.mortar.GCMIntentService3com.halfbrick.fruitninjafree.permission.C2D_ME
permission*com.google.android.c2dm.permission.RECEIVEandroid.permission.INTERNETandroid.permission.GET_ACCOUNTSandroid.permission.WAKE_LOCK#and
oid.permission.READ_PHONE_STATEandroid.permission.ACCESS_NETWORK_STATE
$android.permission.ACCESS_WIFI_STATEandroid.permission.ACCESS_COARSE_LOCATIONandroid.permission.WRITE_EXTERNAL_STORAGEandroid.permission.GET
_TASKScom.android.vending.BILLING
uses-featureandroid.hardware.touchscreenandroid.hardware.faketouchuses-sdkÄhŽÑÔÜø+'$"     Åé
```

Fig 21: Fruit Ninja AndroidMainfest.xml before the monitoring app in place"Permission Manager".



```
AndroidManifest.xml
4
û
R¢_J¥÷.h™Ê0ÇŒ$zǝ
HÃ
versionCode
versionNameinstallLocation
smallScreens
normalScreens
largeScreens
xlargeScreenslabeliconname
debuggabletheme
configChangesscreenOrientationwindowSoftInputMode
launchModeschemevaluehardwareAccelerated
permissionprotectionLevel
glEsVersionrequired
minSdkVersionandroid*http://schemas.android.com/apk/res/androidpackagemanifestcom.halfbrick.fruitninjafree1.8.7
application&com.halfbrick.mortar.MortarApplicationactivity'com.halfbrick.mortar.MortarGameActivity
intent-filteractionandroid.intent.action.MAINcategory
android.intent.category.LAUNCHERandroid.intent.action.VIEWandroid.intent.category.DEFAULT!android.intent.categor
fruitninja     meta-data com.halfbrick.mortar.GCMSenderId
id=9251499090service#com.halfbrick.mortar.BillingServicereceiver
$com.halfbrick.mortar.BillingReceiver)com.android.vending.billing.IN_APP_NOTIFY)com.android.vending.billing.RESP
g.billing.PURCHASE_STATE_CHANGED!com.mopub.mobileads.MoPubActivity!com.mopub.mobileads.MraidActivity
com.mopub.mobileads.MraidBrowsercom.mobclix.APPLICATION_ID$46397573-C748-4CD5-BFCD-
DB612D04A52D.com.mobclix.android.sdk.MobclixBrowserActivitygs_guid$3953f12f-98b9-4aa0-b017-
bf64196f47ad'com.greystripe.sdk.GSFullscreenActivitycom.google.ads.AdActivity"com.halfbrick.mortar.inmobi_app_i
4028cbff39009b2401394187cfa10536'com.inmobi.androidsdk.IMBrowserActivity&com.millennialmedia.android.MMActivity'
VideoPlayer"com.jirbo.adcolony.AdColonyOverlay
%com.jirbo.adcolony.AdColonyFullscreen"com.jirbo.adcolony.AdColonyBrowsercom.vungle.sdk.VungleAdvert3com.flurry.
overActivity/com.halfbrick.mortar.MortarGCMBroadcastReceiver'com.google.android.c2dm.permission.SEND&com.google.
+com.google.android.c2dm.intent.REGISTRATION%com.halfbrick.mortar.GCMIntentService3com.halfbrick.fruitninjafree.
permission*com.google.android.c2dm.permission.RECEIVEandroid.permission.INTERNETandroid.permission.GET_ACCOUNTS
comilion.protected.READ_PHONE_STATEandroid.permission.ACCESS_NETWORK_STATE
$android.permission.ACCESS_WIFI_STATEcomilion.protected.ACCESS_COARSE_LOCATIONcomilion.protected.WRITE_EXTERN
GET_TASKScom.android.vending.BILLING
uses-featureandroid.hardware.touchscreenandroid.hardware.faketouchuses-sdkÄhŽÑÔÜø+'$"     Åé
```

Fig 22: Fruit Ninja AndroidMainfest.xml when the monitoring app in place "Permission Manager".

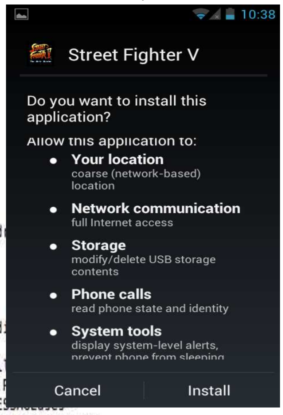


Fig 23: Street Fighter V app AndroidMainfest.xml before the monitoring app in place"Permission Manager".

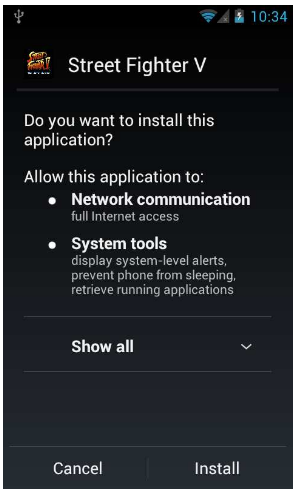


Fig24: Street Fighter V app AndroidMainfest.xml After the monitoring app in place"Permission Manager".

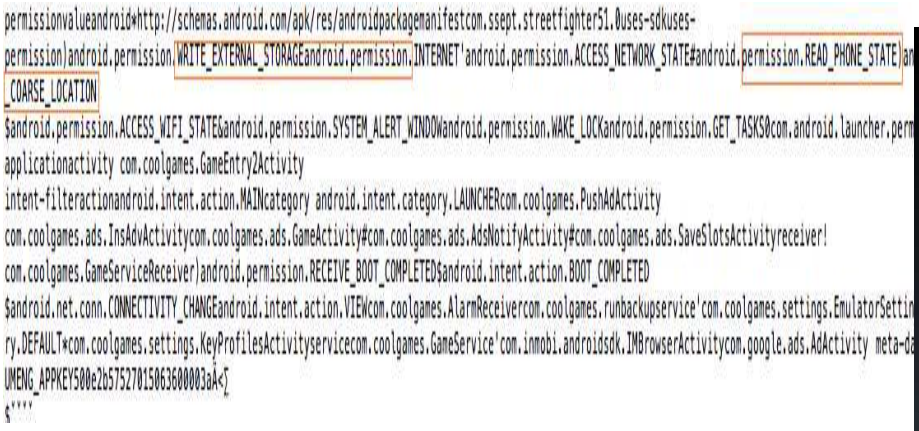


Fig 25: Street Fighter V app AndroidMainfest.xml before the monitoring app in place"Permission Manager".

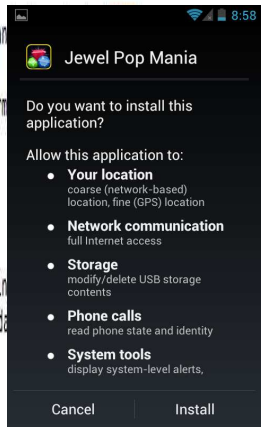


Fig 29: Jewel Pop Mania app AndroidMainfest.xml before the monitoring app in place"Permission Manager".

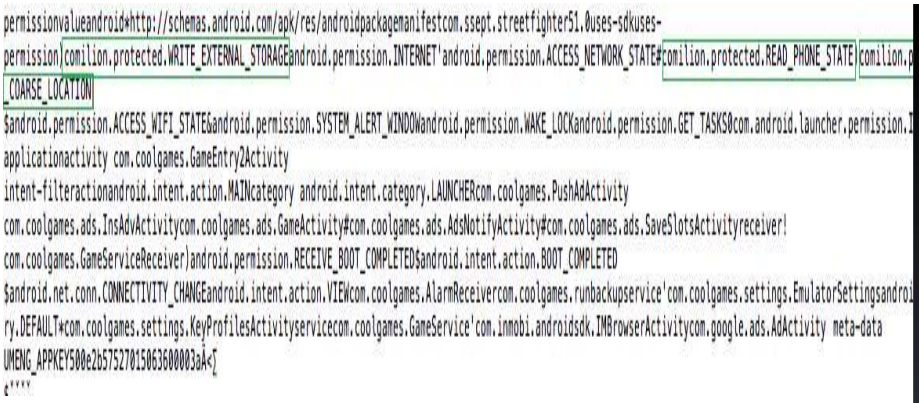


Fig 26: Street Fighter V app AndroidMainfest.xml after the monitoring app in place"Permission Manager".

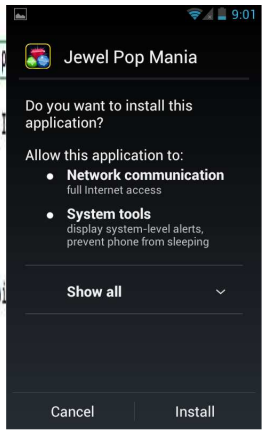


Fig 30: Jewel Pop Mania app AndroidMainfest.xml after the monitoring app in place"Permission Manager".

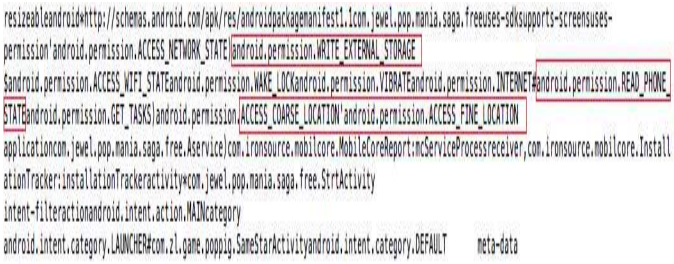


Fig 27: Jewel Pop Mania app AndroidMainfest.xml before the monitoring app in
place"Permission Manager".

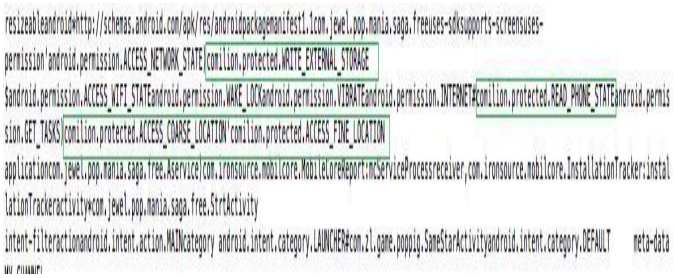


Fig 28: Jewel Pop Mania app AndroidMainfest.xml after the monitoring app in place"Permission Manager".

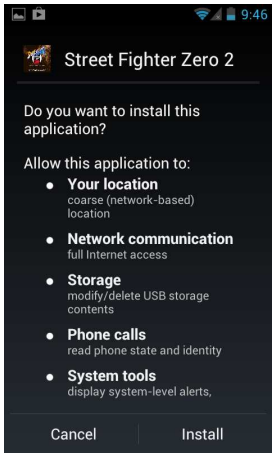


Fig 31: Street Fighter Zero 2 app AndroidMainfest.xml before the monitoring app in place"Permission Manager".

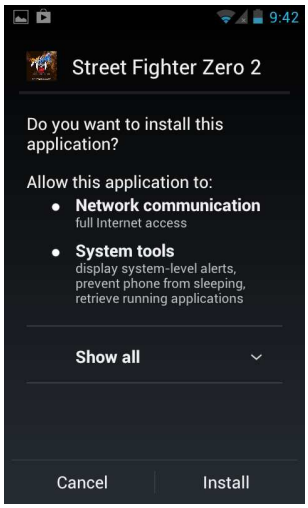


Fig 32: Street Fighter Zero 2 app AndroidMainfest.xml after the monitoring app in place"Permission Manager".

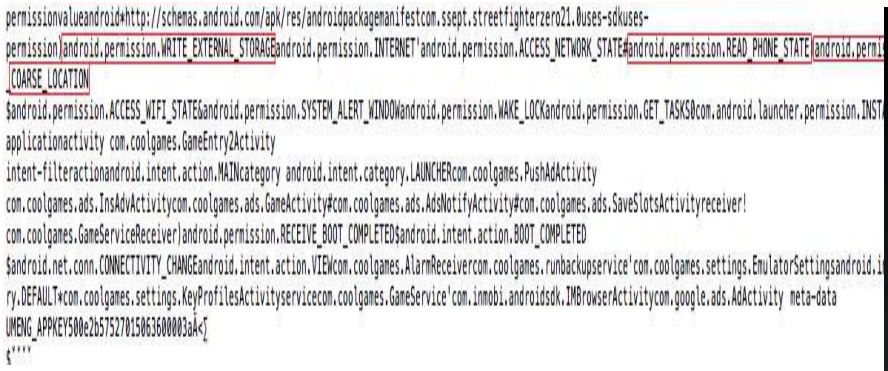


Fig 33: Street Fighter Zero 2 app AndroidMainfest.xml before the monitoring app in place"Permission Manager".

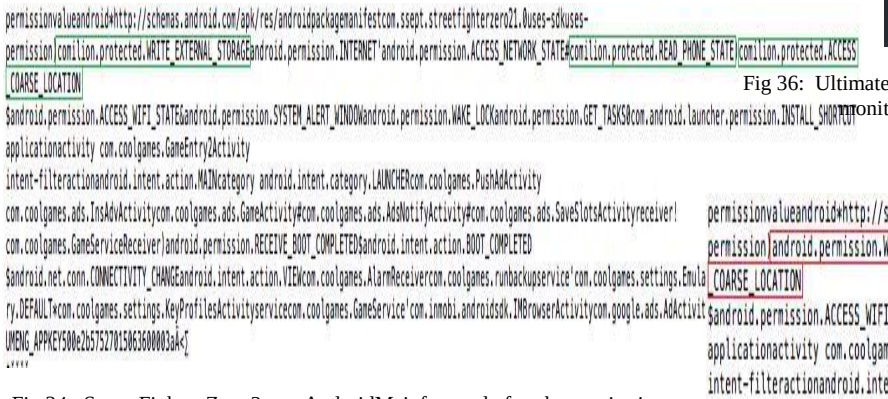


Fig 34: Street Fighter Zero 2 app AndroidMainfest.xml after the monitoring app in place"Permission Manager".

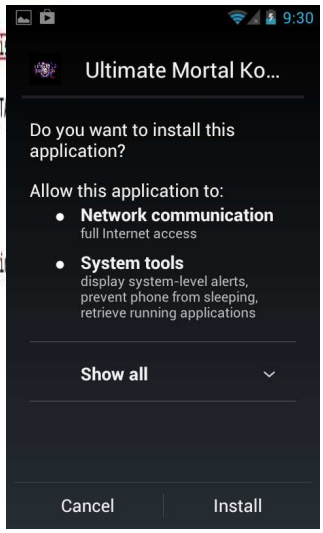


Fig 36: Ultimate Mortal Compact app AndroidMainfest.xml after the monitoring app in place"Permission Manager".

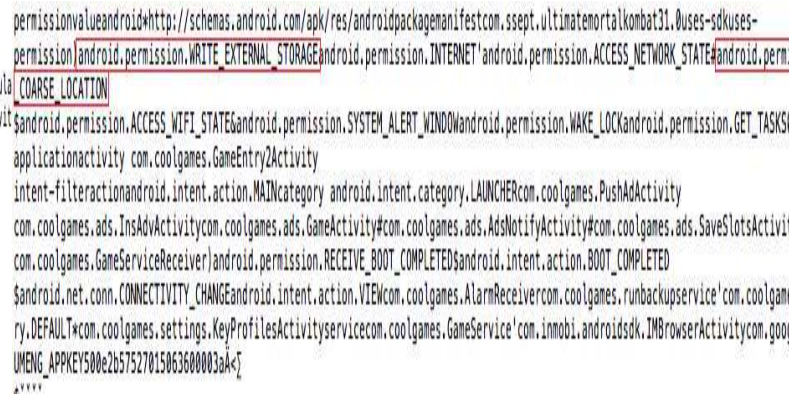


Fig 37: Ultimate Mortal Compact app AndroidMainfest.xml before the monitoring app in place"Permission Manager".

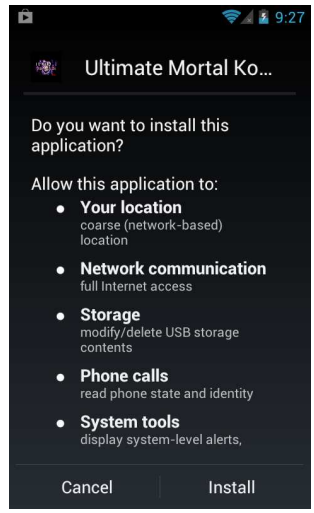


Fig 35: Ultimate Mortal Compact app AndroidMainfest.xml before the monitoring app in place"Permission Manager".

Fig 38: Ultimate Mortal Compact app AndroidMainfest.xml after the monitoring app in place"Permission Manager".

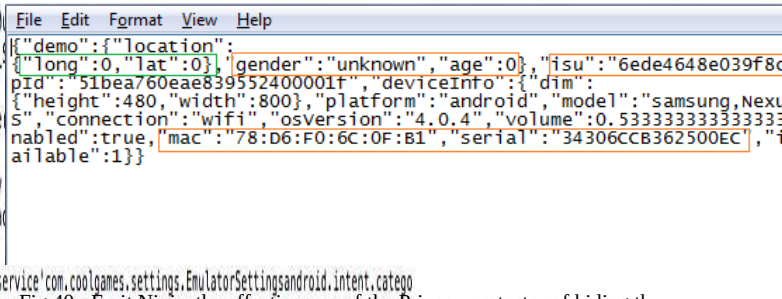


Fig 40: Fruit Ninja, the effectiveness of the Privacy protector of hiding the location

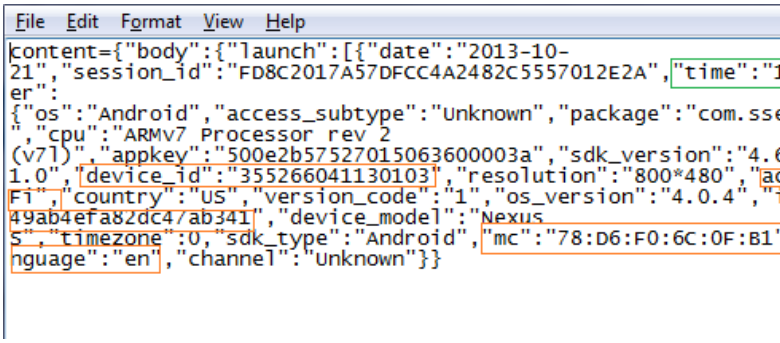


Fig 41: Street figher V, the effectiveness of the Privacy protector of hiding the location

The following figures show the effectiveness of the Privacy protector of hiding the location

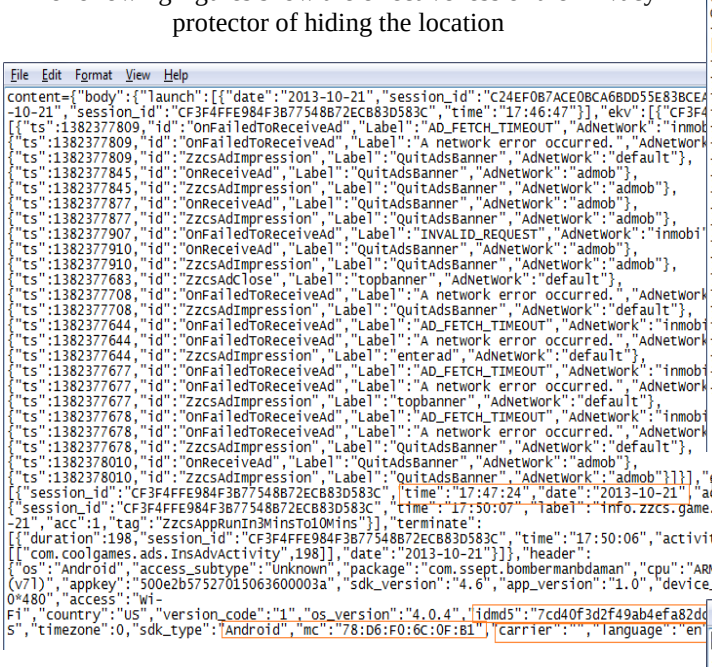


Fig 39: BomberMan, the effectiveness of the Privacy protector of hiding the location

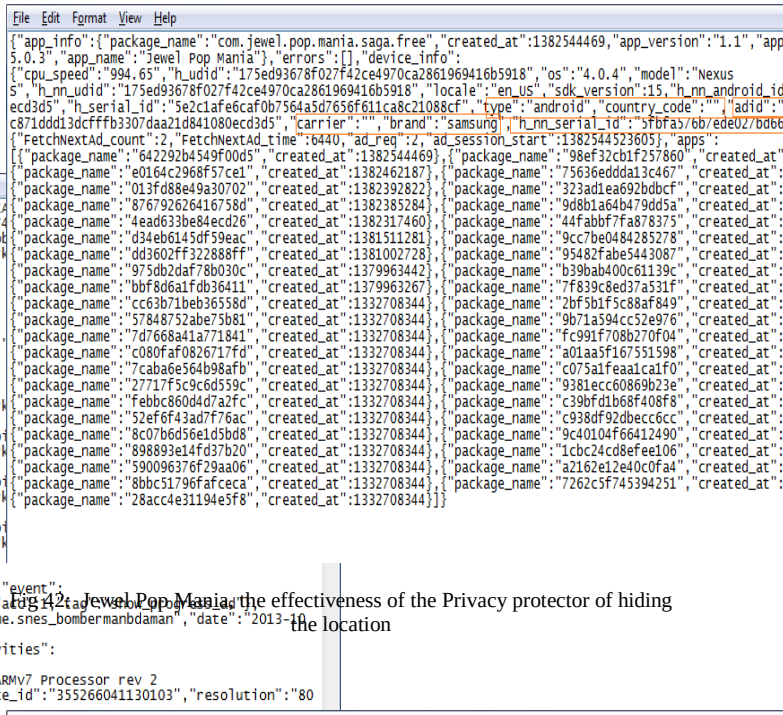


Fig 42: Jewel Pop Mania, the effectiveness of the Privacy protector of hiding the location

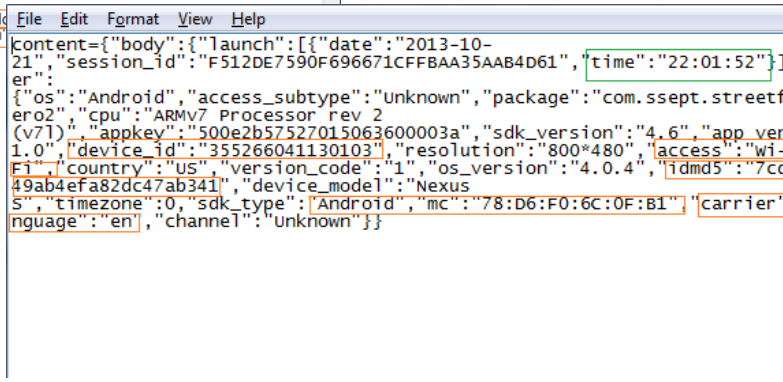


Fig 43: Street Fight Zero 2, the effectiveness of the Privacy protector of hiding the location

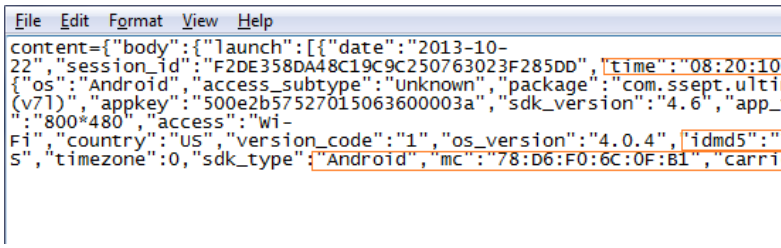


Fig 44: Ultimate Mortal Kompact, the effectiveness of the Privacy protector of hiding the location