ISSN 2303-4521
Periodicals of Engineering and Natural Sciences *Case Study*
Vol. 9, No. 4, September 2021, pp.386-391

Security vulnerabilities in the information management in university student´s smartphones: A case study in south west of Colombia

Cristian Camilo Ordoñez[1], Hugo Armando Ordoñez Erazo[2], Armando Ordoñez[1]
[1] Faculty of Engineering, Foundation University of Popayan
[2] Faculty of Electronic Engineering and Telecommunications, Cauca University

ABSTRACT

Nowadays university student's smartphones are affected by the theft and leakage of information, to address this issue, this research aims at identifying security vulnerabilities in these devices. Moreover, an application was developed to prevent phishing and information leaks. Effectiveness and functioning tests were carried out to identify diverse vulnerabilities and alerting users about them.

Objective: Identify vulnerabilities based on diverse techniques (phishing, DNS poisoning, identity theft, Man in the middle, foot-printing, spyware) in Android smartphones used by university students from the south west of Colombia.

Methodology or method: The following phases were carried out: 1. Definition of the problem. 2. Literature review. 3. Planning; In this phase, the study population, and the methods and instruments were defined. 4. Evaluation: we collected the information and analysed the results.

Results: An application was developed to show the security vulnerabilities regarding the installation of malicious software that extracts information from their devices.

Conclusions: The security of our mobile phones is a priority nowadays. We developed an application to achieve greater security in android smartphones, However, it is crucial to be aware of the importance of self-care.

Keywords: Vulnerabilities, Mobile devices, Android, Phising, Information leak

Corresponding Author:

Cristian Camilo Ordoñez
Faculty of Engineering
Foundation University of Popayan
Address. Cl. 8 #9-51, Popayán, Cauca, Colombia
E-mail: camilo.ordonez@docente.fup.edu.co

1. Introduction

Recent studies show that the adoption of smartphones in Colombia continues to rise. Currently, 96% Colombians use a mobile phone in their daily life [1]. Most mobile phones are used to store personal information such as photos, videos, bank accounts, as well as other confidential documents of their companies.[2].

Recently there have been incidents related to the theft of information from mobile devices. Incidents affecting ordinary citizens decreased by 35% between 2014 and 2016 figure. While at this same time, business attacks had an increase of more than 20% [3].

Many people are affected for this reason, which has consequences on a psychological, physical and financial level. One of the most vulnerable groups to these attacks are university students, who store personal information and carry out transactions with their mobile devices using free and insecure public networks [4]. This article presents an investigation with students from various universities in the south west of Colombia in order to identify the risk to which their mobile devices are exposed.

The remainder of this article is organized as follows: Related work is presented in Section 2. Section 3 presents the case study and the mobile application developed. Section 4 exposes the results and finally Section 5 concludes and exposes future work.

2. Related work

GDroid [6] is a tool to detect malware or vulnerabilities in Android, to do so, GDroid uses convolutional neural networks (CNN) to map Android APIs and applications into a heterogeneous graph, then a node classification task is carried out based on the invocations to the Android API and the usage patterns.TC-Droid [7] presents a framework that feeds on the text sequence of the application reports generated by AndroPyTool, and applies CNN to explore them. In [8] a CCN-based malware detection model is presented, which uses an Android application's operation code sequence, to do so, Dreblin´s dataset is used [9]. DCDroid [10] offers a tool to detect vulnerabilities by combining static and dynamic analysis. In the static analysis, DCDroid focus on identifying various types of vulnerabilities in application code snippets. In the dynamic analysis, DCDroid prioritizes activation of user interface (UI) components based on static analysis results to confirm SSL / TLS misuse.

As can be seen in the related work, existing approaches are based on neural network models with very good results in the classification of malware. However, none of them study directly the behaviour of real users to identify the most frequent vulnerabilities.

3. Case study

The case study is descriptive /holistic and is based on the Runeson methodology (See figure 1) [11]. The main objective is to identify the vulnerability of mobile devices with Android operating system in university students from the southwestern of Colombia.

Figure **1.** Case study phases

3.1. Population selection

The population is made up of university students between 16 - 25 years old, because of their vulnerability. This is due to the fact that many times they do not have the precaution when connecting to free Wi-Fi networks or providing information which causes theft of information or credentials. The students had to use a smartphone with Android operating system. The population consists of 129 participants and is described in Figure 2.

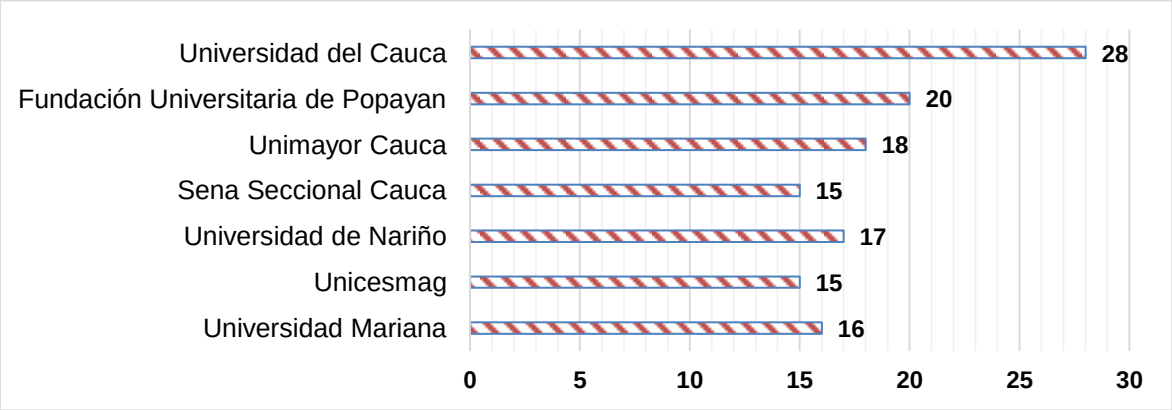

Figure 2. University Students from the southwest of Colombia

3.2. Compliance with restrictions

information about Android vulnerabilities from version 5.0 to 10.0, is collected from the National Vulnerability Database (NVD) provided by the National Institute of Standards and Technology [12].

3.3. Initial meeting

With the analysed data, vulnerabilities, severity and details are classified. Between August 28, 29, 2020, two meetings were held by zoom platform, a meeting with 70 people and another with 59 for a total of 129 students from different universities

3.4. Evaluation with the app

The analysis unit corresponds to the execution of the process, that is, the installation of the application on the devices by a cybersecurity expert, thus measuring qualitative and quantitative aspects. At a quantitative level, the vulnerability identification time was taken into account and for the qualitative aspect, through the evaluation of the expert it was determined to what degree the devices are at the vulnerability level. Next, Table 1 shows the results of the versions found and the app installation messages.

Table 1 Android Versions

VERSION	NUMBER OF DEVICES	MESSAGE DURING INSTALLING
ANDROID 5	10	Message configuration during installing
ANDROID 7	15	Automatically installed
ANDROID 8	40	Confirmation Mensaje during installing
ANDROID 9	30	Message configuration during installing
ANDROID 10	34	Message configuration during installing
TOTAL	129	

3.5. Application of validation instruments

Surveys were used as information collection methods. Two vulnerability identification cycles were defined in order to establish how the repetition of the execution of the process affects the results obtained. At the end of the two cycles, it was decided to conduct a survey to each of the participating students about the process and implementation of the application.

3.6. Mobile application

To carry out the case study, it was necessary to develop the application with the name Android Protect, this has the functionality of identifying whether or not there are vulnerabilities within the mobile device where it is installed, then each of its components is shown.

Figure 3. Icon of the application

Figure 4 shows the application toolbar which illustrates the icons of each module. The first icon from left to right is illustrated with a key which represents the configuration module, the second is illustrated with a circle which is one of the most important because it scans the vulnerabilities according to the version used in the mobile device, the third icon is illustrated with an icon of sheets and graphs which is the results module there are shown the terms related to the configurations and scans of the vulnerabilities clearer and with a brief description, in addition to the most common attack, finally there is the fourth icon illustrated with a light bulb which provides some alternatives as a solution to counteract data leaks. In addition, the sources from which the registered vulnerabilities are extracted in real time according to the version of each device are detailed, where there is a brief description of each menu. In module 1, when the application has already been entered, it asks the user for permissions, which are shown below.

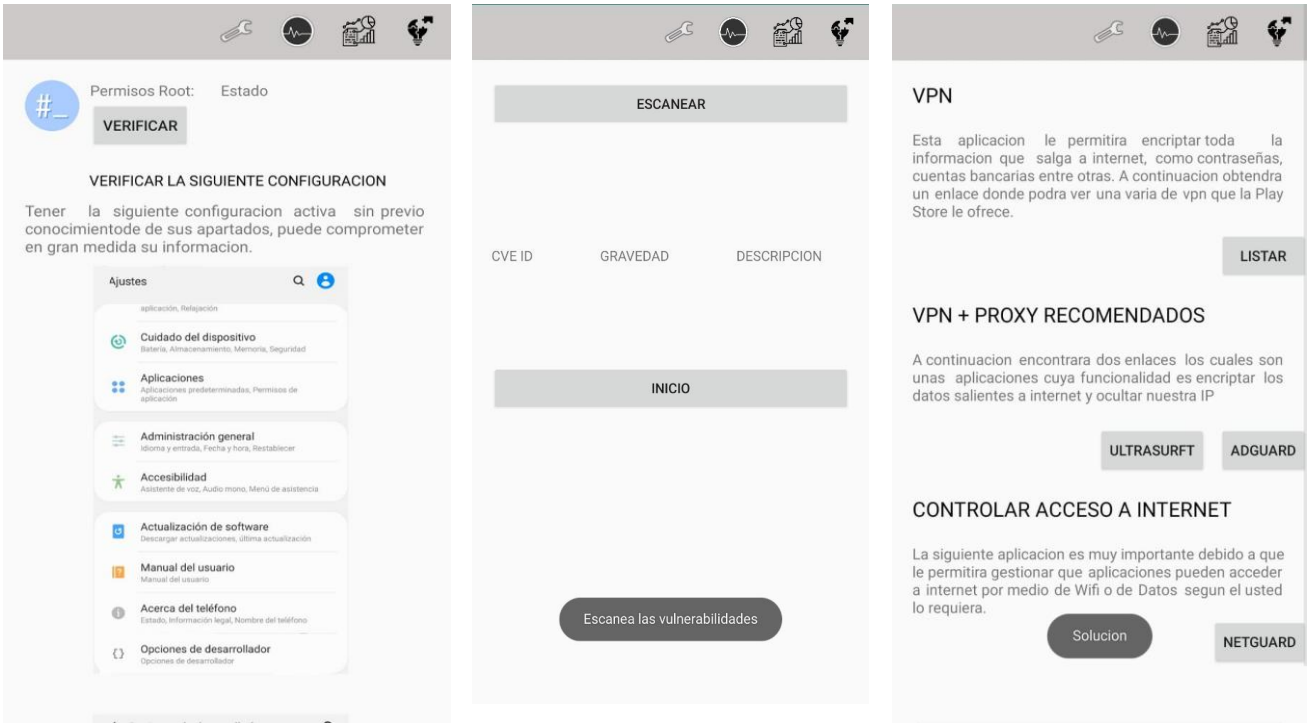

Figure 4. Modules 1,2,3 Scanner, Root

The first module of the application is in charge of verifying if the mobile device is with Root access, that is to say, released with all the permissions to use the device. Module 2 is the most important as it generates the device's scanner, it has a table of 3 items which categorizes the scanned information for a better understanding, this is done in technical terms which help to know the severity, the description of the vulnerability and the code of the vulnerability. The scanning process is done using a technique called Web Scraping which consists of capturing the vulnerabilities registered by the manufacturers in a table according to the Android version of the device, then the application shows the results of the scan. In module 3 displays the vulnerabilities.

Module 3 shows the most frequent attack techniques on mobile devices, it also describes the configurations presented in module one, and informs why these configuration options should be disabled (such as keep the

device without Root access). Moreover, there is a brief description of the module. Finally, in module 4 it shows the recommendations of tools and actions that help to take care of the device such as:

> ***VPN*** maintains control of the internet connection and modifies the geolocation to make an anonymous bridge of the data handled by the device, on the other hand, it allows to encrypt the susceptible information on the internet.
>
> ***VPN + PROXY:*** This service allows the user to encrypt the outgoing data to the internet and hide the user's IP, using Ultrasurft, [13] y Adguard [14].
>
> ***CONTROL OF INTERNET ACCESS***: Here Netguart is used to control the entry and exit of data to the network, it has a firewall that allows controlling the internet access of each application installed on the mobile device, providing better control of the data flow [15].

4. Results

The evaluation of the case study was carried out with 129 students from different universities in the south west of Colombia, the installation of the Android Protect application was carried out on the different devices of the participants in Android 5.0 lollipop versions up to version 10.0. The scan was successfully performed on the versions of those ranges and their derivatives. After that, a laboratory was developed where the devices were infected with spyware and persistent backdoor [16]. The practice consisted of verifying the effectiveness of the Android protect to stop the viruses. Other sections such as Root access detection, configuration enabled in developer mode were also evaluated.

Figure 5 shows that 19% of the devices were detected by Google Play Protect as it is responsible for blocking the installation of malicious applications on their device, 12% allowed the installation without any protection, and 70% display a message requesting for permission for installation. 19% of devices had Root access, allowing them to view any type of data and modify it. 81% of the devices had not Root access. On the other hand, it was identified that 22% of devices had Google Play Protect disabled (The device is pre-configured to accept external applications). On the other hand, 78% have Android protection enabled, so it does not allow applications to be installed automatically. Finally, it is evident that 100% of the devices that carried out the laboratory are vulnerable because none have a stable protection that allows them to block the installation of applications or malicious elements.

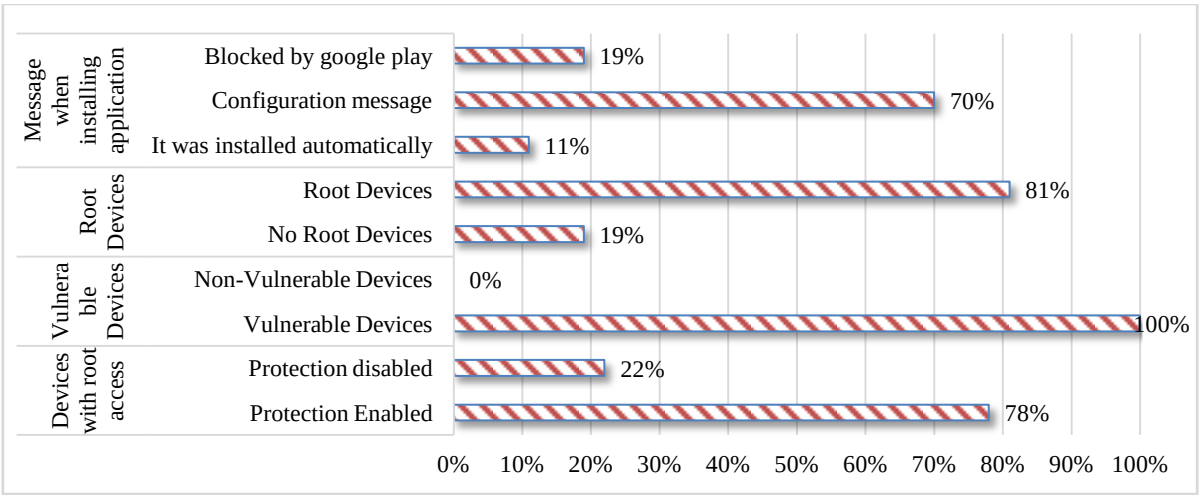

Figure 6. Results of the case study

5. Conclusions and future work

The case study shows the importance of taking care of security issues in our mobile devices. It is recommended to take into account these issues to generate self-care.

A mobile application was developed with four modules to help the user to verify the security of the mobile device. First, an analysis of the internal configuration (Root access, debugging mode). Second, a scanning the registered vulnerabilities. Third a section where the most exploited vulnerabilities are explained. Finally a Section about how to prevent the vulnerabilities.

It is also concluded that 100% of the devices were vulnerable because the devices had a malicious software blocker, however it was not enough to prevent malicious installations. Many devices had a wrong permission assigned by the owner of the mobile device. For this reason, it is recommended to have more knowledge on the subject and in this way avoid these problems

References

[1] Nelson Valero, "Consumo movíl en Colombia," Deloitte, vol. 01. p. 23, 2018.

[2] W. C. Álzate, C. S. Romaña, and Y. Q. Barco, "Factores y causas de la fuga de información sensibles en el sector empresarial.," Cuad. Act., vol. 7, no. 1 SE-Artículos de reflexión, Jan. 2015.

[3] R. Maya, "El cibercrimen y sus efectos en la teoría de la tipicidad: de una realidad física a una realidad virtual," Nuevo Foro Penal, vol. 13, pp. 72–112, 2017, doi: 10.17230/nfp.13.88.3.

[4] A. C. Silva Calpa and D. G. Martínez Delgado, "Influencia del Smartphone en los procesos de aprendizaje y enseñanza," Suma Negocios, vol. 8, no. 17, pp. 11–18, 2017, doi: https://doi.org/10.1016/j.sumneg.2017.01.001.

[5] A. Razgallah, R. Khoury, S. Hallé, and K. Khanmohammadi, "A survey of malware detection in Android apps: Recommendations and perspectives for future research," Comput. Sci. Rev., vol. 39, p. 100358, 2021, doi: 10.1016/j.cosrev.2020.100358.

[6] H. Gao, S. Cheng, and W. Zhang, "GDroid: Android malware detection and classification with graph convolutional network," Comput. Secur., vol. 106, p. 102264, 2021, doi: 10.1016/j.cose.2021.102264.

[7] N. Zhang, Y. an Tan, C. Yang, and Y. Li, "Deep learning feature exploration for Android malware detection," Appl. Soft Comput., vol. 102, p. 107069, 2021, doi: 10.1016/j.asoc.2020.107069.

[8] M. Kinkead, S. Millar, N. McLaughlin, and P. O'Kane, "Towards Explainable CNNs for Android Malware Detection," Procedia Comput. Sci., vol. 184, no. 2019, pp. 959–965, 2021, doi: 10.1016/j.procs.2021.03.118.

[9] Yoshikuni Igarashi, "DREBIN: Effective and Explainable Detection of Android Malware in Your Pocket," J. Jpn. Stud., vol. 36, no. 1, pp. 165–169, 2009, doi: 10.1353/jjs.0.0130.

[10] Y. Wang et al., "Identifying vulnerabilities of SSL/TLS certificate verification in Android apps with static and dynamic analysis," J. Syst. Softw., vol. 167, p. 110609, 2020, doi: 10.1016/j.jss.2020.110609.

[11] P. Runeson and M. Höst, "Guidelines for conducting and reporting case study research in software engineering," Empir. Softw. Eng., vol. 14, no. 2, p. 131, 2008, doi: 10.1007/s10664-008-9102-8.

[12] N. I. of S. and Technology, "Marco de Ciberseguridad del NIST," Ciberseguridad NIST, vol. 1, pp. 1–9, 2019.

[13] R. Al-quraan, A. Hadi, J. Atoum, and M. Al-Zewairi, "Ultrasurf Traffic Classification: Detection and Prevention," Int. J. Commun. Netw. Syst. Sci., vol. 8, pp. 304–311, 2015, doi: 10.4236/ijcns.2015.88030.

[14] D. Howe and H. Nissenbaum, "Engineering Privacy and Protest: a Case Study of AdNauseam," 2017.

[15] A. Skendzic and B. Kovačić, "Open source system OpenVPN in a function of Virtual Private Network," IOP Conf. Ser. Mater. Sci. Eng., vol. 200, p. 12065, 2017, doi: 10.1088/1757-899X/200/1/012065.

[16] J. Dai, C. Chen, and Y. Li, "A Backdoor Attack Against LSTM-Based Text Classification Systems," IEEE Access, vol. 7, pp. 138872–138878, 2019, doi: 10.1109/ACCESS.2019.2941376.