A Survey of Android Attacks Detection Techniques

José Duarte

Lusofona University of Porto, Portugal
fernandoteixo_10@hotmail.com

Abstract. In this age of technology, mobile devices have become indispensable for humans. Most of these mobile devices have android operating system and as this number grows, the number of applications and malware, consequently also grows, which leads to greater concern and prevention, in this case, the android world. Many of these malicious applications are available on android's play store, making it an arduous task for the user to distinguish between which applications are clean. Somehow, there are several malware detection tools, which makes it difficult for malicious applications to penetrate, but malware writers use various techniques to avoid these tools. That said, this paper aims to explore the different attack detection techniques of Android and make some suggestions for defense mechanisms.

Keywords: Malware Detection, Android, Mobile Devices, Applications, Security, Android Attacks, Detection Techniques

1 Introduction

Nowadays, mobile devices are of great importance to humans, having exceeded the 5 billion mark, according to Hootsuite and We Are Social, accounting for a 67% share of the world's population [1]. These mobile devices are getting better technology at both chip design and microprocessor computing power levels, offering a wide range of features, and this is one of the reasons for their soaring popularity. Most users regard them as a reliable and private communication channel having access to various personal information. Within these mobile devices, there are several operating systems, but the most used is Android, with over 2 billion active devices [2]. It has become popular because of its low cost and because Android operating system code is made available by Google under open source license. By having such features and popularity, Android System-based devices inevitably attract the attention of cybercriminals who are creating and distributing malicious programs. According to Maya Horowitz (director of threat intelligence) and Check Point research, "The sharp rise in mobile banking malware is related to the growing use of mobile banking applications", as hackers are increasingly focused on theft of credentials, vigilance, and malicious advertising [3]. In many cases, malware attacks follow distribution strategies like those of desktop users, with applications running silently in the background, without the victim noticing. This article is organized as follows, section 2 addresses security challenges for mobile device users; Section 3 presents the Android security architecture; Section 4 presents the most popular malware types with a brief introduction. Section 5 discusses some malware

10.11228/dpsc.02.01.010

detection techniques; Section 6 presents a real case of a ransomware attack; and to finish we have the conclusion.

2 Security Challenges for Mobile Device Users

Mobile devices were created for the main purpose of communication, but nowadays these pocket computers can be used for various daily needs like searching for any information, making payments, entertainment and many other things, but this level of comfort has brought with it an extreme number of security risks to our personal information.

- Physical Security: Physical security is when the mobile device is lost or stolen. The personal data of the device user may be stolen and misused.
- Insecure Data Storage: Insecure data storage is the most common problem, found in 76% of mobile applications [4], so access to personal data such as name, address, date of birth, bank information, family photos, social network, email address, as well as access to work information (company name, job title and the like). Hackers do not need physical access to a mobile device to steal data as 89% of vulnerabilities could be exploited using malware [4]. Most cases are caused by deficiencies in application security mechanisms, but cyber-attacks also depend on user inattention leading to financial losses for users.
- Mobile Browsing: Normally, on mobile devices it is not possible to see the entire URL or web address, so it is difficult to prevent us from a phishing attack, for example.
- Multiple User Logging: There is a progressive growth in social media and single sign-on (SSO) as most mobile applications are insecure due to the possibility of allowing the user to access various services that require authentication by performing authentication only once. Hackers who gain access to login credentials for a website or application such as Facebook may also have access to a user's profile page.
- Client-Side Injection: The client-side injection results in the execution of malicious code on the client side, the mobile device and this through a mobile app. This malicious code is often provided in the form of data. What they target is the data on the device with SQL injection; the mobile user session with JavaScript injection (XSS, etc.), the application interfaces or functions, and the Binary Code itself.[5]
- Improper Session Handling: Many developers allow long user sessions that do not expire or use session tokens that are too predictable. They often do this because companies want users to have quick access to shopping and checkout so that sales are made immediately, and no second opinion is created. Long-term sessions invite vulnerabilities when performing financial tasks. Poor session management can provide clues about unauthorized access by hijacking sessions on mobile devices.[6][7]
- Weak Authentication and Brute Force Attack: Many applications rely on password-based authentication as a single factor, and often the owners of these applications do not enforce strong passwords and many users are exposed to a host of threats, including brute force attack. About 5% of confirmed data breach incidents in 2017 resulted from brute force attacks [8] and these attacks are simple and reliable as

attackers use computational power to perform their work by testing different username combinations and passwords until you find the key. [6]

3 Android Security Architecture

Android is an open source software platform for mobile devices. It includes a Linux Kernel, middleware framework, and core applications. The android has limited resource, so it's very difficult to implement traditional security services. Therefore, researchers are trying to propose different behavioral approach to guard against malware.[9]

- Permission Mechanism: The purpose of a permission is to protect the privacy of an android user. Android applications must request permission to access user confidential data (such as contacts and SMS) and certain system features (such as camera and internet). A central design point of Android's security architecture is that no application, by default, can perform operations that would adversely impact other applications, the operating system, or the user. [10]
- Sandboxing: On the Android system each application is assigned a unique UserID. Android uses the UID to set up a kernel-level Application Sandbox. This isolates applications from each other by protecting them, for example, if application A attempts to do something malicious, such as reading application B's data without permission, will be prevented from doing so because it does not have the appropriate default user privileges. The sandbox is based on process separation and file permissions. [11]
- Access Control: In access control the mechanism of each archive has specific access rule and each process assigns a UserID. Each process has a specific permission to read, write or execute the file. [9]
- Components Encapsulation: Application components can be specified as public or private. Private components are accessible only by components within the same application. When declared public, components can also be accessed by other applications. However, full access can be limited by requiring calling ap-plications to have specified permissions. [9]
- Application Signing: Android uses cryptographic signatures to verify the origin of applications and establish trust relationships between them, so developers need to sign application code. This allows you to enable signature-based permissions or allow applications from the same source to share the same UserID. There is a certificate that is self-signed by the developer that is validated at application installation time. [9] [12]

4 Android Malware Attacks

Cybercriminals are increasingly focusing on mobile devices, with Android being the hardest hit due to its characteristics and this is because users ignore all or almost everything about mobile apps, or don't care about it. that favors cybercriminals. Therefore,

additional security knowledge of mobile devices is required, as well as better security solutions and policies. To obtain confidential financial information, hackers have developed and spread mobile malware. Malware is software that has been coded to damage devices, harm users, and steal data by infecting the operating system without the user's knowledge or approval. Malware is often developed by hacking teams who are often just looking for a way to make money, either by proliferating their own malware or through auctioning on the Dark Web. This malicious software can be used as protest tools, to test the security of a network, or even as weapons of war between governments [15]. Mobile malware often steals information stored on users 'mobile devices or sends SMS to premium numbers for the hackers' monetary profit. Stolen information may include International Mobile Equipment Identity (IMEI) numbers, International Mobile Subscriber Identity (IMSI) numbers, Sub-scriber Identity Module (SIM) serial number, user credentials for future misuse, contacts or location of the Global Positioning System (GPS). Some mobile malware turns the infected phone into a bot that can be remotely controlled by the Command and Conquer (C&C) server. [13]

4.1 Types of Malware on Android

There are several types of malware, all with different forms of penetration and their list is far from defined, but some of the best known are:

- Virus: Viruses are a piece of code that replicates and is dispatched by the application, so they attach themselves to clean files and infect other clean files. They can spread uncontrollably, damaging a system's core functions and deleting or corrupting files, but they are also used to deceive information and steal money. They usually appear as an executable file (.exe). The most popular examples of viruses on Android are: Universal Cross-Site Scripting Attack (UXSS), Malware Hidden in Downloaded Applications, Lasco, Command & Control (C & C), Card-Block, CardTrap Android Installer Hijacking and Crossover. [13]
- Worms: Worm can replicate and disperse across devices to devices without any user interaction to perform. Worms infect entire device networks, either locally or over the internet using network interfaces. They use each infected machine to infect more others. They are usually received by SMS, MMS or another digital me-dia. The most popular example of the worm on android is the ADB.Miner An-droid. [13]
- Trojan: This type of malware pretends to be a legitimate program or hides in an original program that has been breached. They need to be installed by the user unlike worms. Once installed, Trojans can steal passwords, disable certain apps or lock the mobile device for a certain period. The most popular examples of Trojan are MasterKey, Fake-Player, GantSpy, DownAPK, etc. [13] [15][16]
- Spyware: This is malware designed to spy. It hides in the background and records online activities including passwords, credit card numbers, browsing routine and more. Exploiting vulnerabilities is the most important Spyware import goal. An example of spyware on Android is RedDrop. [13][15] [17].
- Ransomware: This malware is used to lock the device, and to unlock a payment is required to access your data. After payment the malware disappears. The most

popular ransomware Android malware is Xbot, Simpllocker FakeDefender [18] and adultPlayer. [13] [15]

- Botnets: Botnets often use special trojans to breach the security of mobile devic-es. The botnet is a piece of code used to "turn" a device into a bot without the user's consent, then these bots are all connected and thus form a "bot-network" or "botnet". Its purpose is to collect information. The best-known botnets in the Android operating system are Geinimi, Beanboot and DoubleDoor. [13] [18] [43]
- Rootkit: Rootkit is software designed to hide or conceal the existence of certain normal detection methods or processes, Rootkit also has administrative access to run various malicious applications to steal harmful action information and edit the configuration of the rootkit. system. There are some examples of Rootkit malware, but the most popular on Android are Godless, HummingBad, and Checkpoint. [13] [19]
- Backdoor: Backdoor is used to open any port for other applications and is a method that both authorized and unauthorized users can bypass normal security measures and gain high-level access to a computer system, network, or software application. When malicious as by unauthorized users becomes very dangerous malware. Brador is very popular backdoor malware. [13] [20]
- Keylogger: Keyloggers are applications that, once installed on a system, run to monitor all keypad entries. Then you can consult everything that was typed. The most common Malware Key-Logger on Android is FlexiSpy, mSpy. [13] [21]

5 Malware Detection Techniques

Malware in the android operating system is increasing day by day, which leads to a greater need to detect it to make it more secure. Very often mobile devices exhibit abnormal behavior that is driven by malicious malware that can harm the user in very dangerous ways, such as sending the user's private information to an unknown server. The techniques used to detect malware can be broadly categorized into two categories which are Anomaly-based detection and Signature-based detection. In fig.1 is represented the structure.

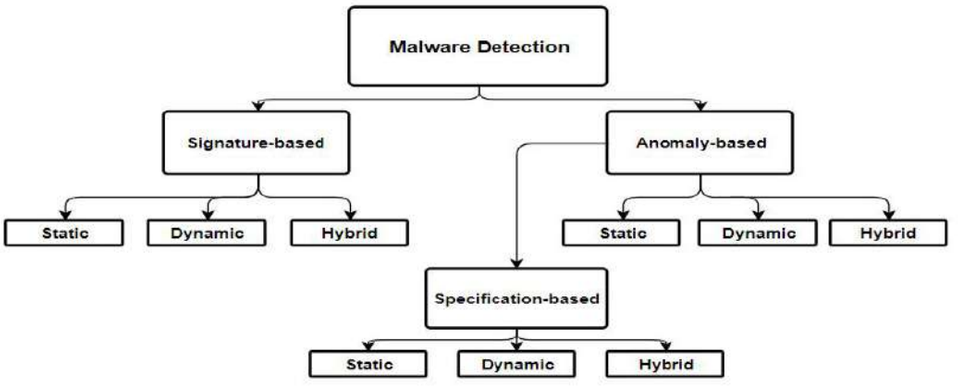

Fig. 1. A classification of malware detection techniques.[21]

5.1 Anomaly-based detection

This technique uses your knowledge of what constitutes normal behavior to decide whether the program under inspection is malware, tracking different parameters and the status of device components. Anomaly-based detection usually needs to work on a statistically significant number of packages, because any package is just an anomaly compared to some baselines. A special type of anomaly-based detection is called Specification-based detection. Specification-based techniques take advantage of some specifications or rulesets of what is valid behavior to decide whether the program under inspection is malware, programs that violate the specification are considered anomalous and generally malicious. [21] [22]

5.2 Signature-based detection

A signature is a sequence of bytes extracted from previously known malware, i.e. it uses the characterization of what is known as malicious to decide the maliciousness of a program under inspection, so if that pattern or signature is discovered again, the file may be marked as infected. This technique becomes a bit more limited as you always must access the signature database for regularly updating newly created signatures, but it offers higher malware detection accuracy. [21] [22]

5.3 The Comparison of different approaches for malware detection

Specific analysis of an anomaly or signature-based technique is determined by how the technique gathers information to detect malware. Each detection technique can employ one of three different approaches: static, dynamic or hybrid. A static approach attempts to detect malware before program execution under inspection; The dynamic approach attempts to detect malicious behavior during program execution or after program execution; The hybrid combines both ways, so static and dynamic information is used to detect malware. [21] [22]

A. Static Analysis: The static analysis deals with the features which are extracted from the application file without executing. The most popular dodging technique is known as Update Attack in which the malicious content is downloaded and installed as part of the update. This is not possible to detect by static analysis techniques. Permission and API calls are the most common features of static analysis as extracted from AndroidManifest.xml, thus can influence the malware detection rate to a high level studied by various researchers especially on meta-data available in Google Play Store.[23] Table 1 shows the advantages and disadvantages of static analysis.[21][24]

Advantages	Disadvantages
-It allows a complete analysis of a given; - It can cover all possible execution paths of a malware sample; - It is generally safer than dynamic approach as the source code is not actually executed.	-It is ineffective against previously unseen attacks and hence it cannot detect new and unknown intrusion methods as no signatures are available for such attacks. – The source code of malware samples is not readily available – It can be extremely timeconsuming and awkward process

Table 1. Advantages and Disadvantages of Static Analysis [22]

B. Dynamic Analysis: Dynamic analysis is a dynamic behavioral detection method that builds the operating environment using a sandbox, virtual machine, and so on, and simulates application execution to acquire the application behavior model. The goal is to find errors in a program while it is running, rather than repeatedly scanning offline code.[24] Table 2 presents some advantages and disadvantages of dynamic analysis.[21]

Advantages	Disadvantages
- It can avoid obfuscation issues, so it is easy to see the actual behavior of a program. - It can detect new intrusion method and can detect new malware	- The main drawback is that usually it monitors only one execution path, so it suffers from incomplete code coverage. - There is also the danger of harming third party systems, if the analysis environment is not properly isolated or restricted respectively. - Furthermore, malware samples may alter their behavior or stop executing at all once they detect to be executed within a controlled analysis environment.

Table 2. Advantages and Disadvantages of Dynamic Analysis [22]

C. Hybrid Analysis: Hybrid analysis is a combination of static and dynamic analysis. It is a technology or method that can integrate runtime data extracted from dynamic analysis into a static analysis algorithm to detect malicious behavior or functionality in applications. The hybrid analysis method involves the combination of static resources obtained by analyzing the application and the dynamic resources and extracted information as the application runs. It uses advantages and reduces the disadvantages of both dynamic and static analysis. [22][21][24]

6 Real Case of Ransomware

A new ransomware has emerged which is propagated via SMS message, which was detected by ESET Mobile Security as Android / Filecoder.C. This new ransom-ware has been distributed through various online forums and affects An-droid versions 5.1 and up. It uses the victims contact list and thus propagates via maliciously linked SMS to all contacts listed on the device. Once the malicious SMS messages are sent, the threat encrypts most files on the user's device and requests a ransom.[26]

6.1 Distribution

This malware is distributed through attractions created by these attackers, for example, with pornography-related publications. In all comments or posts made (in this case on reddit), attackers included links or QR codes that were directed to malicious applications. In fig.2 an example is presented. [26]

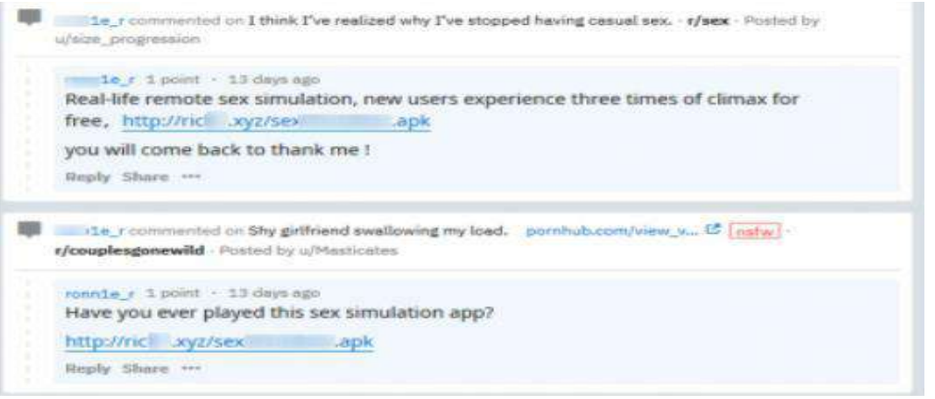

Fig. 2. Comments made on Reddit [26]

The Android / Filecoder.C ransomware to increase credibility, presents a link depending on the theme that is created as bait. In fig. 3 is an example of a link that appears as if it belonged to an application that allegedly uses the victim's photos. To maximize range ransomware has a model of the same message in several different languages. Before the message is sent, the threat chooses the version that matches the victim's device's language setting.[26]

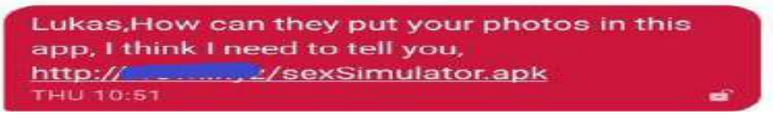

Fig. 3. An SMS with a link to the ransomware.[26]

10.11228/dpsc.02.01.010

After the victims receive the SMS message with the link to the malicious application, it must be installed manually. After the application starts, as promised in the reddit comments, it shows an online simulation game, in this sexual case, but the main goal is to communicate with C&C (command & control), propagating messages. malicious and implementing the encryption / decryption mechanism. Ransomware can send text messages because it has access to the user's contact list. The ransomware then passes through files located in accessible storage and encrypts most of them. After encrypting the files, ransomware displays your ransom note, as shown in Figure 4. [26]

Fig. 4. Rescue message presented by the Android/Filecoder.C. [26]

6.2 File Encryption Engine

Ransomware uses asymmetric and symmetric encryption, generating a public and private key pair. The private key is encrypted using the RSA algorithm with the hard-coded value stored in the code and sent to the attacker's server. To encrypt the files, ransomware generates a new AES key for each file that will be encrypted. This AES key is encrypted using the public key and is placed before each encrypted file, resulting

in the following pattern: "((AES) public key + (File) AES). seven ", Fig. 5 illustrates in a more exemplary manner the above pattern. [26]

Fig. 5. Overview of the structure of encrypted files.[26]

6.3 Decryption Engine

The code to decrypt encrypted files is present in ransom-ware. If the victim pays for the ransom, the ransomware operator can verify it through the website shown in Figure 6 and send the private key to decrypt the files. [26]

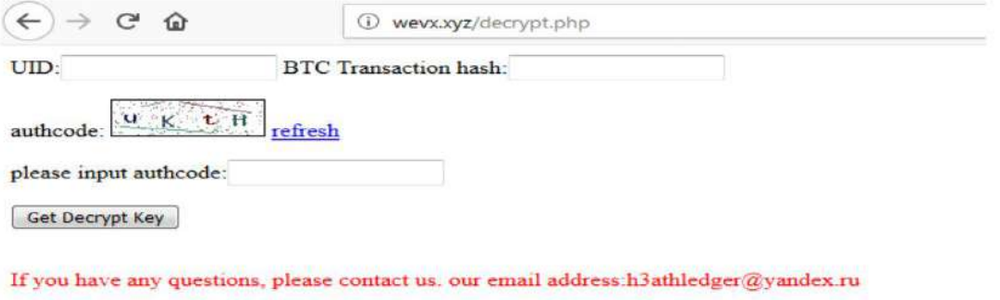

Fig. 6. Redemption payment verification web page.[26]

6.4 How to be protected

There are a few ways to prevent these attacks, which every user should do, which are: to keep devices up to date; download apps only through official stores; Before installing any app, check its rating and comments; Verify the permissions requested by the application; and use a mobile security solution.[26]

7 Conclusion

Most users think that mobile devices are a 100% safe device, or that only malicious cases happen to others, but it is not the best way to think when we talk about technology.

In this article, some security challenges for mobile users have been described, as a secure or informed user is much harder to fool than the uninformed one. Also introduced was the Android architecture, which is the system most subject to vulnerabilities derived from its features, not to mention that increasingly there are mobile banking applications that also catch the attention of hackers. The main point of this paper is presented in section 5, which analyzes the most well-known detection techniques, namely Signature Based Detection and Anomaly Based Detection, which are divided into 3 types which are static, dynamic and hybrid, based on detection. in anomaly a special type that is Specification Based Detection. We hope that with this article mobile users will get more information to be properly informed about possible attacks, and not an easy target to deceive, such as the real case of ransomware (section 6)

References

1. Tecmundo, https://www.tecmundo.com.br/celular/117849-5-bilhoes-pessoas-usam-celular-mundo-pesquisa.htm, [Last visited (20/11/2019)]
2. Tecmundo,https://www.tecmundo.com.br/dispositivos-moveis/141038-android-tem-2-5-bilhoes-usuarios.htm, [Last visited (20/11/2019)]
3. Zdnet, https://www.zdnet.com/article/mobile-malware-attacks-are-booming-in-2019-these-are-the-most-common-threats/, [Last visited (20/11/2019)]
4. Positive Technology, https://www.ptsecurity.com/ww-en/analytics/mobile-application-security-threats-and-vulnerabilities-2019/, [Last visited (22/11/2019)]
5. Appknox,https://www.appknox.com/blog/understanding-owasp-top-10-mobile-client-side-injection, [Last visited (22/11/2019)]
6. Khan, J., Abbas, H., & Al-Muhtadi, J. Survey on mobile user's data privacy threats and defense mechanisms. Procedia Computer Science, 56, 376-383, (2015).
7. Appknox, https://www.appknox.com/blog/understanding-owasp-top-10-mobile-improper-session-handling, [Last visited (22/11/2019)]
8. Varonis, https://www.varonis.com/blog/brute-force-attack/, [Last visited (23/11/2019)]
9. Hur, J. B., & Shamsi, J. A. A survey on security issues, vulnerabilities and attacks in Android based smartphone. In International Conference on Information and Communication Technologies (ICICT) (pp. 40-46), (2017, December).
10. Android Developers, https://developer.android.com/guide/topics/permissions/overview, [Last visited (27/11/2019)]
11. Android Open Source Project, https://source.android.com/security/app-sandbox, [Last visited (27/11/2019)]
12. Davi, Lucas, Alexandra Dmitrienko, Ahmad-Reza Sadeghi, and Marcel Winandy. "Privilege escalation attacks on android." In International Conference on Information Security, pp.346-360, (2010).
13. Khan, M.; Tripathi, R.; Kumar,A.; “A Malicious Attack and Defense Techniques on Android-Based Smartphone Platform”, Internation Journal of Innovative Technol-ogy and Exploring Engineering (IJITEE), Volume 8, Issue-8S3, (2019, June)
14. AVG, https://www.avg.com/pt/signal/what-is-malware, [Last visited (29/11/2019)]
15. Avast Blog, https://blog.avast.com/pt-br/como-detectar-e-remover-um-virus-do-seu-telefone-android, [Last visited (06/12/2019)]
16. Tecmundo, https://www.tecmundo.com.br/seguranca/196-o-que-e-um-trojan-.htm, [Last visited (06/12/2019)]

17. TechTudo, https://www.techtudo.com.br/noticias/2019/07/o-que-e-spyware-entenda-como-age-o-app-espiao-e-veja-como-se-proteger.ghtml, [Last visited (10/12/2019)]
18. Symantec, https://www.symantec.com/connect/blogs/simplocker-first-confirmed-file-encrypting-ransomware-android, [Last visited (12/12/2019)]
19. Canaltech, https://canaltech.com.br/seguranca/O-que-e-rootkit/, [Last visited (13/12/2019)]
20. Malwarebyte, https://www.malwarebytes.com/backdoor/, [Last visited (13/12/2019)]
21. Idika, N., & Mathur, A. P. A survey of malware detection techniques. Purdue University, 48, 2007-2, (2007).
22. Baraiya, D., & Diwanji, H. A Survey on Android Malware and Malware Detection Techniques, pp.47-53, (2016).
23. RIASAT, R., SAKEENA, M., Chong, W. A. N. G., SADIQ, A. H., & WANG, Y. J. A Survey on Android Malware Detection Techniques. DEStech Transactions on Computer Science and Engineering, (2016).
24. Rao, V., & Hande, K. A comparative study of static, dynamic and hybrid analysis techniques for android malware detection. Int. J. Eng. Dev. Res (IJEDR), 5,pp.1433-1436, (2017).
25. Avg, https://www.avg.com/pt/signal/what-is-malware, [Last visited (29/11/2019)]
26. Welivesecurity,https://www.welivesecurity.com/br/2019/08/02/novo-ransomware-para-android-e-propagado-via-mensagens-sms/, [Last visited (19/12/2019)]