Android Botnets on the Rise: Trends and Characteristics

Heloise Pieterse
Defence, Peace, Safety and Security
Council for Scientific and Industrial Research
Pretoria, South Africa
hpieterse@csir.co.za

Martin S Olivier
Department of Computer Science
University of Pretoria
Pretoria, South Africa
molivier@cs.up.ac.za

***Abstract*—Smartphones are the latest technology trend of the 21st century. Today's social expectation of always staying connected and the need for an increase in productivity are the reasons for the increase in smartphone usage. One of the leaders of the smartphone evolution is Google's Android Operating System (OS). The openness of the design and the ease of customizing are the aspects that are placing Android ahead of the other smartphone OSs. Such popularity has not only led to an increase in Android usage but also to the rise of Android malware. Although such malware is not having a significant impact on the popularity of Android smartphones, it is however creating new possibilities for threats. One such threat is the impact of botnets on Android smartphones. Recently, malware has surfaced that revealed specific characteristics relating to traditional botnet activities. Malware such as Geinimi, Pjapps, DroidDream, and RootSmart all display traditional botnet functionalities. These malicious applications show that Android botnets is a reality. From a security perspective it is important to understand the underlying structure of an Android botnet. This paper evaluates Android malware with the purpose of identifying specific trends and characteristics relating to botnet behaviour. The botnet trends and characteristics are detected by a comprehensive literature study of well-known Android malware applications. The identified characteristics are then further explored in terms of the Android Botnet Development Model and the Android Botnet Discovery Process. The common identified trends and characteristics aid the understanding of Android botnet activities as well as the possible discovery of an Android bot.**

***Keywords*—Android, botnet, malware, application, trends, characteristics.**

I. Introduction

The popularity of smartphones is continuously on the rise in the 21st century. Demonstrating functionality similar to that of a traditional computer, it is difficult to believe that these devices have been around for only two decades. The latest smartphones is an all-in-one portable device that combines the best features of both cell phones and computers.

Smartphones are classified according to the operating system (OS) installed on the device. The most prominent OSs include iPhone OS, Google's Android OS, Blackberry's RIM OS and Microsoft Windows's Mobile OS. Android is currently the leader in the market with 56.1% smartphone sales during the first quarter of 2012 [1]. The popularity of Android is mostly due to a lack of new products on alternative OSs [1].

The popularity of Android has lately come under threat due to a sudden rise in malicious applications. During 2010 a malicious application, named FakePlayer, found a way onto the Android platform by infecting a Movie Player application [2]. The escalation of Android malware since 2010 has been swift, with a rise of 13% in only 14 months [2]. This rise of malware does not have an obvious impact on the popularity of smartphones using Android OS but is creating possibilities for new threats.

Botnets are a well-known threat to computers and computer networks. Traditional botnets are responsible for spam delivery [3], distributed denial of service (DDoS) attacks and stealing personal information [4]. Botnets commonly make use of a command and control server and communicate using covert channels such as IRC (Internet Relay Chat) and P2P (Peer-to-Peer) overlays [4]. The above mentioned capabilities are regarded as traditional botnet functionality. Recently Android malware appeared that are demonstrating characteristics similar to that of traditional botnets.

Well-known Android malware such as Geinimi, Pjapps, DroidDream and RootSmart all display traditional botnet functionality. The above mentioned malware appeared on both third-party application markets as well as the official Android market [5]. It is therefore becoming important to create awareness of the threats posed by Android botnets.

An Android botnet is a network consisting of compromised Android smartphones controlled by a botmaster through a command and control (C & C) network [6]. To create awareness of the threats posed by Android botnets, it is necessary to study the underlying structure of Android botnets. Research conducted recently on Android malware only explores characteristics specific to the malware being evaluated and do not identify common characteristics.

This paper evaluates past and present Android malware that display traditional botnet functionality. The purpose of the evaluation is to identify common trends and characteristics between the malware. The evaluation specifically focuses on the code, structure and behaviour of the malware. The final result is a collection of common trends and characteristics pertaining to Android botnets. These trends and characteristics can aid the discovery of new Android botnets.

The remainder of this paper is structured as follows. Section II provides a short overview of the evolution of Android malware. Section III focuses on the identification of trends and characteristics of Android botnets while section IV provides a discussion on the future of Android botnets and possible prevention measures. Section V concludes the paper.

II. Short History of Android Malware

The use of the Android OS has grown rapidly since the first release in September 2008 [7]. Parallel to this growth is the rise of Android malware, which started appearing in August 2010 [2]. The increase in Android malware since 2010 has been significant, with new malware appearing at regular intervals.

Dennis Maslennikov discovered the first SMS Trojan for Android in August 2010 [7]. The Trojan, commonly known as FakePlayer, appeared as a legitimate Movie Player application with a fake Microsoft Windows Media Player icon [2] [7]. The application sent SMS messages to premium-rate numbers without the user's consent [7].

In the same month Symantec discovered the first GPS Spy malware [7] [8]. The malware, disguised as a classic snake game, had the ability to collect and send GPS coordinates to a remote server [7]. This malware was classified as low-risk because of the limited spreading capabilities it presented but it still displayed an important step forward for Android malware evolution.

On 29 December 2010, a new Trojan, named Geinimi, was discovered affecting Android devices [9]. Geinimi is the first malware to display traditional botnet functionalities [8]. The malware is responsible for collecting personal information and forwarding the collected information to a remote server [9]. One significant improvement of the Geinimi malware is the ability to infect legitimate applications [7].

Pjapps is another example of an application containing a Trojan which displays traditional botnet functionality. Pjapps is bundled with applications located on unofficial third-party application markets. The malware allow for the opening of a backdoor on the infected device and so receive commands from a remote server [8].

The first quarter of 2011 saw the arrival of a new generation of Android malware called DroidDream [8]. DroidDream infected more than 50 applications on the official Android Market [2] [5]. The malware exhibited complex functionality such as data theft, root exploits and botnet functionalities [2]. The ultimate goal of the DroidDream malware was to establish a botnet [10] and by affecting nearly 200 000 users, the goal was plausible [7].

ZitMo (Zeus in the Mobile) first appeared on Android devices during July 2011 [8]. It infects legitimate applications and works together with the Zeus banking Trojan to steal banking information [11]. The ZitMo malware intercepts and forwards all SMS messages to a command and control server [11]. ZitMo is a classic example of a Man-in-the-Mobile attack [11].

Towards the end of 2011, the NickiBot malware was discovered [8]. SMS messages are used to command and control this malware [12]. NickiBot supports location monitoring, sound recording and call-log collection [12]. Thus far this malware has only been found on unofficial Android markets [12].

In the first quarter of 2012 a new Android threat emerged. The malware, called RootSmart, interfaces with a botnet called Android.Bmaster. The malware has the ability to gain root access on devices running Android Gingerbread versions before 2.3.4 or Android 3.0. Currently researchers have only identified the malware on unofficial third-party Android markets [13].

The growth of Android malware has been significant, with every new malware that appears showing an improved ability over the previous malware. In the next section, the mentioned Android malware is evaluated and trends and characteristics are identified.

III. Ease of Use

A. Trends of Android Botnets

The sophistication with which Android botnets are developing is increasing at a rapid pace. Although similar techniques are continuously used, the method in which these techniques are applied is constantly changing. It is therefore necessary, from a security perspective, to create awareness of the changes in Android Botnet development.

The first trend of Android botnet development arrived in the form of a simple SMS Trojan. This Trojan, included in a repackaged version of a legitimate application, was primarily responsible for sending SMSs to premium rate numbers. The Trojan did not yet display substantial botnet functionalities but showed the possibility of malware running inconspicuously on Android devices.

Soon afterwards malware started appearing on Android devices that included the ability to communicate to a remote server. This remote server, more commonly known as a C & C server, is responsible for receiving information from the infected Android device as well as sending commands to the device. This particular trend in the evolution of malware is the first to show traditional botnet functionality.

In addition to the communication with a C & C server, Trojan applications install additional, but malicious, applications. The downloading of a malicious application takes place either dynamically or the user is prompt to do the installation. The increase of malware functionality shows that botnets on Android devices is a possibility.

With Android botnets becoming a possibility, the focus shifted towards exploits that can improve the functionality of the Trojan malware. A well-known exploit is the 'rage against the cage' exploit that allows a user to acquire root access on carrier locked Android devices [14]. Such exploits lead to new possibilities for Android botnet evolution.

For many months, Android malware mostly circled around unofficial third-party application markets. In recent times malware has managed to slip past the security doors of the Android into their Official Market. One of the first malware to do so is the DroidDream malware [5]. By being able to infect applications in the Official Android Market allows for more efficient spreading of botnet malware.

The latest trend in Android Botnet development is the use of SMSs to receive botnet commands. The traditional use of IRC and HTTP-controlled botnets has become impractical to use on mobile devices. SMS, which is available on most mobile devices, provides improved possibilities for C & C [15].

B. Characteristics of Android Botnets

In order to identify possible Android characteristics, the following Android malware are evaluated: BaseBridge, BgServ, DroidDream, DroidKungFu, Geinimi, LeNa, Nickispy, Pjapps, RootSmart and SMSspacem [16]. Additional malware that was also evaluated include ADRD (also known as HongTouTou), DroidDreamLight, Tonclank and Golddream [8]. By evaluating the technical reports of the above mentioned Android Malware, it allowed for the identification of common characteristics among the malware. The characteristics include the following: repackaging an application, receiving commands, messaging, stealing information, applications found on third party application markets, downloading additional content and modifying the Android Manifest file. Certain of these identified characteristics relate closely to traditional botnet functionality, such examples are the receiving commands and stealing information characteristics. It is therefore possible to use these identified characteristics to detect botnets on Android devices.

1) Repackaged Application

The distribution of malicious code to drive a botnet usually takes the form of an application. These applications are well-known and legitimate but an attacker reverse engineered and repackaged the original code with additional malicious code. A user installs the application but is unaware of the additional configurations taking place on the device. This characteristic is similar to that of a Trojan horse and is the most common method to distribute botnet code.

2) Receiving Commands

An essential characteristic of any bot is the ability to either receive command automatically or to prompt a remote server for the commands. The current techniques used by Android botnets are very similar to these traditional techniques. The first option is to send the commands directly from a C & C server to the Android bot as needed. The other option is to allow the Android bot to contact the C & C server at regular intervals and ask whether new commands are available. Any contact with a remote server is an obvious indication that a possible Android botnet is at work.

3) Messaging

The traditional notion of a botnet is either to cause destruction at a particular level or for monetary purposes. Current Android botnets are exploiting SMS messages to gather money by sending messages to premium-rate numbers. These premium-rate numbers are phone numbers, used for a certain service and are charged at a higher rate than normal phone calls [17]. By sending SMS messages at regular intervals to such numbers, the botnet can generate substantial amounts of money for its operators.

4) Steal Information

Android botnets do not only receive information from a C & C server but also upload information about the infected device to the server. This type of activity occurs usually after the installation of the malicious application. Information commonly collected by Android botnets can possibly include the following:

- IMEI (International Mobile Equipment Identity) number
- IMSI (International Mobile Subscriber Identity) number
- GPS Location
- Phone Number
- SDK (Software Development Kit) version
- Device Model
- Installed Packages

The above stolen information aids the botmaster in uniquely identifying and controlling a bot.

5) Third Party Application Markets

Traditionally, malicious applications only appeared on unofficial third party application markets. This is no longer the case as malicious applications have surfaced on the Official Android Market recently. The DroidDream malware is one such example [5]. Although the chances are slim of locating a malicious application on the Official Android Market, caution must still be exercised.

6) Additional Content Downloaded

The latest characteristic of Android botnets is the ability to download additional content. This content, usually malicious in nature, aids and improves the performance of the botnet. The additional content is either downloaded dynamically by the application or a prompt asks the user to perform the necessary download.

7) AndroidManifest.xml File: Features and Permissions

Every Android application includes the AndroidManifest.xml file in the root directory [18]. This file presents essential information about the particular application to the Android system [18]. Some of the elements included in the structure of the AndroidManifest.xml file is the <uses-feature> and the <uses-permission> elements [18]. The <uses-feature> element declares a single hardware or software feature used by the application [19]. Android botnets commonly use the following features:

- android.hardware.telephony
- android.hardware.touchscreen
- android.hardware.location
- android.hardware.wifi

All of the above features are self-explanatory and allow the Android botnet improved control over the infected device.

The <uses-permission> element requests a permission that the application requires in order to operate correctly [20]. Android botnets commonly use the following permissions:

- android.permission.READ_CONTACTS
- android.permission.WRITE_CONTACTS

- android.permission.SEND_SMS
- android.permission.WRITE_SMS
- android.permission.READ_SMS
- android.permission.RECEIVE_SMS
- android.permission.READ_PHONE_STATE
- android.permission.INTERNET
- android.permission.WRITE_INTERNAL_STORAGE

The AndroidManifest.xml file provides valuable information to the user about a particular application and contains identifiable characteristics of an Android botnet.

IV. Discussion

Android botnets recently discovered on Android devices already display significant capabilities. As with all other malware, the developers of Android botnets are remaining ahead of the mobile security curve and therefore the future developments of Android botnets look bright.

Thus far security analysts have identified few countermeasures against the threats posed by Android botnets even though the developers continuously use the same techniques to develop the botnets. These techniques are indeed identifiable characteristics, as described in the previous section, and can become valuable detection mechanisms to use during the analysis of an application showing botnet functionalities. The remainder of this section will refer to the seven characteristics as the Android botnet characteristics.

The Android Botnet Development Model describes the phases through which the botmaster iterates to develop an Android botnet and during each phase one or more Android botnet characteristic are formed. The following phases form the Android Botnet Development Model: Infection, Propagation, and Execution.

During the Infection phase, a botmaster alters a legitimate application to allow for the accommodation of the malicious bot code. The botmaster will thus make changes to the code structure of the application by adding additional files and code snippets. Such changes lead to the development of a repackaged application, which is the first characteristic of the Android botnet characteristics. For the changes to the code structure to function in the new application, the botmaster must update the AndroidManifest.xml file accordingly. Any changes to this file form the seventh Android botnet characteristic.

After successful infection of a legitimate application, the botmaster proceeds to the Propagation phase to enable the spreading of the newly infected application. Such an application will serve no purpose if it is unable to propagate to other Android devices and will also limit the growth capability of the Android botnet. Therefore the botmaster returns the newly infected application to an application market for propagation. This phase leads to the fifth Android botnet characteristic (Third Part Application Markets).

The last phase of the Android Botnet Development Model is the Execution phase during which the Android Botnet will serve its purpose. The purpose of the Android botnet can have multiple possibilities including denial-of-service attacks, information stealing, SMS messaging or receiving commands. The remainder of the Android botnet characteristics emerge during the last phase.

A security analyst faces an enormous task when he/she needs to reverse engineer an Android application to determine whether it is malicious botnet or not. This task can become a time consuming process if the security analyst evaluates every line of code without a definite starting point. Since the Android botnet characteristics are formed during the development of an Android botnet, these characteristics can therefore aid the discovery of an Android botnet. Thus the Android botnet characteristics become detection mechanisms that a security analyst can use during the Android Botnet Discovery Process. The Android Botnet Discovery Process describes the steps a security analyst can follow to determine whether a certain application poses any threats relating to that of botnets. The steps followed in the Android Botnet Discovery Process includes: Locate, Explore and Identify.

In order to locate possible malicious Android applications, the security analyst uses the fifth Android botnet characteristic and selects Android applications from Third Party Application Markets. After selecting an application, the security analyst will explore the application and determine whether the application being investigated is a repackaged application or not. To identify a repackaged application the security analyst can follow the prototype developed in [21]. If is indeed a repackaged application, the security analyst next explores the AndroidManifest.xml file. The analysis of the AndroidManifest.xml file and the permissions defined within this file can lead the security analyst to the identification of possible threats posed by the application. For example, Android permissions such as:

- RECEIVE_SMS, INTERNET, and READ_SMS show the possibility of the application receiving commands.
- RECEIVE_SMS, WRITE_SMS, SEND_SMS, and READ_SMS show the possibility of the application sending out SMS messages.
- READ_CONTACTS, READ_SMS, and READ_ PHONE_STATE show the possibility of the application stealing information from the Android device.
- INTERNET, WRITE_EXTERNAL_STORAGE show the possibility of the application downloading additional content onto the Android device.

As Identifying these subsets of Android permissions do not necessarily refer to a threat as these permissions can be part of the original application. It however allows the security analyst to only search for code relating to the above mentioned characteristics rather than evaluating all of the code structures. Only then by assessing the specific code snippets can the security analysts conclude whether the application poses any threats relating to botnets and what malicious tasks the application may perform.

Although these Android botnet characteristics are valuable as detection mechanisms, it is still best to only download from trusted application markets. Additional defensive techniques that a person can follow include the checking of the permissions of Android applications and to be constantly aware of the behaviour of the device and any unusual activities. Then the oldest possible defensive technique is the use of antivirus or mobile security applications.

V. Conclusion

As smartphones are becoming more popular, they become the targets for potential attacks. With the openness of the design of the Android OS and its increasing popularity, a growth in Android malware can be expected. In this paper specific trends and characteristics of Android botnets were identified. The characteristics identified are the following: Repackaged Applications, Receiving Commands, Messaging, Steal Information, Third Party Application Markets, Additional Content Download and AndroidManifest.xml File: Features and Permissions. The characteristics were described in terms of the Android Botnet Development Model and the Android Botnet Discovery Process. These mentioned characteristics can then aid the identification of current Android botnets as well as prevent the rise of new Android botnets. Future research includes the advance study of the internal workings of current Android botnets and malware. The purpose of this research is to explore the development and the underlying structure of Android botnets to aid the discovery process of such botnets. The future focus will be on the identification of Android botnets by means of a signature-based and/or a behavior-based detection model.

References

[1] L. Goasduff, and C. Pettey, "Gartnet Says Worldwide Sales of Mobile Phones Declined 2 Percent in First Quarter of 2012," (Gartner Newsroom), [online] 2012, http://www.gartner.com/it/page.jsp?id=2017015 (Accessed: 3 July 2012).

[2] T. Armstrong, "Android malware is on the rise," (Kaspersky), [online] 2011, http://www.virusbtn.com/pdf/conference_slides/2011/Armstrong-Maslennikov-VB2011.pdf (Accessed: 4 April 2012).

[3] Y. Xie, F. Yu, K. Achan, R. Panigrahy, G. Hulten, and I. Osipkov, "Spamming botnets: signatures and characteristics," *ACM SIGCOMM Computer Communication Review*, vol. 38, (4) pp. 171-182, 2008.

[4] F. Giroire, J. Chandrashekar, N. Taft, E. Schooler, and D. Papagiannaki, "Exploiting temporal persistence to detect covert botnet channels," *Recent Advances in Intrusion Detection*, pp. 326-345, 2009.

[5] T. Bradley, "DroidDream becomes android market nightmare," (PCWorld), [online] 2011, http://www.pcworld.com/businesscenter/article/221247/droiddream_becomes_android_market_nightmare.html (Accessed: 4 April 2012).

[6] G. Geng, G. Xu, M. Zheng, Y. Gou, G. Yeng, and C. Wei, "The design of sms based heterogeneous mobile botnet," *Journal of Computers*, vol. 7, (1) pp. 235-243, 2012.

[7] C.A. Castillo, "Android malware past, present, and future," (McAfee), [online] 2010, http://www.mcafee.com/us/resources/white-papers/wp-android-malware-past-present-future.pdf (Accessed: 1 March 2012).

[8] P. Passeri, "One year of android malware," (Hackmageddon.com), [online] 2011, http://hackmageddon.com/2011/08/11/one-year-of-android-malware-full-list/ (Accessed: 2 March 2012).

[9] T. Wyatt, "Security alert: geinimi, sophisticated new android trojan found in wild," (Lookout Mobile Security), [online] 2010, http://blog.mylookout.com/blog/2010/12/29/geinimi_trojan/ (Accessed: 1 March 2012).

[10] Y. Zeng, "On detection of current and next-generation botnets," Ph.D. thesis, University of Michigan, Michigan, 2012.

[11] D. Desai, "Analysis of zitmo (zeus in the mobile)," *Hakin9*, vol. 4, (4) pp. 14-17, 2011.

[12] X. Jiang, "Security alert: new nickibot spyware found in alternative markets," [online] 2011, http://www.csc.ncsu.edu/faculty/jiang/NickiBot/ (Accessed: 12 March 2012).

[13] M.J. Schwartz, "Android botnet exploits gingerbread root access," (Information Week Security), [online] 2012, http://www.informationweek.com/news/security/mobile/232600576 (Accessed: 15 February 2012).

[14] T. Armstrong, "Malware in the android market," (Securelist), [online] 2012, http://www.securelist.com/en/blog/11184/Malware_in_the_Android_Market (Accessed: 2 March 2012).

[15] Y. Zeng, X. Hui, and K.G. Shin, "Design of sms commanded-and-controlled P2P-structured mobile botnets," *Tech.Rep.CSE-TR-562-10*, 2010.

[16] Current android malware, Forensics blog, [online] 2012, http://forensics.spreitzenbarth.de/android-malware/ (Accessed: 13 March 2012).

[17] Premuim rate numbers info, [online] 2010, http://www.premium-rate-numbers.info/2010/03/premium-rate-numbers.html (Accessed: 19 March 2012).

[18] The AndroidManifest.xml file, *Android Developers*, [online] 2012, http://developer.android.com/guide/topics/manifest/manifest-intro.html (Accessed: 19 March 2012).

[19] <uses-feature>, *Android Developers*, [online] 2012, http://developer.android.com/guide/topics/manifest/uses-feature-element.html (Accessed: 19 March 2012).

[20] <uses-permission>, *Android Developers*, [online] 2012, http://developer.android.com/guide/topics/manifest/uses-permission-element.html (Accessed: 19 March 2012).

[21] W. Zhou, Y. Zhou, X. Jiang, and P Ning, "Detecting repackaged smartphone applications in third-party android marketplaces," *CODASPY '12 Proceedings of the second ACM conference on Data and Application Security and Privacy*, pp. 317-326, 2012.