IOSR Journal of Computer Engineering (IOSR-JCE)
e-ISSN: 2278-0661,p-ISSN: 2278-8727,
PP 05-10
www.iosrjournals.org

Android Botnet: An Upcoming Challenge

Sangeeta Joshi[1], Dr Ravinder Khanna[2], Lalit Kumar Joshi[3]
[1] *(Department of Computer Science, Mata Gujri College, Fatehgarh Sahib, India)*
[2]*(Department of Electronics, Maharishi Markandeshwar University, Sadopur, India)*
[3] *(Department of Computer Science, Mata Gujri College, Fatehgarh Sahib, India)*

ABSTRACT: *Mobile devices are now a day well incorporated with advanced capabilities and technologies such as the Internet, GPS, and Bluetooth etc. Mobile security has become a globally sensitive issue because of the high usage of mobile devices. But as compared to computers and computer networks, they are less protected. The user's of these devices pay less attention to the security updates. The most popular mobile operating system today in the industry is Android. However, with the growing market share of Android smart phone, the malware (malicious software) writers have begun to target the Android operating system. Recently, the most dangerous threat which is targeting Android is 'botnets'. In this paper, we present an introduction of Android botnets and an analysis of currently available families of botnets. By analyzing these samples we reveal common characteristics and behavior of these botnets which will aid in the identification of new malware on Android devices.*

Keywords – *Android, Application, Botnet, Malware, Mobile,*

I. INTRODUCTION

The appearance of new technologies and advance features of mobile devices makes mobile communication an essential part of every aspect of human activity. Mobile devices are now well integrated with the Internet; therefore, with the increasing use of these devices on a global scale, mobile security has become a fundamental issue [1]. Mobile devices need to run lightweight operating system due to limited battery and storage constraints. Android has become the most popular operating system with 79 percent of mobile phone market share in 2013[2]. Along with different network security challenges like viruses, worms and trojans, botnets have become the most dangerous [3]. A bot, invented from the word 'robot', is an application that can perform and repeat a specific task much faster than a human. Botnet is collection of several bots connected with each others with the help of networks. The three basic elements of a botnet are - the bots, the command and control (C&C) servers, and the botmasters. Botnets gain the complete access to the device and its contents as well as providing control to the botnet creator. Mobile botnets provide botmaster with root permissions over the compromised mobile device, enabling botmaster to send mails or text messages, make phone calls, access contacts and photos, etc. Most of the time, mobile botnets perform malicious activities in hidden and spread themselves by forwarding their copies to other devices using messages and e-mails. Therefore, the detection of botnets has become a difficult issue in the field of mobile network security. This paper aims to provide an overview on Android botnets and their characteristics. Based on theses common characteristics analysis of some malwares having botnet functionality is done. These characteristics will also aid the discovery of new botnets present in the applications.

II. CHARACTERISTICS OF MOBILE BOTNETS

In order to identify possible botnet characteristics, the following Android botnets are evaluated: Zitmo, BaseBridge, AnServerBot, DroidDream, DroidKungFu, Geinimi, Nickispy, Pjapps, RootSmart, FakePlayer , SMSspacem, TigerBot and ADRD [4]. We downloaded the above malware samples from contagio mobile dump: a shared library of malwares for research purpose. We evaluated the malware samples using permission based filtering approach to identify common characteristics among the botnet. The common characteristics include the following: repackaging an application/software, receiving commands, messaging, stealing information, applications found on third party application markets, permissions, downloading additional content and updating device. Certain of these identified characteristics relate closely to traditional botnet functionality, such examples are the receiving commands, stealing information characteristics and downloading content. It is therefore possible to use these identified characteristics to detect botnets on Android devices. Table 1 shows malicious activities of some Android botnets.

Table.1. Examples of Android botnet activities.

Name	Activities
Zitmo	• Infected SMS Messages • Mobile Banking Attacks • TAC Thefts • Illegal Transactions
TigerBot	• Theft of Private Data • Change Device Setting
AnserverBot	• Theft of Private Data
Geinimi	• Send Email & SMS, • Make phone calls, • Update C&C address, • Add new Application Shortcuts
PjApps	• Send SMS • Theft of Private Data • Install a new application, • Open URL in phone browser
DroidDream	• Theft of Private Data • Download Malicious Applications • Attempts to Root Device.
RootSmart	• Revenue Generation • Theft of Private Data
DroidKungFu	• Download Malicious Applications • IMEI Theft
SMSspacem	• Send SMS to all contacts on phone containing an HTTP link.
FakePlayer	• Content delivery manipulation • Send SMS
ADRD	• Send IMEI/IMSI information to a remote server. • Receive commands from a remote server. • Download Packages. • Receive URL entry.

- Application Repackaging

All mobile operating system like Android, iPhone provide the user to download applications from their official stores. But in some operating systems like Android users can download applications from unofficial stores also. The distribution of malicious code to drive a botnet usually takes the form of an application. The botnet authors perform reverse engineering on these legitimate applications and repackage them with malicious code along with the original code. This is the most common method to distribute botnets. 'Pjapps' is an example of an application containing malicious programme which displays traditional botnet functionality. Pjapps is bundled with applications available on unofficial third-party application stores. The malware allow for the opening of a backdoor on the infected device and so receive commands from a remote server [5]. Another such example is "OPFAKE'. More than 500 OPFAKE variants can be downloaded from some third-party application stores. It contains malicious advertising code and sends messages to premium rate numbers.

- Receiving Commands

An important characteristic of a botnet is the ability to receive commands from a remote server. Command and control (C&C) methods are used as an interface to send botmasters' commands to targeted devices and receive responses from them [6]. The current techniques used by mobile botnets are very similar to these traditional techniques. The first option is to send the commands directly from a Command and Control(C & C) server to the bot as needed. The other option is to allow the bot to contact the C & C server at regular intervals and ask whether new commands are available. Any contact with a remote server is a clear clue that a mobile botnet is present in the device. 'AnserverBot' is one example of such mobile botnet. In addition to detecting and disable the security solution in infected device, the AnserverBot regularly checks its signature to verify its integrity in

order to protect itself from any type of changes. 'TigerBot' is a bot which is controlled by SMS instead of any web technologies. It detects the C&C messages and keeps them hidden from the mobile device owners. Instead of collecting private data like SMS messages, it records voice calls and even surrounding sounds.

- Messaging

The prime motive of traditional botnet is to make financial gains. Mobile botnets are using SMS messages to collect 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. By sending SMS messages at regular intervals to such numbers, the botnet can cause considerable amounts of money.'Zitmo' is a bot that infects different mobile operating systems, such as Symbian, Windows Mobile, BlackBerry, and Android. It mainly intercepts SMS's which are sent by banks to customers by stealing mobile Transaction Authentication Numbers (TAC). 'Android.Bmaster' has gained millions of dollars through premium SMS.

- Data Theft

Botnets sends information about the personal data to a remote server. This type of activity occurs usually after the installation of the malicious application. Information usually collected by botnets can probably include: IMEI (International Mobile Equipment Identity) number, IMSI (International Mobile Subscriber Identity) number, GPS Location, Phone Number, Device Model, Contact List, Installed apps, Email addresses, Browser histories and SDK Version. 'Geinimi' is the first malware to display traditional botnet functionalities. This malware collects personal information of the device and forwards this information to a remote server. 'PjApps.A' collects IMEI, IMSI, Phone Number, SMS service canter, ICCID information and sends it to a remote server. 'TigerBot.A' forward SMS and IMEI no to a remote server. Another fundamental characteristic of a botnet is location awareness, i.e. the ability to determine geographical position. Even if this feature has significant benefits, it raises some important privacy implications for users of mobile phones. A comprehensive survey on different privacy issues related to location-aware mobile phones is presented in [7].

- 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. The 'DroidDream' malware download another application on infected targets. These new downloads prevents 'DroidDream' from removal.

- Root Exploit:

With the growth of botnets, their motivation shifted towards exploits that can improve the functionality of the malware. A well-known exploit is the 'rage against the cage' exploit that allows a user to gain root access on targeted devices. Such exploits lead to new possibilities for mobile botnet evolution. The malware, called RootSmart, has the ability to gain root access on devices running some versions of Android operating system. 'DroidDream' infected more than 50 applications on the official Android store. With the functionalities such as data theft, downloading contents, it also takes over the root access of the device. The ultimate goal of the 'DroidDream' malware was to establish a botnet and by affecting almost 200 000 users [8]. The DroidKungFu malware also remount the system to gain the root access of the device.

- Third Party Application Stores

The easiest way to download applications is to go to the official store such as the Android Market or BlackBerry App World. But a complete universe of third-party, independent mobile app stores are out there, too. These stores sell apps across platforms and offer some unique features that you can't find in the official app stores. Be aware Android users can buy applications at third-party app stores in addition to the official stores on their phones. These stores contain huge amount of malware available for Android.

- Permissions

Every Android applications can define permissions that control the access to sensitive resources and functionalities. There are 130 Android defined permissions [9], amongst which 122 permissions are available to third party applications. The permissions are to be included in AndroidManifest.xml file. Android permissions are used to inform users about the security risks of installing applications [10] .But the researchers have analyzed that Android permission notifications are mostly ignored by users [11, 12]. The structure to define permissions in manifest file is as follows:

```
<manifest xlmns: android...>
...
<uses-permission android: name="android.permission.INTERNET" />
<application>
```

</manifest>

Android botnets commonly use the following permissions:

- READ_CONTACTS
- READ_PHONE_STATE
- ACCESS_FINE_LOCATION
- WRITE_CONTACTS
- SEND_SMS
- WRITE_SMS
- READ_SMS
- RECEIVE_SMS
- READ_PHONE_STATE
- INTERNET
- ACCESS_GPS
- INSTALL PACKAGES
- WRITE_INTERNAL_STORAGE
- EXTERNAL STORAGE

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

III. DISCUSSION

Recently discovered mobile botnets are becoming tough to takedown. Each new variant of these malwares use multiple C&C servers to run commands on the targeted device. The future of mobile botnets look bright as the prime motivation behind such threats is financial gains by sending messages to premium numbers or reading messages from the banks. As far the security measures are concerned there are some countermeasures against the threats posed by mobile botnets. To find out that whether an application is benign or having botnet functionalities, the characteristics described in the previous section, may prove to be valuable detection methods. The rest of this section will discuss Mobile Botnet Development Model (i.e. the stages through which a botmaster develop a botnet) and during each stage the characteristics of the botnet are emerged. The following are the phases as shown in Fig1 form the Android Botnet Development Model:

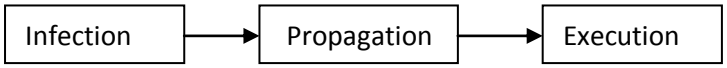

Fig.1. Android Botnet Development Model

The lifecycle of a botnet begins with the infection phase where a botmaster modifies a legitimate application to make room for malicious bot code [13]. The botmaster will get the source code of legitimate application using reverse engineering techniques. He will then make changes in the code by adding malicious contents. It will result into a repackaged application, which is the first botnet characteristic. The Propagation phase, which is next in turn, is all about the distribution of this repackaged application. In case of computer devices the infected code can be spread through emails, file sharing or malicious URLs etc. [14]. But in case of mobile devices the most common and effective transmission medium for botmasters is third party application stores. There are ample third party application stores available for Android devices. The developers can upload their applications on these stores very easily without any security botheration. Hence the botmaster uploads the repackaged application to an application store for propagation. This phase leads to the seventh botnet characteristic (Third Part Application Markets). The last phase of the Android Botnet Development Model is the Execution phase during which the botnet will accomplish its goal. The purpose of the botnet can have multiple possibilities including denial-of-service attacks, information stealing, SMS messaging or receiving commands etc. The rest of the botnet characteristics appear during the last phase. For malware identification in Android application, reverse engineering is performed. In reverse engineering some tools ApkTool, Dex2Jar, Notepad++, AndroGuard, etc are used to get the source code and then to scan every line of the code to conclude whether it is malicious botnet or not. This can be a time consuming task if the security analyst evaluates every line of code. But the common characteristics of botnet can therefore help the detection of mobile botnet easily. Thus the botnet characteristics can be used during the Botnet Discovery Process. The Botnet Discovery Process describes the steps a security analyst can pursue to determine whether a certain application benign or malicious relating to that of botnets. The steps followed in the Botnet Discovery Process [15] include: Find, Explore and Identify.

With the intention to find possible malicious applications, the security analyst uses the seventh botnet characteristic and selects an application from Third Party Application Stores. After downloading of the application, the security analyst can use prototype [16] to identify whether the application being investigated is a repackaged application or not. In case of repackaged applications by exploring permissions defined in AndroidManifest.xml can lead to the identification of possible threats posed by the application. For example, the following essential Android permissions are indicators for Android botnet:

- INTERNET, ACCESS_NETWORK_STATE RECEIVE_BOOT_COMPLETED: Application is gaining root access.
- INTERNET_RECEIVE SMS, SEND_SMS: Application is receiving commands.
- ACCESS_WIFI_STATE, CHANGE_ WIFI_STATE: Application is making changes on device.
- INTERNET, READ_ PHONE_STATE ,READ_CONTACTS: Application is stealing device information
- INTERNET, SEND_SMS: Application is sending SMSs.
- INTERNET, INSTALL_PACKAGES: Application is downloading packages.

Presence of these sets of permissions do not always indicate to a malware however it will help the security analyst to focus on the code (API calls) of the above mentioned characteristics rather than evaluating all of the code structures. By assessing the specific code area can the security analysts conclude whether the application poses any threats relating to botnets and what malicious behavior the application is performing? Although these botnet characteristics are very helpful, but the users should always download applications from official Android store (Google Play) as it is very secure. Additionally at the time of installation of an application a user must pay attention to the permissions asked by that application. Then the oldest possible defensive technique is the use of antimalware or mobile security applications. Users must regularly update their anti-malwares to identify new threats.

IV. CONCLUSION

As mobile phones are becoming more popular, they become the targets for malicious activities. In case of Android operating systems the availability of large number of application stores and very few restrictions for developer to upload their applications increases the security risk. Android botnet is a current challenge. In this paper we analyzed some popular malware families of Android that perform botnet type functionalities. Based on this analysis some common characteristics of mobile botnets were identified. These characteristics can help the identification of botnets as well as stop the rise of new botnets.

REFERENCES

[1] A. Flo and A. Josang, Consequences of Botnets Spreading to Mobile Devices, Proc. 14th Nordic Conference on Secure IT Systems ,(NordSec), 2009, pp. 37-43.
[2] Android climbed to 79 percent of smartphone market share in 2013, but its growth has slowed. [Online] http://www.engadget.com/2014/01/29/strategy-analytics-2013-smartphone-share/
[3] L. Jae-Seo, J. HyunCheol, P. Jun-Hyung, K. Minsoo, and N. Bong-Nam, The Activity Analysis of Malicious HTTP-Based Botnets Using Degree of Periodic Repeatability, Proc. of the International Conference on Security Technology (SECTECH), 2008, pp. 83-86.
[4] Current android malware, Forensics blog, [online] 2012, http://forensics.spreitzenbarth.de/android-malware/ (Accessed: 13 March 2012).
[5] C.A. Castillo, Android malware past, present, and future, (McAfee), [online] 2010, http://www.mcafee.com/us/resources/white-papers/wpandroid-malware-past-present-future.pdf
[6] M. Eslahi, R.Salleh and N.B Anuar, MoBots: A New Generation of Botnets on Mobile Devices and Networks, International Symposium on Computer Applications and Industrial Electronics (ISCAIE), Kota Kinabalu Malaysia, 2012, pp. 262-266.
[7] R. P. Minch, Privacy Issues in Location-Aware Mobile Devices, Proc. 37th Annual Hawaii International Conference on System Sciences (HICSS'04) - Track 5 - Volume 5. Washington, DC, USA: IEEE Computer Society, 2004, pp. 50 127.2.
[8] Y. Zeng, On detection of current and next-generation botnets, Ph.D. thesis, University of Michigan, Michigan, 2012.
[9] K. W. Y. Au, Y. F. Zhou, Z. Huang, P. Gill, and D. Lie, Short paper: a look at smartphone permission models, Proc. 1st ACM workshop on Security and privacy in smartphones and mobile devices, ser. SPSM '11, 2011, pp. 63–68.
[10] Azzife Khalo, A Guide to Understanding Android App Permissions (& How to Manage Them) [Online] http://www.hongkiat.com/blog/android-app-permissions/
[11] A. P. Felt, K. Greenwood, and D. Wagner, The Effectiveness of Application Permissions. Proc. Of the 2nd USENIX Conference on Web Application Development (WebApps), USENIX Association, 2012, pp. 75-86.

[12] W. Enck, M. Ongtang, and P. McDaniel, On lightweight mobile phone application certification, Proc. Of the 16th ACM Conference on Computer and Communication Security (CCS), New York, 2009, pp. 235-245.
[13] M. Bailey, E. Cooke, F. Jahanian, X. Yunjing, and M. Karir, A Survey of Botnet Technology and Defenses, in Proceedings of the Cybersecurity Applications & Technology Conference for Homeland Security (CATCH), 2009, pp. 299-304.
[14] N. Hachem, Y. Ben Mustapha, G. G. Granadillo, and H. Debar, Botnets: Lifecycle and Taxonomy, in Proceedings of the Conference on Network and Information Systems Security (SAR-SSI), 2011, pp. 1-8.
[15] H. PIETERSE AND M. S. OLIVIER, Android botnets on the rise: Trends and characteristics, In: Information Security for South Africa (ISSA), IEEE, 2012, pp. 1–5
[16] W. Zhou, Y. Zhou, X. Jiang, and P Ning, Detecting repackaged smartphone applications in third-party android marketplaces, Proc of the 2nd ACM conference on Data Application Security and Privacy, San Antonio, TX, USA, 2012, pp. 317-326.