International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 07 Issue: 05 | May 2020 www.irjet.net p-ISSN: 2395-0072

Android Spy Agent - Remote Access Trojan

Kshitij Barapatre[1], Prof. Priya Parkhi[2]

[1,2]Department of Computer Engineering, St. Vincent Pallotti College of Engineering & Technology. Nagpur, India

---

Abstract - *In the era of digitalization, the crime rates were increases rapidly over the internet even in the real-world scenario, the level of cybercrime is increases aggressively which makes it difficult to identify the criminals among various suspects just by interrogating them.*

The Android is one of the most popular Operating system in the field of smart phones, which occupies over 80 percent of the market share.[1] By looking at the popularity and usability, cybercriminals are targeting android device with Malware (malicious programs); in which, the most popular one was Remote Access Trojan (RAT). It allows a potentially malicious user to remotely control the system, by using specific Reverse TCP port connection on suspect machine. RAT is sequence of code, once installed on Android device can allow remote computer to take control of device usually unnoticeably and perform unpredicted or even unauthorized operations and can use an array of techniques to remain undetectable for a long haul. The infection of the malware can be caused by mean of Social Engineering. The Malware developers use Fully Undetected (FUD) techniques in order to make it unsuspicious to users. example: firewall, antivirus and Google play protection bypass, making persistence application, sometime by legitimate-looking binaries with signature bypass. The malwares detection occurs mainly by the three methods i.e. Signature, Behavior and Bit Parity. In this paper, our objective is to construct a RAT for law enforcement agencies to spy on suspicious people, limiting time, man power and others assets to gain sensitive information of suspects.

***Key Words*: Android Malware, Remote Access Trojan (RAT), Fully Undetected (FUD), Reverse TCP, Social Engineering, Malware Analysis & Detection.**

1. INTRODUCTION

These days, law enforcement Department are working manually over the criminal cases, which involves lot of confusion with the evidences and suspects. Remote Access Trojans (Malware) can make their work easier by finding digital evidences very easily.

The word "Mal" stands with meaning "bad, ill, wrongful", on the other side "ware" refers to software, combining it as Malware i.e. malicious software. The Malware includes Virus, Trojan, Adware, Spyware, Rootkits, Zero-day, Worm, Botnet, Keyloggers and Ransomware.

1.1 Remote Access Trojan

The Remote Access Trojan (RAT) is a type of Backdoor, through which, the law enforcement Departments can maintain peace in nation by observing the suspicious activities remotely. The RAT application mainly works with two modules, i.e. server and client. The RAT server is installed on the suspects device where the RAT client is present on the attacker machine and can listen for the server program on the specified port to make connection.[3] Thus, the attacker can connect to server vie TCP/UDP protocol. Where, server program can run in the background of the Suspect's machine, which is invisible to user. For the successful attack, server needs to bypass with firewall, Antivirus, Google Play Protection and Signature of the application, also it needs to make an application persistence. So, it can last for longer haul.

The Reverse TCP connection can help the attacker to bypass firewall of the Android device.[3] Usually, firewall blocks In-bounded Traffic, can concern less with, out-bounded traffic. Similarly, heuristics or signature-based approaches on arbitrary file system objects results as limitation of android antiviruses, the detection of malware were occurred by the patterns or behavior of the code present in file.[11] The pattern and behavior included in the databases will detect the malware while downloading malicious file on its own directory.[11] The signature of the application could be bypass for the purpose to use legitimate application as a weapon.

It can prove to be dangerous; if gets into negative hands and can be used for the offensive purposes.

2. LITRATURE REVIEW

Over the past few years, the number of attacks over the personal data of Android device users increased by 40,386 in 2018 to 67,500 till May,2019.[10] Attacks on users' personal data and Detection of Trojans became more frequent.[10] significantly, the increasing of adware threats over past few years shows the purpose of being to harvest personal data on mobile devices.[10] The statistics show that, the number of users attacked by adware in 2019 is roughly unchanged from 2018.[10]

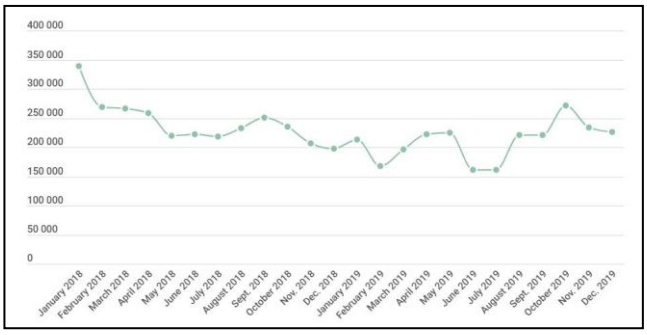

Figure 2.1: Attacks by Adware

For the past three consecutive years, it is to seen that, overall decline in the number of mobile threats distributed as installation packages (RAT).[10] The situation largely depends on specific cybercriminal campaigns: some have become less active while others have completely ceased.[10]

The situation shows the similarity with increasing as well as decreasing in number of attacks using mobile threats: In 2018, it is to observed a total of 116.5 million attacks whereas in 2019 the figure was down to 80 million.[10]

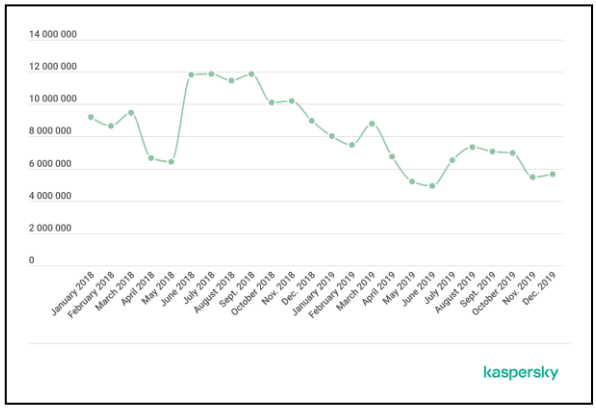

Figure 2.2: Mobile Threats

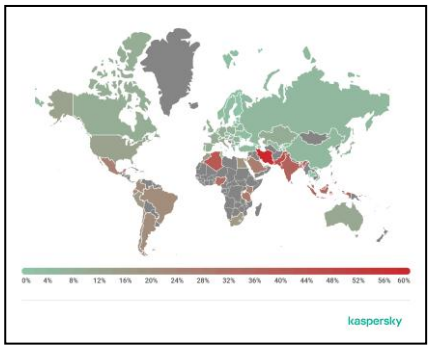

Figure 2.3: Geography of attacked users in 2019 [10]

COUNTRY	PERCENTAGE
Iran	60.64 %
Pakistan	44.43 %
Bangladesh	43.17 %
Algeria	40.20 %
India	37.98 %
Indonesia	35.12 %
Nigeria	33.16 %
Tanzania	28.51 %
Saudi Arabia	27.94 %
Malaysia	27.36 %

Table 2.1: Top 10 countries attacked by mobile Malware[10]

2.1 Methodology

The infection of the malicious program is encapsulated inside a deceivingly harmless executable file. An internal node is used to open the encrypted connection to a connection manager based on the access request.[8] A communication session is established in between the suspect device and the remote computer are forwarded using the encrypted connection.[8] The connection manager forwards the host and the port to establish the connection between suspect device and remote device. This method of accessing the device is known as Remote Access.[8]

Receiving input identifies an Android device, presenting with suspect and remote commands corresponding to an Android device, the receiving user input select remote commands from the one or more presented remote devices, generating a remote command message instructing the Android device to execute the commands for selected device.[6]

The Metasploit frame is the methods for realizing internet security, are related to field of information security technology.[9] The invention calls realization test process by carrying out batch to the penetration attack tool in Metasploit frame. a large amount of penetration attack tools integrates in Metasploit frame, load is attacked, and other dependent test tools are combined to realize the security test to target network.[9] The attack payload module is the set of a plurality of attack load, is the code for completing actual attack function, in success It is run after infiltration loophole, a connection can be established between attacker and target network.[9]

The detection of malware can be examined by three major factors i.e. signature, behavior, bit parity but still there are number of techniques to compromise Android devices. One

of them is Update attack. The update attack is difficult to detect. the previous version which were installed on the device is benign and it is not sure when the malicious activity performed because while updating the application it is difficult to detect the function of the previous benign application that was installed in the device.[9] Even Antivirus cannot access or monitor an Android device's file system, or dynamic behavior of installed apps; that needs permission to access all files. This includes the downloading of malicious files after installation, and other file system alterations.[7]

3. PROPOSED SYSTEM

3.1 Objective

To construct an Android Application for law enforcement agencies to spy over suspicious people or criminals with respect to law and order, while valued towards time, man power and others assets to gain sensitive information of the suspects.

3.2 How objective can be achieved

3.2.1 Find existing vulnerability

Android partners were notifying of all issues related to security at least a month before the publication. [16] Source code patches of these vulnerability issues have been released to the Android Open Source Project (AOSP) repository and linked from the bulletin.[16] This bulletin includes the links to patch the outside of AOSP.[16]

The severity of these issues is a critical security vulnerability in Media frameworks which could enables a remote attacker to use a specially crafted file to execute arbitrary code within the context of a privileged processes.[16] The severity assessment is based on the effect that exploiting the vulnerability would possibly have on an affected device.[16]

SEVERITY BASIC	SCORE RANGE
NONE	*0-0*
LOW	*0.1-3.9*
MEDIUM	*4.0-6.9*
HIGH	*7.0-8.9*
CRITICAL	*9.0-10.0*

TABLE 3.1: Common Vulnerability Severity Score (v3.0 RATING) [17]

3.2.1.1 Vulnerabilities

CVE-2017-13156 (Android Janus APK Signature bypass): The module exploits "CVE-2017-13156" in Android Operating system to install a payload into another legitimate application.[14] The payload APK will have the same signature as other APK and can be installed as an update and can preserve the existing data. The vulnerability was fixed in 5th December 2017 security patch, also additionally fixed by the APK Signature scheme v2. So, the only APKs which is signed with the v1 scheme are still vulnerable.[14]

SEVERITY: HIGH-7.8 [15]

CVE-2016-5195 (Dirty Cow-Privilege Escalation): This vulnerability is a race condition, found on the way to Linux kernel's memory subsystem, which handles the copy-on-write (COW) breakage of private read-only memory mappings. This flaw helps to get the more privilege to local user and gain write access to otherwise read-only memory mappings. Thus, this can increase their privileges on the system.[18] [12] this vulnerability is also known as Dirty Cow.

SEVERITY: CRITICAL [18][19]

There were few more latest vulnerabilities of android which helps to compromise the Android device.

3.2.2 Creation of malicious application

Android Package (APK) is a software, through which Android OS distributes and installs mobile applications. These packages may create an issue for the harmful malicious applications.

3.2.2.1 Firewall Bypass

The Reverse TCP connection can help the attacker to bypass firewall of the Android device.[3] Usually, firewall blocks In-bounded Traffic, can concern less with, out-bounded traffic. Firewall works on the principle of blocking in-bounded traffic. basically, any of the incoming connection towards the host is blocked by the firewall only if it malicious. However, return traffic towards the connection initiated by the device will be permitted. So, whenever our device trying to initiate a connection to any device, we call it as forward connection, the connection goes from client and server. But when server wants to initiate the connection to client, we call it as reverse connection.

In Reverse TCP connection, the connection is established from Android device instead of attacker, which cannot gets blocked by firewall. The device will initiate connection to the attacker, which will be allowed by the firewall.

3.2.2.2 Antivirus Bypass

Malwares gradually improve their codes to make them invisible. where, anti-virus software continually follows the new logics and methodologies to overcome the threats caught by victims to infected without the knowledge of victim. Since, if the malware is programmed manually, it is less likely to be detected by any Antivirus as its signature will not be present in the Databases of the Antivirus.[3]

3.2.2.3 Signature Bypass

If Android users installs an older version of APKs of legitimate applications, due to any reason such as, features or previous UI which are preferred by the user.[18] It is possible for an attacker to then create a malicious version of legitimate application which hides a payload and allow an Android device to connect back to the attackers' machine through the backdoor. The CVE-2017-13156 - Android Janus APK Signature bypass, can allows an attacker to install a payload to any another legitimate application, which is signed with the v1 scheme.[18]

3.2.2.4 Persistence Application

The persistence application is a concept of server (android device). In which, application can continuously run at background of the server. as the device booted, it starts the activity, and runs in the background of the suspect's machine. Which can establish a connection request to the attacker's machine, whenever the suspect's machine is online and connected to the internet. The attacker's machine is constantly online for the connection and can listening for active servers.

Our application can be persistent by using the superclass activity of BroadCastReceiver to our activity and by adding few permissions to Android-manifest (i.e. android.permission.RECEIVE_BOOT_COMPLETE) as well as we have to add actions to the receiver i.e. BOOT_COMPLETED and QUICKBOOT_POWERON.

StartActivityOnBootReceiver | onReceive()

```
package com.codinginflow.onbootreceiverexample;

import android.content.BroadcastReceiver;
import android.content.Context;
import android.content.Intent;


public class StartActivityOnBootReceiver extends BroadcastReceiver {

    @Override
    public void onReceive(Context context, Intent intent) {
        if (Intent.ACTION_BOOT_COMPLETED.equals(intent.getAction())) {
            Intent i = new Intent(context, MainActivity.class);
            i.addFlags(Intent.FLAG_ACTIVITY_NEW_TASK);
            context.startActivity(i);
        }
    }
}
```

Figure 3.2: Broad Cast Receiver

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.codinginflow.onbootreceiverexample">

    <uses-permission android:name="android.permission.RECEIVE_BOOT_COMPLETED" />

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="On Boot Receiver Example"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/AppTheme">
        <activity android:name=".MainActivity">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>

        <receiver android:name=".StartActivityOnBootReceiver">
            <intent-filter>
                <action android:name="android.intent.action.BOOT_COMPLETED" />
                <action android:name="android.intent.action.QUICKBOOT_POWERON" />
            </intent-filter>
        </receiver>
    </application>

</manifest>
```

Figure 3.3: Manifest Permissions

The permissions, which would add to the manifest of the android application should be lesser than the android version 6 (i.e. marshmallow), so these permissions would be given itself with other permissions during the installation of an application. Where, in android version greater than 6 will take permission at runtime to each individually.[3]

3.2.3 Set-up Listener

Finally, after setting-up each individual, we have to start listener for the specific port and IP-Address, The IP-Address should be static or use dynamic DNS with port forwarding. Due to static IP address or Dynamic DNS, the application will always response to system either it will change the IP Address as the internet switch from the listeners' device and the connection with suspect will never be establish. The listener should be start to listen port and IP address before the Application code (Android) runs so the connection could be established.

4. RESULT

The proposed system is used to gather active information, while getting undetectable. The system can bypass the security flaws in antiviruses, firewall, Google play protection and signature of the application which helps RAT to be undetected. It also proposed the concept of persistence application. In which, the script runs on the background of the system and maintains remote connectivity with monitoring device through the backdoor.

5. CONCLUSION

The described methodology can help law and enforcement departments to keep eye on criminals as well as track their malicious activities, to maintaining peace in the nation. The Remote Access Trojans (RAT) are mainly used for offensive purposes, but can also be used for defensive purposes to spy on criminals and suspicious people. It can create much value in limiting time, man-power and other assets and can also help to identify the digital evidences with their preservatives.

REFERENCES

[1] Akshitasinh Chauhan, Dr. Ravi K Sheth. "Fully Undetectable Remote Access Trojan: Android." International Journal for Research in Applied Science & Engineering Technology (IJRASET), Volume 7 Issue V, May 2019.

[2] Manjeri N. Kondalwar, Prof. C.J. Shelke. "Remote Administrative Trojan/Tool (RAT)." International Journal of Computer Science and Mobile Computing, Vol.3 Issue.3, March- 2014, pg. 482-487.

[3] Prakhar Ahlawat, Sushant Dhar, Samruddha Wagh, Amit Koppad. "Remote Access Tool Using Metasploit." International Journal on Recent and Innovation Trends in Computing and Communication, Volume: 5 Issue: 4 425 – 427.

[4] İlker Kara, Murat Aydos. "THE GHOST IN THE SYSTEM: TECHNICAL ANALYSIS OF REMOTE ACCESS TROJAN." International Journal on Information Technologies & Security, № 1 (vol. 11), 2019.

[5] Andi Fitriah Abdul Kadir, Natalia Stakhanova and Ali A. Ghorbani. "Understanding Android Financial Malware Attacks:Taxonomy, Characterization, and Challenges." Journal of Cyber Security and Mobility, Vol. 7_3, 1–52. River Publishers 2018.

[6] Christopher Brooke Sharp, San Jose, CA (US); Brendan A. McCarthy, San Francisco, CA (US): Stuart Slack, Cupertino, CA (US); Carsten Guenther,San Francisco, CA (US); Jeff Lin, Cupertino, CA (US); Rob Butler, Middletown, DE (US) . "United States Patent." Patent No.: US 8,660,530 B2, Date of Patent: Feb. 25, 2014.

[7] Rafael Fedler, Marcel Kulicke and Julian Sch¨utte. "An Antivirus API for Android Malware Recognition." Conference Paper · October 2013, DOI: 10.1109/MALWARE.2013.6703688.

[8] Irfan Z. Khan, Minneapolis, MN (US); Mitchell Y. Coopet, Inver Grove Heights, MN (US); Matthew D. Dornquast, Minneapolis, MN (US); Richard C. Baker, Mahtomedi, MN (US); Peter J. Lindquist, Saint Paul, MN (US), "United States Patent Application Publication" Pub. No.: US 2007/0061460 A1, Pub. Date: Mar. 15, 2007

[9] https://securelist.com/mobile-malware-evolution-2019/96280/

[10] https://economictimes.indiatimes.com/news/politics-and-nation/police-in-states-across-indiaare-relying-on-private-firms-and-consultants-to-solve-cybercrimecases/articleshow/72499885.cms?utm_source=contentofinterest&utm_medium=text&utm_campaign=cppst

[11] https://www.exploit-db.com/exploits/40839https://en.wikipedia.org/wiki/Reverse_connection

[12] https://www.rapid7.com/db/modules/exploit/android/local/janus

[13] https://nvd.nist.gov/vuln/detail/CVE-2017-13156

[14] https://source.android.com/security/bulletin/2019-05-01

[15] https://nvd.nist.gov/vuln-metrics/cvss

[16] https://cve.mitre.org/cgi-bin/cvename.cgi?name=cve-2016-5195

[17] https://source.android.com/security/bulletin/2016-12-01.html