SECTRIO

MALWARE REPORT

Android: Anibus Malware

Date: 30 /09 /2021

K. Narahari

Overview:

The Anibus malware was identified in our Honeypot. Anubis is a malicious application classified as a banking Trojan. It targets smartphones running the Android Operating System. The malware infects the mobile devices of the user by tricking them to download Anibus apps disguised as other popular applications. Most infection occurs when the android user downloads and installs malicious applications from the third-party store.

This malware attempts to steal banking information and can lead to victim's experiencing financial loss and privacy issues. It can also lead to decreased device performance; quick battery drains and other problems.

File Hash: **dc030efa5973ba809bad2f544d9b18d2**

Technical Analysis:

We downloaded and reverse-engineered the malware sample and then performed static analysis on the .apk file and went through the manifest.xml, which is the primary file to begin the analysis.

```
<uses-permission android:name="android.permission.FOREGROUND_SERVICE" />
<uses-permission android:name="android.permission.REQUEST_DELETE_PACKAGES" />
<uses-permission android:name="com.sec.android.provider.badge.permission.WRITE" />
<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />
<uses-permission android:name="android.permission.GET_TASKS" />
<uses-permission android:name="android.permission.RECEIVE_SMS" />
<uses-permission android:name="android.permission.READ_SMS" />
<uses-permission android:name="android.permission.WRITE_SMS" />
<uses-permission android:name="android.permission.PACKAGE_USAGE_STATS" />
<uses-permission android:name="android.permission.USE_FULL_SCREEN_INTENT" />
<uses-permission android:name="android.permission.ACCESS_NOTIFICATION_POLICY" />
<uses-permission android:name="android.permission.SYSTEM_ALERT_WINDOW" />
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
<uses-permission android:name="android.permission.CALL_PHONE" />
<uses-permission android:name="android.permission.INTERNET" />
<uses-permission android:name="android.permission.SEND_SMS" />
<uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
<uses-permission android:name="android.permission.READ_EXTERNAL_STORAGE" />
<uses-permission android:name="android.permission.RECORD_AUDIO" />
<uses-permission android:name="android.permission.READ_CONTACTS" />
<uses-permission android:name="android.permission.READ_PHONE_STATE" />
<uses-permission android:name="android.permission.WAKE_LOCK" />
<uses-permission android:name="android.permission.RECEIVE_BOOT_COMPLETED" />
<uses-permission android:name="android.permission.REQUEST_IGNORE_BATTERY_OPTIMIZATIONS" />
```

Figure 1

From the analysis of the manifest file, we observed that there are plenty of permissions given to the application. Some of the dangerous permissions are "REQUEST_DELETE_PACKAGES", which is used to delete already installed applications or

packages from the victim device. “PACKAGE_USAGE_STATS” is used to gather information related to the applications which are installed on the device and their usage statistics.

“RECORD_AUDIO” is used to record the audio clips from the victim’s device. “READ, WRITE, RECEIVE, SEND_SMS” can be used for malicious purposes to send or receive OTP, some important messages. The attacker can even send, receive, and manipulate the messages from the victim’s device.

Also, this application has many permissions and services to access like, contact details can be accessible, phone details like dialled, received, and missed calls information, Wi-Fi connectivity information, internet related and many more malicious permissions.

The permissions used by this application seem to be very dangerous and they are used to perform malicious operations. This type of malware is very advanced and can be seen in rare scenarios.

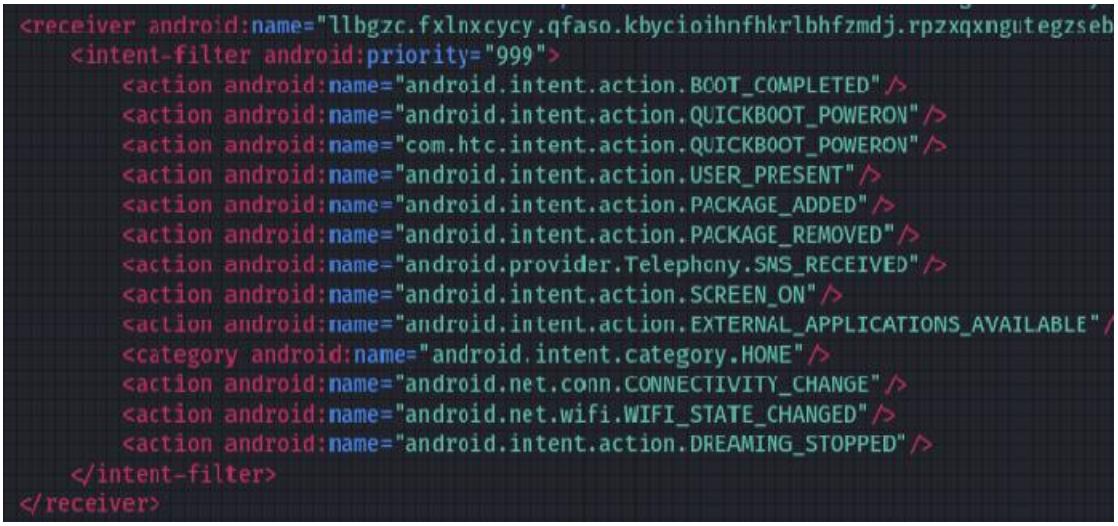

Figure 2

From the above screenshot, we can observe that this application is using broadcast receivers to pass the data from the application to other malicious application which are installed on the device. In general, broadcast receivers are used to pass the data from one application to another in the device. For example, OTP will be sent from messaging application to other applications like Uber and Facebook, this happens through the usage of the broadcast receiver.

Now in this case, attacker is using receivers to pass the phone state, power details, SMS, contacts, user details, connectivity details, OTP, login credentials, credit card details, battery information. This application uses many receivers and the receiver displayed above is one of the many receivers used by this malicious application.

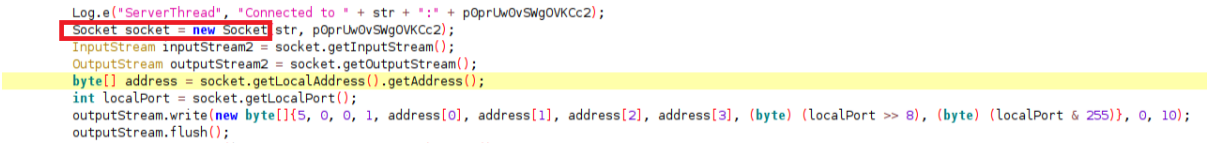

Figure 3

Sockets are used in android to connect to the remote device for sending and receiving data or commands from the remote host. Here in this application, the attacker connects to a command-and-control server for performing malicious activities using the data streams. The attacker steals all the data from the victim device and stores that in data stream buffers and connects to the C&C server and steals the information and executes remote commands.

Figure 4

The attacker steals the SMS from the victim device, pass it into JSON object and sends it to the command-and-control server. Along with this, we have found different code snippets on sending, receiving, manipulating, forwarding sms related to the attacker server. In this case, the attacker places all the SMS related data in JSON object, and the below highlighted one is encoded attacker URL.

Figure 5

Here are some of the malicious activities performed by the malware:

- This malware can encrypt and transmit to a C2C server.
- It can install and uninstall the applications on the victim device.
- Using the exported activities and broadcast receivers, it can steal data from other applications.
- It can take control of the calls, SMS, contacts information.
- It can control audio, microphone on the victim device.
- It can enable or disable the push notifications or control some of its settings.
- Execute commands from the C2C server.

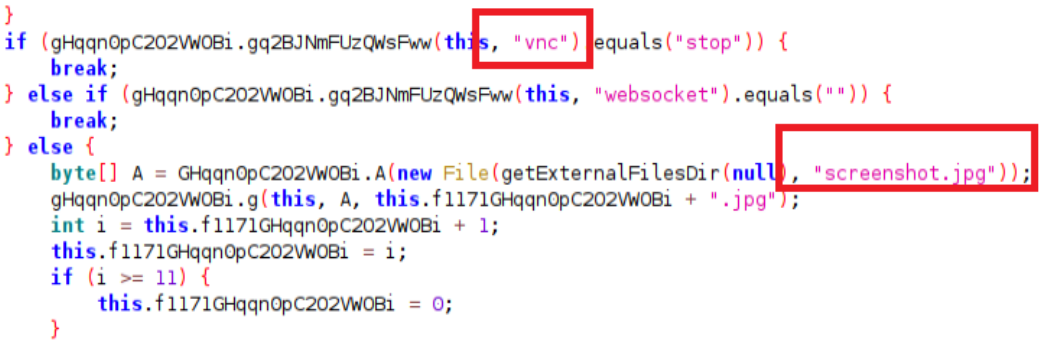

Figure 6

It is also capable of establishing VNC session. If the VNC session is started, the attacker can also see the screen of the victim's device.

IOC Url's:

https://buratino.tokyo/dope
http://gluer.gansandroses.club/php/action_get_apk_statistic.php
http://kazanin20gbturkiye.com
http://160.153.129.239
http://xn--20gb-tanmla-kullan-l0c.com

Mitre Techniques:

Deliver Malicious App via Authorized Store (T1475)
Broadcast receivers (T1402)
Application Discovery (T1418)
Capture SMS Messages (T1412)
File and Directory Discovery (T1420)
Data Encrypted for Impact (T1471)
Web Service (T1481)

Sectrio Protection:

Sectrio detects the android sample as “SS_Gen_AnibusMalware_A”.

Our Honeypot Network:

This report has been prepared from the threat intelligence gathered by our honeypot network. This honeypot network is today operational in 72 cities across the world. These cities have at least one of the following attributes:

- Are landing centers for submarine cables
- Are internet traffic hotspots
- House multiple IoT projects with a high number of connected endpoints
- House multiple connected critical infrastructure projects
- Have academic and research centers focusing on IoT
- Have the potential to host multiple IoT projects across domains in the future

Over 12 million attacks a day is being registered across this network of individual honeypots. These attacks are studied, analyzed, categorized, and marked according to a threat rank index, a priority assessment framework that we have developed within Sectrio. The honeypot network includes over 4000 physical and virtual devices covering over 400 device architectures and varied connectivity mediums globally. These devices are grouped based on the sectors they belong to for purposes of understanding sectoral attacks. Thus, a layered flow of threat intelligence is made possible.