A Characterization of Malicious Android Applications

Lorenzo Gomez Iulian Neamtiu
Department of Computer Science and Engineering
University of California, Riverside
{gomezl,neamtiu}@cs.ucr.edu

June 2011

Abstract

Smartphones are becoming pervasive–2010 sales have jumped 55% compared to 2009, and IDC estimates that 269 million units will be sold worldwide in 2010. Smartphone applications (apps) offer a wide range of financial, social, health, scientific, and even military capabilities on the go. However, mobile access to GPS location, camera, Internet, calendar, contacts, and other sensitive information can lead to inadvertent security risks, and this problem is exacerbated by the rapid evolution of smartphone hardware and software platforms. Today, smartphone application developers are largely on their own to ensure that they access sensitive resources safely and that they do not inadvertently allow access by untrusted third parties. Malicious Android apps masquerade as legitimate applications, but use the phone for nefarious purposes, e.g., for financial gains. These malicious apps are able to take advantage of the rapid evolution and developer freedom of the Android market to exploit applications to gather security-sensitive data, enlist the phone into premium services, and more. To effectively thwart malicious apps, their behavior must be further studied and dissected to understand exactly what the specific exploits are, what they do, and what reoccurring patterns and structures these malicious applications use. In this paper we perform such a study, that provides a characterization of the behavior of 12 malicious apps. This study is a step towards recognizing and mitigating the threat posed by malicious Android apps.

1 Introduction

Smartphones are becoming pervasive, with more than 115 million sold worldwide in the first half of 2010 alone [5, 4]. A major draw of any smartphone is its ability to run applications. Such applications, which exploit the rich information and capabilities of the phone and the immense resources of the Internet, have exciting potential for a variety of social, health, scientific, and military purposes.

Hundreds of thousands of applications are already in use across several smartphone platforms. Unfortunately, the properties that give smartphone applications the potential to improve our lives also pose a serious threat to our privacy and security. Applications can collect sensitive data from the user (e.g., usernames and passwords to connect to online services); access personal data stored on the device (e.g., calendar and contact information); and use sensitive device capabilities (e.g., the internal GPS or camera). Moreover, smartphone hardware and software platforms are evolving rapidly, and applications are being developed every day that use smartphone capabilities in new and unexpected ways.

The motivating observation behind our proposed work is that todays smartphone platforms fail to help developers write secure applications. Instead, at best they provide coarse permission models that grant all-or-nothing access to sensitive data and resources. Once given access, developers are entirely on their own to ensure they use that access safely and only to the extent necessary for their applications.

This kind of violation of the principle of least privilege is a well-known problem with any operating system. However, it is particularly important and challenging for smartphone platforms, where applications can interact with third-party components in subtle and indirect ways, sensitive data and capabilities on the phone can be combined in ways that have unintended consequences, and rapid platform evolution makes the goal of smartphone security a moving target.

These properties combine to foil the development of a standard, fixed vocabulary of permissions that is precise enough to capture policies of interest, yet adaptable enough to support the pace of innovation.

2 Background Information

2.1 Security-sensitive Data on Android

The Android operating system contains many security-sensitive pieces of data that identify the user's identification information and user specific settings. The most sensitive pieces of data are described below:

- International Mobile Equipment Identity (IMEI)
- International Mobile Subscriber Identity (IMSI)
- Android ID
- Mobile Subscriber Integrated Services Digital Network Number (MSISDN, phone number)
- Contacts list
- Contents of external SD card (user files, application data)

With the combination of the IMEI, IMSI, and Android ID, an Android mobile phone can be uniquely identified down to the physical device and the subscriber.

The IMEI is a unique number that identifies the cell physical phone device. The number is used by the GSM, Global System for Mobile Communications, to identify valid phones in its cellular network [18].

The IMSI is a unique number securely stored inside the phone's SIM (subscriber identification module). The number is sent from the phone to the network and identifies the user's mobile subscription and provider [19].

The Android ID uniquely identifies an Android device with a 64 bit hex string that is randomly generated on the device's first boot and normally remains constant for the lifetime of the device [8].

The phone number is stored securely as the MSISDN of the mobile and uniquely identifies the subscription of the phone inside the mobile network [20].

The contacts list of the phone identifies all contacts stored by the user which may include, names, phone numbers, addresses, emails, etc.

The contents of the SD card (external storage) may contain personal files of the user, including: photos, music files, and document files. The card may also contain application data that is stored by applications as a temporary location or settings location. The information contained in these is dependent on the application.

2.2 Revenue Generating Schemes

Malicious applications usually use a revenue generating scheme that will benefit their developer in some way.

These schemes are in the form of generating charges for the phone's user in call charges or SMS messaging charges.

- Calls to 1-900 premium phone numbers
- Subscriptions to SMS messaging services

2.2.1 Calls to 1-900 premium phone numbers

If an Android application is given access to make phone calls, it is able to call phone numbers without the user's intervention.

Therefore, malicious applications can accrue phone call charges on the user's bill by dialing premium phone numbers.

Oblivious to the user, a list of premium phone numbers are often times hidden inside the malware in order for the application to retrieve and call the number [21].

2.2.2 Subscriptions to SMS messaging services

If an Android application is given access to send and receive SMS messages, the application is capable of sending and reading messages without the user's intervention.

Therefore, malicious applications can utilize SMS communication to fraudulently register the unsuspecting users device to premium SMS services. The services can be used to deliver digital content such as news alerts, financial information, logos, and ring-tones.

Those affected with the malware would notice unsolicited charges to their mobile accounts due to their device being registered for those premium services [21].

2.3 Android Permissions

In Android, access to sensitive resources is controlled by permissions. Each application bundle includes an XML manifest file that lists the permissions requested by the application. When an application is installed, the permissions in the applications manifest are shown to the user, who then decides whether to proceed with the installation (i.e., grant the permissions), or to cancel it. No additional permissions may be acquired when an application runs, and an application is killed if it tries to access a resource for which it does not have permission.

Below are some of the most important permissions of the Android operating system that define what an application has access to [6].

Currently, Android's security model does not protect against misuses of these permissions. Once a permission is granted, it is up to the developer of the application to ensure that the data is being used in a safe manner [7].

- ACCESS_FINE_LOCATION, ACCESS_COARSE_LOCATION
- CALL_PHONE
- READ_SMS, SEND_SMS
- READ_PHONE_STATE
- INTERNET
- READ_CONTACTS, WRITE_CONTACTS
- WRITE_EXTERNAL_STORAGE

2.3.1 ACCESS_FINE_LOCATION, ACCESS_COARSE_LOCATION

These permissions handle the information of the phone's location. The coarse permission allows access to the phone's location based on the CellID or connection to WiFi.

The fine permission will allows access to the mobile's GPS location, which is more exact than the coarse.

The phone's location can be used in many malevolent ways, if desired. For example, a combined list of reported locations could be used for tracking purposes.

2.3.2 CALL_PHONE

This permission allows the application to be able to make phone calls on the user's behalf with or without their interaction (through the dialpad). Giving access to this permission does not allow the application to dial emergency numbers.

As mentioned above, malicious applications can use this permission to call phone numbers that charge a premium to compile charges for the user.

2.3.3 READ_SMS, SEND_SMS

With these permissions an application will be able to both read and send SMS messages with or without their intervention.

As noted above, malicious applications can use this permission to subscribe to premium SMS services on the user's behalf.

2.3.4 READ_PHONE_STATE

This permission will give access to a great deal of information including the user's phone physical identifier (IMEI), the subscriber identifier (IMSI), phone number (MSISDN), language, and country [9].

Accessing the phone's internal information may seem harmless at first glance, but if the intent is malicious, this information can be used in potentially harmful ways. Collections of this type of data could compile a "blacklist" of phone numbers and phone identifiers that could be used for whatever reason, without the user's knowledge or consent.

Classification	Malicious Applications
DroidDream	Basketball Shot Now Super Stopwatch & Timer Scientific Calculator
DroidDreamLight	Bubble Buster Free HOT Girls 1 Floating Image Free
Zsone SMS	Sea Ball iCalendar ShakeBanger
Geinimi	Geinimi StickFood Monkey Jump 2

Table 1: A table showing the classifications of each of the analyzed applications.

2.3.5 INTERNET

The permission to grant internet access is one of the most important permissions on the Android operating system. With this permission, the application can open network sockets, which enables the application to have full access to the internet for both sending and receiving data.

Many malicious applications involve using the internet to connect to an external source. Whether it be to send data off the phone or bring in new data to the phone, they can use the internet permission to their advantage to engage in malicious activities.

2.3.6 READ_CONTACTS, WRITE_CONTACTS

These permissions allow the application read and write access to the mobile phone's contact list. The application will be able to read the data stored for all the user's contacts, this information usually includes names, phone numbers, and email addresses. The application will also be able to create new contacts and add them into the contacts list.

Malicious applications may use the information gained from having this permission enabled in many different ways. When coupled with the internet permission, there is no limit as to where the phone's contact list is being sent and/or being stored externally.

2.3.7 WRITE_EXTERNAL_STORAGE

This permission is also of high importance. It allows an application to read and write to external devices that are connected to the phone. In most cases, the external device is a SD card that is inserted into the phone to act as an media device.

Allowing an application access to read and write to the external storage could be harmful if the application is malicious. A malicious application may attempt to delete or overwrite the data stored on the card, or it may gather data on the card and try to send it to an external source.

3 Malware Classifications

3.1 Classifications

Of the 12 Android applications analyzed, there was a total of four categorizations of malware. These included: DroidDream, DroidDreamLight, Zsone SMS, and Geinimi. Below are detailed descriptions of the malware including why they are harmful and what they do.

	DroidDream	DroidDreamLight	Zsone SMS	Geinimi
Location			✓	✓
Call phone				✓
SMS			✓	✓
Phone state	✓	✓		✓
Internet	✓	✓	✓	✓
Contacts				✓
Storage				✓

Table 2: A table showing the permissions used by each of the classifications of malicious applications.

3.2 Description of Classifications

1. DroidDream
2. DroidDreamLight
3. Zsone SMS
4. Geinimi

3.2.1 DroidDream

The pirated and trojanized applications have the ability to root an Android device using the "rageagainstthecage" root exploit to allow the trojan access to data and services only obtained with root access [10].

DroidDream then sends personal information such as the phone's information to external servers. The information sent includes personal information such as the International Mobile Equipment Identity (IMEI), product ID, language and country, Subscriber, and Subscriber Identity Module serial number.

DroidDream also is able to download and install additional applications at will, all without informing or obtaining permission from the user to do so [13, 22].

3.2.2 DroidDreamLight

DroidDreamLight will gather information about the device and try to upload the data to a list of websites defined by the trojan. The information taken includes: device model, language and country, International Mobile Equipment Identity (IMEI) number, International Mobile Subscriber Identity (IMSI) number, Software development kit (SDK) version, and a list of installed apps.

The trojan is launched by the receiver it creates, $< package > .lightdd.Receiver$. The receiver then launches the service $< package > .lightdd.CoreService$ which contacts remote servers and supplies the stolen info.

DroidDreamLight is also capable of downloading and prompting installation of new packages, though unlike its predecessors it is not capable of doing so without user intervention [17, 3, 16].

3.2.3 Zsone SMS

The Zsone trojan utilizes SMS communication to subscribe users to premium rate QQ codes via SMS without their knowledge. A QQ code is a form of short code that is able to subscribe users to SMS updates and/or instant message services.

Once the application is started, the trojan will silently send a single SMS message to initially subscribe the user to a premium-rate SMS service without their authorization or knowledge [14, 15]. After the user is subscribed, the trojan ensures that the broadcast is never completed again and runs normally performing whatever features that the application offers.

Interestingly, the trojan also has the ability to divert being detected when running in the Android developer's virtual device. By checking the prefix of the internal phone number, the broadcasting of the message will be aborted if the virtual device's number is found [2].

3.2.4 Geinimi

Geinimi is an Android trojan that has the ability to harvest and transmit personal and device identifying information such as the IMEI and IMSI to remote servers. The trojan can also monitor and send location data of the mobile. Other information can also be sent, such as an enumerated list of applications installed on the infected device.

Geinimi can also potentially accumulate charges for the mobile user by being able to monitor and send SMS messages, delete selected SMS messages, and place phone calls.

Furthermore, the trojan can silently download files and prompt the user to install 3rd party applications. The trojan can also launch the web browser with pre-defined URL's [12, 1, 11].

3.3 Patterns

Several of these classifications involve using the same permissions and performing the same malevolent actions.

All of the malicious applications analyzed allowed the permission to access the internet. Enabling this permission allows the malicious applications to send both phone data and files away from the phone. Depending on what other permissions are enabled, the phone's identifiers, contacts list, or user files all could be harvested. Allowing open internet access also allows the application to bring in new data to the phone, whether it be to bring in new information to aid the trojan or files to install 3rd-party software.

Three out of four of the classifications (9 applications in total) involve accessing the phone's state and having access to the identifiers hidden inside the phone. The information gathered from this permission could be sent to external servers identified by the malicious app.

Half of the applications analyzed would attempt to sign the user up for some sort of premium service. These applications could potentially have a great impact on the user. Without intervention, charges assembled by the malicious application could reach unexpected amounts.

Acknowledgements

We would like to thank Lookout Labs for providing the malicious applications to analyze, and Xuetao Wei for making this analysis possible. This work was supported in part by National Science Foundation award CNS 1064646 and by a Google Research Award.

References

[1] BitDefender. Trojan.Adroid.Geinimi.A Removal. `http://www.bitdefender.fr/VIRUS-1000648-fr--Trojan.Android.Geinimi.A.html`, January 2011.

[2] A. Brandt and A. Orozco. Chinese Android Trojan Texts Premium Numbers. `http://blog.webroot.com/2011/05/11/chinese-android-trojan-texts-premium-numbers/`, May 2011.

[3] D. Fisher. DroidDream Returns, Dozens of Apps Pulled from Android Market. `https://threatpost.com/en_us/blogs/droiddream-returns-dozens-infected-apps-pulled-android-market-060111`, June 2011.

[4] Gartner Corporation. Gartner Says Worldwide Mobile Device Sales Grew 13.8 Percent in Second Quarter of 2010, But Competition Drove Prices Down. `http://www.gartner.com/it/page.jsp?id=1421013`, August 2010.

[5] Gartner Corporation. Gartner Says Worldwide Mobile Phone Sales Grew 17 Per Cent in First Quarter 2010. `http://www.gartner.com/it/page.jsp?id=1372013`, May 2010.

[6] Google Inc. Android Developers. Manifest.permission. `http://developer.android.com/reference/android/Manifest.permission.html`, 2011.

[7] Google Inc. Android Developers. Security and Permissions. http://developer.android.com/guide/topics/security/security.html, 2011.

[8] Google Inc. Android Developers. Settings.Secure — Android Developers. http://developer.android.com/reference/android/provider/Settings.Secure.html, 2011.

[9] Google Inc. Android Developers. TelephonyManager. http://developer.android.com/reference/android/telephony/TelephonyManager.html, 2011.

[10] Intrepidus Group. Android Root Source Code: Looking at the C-Skills. http://intrepidusgroup.com/insight/2010/09/android-root-source-code-looking-at-the-c-skills/, September 2010.

[11] J. Kirk, ComputerWorld. Monkey Jump apps may be wrapped with malware. http://www.computerworld.com/s/article/9209401/Monkey_Jump_apps_may_be_wrapped_with_malware, February 2011.

[12] Juniper Global Threat Center. Geinimi Trojan. http://globalthreatcenter.com/?p=2056, January 2011.

[13] Juniper Global Threat Center. Myournet Pirated Apps Root, Steal Data, Install CodeJust to Start. http://globalthreatcenter.com/?p=2091, March 2011.

[14] Juniper Global Threat Center. Zsone SMS Trojan Apps Pulled From Android Market. http://globalthreatcenter.com/?p=2280, May 2011.

[15] Lookout Mobile Security. Security Alert: Zsone Trojan found in Android Market. http://blog.mylookout.com/2011/05/security-alert-zsone-trojan-found-in-android-market/, May 2011.

[16] Lookout, Mobile Security. Update: Security Alert: DroidDreamLight, New Malware from the Developers of DroidDream. http://blog.mylookout.com/2011/05/security-alert-droiddreamlight-new-malware-from-the-developers-of-droiddream/, May 2011.

[17] TrendLabs of TrendMicro. Analysis of DroidDreamLight Android Malware. http://blog.trendmicro.com/analysis-of-droiddreamlight-android-malware/, June 2011.

[18] Wikipedia contributors. International Mobile Equipment Identity. http://en.wikipedia.org/wiki/Imei, June 2011.

[19] Wikipedia contributors. International Mobile Subscriber Identity. http://en.wikipedia.org/wiki/IMSI, May 2011.

[20] Wikipedia contributors. MSISDN. http://en.wikipedia.org/wiki/MSISDN, April 2011.

[21] Wikipedia contributors. SMS. http://en.wikipedia.org/wiki/SMS, June 2011.

[22] XDA Developers. Malware Exploit for all pre-Gingerbread phones. http://forum.xda-developers.com/showthread.php?t=977154, March 2011.

Appendix A Malicious Applications Analyzed

#	Application	Package name
1	Basketball Shot Now	Basketball Shot Now.apk
2	Super Stopwatch & Timer	com.droiddream.stopwatch.apk
3	Scientific Calculator	Scientific Calculator.apk
4	Bubble Buster Free	com.BubbleBuster.apk
5	HOT Girls 1	com.japanexe.girl-1.apk
6	Floating Image Free	dk.nindroid.super.rss-1.apk
7	Sea Ball	com.mj.ball.apk
8	iCalendar	com.mj.icalendar.apk
9	ShakeBanger	com.rzstudio.ishakebanger.apk
10	Geinimi	com.sgg.sp.apk
11	StickFood	HappyChain_android21.apk, com.dunliu
12	Monkey Jump 2	MonkeyJump .apk, com.dseffects.MonkeyJump2