International Journal of Information and Computation Technology.
ISSN 0974-2239 Volume 4, Number 4 (2014), pp. 373-380

http://www. irphouse.com /ijict.htm

An Overview of Security Challenges of Android Apps Permissions

Sakshi Dhama

University School of Information and Communication Technology, Guru Gobind Singh Indraprastha University, Sector 16 - C, Dwarka, Delhi, INDIA.

Abstract

In the last five years there has been observed a drastic shift from PC, laptops to smart phones for accessing internet services. The increased dependence on mobile apps brings into light the security risks associated with these apps. The large number of freely available apps in market days sometimes request more permissions than they use, and this fact is usually unknown to the user. Open source platform android makes it easier to introduce such flaws intentionally and steal the confidential information such as personal contacts, passwords etc. The paper presents the analysis of the survey work of vulnerabilities in android apps. The vulnerability analysis of a sample designed android app with over privileged permissions, and content leaks associated with such apps. The security measures practices that should be followed while setting the permissions have also been discussed.

Keywords: Android Security; Android permissions; Vulnerabilities.

1. Introduction

As the smart phone industry is growing and becoming more widespread, so are the security challenges on the mobile apps .In spite of availability of techniques like password protection, cryptography, data encryption, there exist some other mechanisms that pose threat unknown to the user. Android is a well known Operating system supported by a large number of companies HTC dream, Nokia, Samsung Galaxy, Motorola mobility, and many more. To use the basic available features such as contacts, camera, networking stack, GPS every app needs to quire permissions. For a calculator app which is designed for calculation purpose may not need permission for camera API. However apps which are malicious may target the user by acquiring the un necessary permissions. Sometimes user have no idea that apps which they are using are over privileged. To prevent the user from getting exploited the app designer needs to design using an ethical approach .However the apps are sometimes decompiled by hackers to intrude the unnecessary permissions to harm the user.

In this paper section 1 describes the security model of android, Section 2 gives a brief about permission in android, in Section 3 sample gravity calculator app permissions are described, then the results and conclusions are presented.

2. Security Model of Android

In the android kernel a privilege separation model is implemented, while executing the application .The android operating system has its own user id and group id for every application that runs on it .The privilege separation model ensures that no application can read or write the code or data of other applications such as device user, or of the OS itself. It prevents an application from arbitrarily using devices, networking stack to have a connection with remote server. This sandbox model shields the contact list of device to be directly read by an application. Two processes having their own sandboxes can communicate only by explicitly requesting permission to access data.

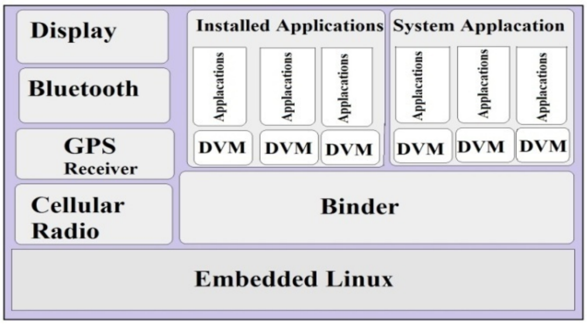


Fig. 1: Android Architecture shows instance of various application being executed on DVM.

2.1 Signing of Code in Android Application

For an application to be executed on android, it must be signed. The certificate of individual developers is verified by android for identity verification and establishment of trust relationship among other application of android operating system. The OS prohibits unsigned applications from execution. However, android allows a self signed certificate application to execute without any restriction.

2.3 Purpose of Permissions in Android

Android provides an environment for application to run with their own set of user identifier and group identifiers. However this makes an application restricted from reading and writing data of another. To overcome this problem and make a way of inter process communication, android provides the system of permissions. An application created in android has no permission to perform any task such as to call

API for GPS, or networking stack, call API to use camera. Any application interaction with android OS may affect the other application or can cause damage. Therefore to have access to the protected API, the android permission architecture is handled by kernel.

2.4 Permission Check Mechanism in Android

An application uses permission to interact with system. In this mechanism, when an application makes a call to API, the permission validation process checks whether the application has gathered permission needed for completion of call. The user at this point has a choice to deny or grant the permission to the application. The API call undergoes three steps:

1. Invocation of API library
2. Library invokes private proxy interface (part of API Library)
3. Private proxy interface using inter process communication queries the service executing in system process to perform the required API call operation.

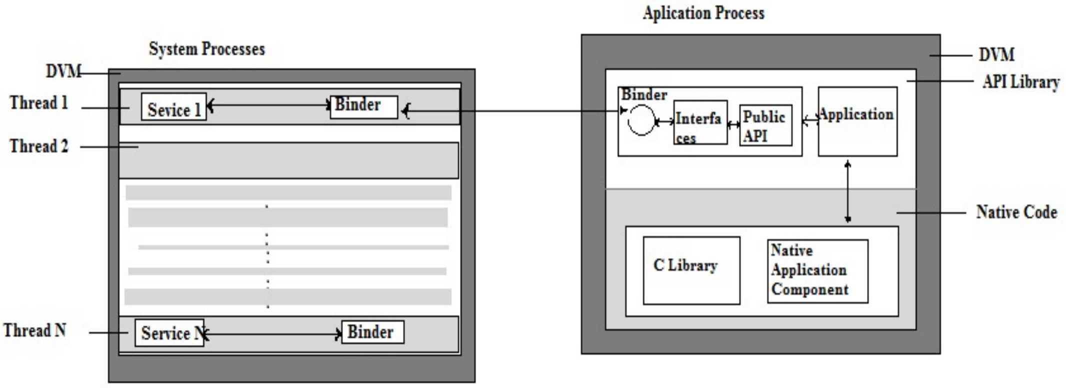


Fig. 2.1: The API Call Process in Android.

Validation process of permission is taken care of by system process. There are some applications where BLUETOOTH, WRITE_EXTERNAL_STORAGE, and INTERNET access permission are required; such permissions are handled by the Linux kernel.

2.5 Self Defined Permissions

Android is a open source platform, that provides option to create and use self created permissions. These permissions are defined to provide protection to the methods and classes of an application from other applications. These permissions are declared in Android.manifest.xml file.

```
    <manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.sakshiapp" >
    <permission
android:name="com.example.sakshiapp.permission.SAKSHI_DATABASE"
```

```
android:label="@string/label_sakshiDatabase"
android:description="@string/description_sakshiDatabase"
android:protectionLevel="normal" />
</manifest>
```

In the above example mydatabase has the permission to access SD card by adding it to the group that has already been assigned custom permission .This can be done by adding the attribute given below in Android.manifest.xml file.

```
android:permissionGroup=" android.permission-group.STORAGE"
```

2.6 Levels of Permissions in Android

A developer during the development process can categorize a level of protection to a self created permission .Basically there are four levels in android.

Any permission protection level 1 or higher triggers the OS to notify the end user that application is being executed can cause harm. Therefore user can deny or grant such permission whenever prompted.

0
normal
1
dangerous
2
signature
3
signatureOrSystem

Fig. 2.2: Permission protection levels in android.

Table 1: Comparison among the four protection levels in android.

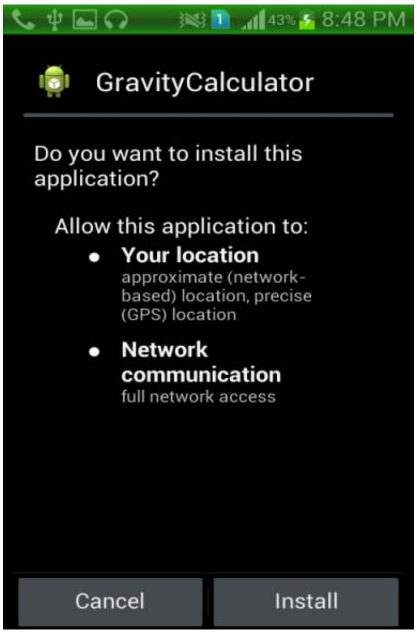
Permission Protection level	Value	Risk	Access provided for
normal	0	low	To isolated application feature
dangerous	1	high	To private user or data
signature	2	Very high	Only signed requesting application
signatureorSystem	3	Vey high	Only to application signed with same certificate

3. Gravity Calculator APP and Sample App Permissions

A sample test app named as gravity calculator is created whose task is to calculate gravity at some height above the earth. As we all are aware that gravity values changes from location to location. The value of gravity is maximum at the equator and minimum at the poles .This app uses GPS and then calculates the value of gravity using latitude and longitude values. The app requires permission to use GPS API. The permission pseudo code in android.manifest.xml file is given below.

```
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
 package="com.example.gravitycalculator"
```

```
    android:versionCode="1"
    android:versionName="1.0" >
    <uses-sdk
    android:minSdkVersion="8"
    android:targetSdkVersion="18" />
    <uses-permission
android:name="android.permission.ACCESS_FINE_LOCATION"/>
    <uses-permission
android:name="android.permission.ACCESS_COARSE_LOCATION"/>
    <uses-permission android:name="android.permission.INTERNET"/>
   ...
   .. </activity>
    </application>
   </manifest>
```



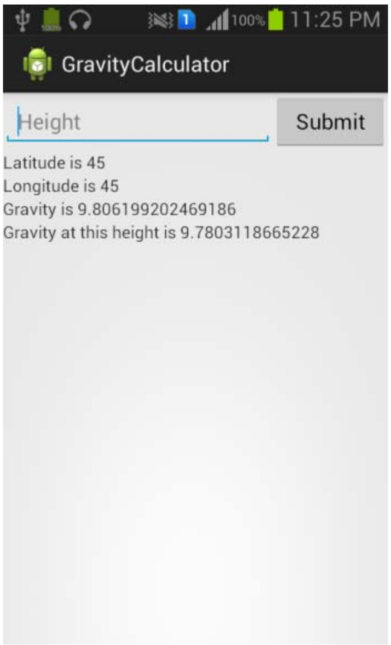
GravityCalculator

Do you want to install this application?

Allow this application to:

- **Your location**
  approximate (network-based) location, precise (GPS) location
- **Network communication**
  full network access

Fig. 3.1: Screen shot of Gravity Calculator notifying the user for permission to use GPS API.



GravityCalculator

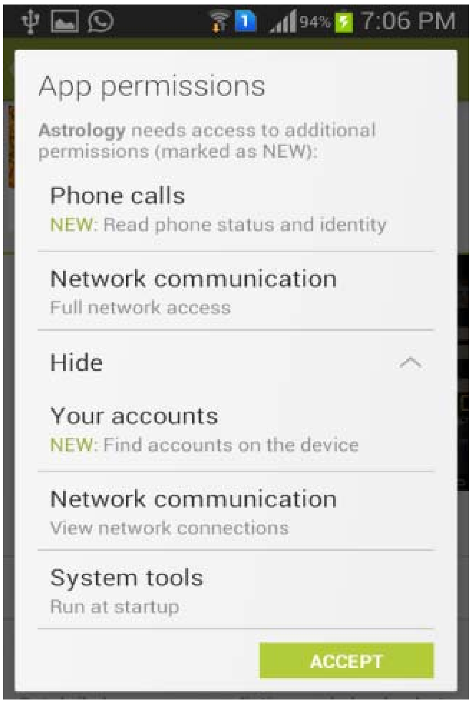
Height

Latitude is 45
Longitude is 45
Gravity is 9.806199202469186
Gravity at this height is 9.7803118665228

Fig. 3.2: Screen shot of Gravity Calculator after permission is granted by end user.

Besides this there are other permissions as well which we have tested on other sample apps.

```
   <uses-permission
android:name="android.permission.ACCESS_FINE_LOCATION"/>
    <uses-permission android:name="android.permission.INTERNET"/>
   <uses-permission
android:name="android.permission.ACCESS_COARSE_LOCATION"/>
```



7:06 PM

App permissions

Astrology needs access to additional permissions (marked as NEW):

Phone calls
NEW: Read phone status and identity

Network communication
Full network access

Hide

Your accounts
NEW: Find accounts on the device

Network communication
View network connections

System tools
Run at startup

Fig. 3.3: Sample App.

4. Conclusion

In this paper we created a sample app to show the usage and level of protection of permissions in android. We have studied various permissions and their protection levels. The importance of in self created permissions was analyzed and was studied in detailed manner. These permission protection levels when used at development level can guard the other application and data from hacking and malicious apps. The apps should not over privileged to access the content from the phone.

References

[1] Google; Android; http://www.android.com.

[2] Google; Android permissions; http://developer.android.com/reference/android/Manifest.permission.html.

[3] . Sheran A. Gunasekera(2013) Android Apps Security

[4] I. Rassameeroj and Y. Tanahashi,(2011) "A Various Approaches in Analyzing Android Applications with its Permission-Based Security Models", Proceedings of 2011 IEEE International Conference on Electro/Information Technology, Mankato, MN, USA

[5] Aditi Tripathy and G.P Potdar,(2012) " Framework for Providing Selective Permissions to Android Applications" IOSR Journal of Computer Engineering, p- ISSN: 2278-8727Volume 13, Issue 3, PP 53-58

[6] Adrienne Porter Felt, Elizabeth Hay, Serge Egelman, Ariel Haneyy, Erika Chin, and David Wagner, (2012) “Android Permissions: User Attention, Comprehension, and Behavior”Symposium on Usable Privacy and Security (SOUPS) 11-13, Washington, DC, USA

[7] Android Open Source Project. Android Security Overview,2012.

[8] P. G. Kelley, S. Consolvo, L. F. Cranor, J. Jung, N. Sadeh, and D. Wetherall(2012), “A Conundrum of Permissions: Installng Applications on an Android Smartphone”. In Proceedings ofthe Workshop on Usable Security (USEC), 2012

[9] A. P. Felt, E. Chin, S. Hanna, D. Song, and D. Wagner.(2011) “Android Permissions Demystified” In Proceedings of the ACM Conference on Computer and Communication Security.