DROIDHOOK: Android Malware Detection Through Context Filtering

Aisha Ali-Gombe, Irfan Ahmed, Golden G. Richard III
Dept. of Computer Science, University of New Orleans
{aaligomb, iahmed4, golden}@uno.edu

ACSAC 2013

Abstract

Android malware has evolved over time with significant increases in both number and stealth. A large proportion of this malware masquerades as benign applications, which are downloaded by users through both primary and secondary Android markets. When installed, malware performs a benign activity that interacts with the user, while at the same time creating a malicious service in the background. A recent survey of over 1260 malware instances revealed how background services are used to steal user data like the device ID, pictures, GPS location information, etc. Quite a number of these malware instances also used the telephony functionality to send unauthorized messages and calls without the user's Knowledge (Zhou et al, 2012).

Some of the detection mechanisms proposed by researchers revolve around the use of machine learning through static analysis of APK files, dynamic analysis using virtual machine introspection and analysis of log files.

The drawbacks of these approaches include high chances of false positives and the inability to detect malicious activity due to obfuscation (encryption, payload downloaded after installation, etc.)

PROPOSED STRATEGY

DroidHook is an Android API hooking mechanism that attempts to deter services unknown to an Android user from transferring "user sensitive data". The aim of this technique is to compliment Android's permission-based security architecture. Research has shown that the permission system is ineffective, largely due to its overbroad nature and the user's misunderstanding of its importance or lack of general security awareness.

DroidHOOK

The proposed strategy modifies how services are implemented within the Android system. When a service makes an API call, DroidHOOK gets alerted and determines if the service is a user service or system service. DroidHOOK only interacts with user services running in the background for which IBinder returns NULL or has no Binder to the main Application activity. In essence, for which no activity has set up a listener to that particular service.

1. Calls to hardware driver for the camera, GPS, and Bluetooth radio by such services get intercepted and user authorization is required to proceed, else the service is destroyed.
2. Calls to READ operations on Content Provider (e.g., Contacts) or Telephony data (e.g., IMEI Number) also get intercepted. DroidHOOK hashes return value(s) and/or files and saves both Service Information and hash values.
3. Calls to open SOCKET for Internet transfers triggers packet inspection by DroidHOOK. If the payload value hashes to any hash stored for the same service, the user receives an alert. A decision is then made based on user response.
4. Calls to Telephony manager to sendSMS and sendMMS also get intercepted and the user is notified. Again, a decision is made based on user response.

DEVELOPMENT

Presently, DroidHook is under development as an Android kernel module to carry out the specified functionalities. The system is set to be complete and available for testing early October 2013.

DETECTION ARCHITECTURE