Curbing Android Permission Creep
Encouraging Least Privilege in development

Timothy Vidas
Nicolas Christin
Lorrie Faith Cranor

W2SP 2011
Motivation

- Developers specify required permissions
- Applications must be granted permissions at install time by the user – it's “all or nothing”
- Why does a Battery app require internet access?
Motivation

- Unlike the iPhone's model, users are not prompted when restricted objects are accessed.
- If an app attempts access w/o having already been granted a permission, it crashes.
Android Permissions

- More than 100 application level permissions govern access to resources
  - Some are very broad:
    - INTERNET
  - Some are very specific
    - READ_SMS
  - Some are very obtuse
    - ACCESS_SURFACE_FLINGER
  - Some are very generic / ambiguous
    - DIAGNOSTIC
Android Permissions

- As a developer, knowing when to use them is not always clear
- API documentation has some guidance
- Source provides some
- Debugger...

```java
/**
 * Get the friendly Bluetooth name of the remote device.
 *<p>The local adapter will automatically retrieve remote names when
 * performing a device scan, and will cache them. This method just returns
 * the name for this device from the cache.<p>
 * Requires {link android.Manifest.permission#BLUETOOTH}
 *
 * @return the Bluetooth name, or null if there was a problem.
 */
public String getName() {
    ...
}
```

Class Overview

A connected or connecting Bluetooth socket.

The interface for Bluetooth Sockets is similar to that of TC
new BluetoothSocket to manage the connection. On

The most common type of Bluetooth socket is RFCOMM

To create a BluetoothSocket for connecting to a know
connection fails.

To create a BluetoothSocket as a server (or "host"),

Once the socket is connected, whether initiated as a clie
connected to the socket.

BluetoothSocket is thread safe. In particular, close/

Note: Requires the BLUETOOTH permission.
Android Permissions

- What permission does restartPackage() need?

```java
public void killBackgroundProcesses (String packageName)

Have the system immediately kill all background processes associated with the given package. This is the same as the kerr
You must hold the permission KILL_BACKGROUND_PROCESSES to be able to call this method.

Parameters
   packageName   The name of the package whose processes are to be killed.
```

```java
public void restartPackage (String packageName)

This method is deprecated. This is now just a wrapper for killBackgroundProcesses(String); the previous behavior here is no longer availab
```

```java
killMethod = ActivityManager.class.getMethod("killBackgroundProcesses", String.class);
```

- What does killMethod() need?
Further Motivation

- Downloaded ~34,000 applications from the Android Market
  - More than 4% contained *duplicate* permission specifications
  - Every category contained some applications that contained duplicates

Note: these applications are *packaged applications* that typically have no associated source code
Android Development

- Android Development Tools (ADT) is an Eclipse extension that facilitates development
  - Developers can choose to not use an IDE, but ADT integrates all the additional tools such as the emulator, debug bridge, etc
Developer Aid

- Enable Eclipse to notify the user that they are including an unnecessary permission (or missing a necessary one)
- Performed via static code analysis along with a permission-API database
- Does not require any alteration to existing devices or the Android framework
Permission Check Tool

End result is an Eclipse notification denoting an error when a permission is not needed.
Tool Analysis

- Without large sets of source code, it is difficult to perform extensive empirical analysis of a source code oriented tool
- Even so, several open source applications were found to have extraneous permissions
  - Quite likely due to common circumstances such as the removal of a feature and forgetting to remove the associated permission in the manifest
Conclusions

- The tool is a functional proof of concept that is compatible with Android ADT Eclipse extension
- The tool highlights permission discrepancies so that the developer has to “go out of the way” in order to include additional permissions
Future Work

- Utilize a more robust permission database
- Extend the Eclipse plugin to be even more user friendly such as using an Eclipse nature to provide real time nudges
- Create corpora of source and packaged applications to facilitate future research projects
- Adjust the definition of proper operation, to include required permissions that the user deems unnecessary (not the source code)
Cylab Usable Privacy and Security Laboratory

http://cups.cs.cmu.edu/

Carnegie Mellon