**CSCI 5800 Cybersecurity Programming**

**Group Fork Bomb**

**Design Document**

**Team Members**

Ramiro Arenivar, Sindhuja Nandikonda and Karthish Muthappa Ponnakachira

**Project Description**

Our team will design and implemented an Android application that will be a bad practices detector (in regards to security), enforce pre-determined security settings and perform a general security audit that will grade the overall security of the Android device. Since Android is an open source platform, companies all over the world use it as the operating system for their devices. Therefore, the default security settings are not uniform across all Android devices. This can result in potential security vulnerabilities in some devices. The goal of this project is to contribute an application to the community that will make Android devices more secured.

In order to accomplish this, the Android application will do the following:

1. **Enforce the use of a device passcode**. One of the most important and simple features for securing a mobile device is to add a passcode to unlock it.
2. **Define a minimum length for the device passcode**. Having a passcode that is only two characters defeats the whole purpose of a passcode since it will be relatively easy to crack. By enforcing a minimum length passcode, we will greatly reduce the possibility of someone cracking it.
3. **Enforce passcode expiration time out.** By setting a passcode expiration, we will make sure that it is regularly changed.
4. **Set maximum inactivity time out before device is locked**. Ensures that the device is locked after a certain amount of time.
5. **Wipe device data after a maximum number of passcode attempts**. In case the device is lost, to make certain that someone trying to get access to the device is not successful, after a number of tries the device will automatically wipe its data.
6. **Require that the device is encrypted**. Protects all data in the device.
7. **Provide option for the user to disable camera**. There is a possibility of a hostile application to turn on and use the device’s camera without the user knowing. The application will give the user an option to disable the device camera to prevent this.
8. **Scan all current apps and their permissions (provide feedback about risky permissions)**. By scanning all the installed apps and their current granted system permissions, we can give the user feedback about which permissions represent more risk. We will rank the risk on the different permissions using the research paper “Exploring Permission-Induced Risk in Android Applications for Malicious Application Detection” which ranks each of the Android permissions by how much risk they represent.
9. **Provide feature for the user to disable unsecured WIFI connections**. The application will give the user feedback when he/she connects to an unsecured WiFi access point. Connecting to an unsecured WiFi opens the user to potential hackers using tools such as Packets Sniffers to get information (specially when using sites that don’t establish a secured connections, such as HTTP, FTP, etc.).
10. **Check if install from unknown sources settings is checked**. There are a number of security settings that are hard to find for the average mobile user, one of these is the install from unknown sources. By letting the user know if this setting is enable, we will greatly increase the device security by preventing the installation of apps that have not been signed by a developer or vetted by the Google Play store.
11. **Add feature that audits the general security of a device by doing the following:**
    1. **Grade permissions granted by apps**
    2. **Check if passcode is strong (length, type of characters)**
    3. **Check if device is encrypted**
    4. **Check if camera is disable**
    5. **Check type of networks used in the past (tell the user when they have been using an unsecured WiFi connection)**

**Tool Uniqueness**

Currently, based on our research, there isn’t an application in the market that combines all of these features:

* A bad practice security detector.
* Enforcement of predetermined security settings.
* Performs a general audit which grades the overall security of the device so that the users can have an idea on how secured their mobile device is. Additionally, they can view and set recommended settings to further increase the security of the mobile device.

**Programming Languages**

The programming language use for Android development is Java. Additionally, the user

interface is defined using XML files to describe and defined the different UI elements. Android provides the Android SDK library to access the platform’s API.

**Team members experience with Java, Android platform and SDK**

* Ramiro Arenivar is fairly comfortable using the Java programming language (he has some previous experience using Java). As far as the Android platform, he’s had some experience develop a fairly simple application in a Mobile Computing class.
* Sindhuja Nandikonda is comfortable programming in Java. She’s done projects using Java in previous courses. She’s new to work on Android Platform, but as Android development uses Java she will be able to work on developing this project.
* Karthish Muthappa Ponnakachira is fairly comfortable C# which is similar to Java and also has a fair amount of experience with the android platform and SDK through a simple application developed in a Mobile Computing class.

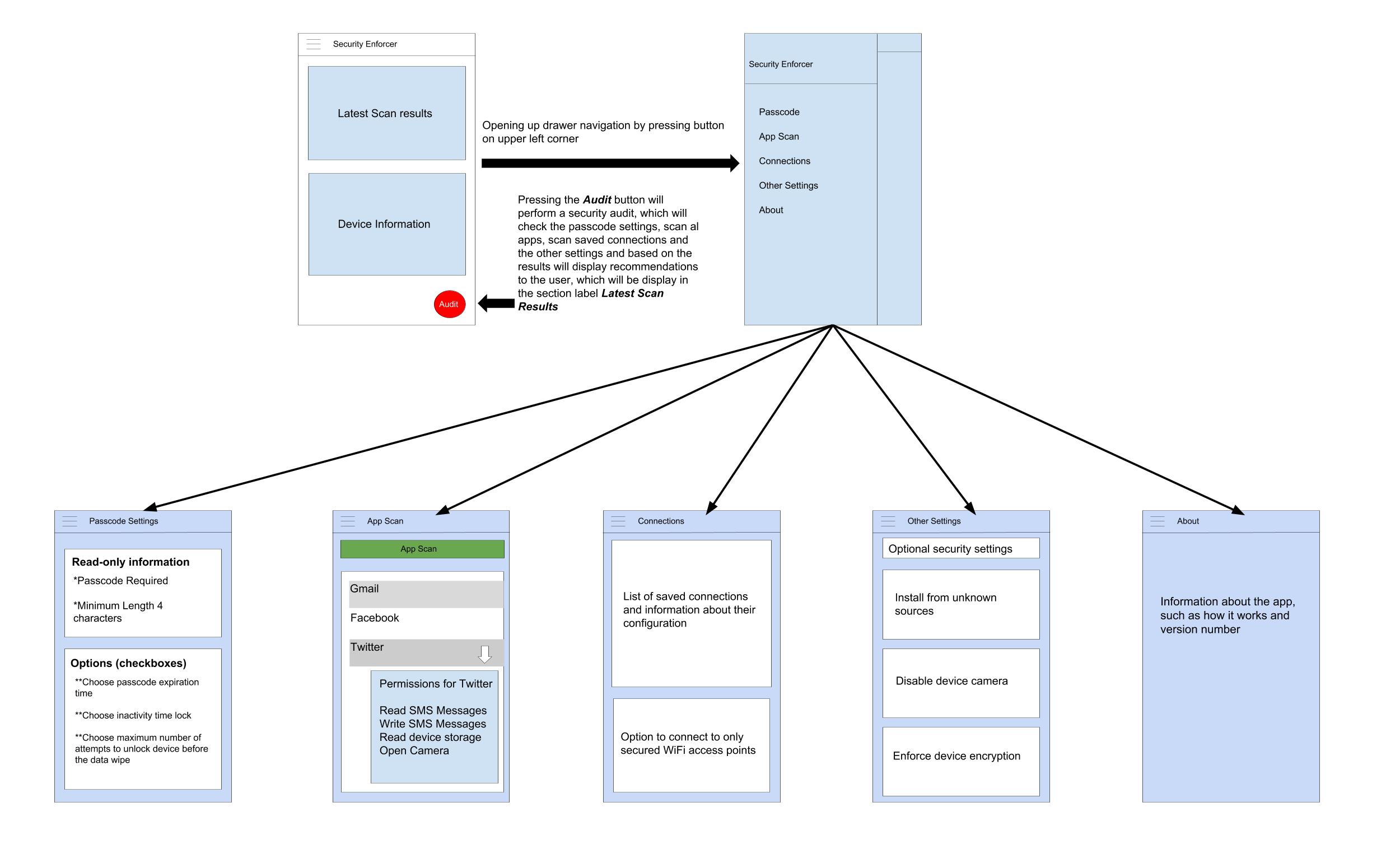
**Test Environment**

Since the project is a mobile application in the Android platform, our test environments are:

* Physical Android devices
* Android emulators provided by Android Studio

Additionally, the minimum Android OS version our project will support is Android 4.4 (KitKat) which was release almost four years ago. According to Google, approximately 86% of all Android devices are running KitKat or later versions.

**Storyboard**



The app implementation will be divide it up between the team members in the following manner:

* Ramiro Arenivar
  + Responsible for the user experience and user interface design
  + Implement the features of the Other Settings and Connections screens
* Sindhuja Nandikonda
  + Responsible for the features in the Passcode screen
* Karthish Muthappa Ponnakachira
  + Will work on the App Scan screen and the Audit functionality