April 8, 2013

Joseph Yan
Professor and Technical Coordinator, Electrical and Computer Engineering
University of British Columbia

Subject: Final Report for EECE 496

Dear Professor Yan:

I am submitting my report entitled DroidWatcher: Final Report as the final report to meet the EECE 496 course requirements.

The main purpose of this report is to describe the system components the purpose of the project I have been working on, the DroidWatcher Android application. The report will go into detail about the project objectives, system components, blocking issues, application deployment and the future of the application.

I hope that this report will meet your approval.

Respectfully yours,

Vinson Yu

Droid Watcher Final Report

COURSE: EECE 496
ASSIGNMENT: FINAL REPORT
STUDENT NAME: VINSON JIA HUI YU
TUDENT NUMBER: 10190072
ECE TECHNICAL SUPERVISOR: ILLDAR MUSLUKHOV
DATE: APRIL 8, 2013

J HUI YU

Abstract

The Android operating system does not currently provide any accountability in the login system. A history of when the user has logged into their mobile device is not available for users to view after they have logged in. The DroidWatcher Android application has features to provide accountability for the login system of Android mobile devices. This application was developed for the studies of accountability and information security in Android mobile devices. The DroidWatcher has been tested for functionality and will be used to aid studies in the near future. The results collected in these studies will aid in providing suggestions to improving the security features of Android mobile devices.

Glossary

Activity – An activity class provides a presentable layer that is visible to the user and allows the user to interact with it. For example, an e-mail application that displays the email and allows the user to delete the email would be an activity.

Intent - Intents are a data structure defined by the Android operating system that is passed between activities, services and broadcast receivers. Additional information can be appended to intents for a customized message before they are sent.

Broadcast Receiver - Broadcast receivers are components in the Android operating system that receives messages from the operating system or other applications. Broadcast receivers will listen to specific messages sent by the system and start the set of tasks it is programmed to do. Broadcast receivers are initiated when they are registered into the system. Registering the broadcast receivers notifies the system that specific messages will be handled by registered broadcast receivers. When the desired tasks are completed, the system will unregister the broadcast receivers from the system. For example, the operating system may broadcast a message to the entire system that it is low on battery and a broadcast receiver that is listening for a low battery message will hear the message then respond by shutting down applications.

Service - A service is a component of the system that is running long term without interacting with the user directly. Services run in the background of the application. For example, a service may be a component in the background of an e-mail application syncing the inbox for new messages.

Contents

List of Figures

1.0 INTRODUCTION

This report presents the DroidWatcher application and its system components as well as how it will be used to aid studies in information security.

The main objective of this engineering project is to create a software application on the Android platform for mobile devices. The purpose of this application is to collect information from the system in real-time and log this information periodically. The information logged by this application are as follows:

1) The time it takes the user to log into their device
2) Duration of time the screen of the device was on for
3) Applications that the user has opened and when the applications were opened

To log real-time data, the application must be designed such that its uptime is maximized. Hence it is an objective to keep the uptime of the application as high as possible.

There are over 400 million Android devices activated around the world as of 2012 and this number continues to grow. The security and accountability of how Android devices are accesses on a daily basis need to continue to improve. The data gathered by this application aims to aid in future studies in security of mobile devices and provide suggestions for improving accountability of mobile devices.

By default, the Android platform does not provide the user with any login accountability. If multiple people were to access the same device, the device would not be able to determine the different users. That is, a potential adversary can take the device and be able to access information about the victim if they knew the login passcode and no digital traces of these actions would be left behind. This application is intended to fill this void by logging when a user has logged into their device, how long it took them to do so and what applications activated during the usage session.

The scope of the project is limited to hardware and the software. This project is the first step and is intended to be expanded in the future. This software application is expected to be able take pictures with the front camera in the future time so the hardware should have a front camera. This software application is expected to be able to support screenshots in the future and to support this feature the Android operating system version must be over 4.0. This version of the application does not yet have these features implemented yet but does provide support for these features in the future. The remaining sections of this final report will discuss the DroidWatcher system itself and results of this project.

2.0 DROIDWATCHER SYSTEM

This section of the report will discuss the components of the system. The system is divided into 5 components: 2 broadcast receivers, 2 services and 1 activity. The system components communicate with each other by exchanging objects called intents. All these components are defined in the glossary. Figure 1 shows an overview of the entire system and how the components interact with each other.

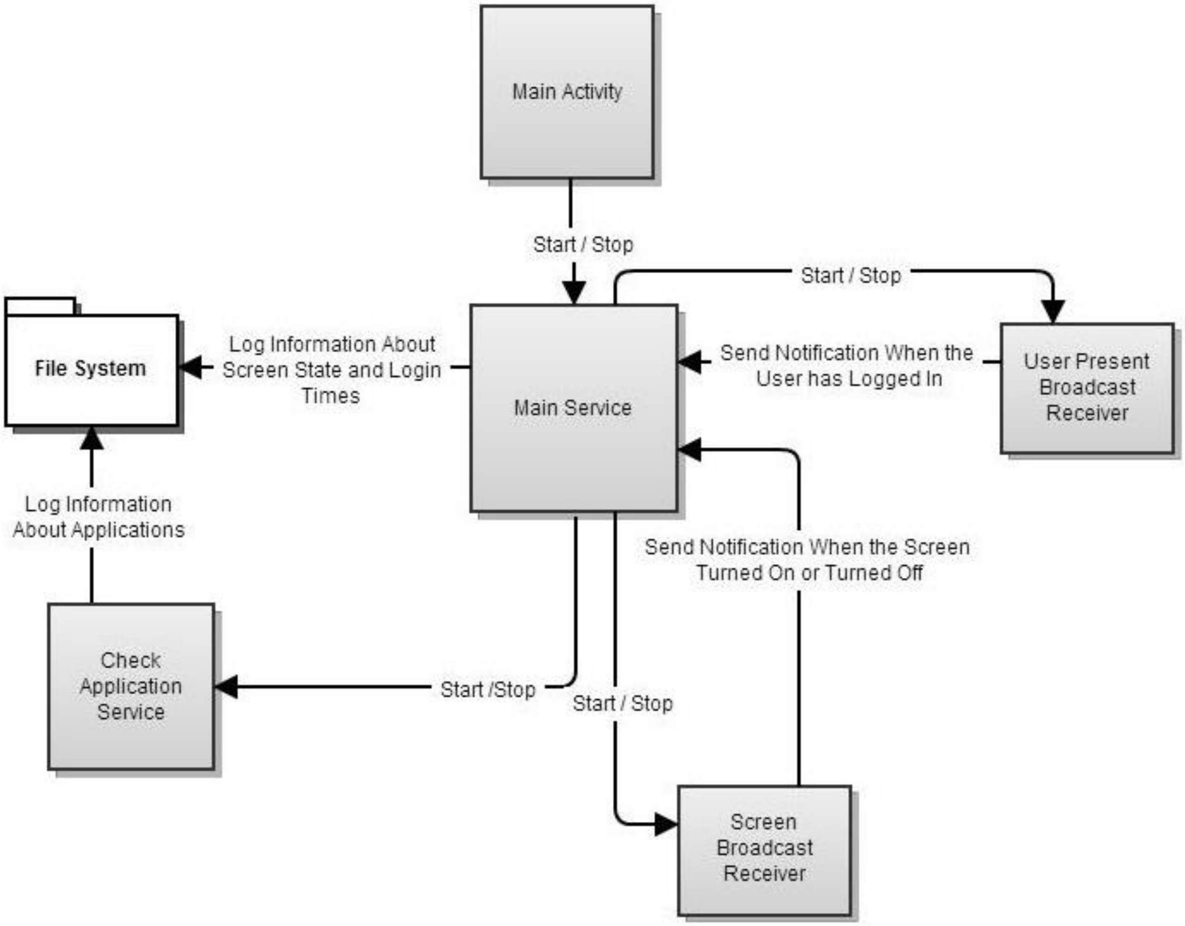
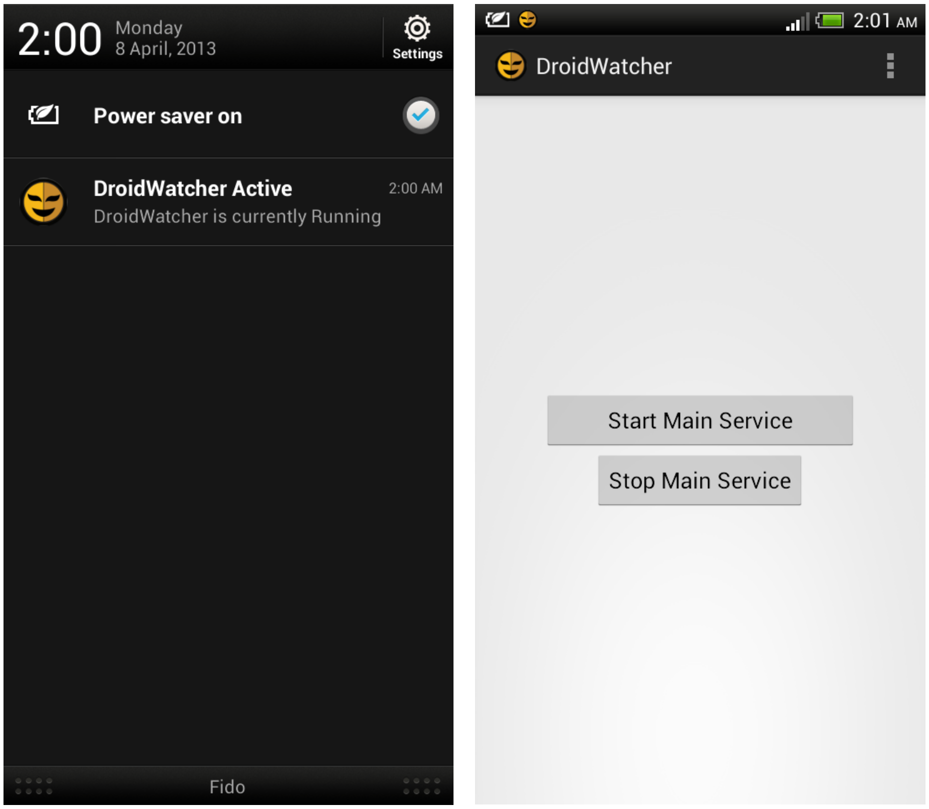

Main Activity
Start / Stop
Start / Stop
File System
Log Information About Screen State and Login Times
Main Service
Send Notification When the User has Logged In
User Present Broadcast Receiver
Log Information About Applications
Send Notification When the Screen Turned On or Turned Off
Check Application Service
Start /Stop
Start / Stop
Screen Broadcast Receiver

Figure 1 Overview of the system architecture

2.1 Main Activity

The Main Activity provides a user interface for the application. When the user opens the application, the Main Activity is started. From the Main Activity, the user can turn on or turn off the Main Service. Once the application has been started the Main Activity does not need to be present for the user to continue to use the application as the main purpose of this application is to have a service running in the background collecting information. A notification is shown if the Main Service is running in the background. The Main Activity can be started any time by pressing on the icon in the menu or by pressing the notification. Figure 2 shows a screenshot of the notification on the left and the Main Activity on the right. Notice that a notification icon is shown on the top left of the user interface of the device on the top of the Main Activity. This is also present at all times to let the user know that a background service is running.



2:00 Monday 8 April, 2013

Power saver on

DroidWatcher Active 2:00 AM
DroidWatcher is currently Running

Figure 2 Notification from the DroidWatcher and the Main Activity

2.2 Broadcast Receivers

The DroidWatcher Application contains two broadcast receivers: the Screen Broadcast Receiver and the User Present Broadcast Receiver. The Screen Broadcast Receiver will listen for messages sent by the operating system about when there is a change in the state of the screen. When the screen is turned off or is turned on the Screen Broadcast Receiver will be sent a message to handle the event. The User Present Broadcast Receiver will listen for messages sent by the operating system when a user has entered the correct password that unlocks the mobile device.

2.2.1 Screen Broadcast Receiver

The Screen Broadcast Receiver will listen for messages from the operating system every time there is a change in state of the mobile device screen. This broadcast receiver is registered into the system by the Main Service. The purpose of this broadcast receiver is to notify the Main Service of when the screen of the device is turned on and when the device has been turned off. Figure 4 summarizes the sequence of events of how the broadcast receiver does this.

First the screen state change occurs and the operating system sends a message to the broadcast receiver. The broadcast receiver will then create an intent with the time that the screen was turned or turned off in milliseconds. The broadcast receiver will then send the intent to the Main Service. When the Main Service is turned off, it will unregister this broadcast receiver so that it longer receives messages from the operating system.

2.2.2 User Present Broadcast Receiver

The User Present Broadcast Receiver will listen for messages from the operating system every time the user successfully logs into the device. This broadcast receiver is registered into the system by the Main Service. The purpose of this component is to notify the main service when the user has logged onto the device. Once the user has entered the correct passcode to log into their device, a message is sent to this broadcast receiver. The broadcast receiver will create an intent with the time the user has logged onto the device and send the intent to the main service.

2.3 Service

DroidWatcher has two services: the Main Service and the Check Application Service. The Main Service manages the Check Application Service and is the key component of this application. Figure 4 shows how every component interacts with the Main Service.

2.3.1 Main Service

The Main Service is started by the Main Activity of the application and remain running until it is turned off by the Main Activity. This component is persistent and will continue to restart itself if it crashes or if the operating system attempts to shut it down. It is important that the service remains on as long as possible as its goal is to collect and log information real-time. This component is connected to the two broadcast receivers, the

Main Activity, the other service and the file system.

Once the Main Service has been created, it will also register the two broadcast receivers and create a notification icon to notify the user that the service is running in the background. The Main Service does not do anything until it receives an intent from the broadcast receivers or the Main Activity. Once it has received an intent the Main Service will determine what sort of intent it is to perform the correct set of tasks.

If the intent is sent from the Screen Broadcast Receiver, this service will check if it is indicating that the screen has been turned on or turned off. If the intent indicated that the screen has been turned on, this service will create a folder with the time stamp as its name. If the intent sent from the Screen Broadcast Receiver indicates that the screen has been turned off a file is created indicating the duration the screen was on for. This file will be saved in the folder that was created when the screen was turned on. The Main Service will shut down the Check Application Service if the screen has been turned off. In the “Alt” box of figure 4, we see the 3 different intents based on the events of the phone and how the Main Service is involved with each event.

When the intent is received from the User Present Broadcast Receiver the service records the time taken for the user to log into their device. The service will then start the Check Application Service by sending it an intent to start up. Figure 3 shows what is saved into the file system every time the screen has been turned on to the point it has been turned off.

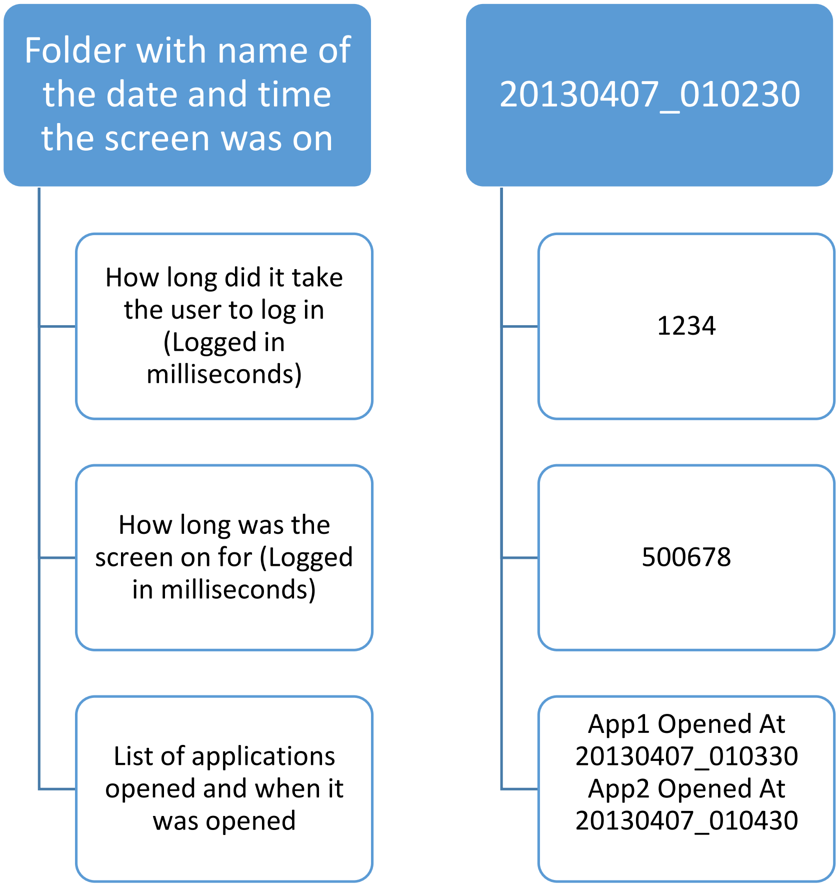

Folder with name of the date and time the screen was on
How long did it take the user to log in (Logged in milliseconds)
How long was the screen on for (Logged in milliseconds)
List of applications opened and when it was opened
20130407_010230
1234
500678
App1 Opened At 20130407_010330 App2 Opened At 20130407_010430

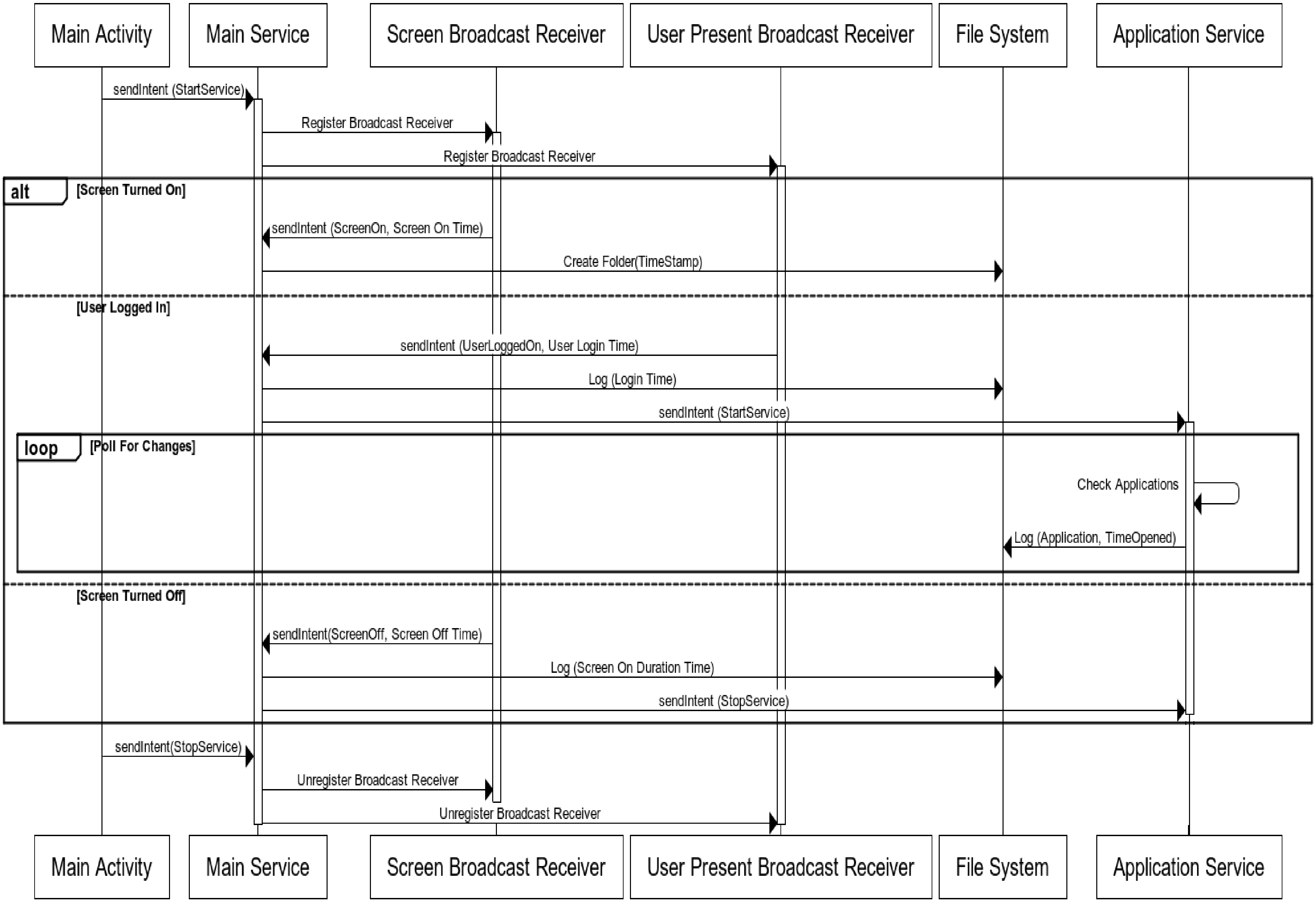
Figure 3 The file and its contents logged in the file system

2.3.2 Check Application Service

The Check Application Service is a service that intended to check for when applications have been activated. This service runs on a separate thread so that the system uses less resources. This service polls the system task manager every 2 seconds for changes in the applications list. That is, the service will perform a series of tasks every 2 seconds to track what applications the user is using. If the service has detected that the user has opened a new application this service will log what application was opened and time it was opened into the file system. More specifically, the information will be logged in the folder that was created by the Main Service when the screen was turned on. The Check

Application Service is turned off by having the tread destroyed by the Main Service every time a screen off intent is sent to the Main Service. When the screen is off, the user cannot change applications hence the Check Application Service should be turned off.

DroidWatcher Sequence Diagram


Main Activity
Main Service
Screen Broadcast Receiver
User Present Broadcast Receiver
File System
Application Service
sendIntent (StartService)
Register Broadcast Receiver
Register Broadcast Receiver
alt
[Screen Turned On]
sendIntent (ScreenOn, Screen On Time)
Create Folder(TimeStamp)
[User Logged In]
sendIntent (UserLoggedOn, User Login Time)
Log (Login Time)
sendIntent (StartService)
loop
[Poll For Changes]
Check Applications
Log (Application, TimeOpened)
[Screen Turned Off]
sendIntent(ScreenOff, Screen Off Time)
Log (Screen On Duration Time)
sendIntent (StopService)
sendIntent(StopService)
Unregister Broadcast Receiver
Unregister Broadcast Receiver
Main Activity
Main Service
Screen Broadcast Receiver
User Present Broadcast Receiver
File System
Application Service

www.websequencediagrams.com

3.0 RESULTS

This section will discuss the results of the project. After the initial stages of development for this project, the application will be used for field studies. The application will go through deployment and collect results for the studies. The application is expected to change in the future. The remaining of this section will be divided into the following subsections: Application Deployment, Project Evaluation, Difficulties and Blocking Issues and future development of DroidWatcher.

3.1 Application Deployment

This application has been in testing on a HTC ONE S, a mobile device is running on Android version 4.1.1. This application has been successfully running on this mobile device for over three weeks and has collected over 1400 data points. The main service has been shut down 8 times over two weeks during heavy usage of the mobile device and the system was low on memory. The service would restart itself within 2-3 seconds every time it was shut down. This application is expected to be deployed to more systems in the near future for studies.

3.2 Project Evaluation

The development phase of this project has been successful but the overall success of this project has yet to be measured because the application has not been deployed to other devices yet. The application has meet all the original objectives stated in section 1.0. There were objectives added to the application that were not met. The following objectives were discussed and added to the project for future development:

1) Taking a picture with the front camera upon login into the device without the user knowing
2) Taking several screenshots of the display of the device after the user has logged into their device
3) Post processing features for the images taken from objective 1 for a more clear picture if needed
4) Create daily logs and send them to a database when the user has turned on their wireless internet

Objectives 1-3 were attempted but have not been fulfilled due to constraints build into the operating system. These constraints will be discussed in the next section.

3.3 Difficulties and Blocking Issues

The lack of understanding of the Android operating system is the prime reason for some of the problems that have been experienced in the development of this application. Progress was slow at the start of this project as a lot of time was spent studying. Two major blocking issues remain that prevent some of the other objectives from section 3.2 from being developed.

Objective 1 from section 3.2, taking a picture with the front camera without the user knowing is a difficult task because in order to take a picture or record a video with the camera a preview must be shown. This is a constraint built into the Android platform and one known workaround is to create a camera preview with 1 pixel high by 1 pixel wide. With a preview that small the user will not notice the preview.

Objective 2 from section 3.2, taking screenshots of the display of the device is a difficult objective because the Android platform does not allow it. The reason why screenshots are generally not permitted is because of security reasons. One application may use screenshot features to obtain information from another application on the device if screenshot features were built into the platform. However, specific devices from certain original equipment manufacturers such as Samsung and HTC provide ways to obtain a screenshot that is triggered by the user. Because screenshot features are provided through the manufacturers and not from the developers of the Android platform, the methods of obtaining a screenshot may differ from device to device.

3.4 Future Development of DroidWatcher

The foundation of the DroidWatcher project has been completed and the objectives from section 3.2 are the next objectives to be fulfilled in this project. To support future studies, the application may be modified to suit these studies. Modifications may include collecting more or less information about the user or notifying the user about their actions. The application will then be used to collect information from users to aid the studies of security and accountability in information security.

4.0 CONCLUSION

This report covers the system components of the DroidWatcher Android application and the motivation behind the project as well as how the application is expected to continue to develop in the future.

The developed application has 5 major components, Main Activity, Main Service, User Broadcast Receiver, Screen Broadcast Receiver and Check Application Service. All of the components are related to the Main Service and communicate with it with intents. These intents let the Main Service know when the screen has been turned on, when it has been turned off, how long it took the user to log into their device, how long the screen was on for and what applications were opened after the user has logged in.

This application is expected to grow and have more features in the future and may be tailored to fit different studies in information security. The main features to be added for the DroidWatcher Application is the ability to take a photograph with the front camera once the user has logged into their device without them knowing and the ability to take screenshots.