

Tether Transpose

Divyanshu Bhartiya (10250)

Vaibhav (10780)

Advisor: Dr. T.V Prabhakar

Dept. of Computer Science and Engineering

{divbhar, vaibhavv, tvp}@iitk.ac.in

November 18, 2013

CS455 Project

1 Introduction

In this era of smart phones, the optimum use of phones can be targetted only when the phone has a working data plan activated on it. Most of the users feel reluctant in buying a data pack for their phone, if they have a working wifi connection on their laptops or their computers. Users prefer to create a wifi hotspot and use that service on their phone. However it becomes tedious to create a hotspot while using softwares such as Connectify. It requires a lot of troubleshooting and most of the apps don't seem to work properly. Hence, we propose a software application that will do this task using a USB cable. This software system will be an android application which will target a large number of smart phone users. This application aims to provide internet sharing from your laptop to mobile when they are connected through USB cable. More specifically, this application will prove to be advantageous over the traditional Wi-Fi sharing as your smartphone will not get drained while using internet from the laptop, and in fact it will continuously be getting charged. No database will be used.

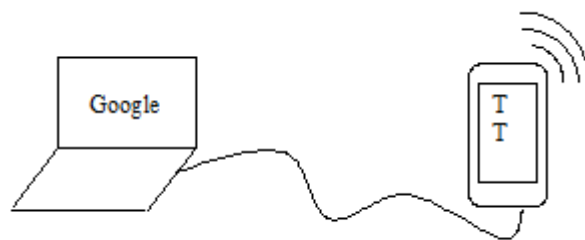


Figure 1: Overview

2 User Stories

- There are situations where one can't do tethering over wifi, for example, your company doesn't let you make any hotspot at work.
- Suppose you don't want to spend money for a mobile plan, and you want a more stable and faster Internet connection than wifi.

- You don't want your PC and your Android to drain battery because of wifi.
- There may be times when you don't have mobile/wifi network and your PC doesn't have any wifi adapter or you want your Android to be charged while in use.
- Many times when we use internet over wi-fi it happens that not all applications run properly.
- Hotspot creation and sharing of internet is not always possible and troubleshooting is required.

3 Requirements

3.1 Hardware

- Laptop : Supported Hosted network adapter
- Smartphone : USB cable.

3.2 Software

- Ubuntu : iptables and ifconfig binaries
- Windows : Network Connection Manager
- Smartphone : ROOTED phone, ip and ifconfig binaries

3.3 Constraints

- The phone must be rooted as the change in network configurations is done at system level not at application level to facilitate working of most of the applications.

4 Functionalities

4.1 Internet Sharing

This was the foremost requirement of the application and is implemented. The challenge here was that the sharing of internet was to be done through the USB and also internet is not application specific, i.e once the reverse tethering is started all the applications on the mobile can use the internet.

4.2 Traffic Counter

As soon as one starts the application, the counter for uploaded and downloaded data starts, so one can keep track of the amount of data he/she is using. This can be particularly useful when you are running low on your data plan.

4.3 Log Generation

This function is for the developers of the application, where for each session ,a log will be generated and sent to the developer for bug fixing and further improvement.

4.4 Feedback

This functionality allows users to send feedback to the developer in case the app doesn't respond and is not able to complete the task. The feedback option will allow the user to enter his message and append the text to the system generated log and send to the user.

4.5 User Friendly

As there are few things that needs to be done prior to using the application, which are different for both linux and windows. So for the same the user will be guided through this user friendly guide for a smooth experience.

5 Design

Here are the (4+1) architecture views of the system.

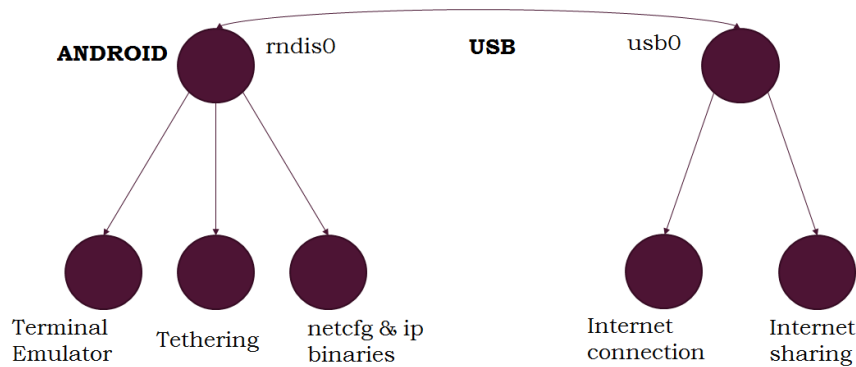


Figure 2: Logical View

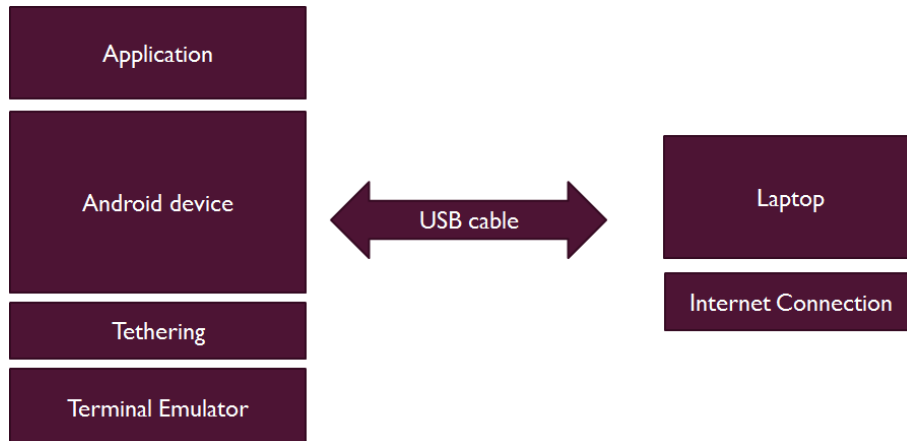


Figure 3: Physical View

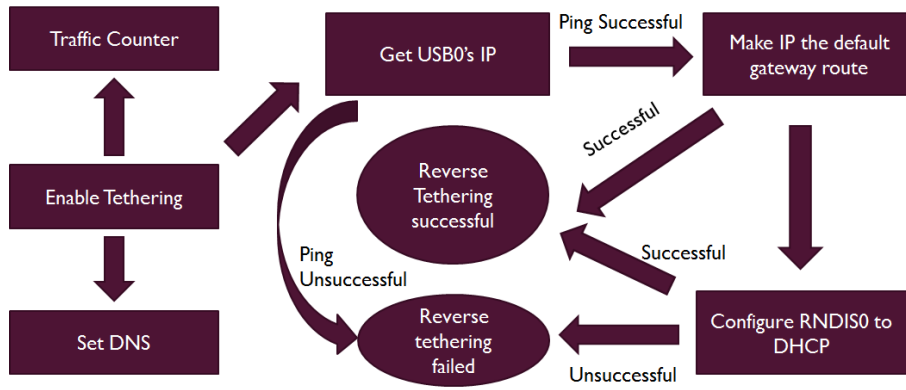


Figure 4: Process View for Android Device



Figure 5: Process View for the Computer

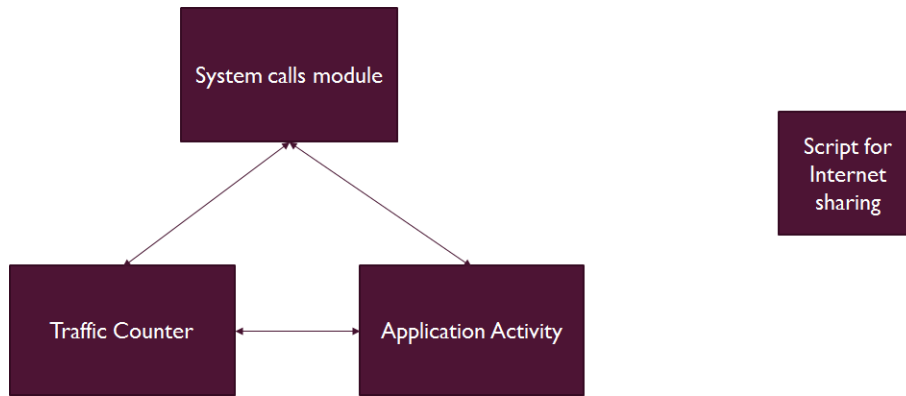


Figure 6: Development View

6 Implementation

- Android ADT on Eclipse IDE
- Both static ip and dhcp configuration
- RootTools.jar used for running root commands on android
- Traffic Counter
- Feedback
- User Guide

7 Testing

The application has been tested on the following basis

- Checked on both rooted and non rooted phones
- Validation of USB plugged is made

- Validation of binaries and tethering support is made
- Validation of gateway by ping
- Most of the applications work
- Works on both Windows and Linux

8 Quality Attributes

- The application is built fault tolerant, and handles all the boundary cases.
- The application does the configuration briskly and there is not a lot of delay.
- The application is highly usable due to the presence of "Guide" which allow users to get a way about working on the app.
- The application is scalable as it works for both windows and linux.
- The application provides for user feedback and hence provides usefulness too.

9 Acknowledgement

We thank Prof. T.V Prabhakar for his valuable support throughout the project, guiding us from time to time and looking into the project when it was needed. We are also thankful to the course teaching assistant Ashish Agrawal.

10 Snapshots

