**Software Requirements**

**Specification**

**for**

**Android Network Sniffer**

**Version 1.0**

**Prepared by Soh Yu Xuan, Timothy Chin, Kenneth Huang, Kendrick Tan**

**22 June 2018**

# Table of Contents

[Table of Contents ii](#_Toc5834)

[Revision History ii](#_Toc5835)

[1. Introduction 1](#_Toc5836)

[1.1 Purpose 1](#_Toc5837)

[1.2 Document Conventions 1](#_Toc5838)

[1.3 Intended Audience and Reading Suggestions 1](#_Toc5839)

[1.4 Project Scope 1](#_Toc5840)

[1.5 References 1](#_Toc5841)

[2. Overall Description 2](#_Toc5842)

[2.1 Product Perspective 2](#_Toc5843)

[2.2 Product Features 2](#_Toc5844)

[2.3 User Classes and Characteristics 2](#_Toc5845)

[2.4 Operating Environment 2](#_Toc5846)

[2.5 Design and Implementation Constraints 2](#_Toc5847)

[2.6 User Documentation 2](#_Toc5848)

[2.7 Assumptions and Dependencies 3](#_Toc5849)

[3. System Features 3](#_Toc5850)

[3.1 System Feature 1 3](#_Toc5851)

[3.2 System Feature 2 (and so on) 4](#_Toc5852)

[4. External Interface Requirements 4](#_Toc5853)

[4.1 User Interfaces 4](#_Toc5854)

[4.2 Hardware Interfaces 4](#_Toc5855)

[4.3 Software Interfaces 4](#_Toc5856)

[4.4 Communications Interfaces 4](#_Toc5857)

[5. Other Nonfunctional Requirements 5](#_Toc5858)

[5.1 Performance Requirements 5](#_Toc5859)

[5.2 Safety Requirements 5](#_Toc5860)

[5.3 Security Requirements 5](#_Toc5861)

[5.4 Software Quality Attributes 5](#_Toc5862)

[6. Other Requirements 5](#_Toc5863)

[Appendix A: Glossary 5](#_Toc5864)

[Appendix B: Analysis Models 6](#_Toc5865)

[Appendix C: Issues List 6](#_Toc5866)

# Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Date** | **Reason For Changes** | **Version** |
|  |  |  |  |
|  |  |  |  |

# 1. Introduction

## 1.1 Purpose

The purpose of this document is to present a detailed description of the Android Network Sniffer. It will explain the Features of the system, Interface, what the system can do, what are the possible constraints and limitation when using the Network Sniffer, possible additional features of the Network Sniffer compared to currently known Android Network Sniffer. The document is intended for developers and users of the system.

## 1.2 Document Conventions

This document follows the basic SRS convention methodology and also based on templates that are online in which to draft this document

## 1.3 Intended Audience and Reading Suggestions

This document is intended for developers, project managers, users, testers and documentation writers. The SRS below contains information regarding the project, scope of the project, references used in writing the SRS, testers who if possible solve any issues that the current developers may have faced, and also for users who wish to understand what the project created was about. It is suggested when reading to have knowledge with regards to Networking, Android, Android Programming, Java, C Programming, and System Security related knowledge. As these knowledges will help in understanding the idea, mechanism and features of the Project

## 1.4 Project Scope

The scope of the project is to develop a android network sniffer application whereby network traffic can be sniffed and examined. However, there may be limitations that may be faced which will be further explained in the later sections

Objectives are to have the ability to perform real time scanning of network traffic and save the data into a file which can be viewed later, additional features such as filtering of traffic, and possibly achieve cracking of WEP

## 1.5 References

# *<https://krazytech.com/projects/sample-software-requirements-specificationsrs-report-airline-database>*

<https://aakashtechsupportdocs.readthedocs.io/en/latest/prodpersp.html>

# 2. Overall Description

## 2.1 Product Perspective

The product is considered a follow-on member of a product family, as there are currently existing Personal Computer as well as Android Network Sniffer available. Our product is Android based on the implementation of a Network Sniffer.

The main features of our product that are included

* Network Sniffing: To be able to Start & Stop Network Sniffing
* To be able to provide Real Time Viewing of Sniffed Data
* To be able to Save the Sniffed Data into a File
* To be able to read from the saved File
* To be able to filter Network Traffic
* Additional Feature if possible: Being able to Crack WEP

## 2.2 Product Features

## *The features of our product allow the users to do Sniff for Network Traffic, and see the data packets and can save it to a file for viewing at a later time. Cracking of WEP, have a GUI, being able to manipulate captured data as well as display it*

## 2.3 User Classes and Characteristics

The type of User classes for our product would be mostly IT users, developers, students.

IT Users: users that work in the field of IT such as System Administrators, who constantly have work involving Network such as Monitoring of Network Traffic for anomaly, or even regular data collection of network traffic to improve the system such as avoiding overhead. IT users would be the type of users that most frequently use Android Network Sniffers

Developers: Users that wish to develop a similar type of Application may want to user our product as reference or possibly create a more improved version of ours as the technology improves

Students: Students are users who are similar to IT users but use our product for personal use and for studies and also wish to approach this field of work in the future

## 2.4 Operating Environment

The device of our product choice must support Android, programming language used are C and Java.

Hardware of the device must have a chipset that allows NIC in monitor mode, Device also needs to be rooted

## 2.5 Design and Implementation Constraints

Because of the platform in which our product is on which is Android there will be limitation and constraints compared to the Personal Computer type of Network Sniffer. When designing our product, the constraints are that the device must be rooted which is to give Super User Access to the phone. However, it is to be noted that rooting a phone would be similar to what would be known as Jail Breaking a phone. once a phone has been rooted, all forms of warranty is gone. Other limitations would be once a phone has been rooted it will be susceptible to Malware as there would be no Anti-virus for the phone. Also, the phone’s chipset will play an important role in the choice of device that will be compatible with the product.

## 2.6 User Documentation

Other than the product of an Android Network Sniffer there will also be a Technical Documentation that elaborate clearly the entire technical aspect of the product

Other than the Technical Document there will also be a User Documentation to allow ease of use for the users who will be using our application

## 2.7 Assumptions and Dependencies

To be filled.

# 3. System Features

Below list our system features of the product and all possible features available

## 3.1 Network Sniffing Feature

Users should be able to start and stop the network sniffing of the application

#### 3.1.1 Description and Priority

This Feature is considered the core of our entire product where by the user is able to start and stop sniffing of network and once the High Priority

#### 3.1.2 Stimulus/Response Sequences

User Starts the Network Sniffing (Please Fill in)

#### 2.1.3 Functional Requirements

<Itemize the detailed functional requirements associated with this feature. These are the software capabilities that must be present in order for the user to carry out the services provided by the feature, or to execute the use case. Include how the product should respond to anticipated error conditions or invalid inputs.

Requirements should be concise, complete, unambiguous, verifiable, and necessary. Use “TBD” as a placeholder to indicate when necessary information is not yet available.>

<Each requirement should be uniquely identified with a sequence number or a meaningful tag of some kind.>

REQ-1:

REQ-2:

## 3.2 System Feature 2 (and so on)

# 4. External Interface Requirements

## 4.1 User Interfaces

Screenshots of the GUI based on the app & short description

## 4.2 Hardware Interfaces

Devices used must have chipset that allows Monitor/Promiscuous mode, device also needs to have root access. After Sniffing for packets, they may be saved in a Pcap format that can only be run by applications that can read this particular format.

## 4.3 Software Interfaces

<Describe the connections between this product and other specific software components (name and version), including databases, operating systems, tools, libraries, and integrated commercial components. Identify the data items or messages coming into the system and going out and describe the purpose of each. Describe the services needed and the nature of communications. Refer to documents that describe detailed application programming interface protocols. Identify data that will be shared across software components. If the data sharing mechanism must be implemented in a specific way (for example, use of a global data area in a multitasking operating system), specify this as an implementation constraint.>

# 5. Other Nonfunctional Requirements

## 5.1 Performance Requirements

<If there are performance requirements for the product under various circumstances, state them here and explain their rationale, to help the developers understand the intent and make suitable design choices. Specify the timing relationships for real time systems. Make such requirements as specific as possible. You may need to state performance requirements for individual functional requirements or features.>

## 5.2 Safety Requirements

Rooting the phone is one of the requirements of the device however it is to be noted that when rooting the phone there will be some effects that the user must be aware of

Disadvantages when Rooting phone

* Rooting immediately voids your phone’s warranty
* Rooting have a risk of “bricking” the phone: Device might become dead and unusable
* Poor performance: when additional features that need to be added might cause the device to lose performance speed and features
* Viruses: custom programs used might make changes to software codes which might have a chance of introducing virus

## 5.3 Security Requirements

There is a security issue when using the phone in which if the device is rooted there will be a risk of viruses because rooted device allows for customization in which the user might attempt to go for which might allow viruses to be more common and it is unlikely that a rooted device has anti-virus protection

## 5.4 Software Quality Attributes

<Specify any additional quality characteristics for the product that will be important to either the customers or the developers. Some to consider are: adaptability, availability, correctness, flexibility, interoperability, maintainability, portability, reliability, reusability, robustness, testability, and usability. Write these to be specific, quantitative, and verifiable when possible. At the least, clarify the relative preferences for various attributes, such as ease of use over ease of learning.>

# 6. Other Requirements

n/a

# Appendix A: Glossary

IT: Information Technology

Pcap: a type of file extension similar to .txt .html

# Appendix B: Analysis Models

<Optionally, include any pertinent analysis models, such as data flow diagrams, class diagrams, state-transition diagrams, or entity-relationship diagrams.>

# Appendix C: Issues List

< This is a dynamic list of the open requirements issues that remain to be resolved, including

TBDs, pending decisions, information that is needed, conflicts awaiting resolution, and the like.>