**Software Requirements**

**Specification**

**for**

**PacketEye**

Version 1.0

Prepared by

N120155

N120170

N120382

N120642

February 28, 2017

Table of Contents

Table of Contents ii

1. Introduction 1

1.1 Purpose 1

1.2 Project Scope and Product Features 1

1.3 References 1

2. Overall Description 1

2.1 Product Perspective 1

2.2 User Classes and Characteristics 1

2.3 Operating Environment 2

2.4 Design and Implementation Constraints 2

2.5 User Documentation 2

2.6 Assumptions and Dependencies 2

3. System Features 2

3.1 Network monitoring 2

3.2 Packet Capturing 6

3.3 Filter Packets based on IP address 6

3.4 Save captured Packets for future analysis 6

3.5 Load already saved files 6

4. External Interface Requirements 6

4.1 User Interfaces 6

4.2 Hardware Interfaces 7

4.3 Software Interfaces 7

4.4 Communications Interfaces 7

5. Other Nonfunctional Requirements 7

5.1 Performance Requirements 7

5.2 Safety Requirements 8

5.3 Security Requirements 8

5.4 Software Quality Attributes 8

Appendix A: Data Dictionary and Data Model 8

Appendix B: Analysis Models 12

# Introduction

## Purpose

The purpose of this SRS document is to describe the functional and nonfunctional requirements for release 1.0 of the PacketEye (Packet Sniffer) application. This document is intended to provide an unambiguous, concise, and complete list of requirements to help design the PacketEye application. This document will include constraints and show how to use the application.

## Project Scope and Product Features

PacketEye software is extensively used for protocol analysis and security. More precisely, this application can capture, decode, and display the required information that is passing through the network to the user. The information that the user can capture includes protocol, time of capture, source and destination IP, source and destination MAC, port numbers and the data. Features of this software includes a) Network monitor for wired networks (Ethernet)

b) Packet store/retrieve

c) Packet Filtering

d) Platform independent GUI

## References

1. Javatpoint website for GUI design

<http://www.javatpoint.com/java-swing>

# Overall Description

## Product Perspective

From a technical perspective, packet sniffers are the most sophisticated tools for data collection. It collects every possible bit of data that can be recorded, which creates security risks.

## User Classes and Characteristics

|  |  |
| --- | --- |
| Sender | A sender is any device that sends any request or data to the receiver through the network. Whenever the data is transmitting, the data is encrypted for the security purpose. |
| Receiver | A sender is also a device that gives response to the devices which sent the request to that device. |
| Intruder | Any device or person that uses the captured information for his own evil purposes like stealing the usernames and passwords. |
| Administrator | Any device or person that uses the captured information to enhance the security of the network. |

## Operating Environment

OE-1: The PacketEye application can work in any operating system (Linux, Windows, etc.) in which JDK (Java Development Kit) is installed.

## Design and Implementation Constraints

PacketEye software captures network data as well as provides sufficient means for the decision making process of an administrator. This application designed in a new model and defined its benefits over existing packet sniffers; the model was developed in Java totally. The aim of this model is to rewrite C language sniffer models into Java, and also develop a model that consumes little memory on the hard disk. In this model we are placed packet sniffer on a network in promiscuous mode to capture and analyze all of the network traffic.

CO-1: The system comprises of five independent modules that handles different tasks efficiently using JPCAP for sniffing.

CO-2: The system will work in promiscuous mode to capture and analyze all of the network traffic.

CO-3: All the GUI components shall design in Java awt and swings.

CO-4: All the back-end shall write in Java servlets.

CO-5: All the captured data shall save in .txt format.

## User Documentation

UD: The system shall provide an ‘help’ button to instruct the user about PacketEye.

And also the system shall provide ‘about’ button that describes the use of PacketEye and also designers.

## Assumptions and Dependencies

AS-1: The PacketEye captures all the ongoing traffic in the network.

AS-2: The PacketEye software shows a message to the user after every one hour that if he/she still wants to capture the packets.