

**Ain Shams University**

**Faculty of Engineering**

**4th CSE Computer Networks**

**Project documentation**

Names:

**Ahmed Hawaf Abd El-Hakim** Section: 1

**Amr Essam Mohamed Barakat** Section: 2

**Mohamed Ossama Abu Al-Hassan** Section: 3

**Mohamed Hatem Sayed** Section: 3

Submitted to:

**Dr. Ayman Bahaa**

**Eng. Aly Osama**

Table of Contents

[Overview 3](#_Toc501843445)

[User manual 4](#_Toc501843446)

[Code Notes **Error! Bookmark not defined.**](#_Toc501843447)

# Overview

This project is an application that sniffs packets sent and received by your device. It supports both Windows and Unix platforms.

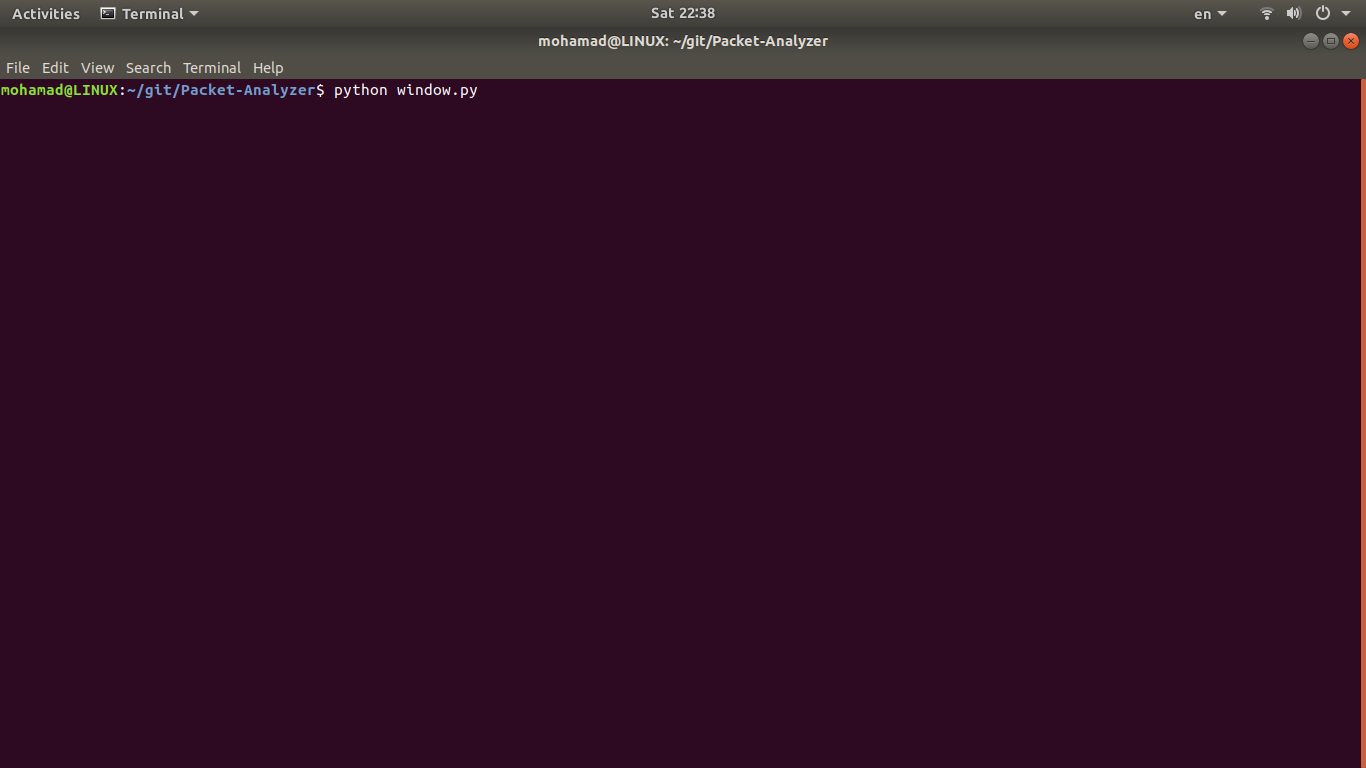
Inside the GUI you can choose an interface to sniff on (e.g. Wireless or Ethernet). On clicking the packet, you will have more details about, that will be described later.

This project is implemented using Python 2.7 and PyQt 4.0, so you’ll have to install both before running this application.

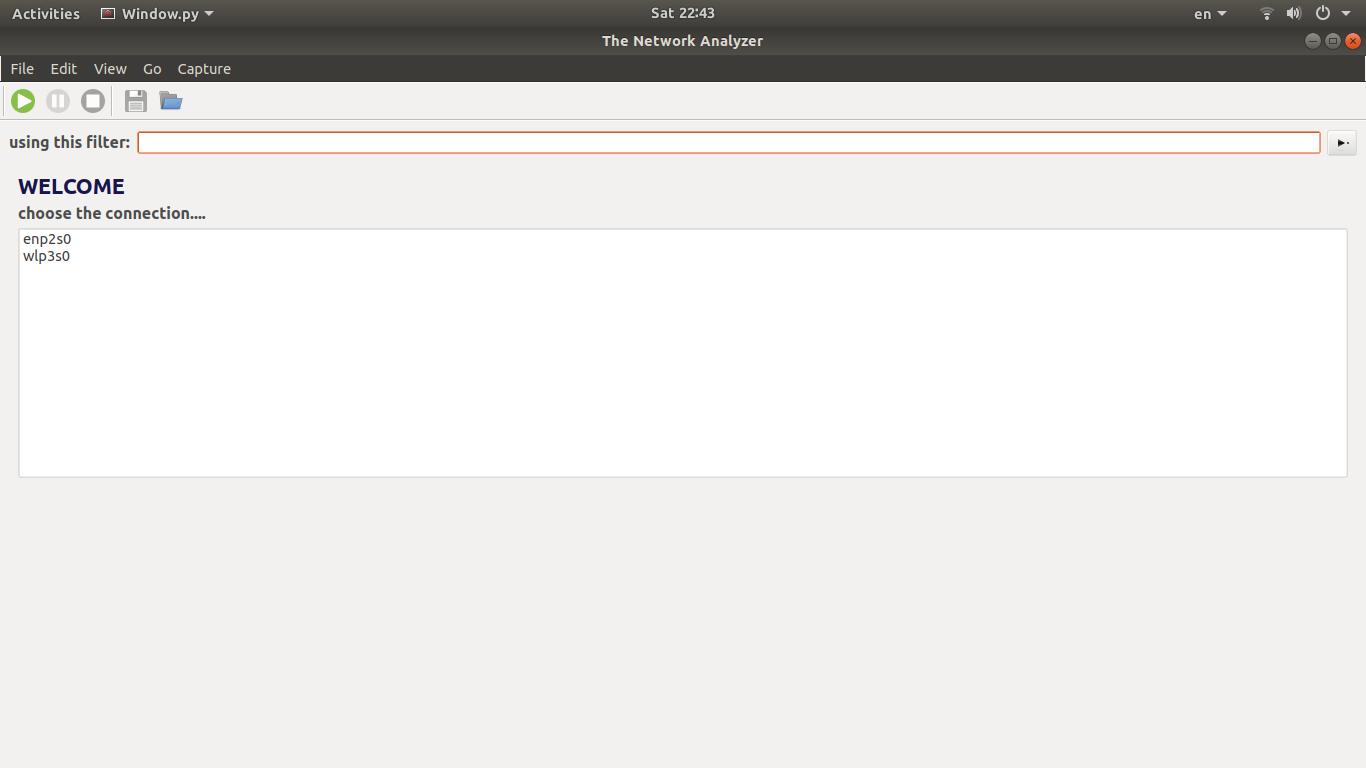
# User manual

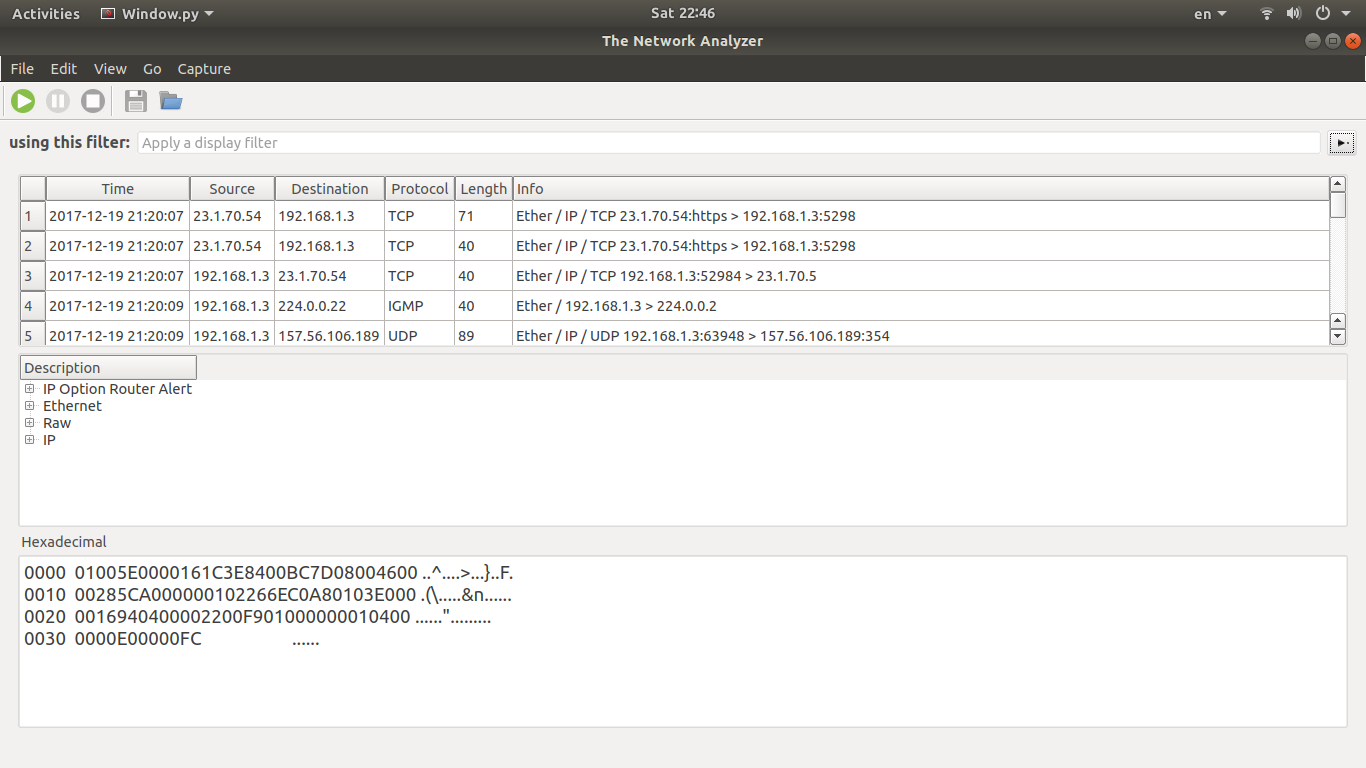
In this section we’ll show how to use this application’s features.

1. Run window.py file using an appropriate way. For example, run it through CLI of UNIX as following.

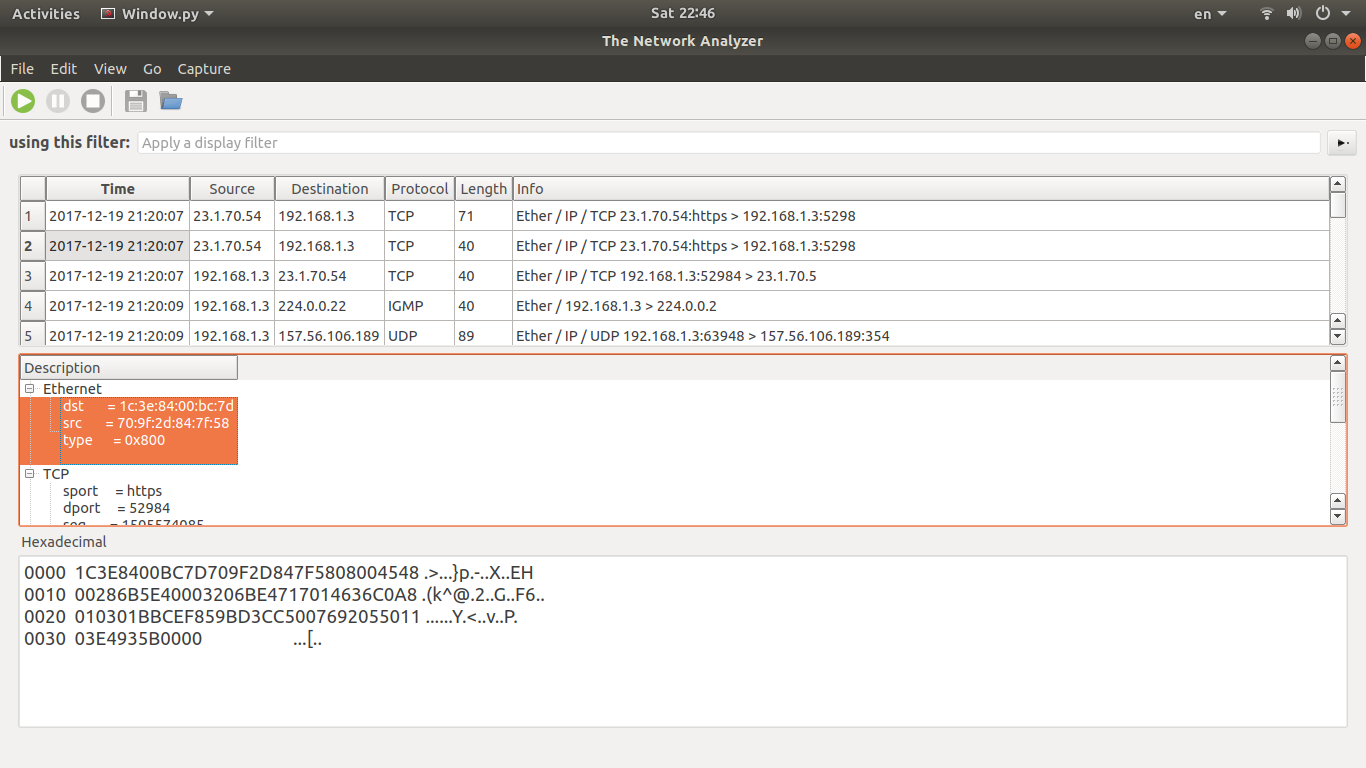


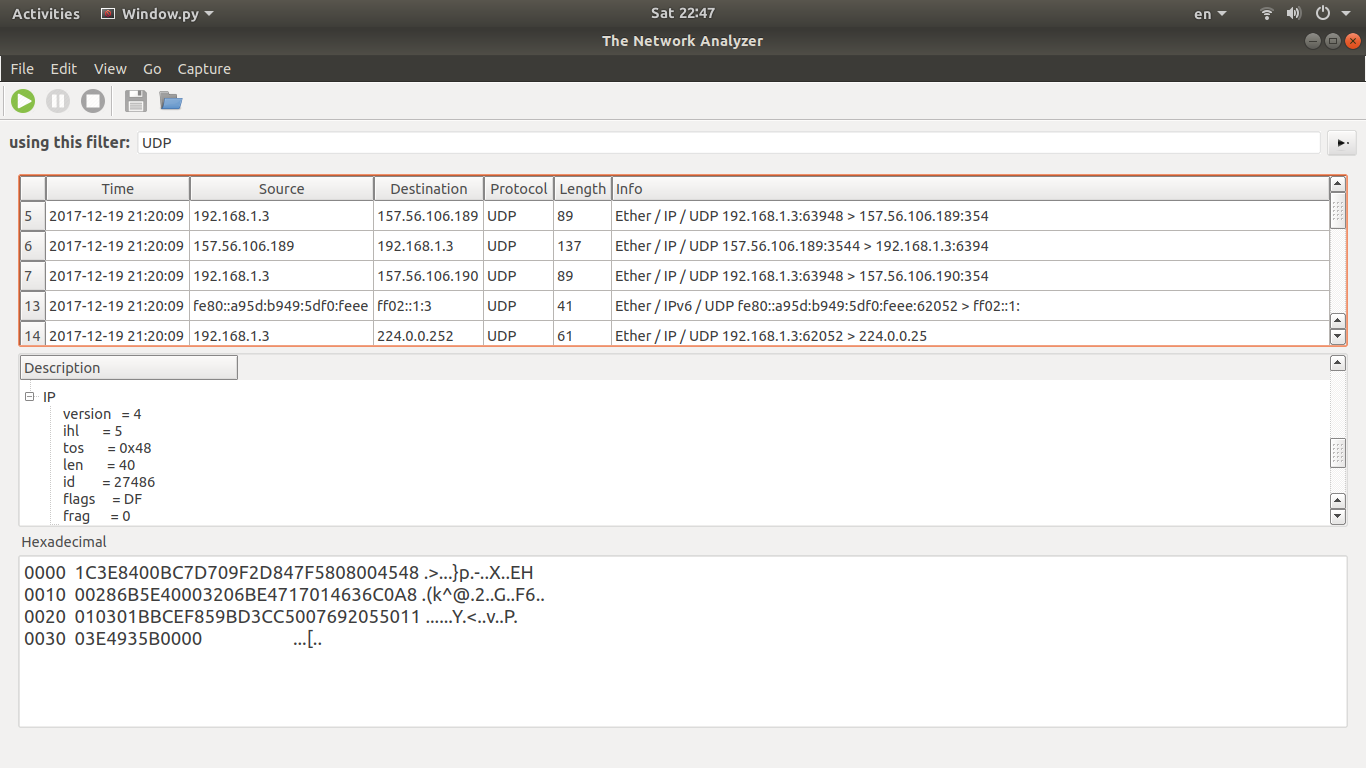
It should be noted that you may need to run the python file using sudo.

1. A window will appear that shows the available interfaces that you can sniff on.
2. Choose an interface then hit the green start button at the top of the window.
3. Packets sent and received will be shown hit the stop button to stop sniffing.
4. A window like the following will appear showing packets and details about it like source, destination, length, protocol and time.

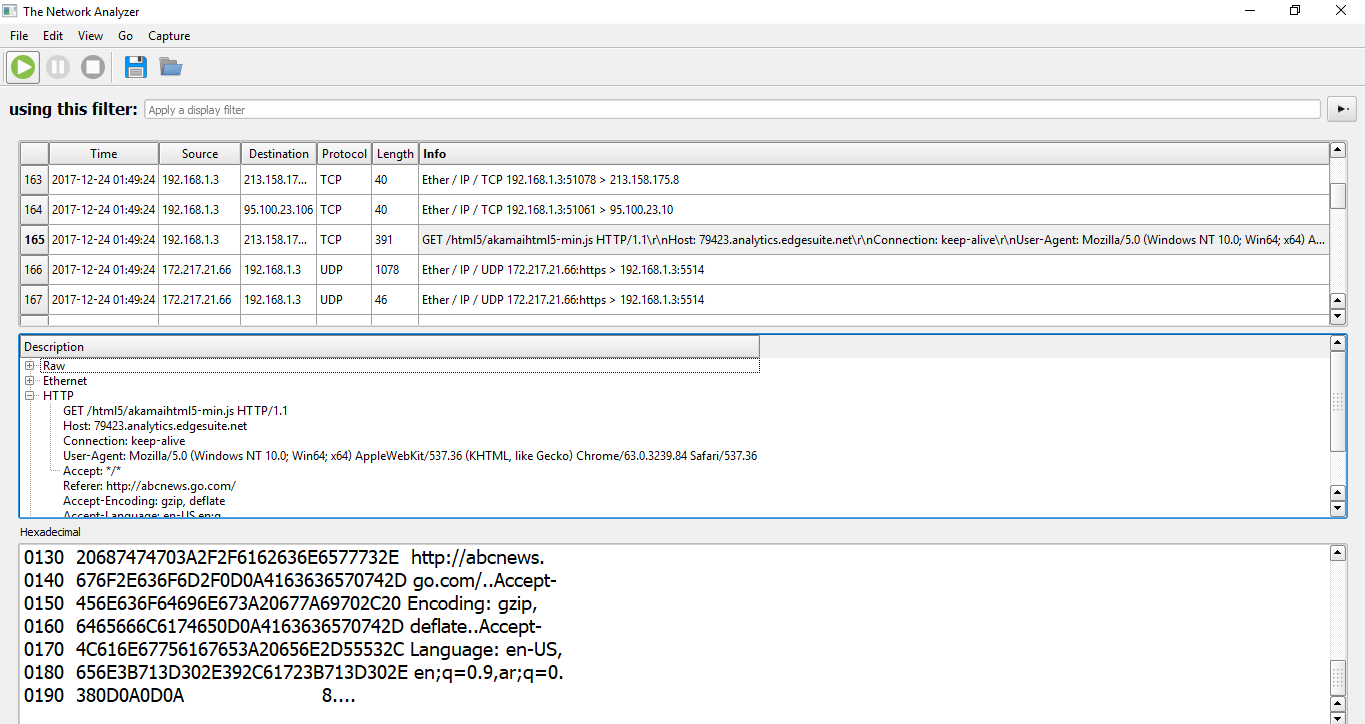


1. The previous window shows more details about any selected packet. To show these details all you need to do is to click on the packet and expand the fields shown in the second field to view these details.

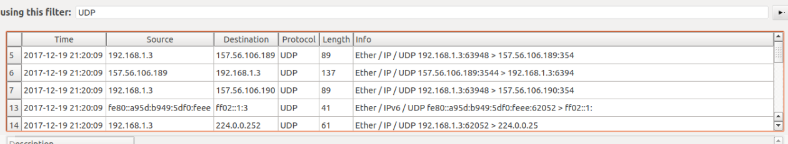




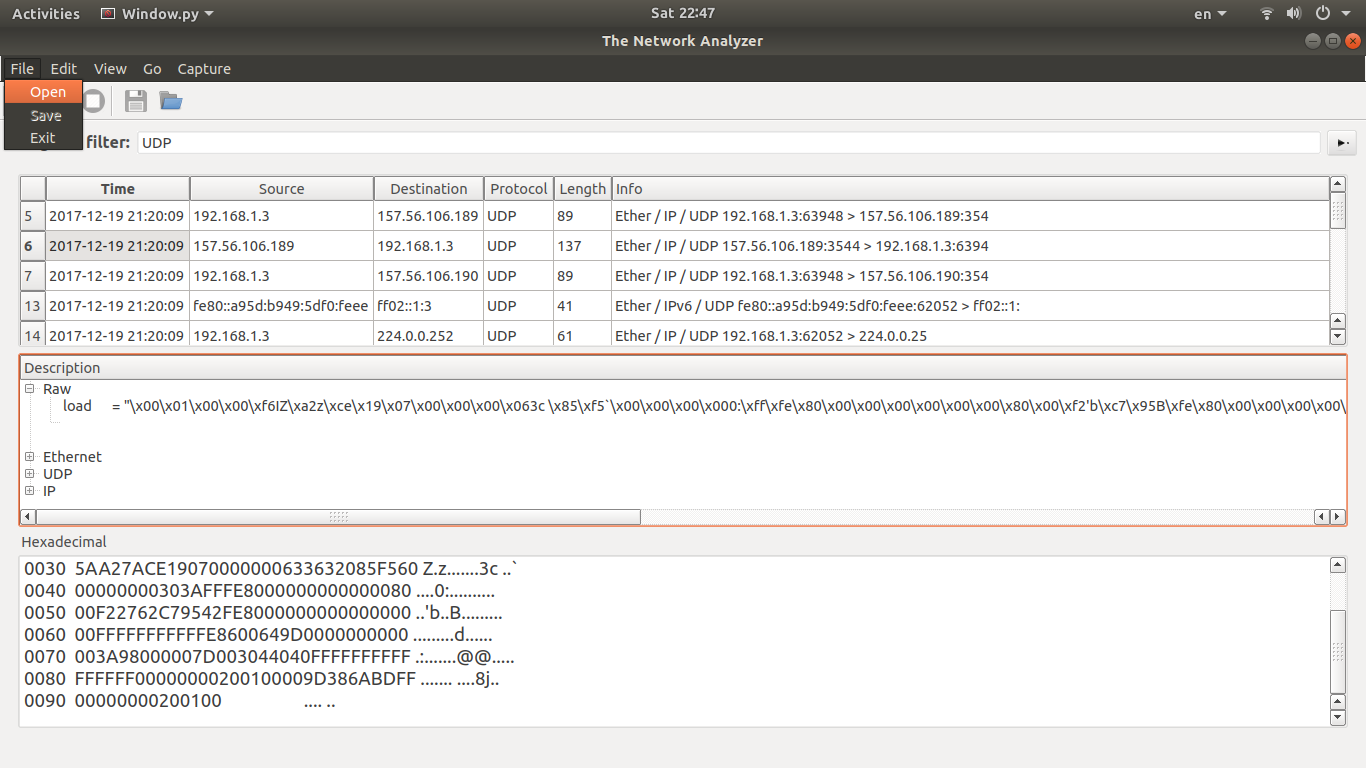
Another Example of showing the details of an Http packet is shown below

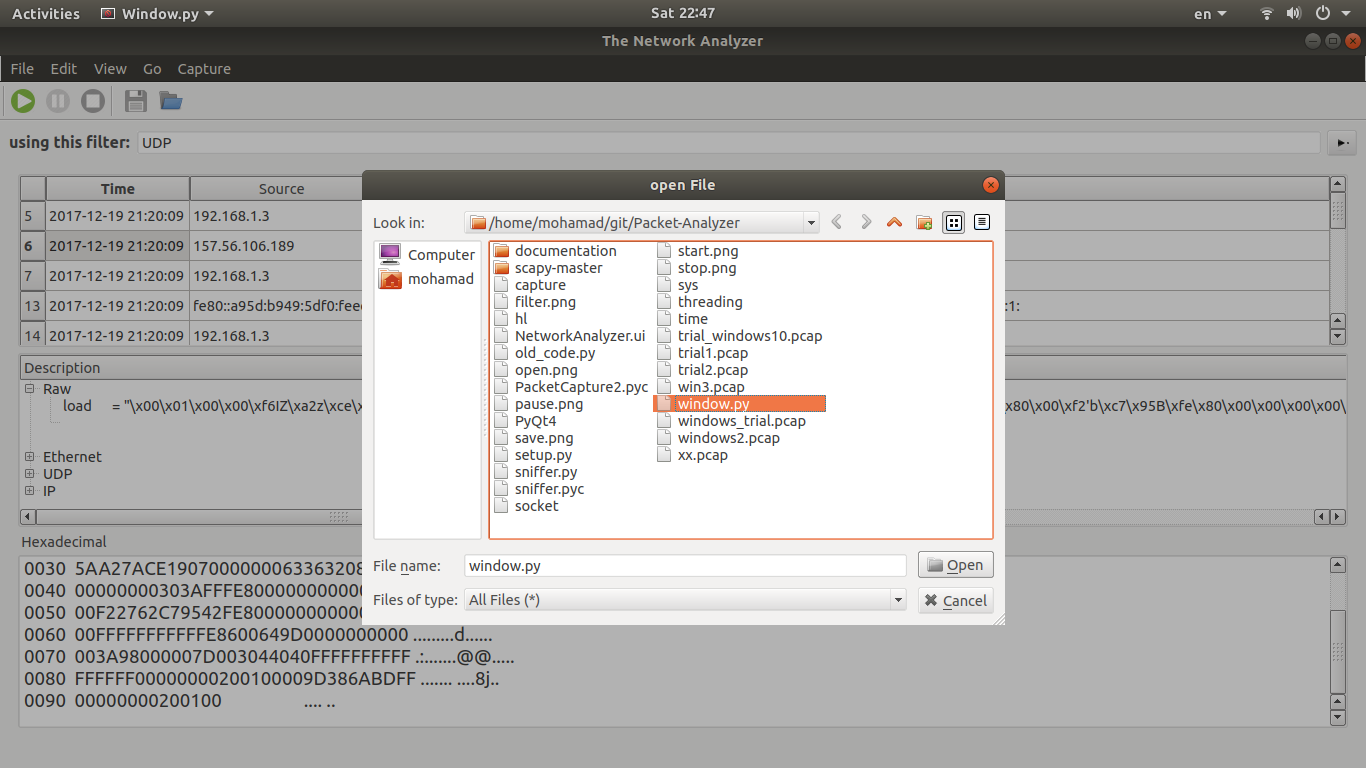


1. You can also apply a filter by entering the filtering value in the text box and press the button on the right to show the packets after applying this filter.



1. To save or load packets details: press the file button and select save. Then you will be directed to a window to browse the save or load location for this file as shown.





# Implementation and Features Notes

* The code relies on a python library called Scapy , but this library don’t handle http so it was handled by modifying the code in Scapy library and parsing the output.
* The GUI is made using PyQt4 and the used python version is python 2.7
* The GUI runs in the main thread and the sniffing (Background process) runs in another thread that sniffs using the Scapy library.
* The using of Scapy is abstracted by a class called Sniffer that handles and parse all the output from the Scapy library and passes it to the GUI.
* You can save and load file in pcap extention and it was tested on wireshark.
* The length column specifies the Packet length.
* You can apply filter to any column.
* All types of packets are sniffed, some of them were handled manually by parsing the output of Scapy such as Http and IPv6.