

Manual de Usuario

Manual de Usuario

Índice

Introducción.....	3
1.1. Descripción del sistema	3
1.2. Novedades	3
Requisitos previos	3
2.1. Conocimientos previos	3
2.2. Requisitos técnicos	3
Instalación y configuración de la aplicación	3
3.1. Instalación de Python	4
3.1.1 Instalación en Windows.....	4
3.1.2 Instalación en Linux	4
3.2. Instalación de PyQt4.....	8
3.2.1 Instalación en Windows.....	8
3.2.2 Instalación en Linux	9
3.3. Instalación de Pcapy	12
3.3.1 Instalación en Windows.....	12
3.3.1.1 WinPcap	13
3.3.1.2 Microsoft Visual C++ Compiler for Python 2.7.....	13
3.3.1.3 Pcapy	14
3.2.3 Instalación en Linux	19
3.4. Instalación del resto de las librerías	21
3.5. Configuración de la aplicación.....	23
Interfaz de la aplicación.....	30
4.1 File	30
4.1.1 Start Sniffing	31
4.1.2 Select device.....	32
4.1.3 Insert routing table	33
4.1.4 Edit config file.....	33
4.1.5 Activate system update table.....	33
4.1.6 Clear data base.....	34
4.1.7 Exit.....	35
4.2 Graphics.....	35
4.3 Help	36
Problemas	37
Contacto.....	37

Introducción

1.1. Descripción del sistema

Como se ha expuesto en la documentación este sistema es un apoyo a los IDS tradicionales que se centran en el análisis del contenido de los paquetes, con la finalidad de proporcionar una mayor seguridad al sistema informático que queremos proteger.

1.2. Novedades

Esta es la primera versión de la aplicación.

Requisitos previos

En este epígrafe vamos a desgranar los conocimientos previos y requisitos mínimos del sistema sobre el que vamos a ejecutar el sistema.

2.1. Conocimientos previos

Esta aplicación no requiere de ningún conocimiento previo por parte del usuario.

2.2. Requisitos técnicos

Los requisitos técnicos mínimos de cualquier equipo en el que se quiera ejecutar el sistema, se encuentran definidos en el apartado del presupuesto de la documentación (apartado 4.1.3.1.1).

Dentro del software necesario encontramos todas las librerías que se utilizan en esta aplicación, y las cuales son condiciones sine qua non se puede ejecutar el sistema. Estas también están recogidas en la documentación en el epígrafe 4.2.1.

Instalación y configuración de la aplicación

En esta aplicación tenemos el software necesario para su funcionamiento y las librerías que debemos instalar, ya que sin ellas no podría ejecutarse la aplicación.

Para poder instalar la aplicación se deben seguir los siguientes pasos:

3.1. Instalación de Python

3.1.1 Instalación en Windows

El primer paso es la instalación del intérprete de *python*, ya que este es un lenguaje interpretado y no compilado.

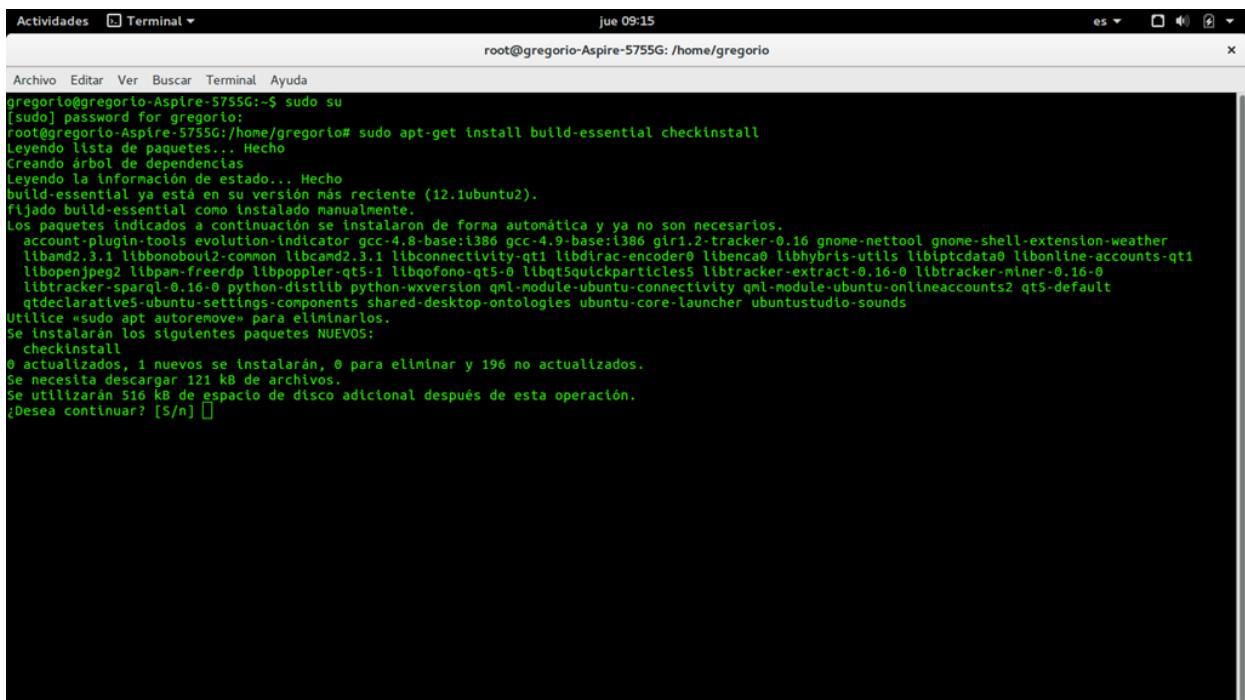
La versión que debemos instalar es la 2.7, aunque dentro de esta versión se recomienda la última disponible. Podemos descargar *python* siguiendo el siguiente enlace: <https://www.python.org/downloads/>

Una vez descargado seguimos los pasos que nos marca el instalador de *python*.

3.1.2 Instalación en Linux

Lo primero es instalar varias librerías dependientes. Remarcar que los sistemas Linux ya llevan instalado *python*, pero siga los pasos para asegurarse la instalación de tener todo lo necesario.

Para instalar estas dependencias abrimos un terminal con permisos de súper usuario e introduzca el siguiente comando: “*sudo apt-get install build-essential checkinstall*”.



The screenshot shows a terminal window titled "Terminal" with the command "root@gregorio-Aspire-5755G: /home/gregorio". The terminal output shows the user running "sudo su" and then executing "sudo apt-get install build-essential checkinstall". The process continues with package dependencies being listed and installed, including various libraries and tools. The terminal ends with a question "¿Desea continuar? [S/n]".

```
jue 09:15
root@gregorio-Aspire-5755G: /home/gregorio
Archivo Editar Ver Buscar Terminal Ayuda
gregorio@gregorio-Aspire-5755G:~$ sudo su
[sudo] password for gregorio:
root@gregorio-Aspire-5755G:/home/gregorio# sudo apt-get install build-essential checkinstall
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
build-essential ya está en su versión más reciente (12.1ubuntu2).
fijado build-essential como instalado manualmente.
Los paquetes indicados a continuación se instalaron de forma automática y ya no son necesarios.
account-plugin-tools evolution-indicator gcc-4.8-base:i386 gcc-4.9-base:i386 gir1.2-tracker-0.16 gnome-nettool gnome-shell-extension-weather
libbamd2.3.1 libbonoboui2-common libcamd2.3.1 libconnectivity-qt1 libdirac-encoder0 libencm4 libhybris-utils libiptcdata0 libonline-accounts-qt1
libopenjpeg2 libpam-freerdp libpoppler-qt5-1 libqofono-qt5-0 libqt5quickparticles5 libtracker-extract-0.16-0 libtracker-miner-0.16-0
libtracker-sparql-0.16-0 python-distlib python-wxversion qml-module-ubuntu-connectivity qml-module-ubuntu-onlineaccounts2 qt5-default
qtdeclarative5-ubuntu-settings-components shared-desktop-ontologies ubuntu-core-launcher ubuntustudio-sounds
Utilice «sudo apt autoremove» para eliminarlos.
Se instalarán los siguientes paquetes NUEVOS:
checkinstall
0 actualizados, 1 nuevos se instalarán, 0 para eliminar y 196 no actualizados.
Se necesita descargar 121 kB de archivos.
Se utilizarán 516 kB de espacio de disco adicional después de esta operación.
¿Desea continuar? [S/n]
```

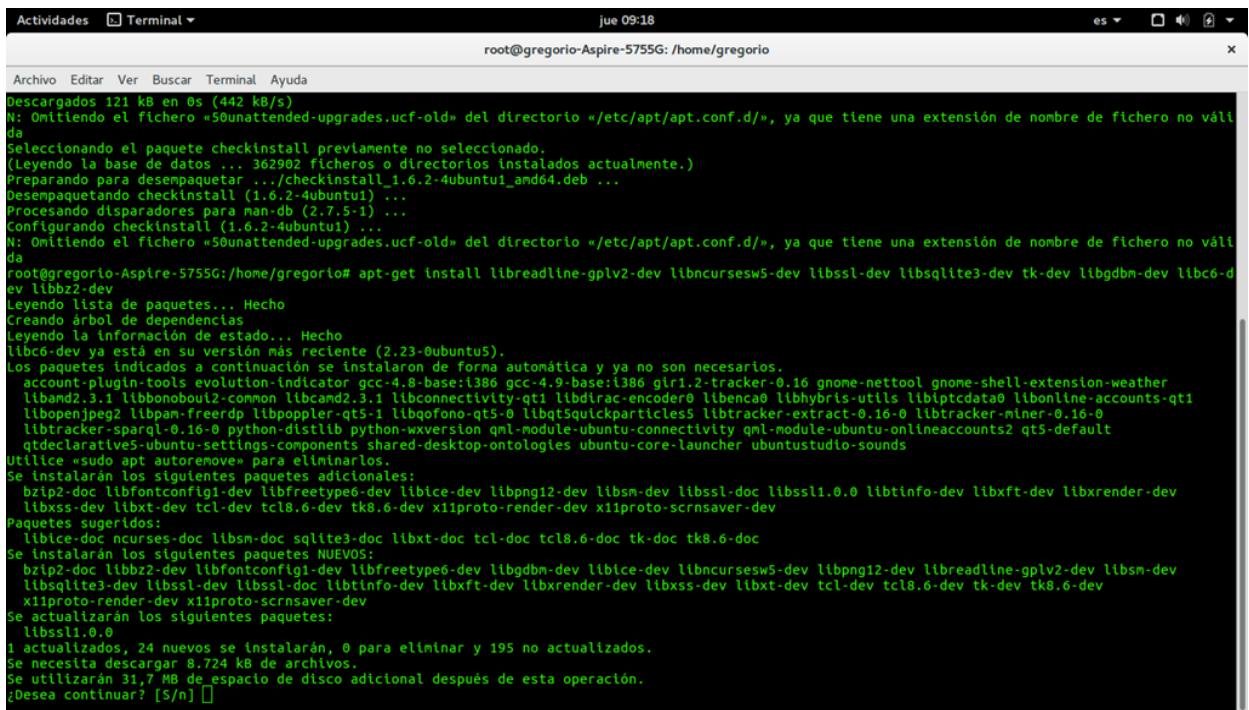
Introducimos “S” para proseguir con la instalación.

```
jue 09:15
root@gregorio-Aspire-5755G: /home/gregorio
Archivo Editar Ver Buscar Terminal Ayuda
gregorio@gregorio-Aspire-5755G:~$ sudo su
[sudo] password for gregorio:
root@gregorio-Aspire-5755G:/home/gregorio# sudo apt-get install build-essential checkinstall
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
build-essential ya está en su versión más reciente (12.ubuntu2).
Fijado build-essential como instalado manualmente.
Los paquetes indicados a continuación se instalaron de forma automática y ya no son necesarios.
 account-plugin-tools evolution-indicator gcc-4.8-base:i386 gcc-4.9-base:i386 gir1.2-tracker-0.16 gnome-nettool gnome-shell-extension-weather libband2.3.1 libbonobout2-common libcamd2.3.1 libconnectivity-qt1 libdirac-encoder0 libenc2a libhybris-utils libiptcdata0 libonlines-accounts-qt1 libjpeg2 libpan-freerdp libpoptler-qt5-1 libqofono-qt5-0 libqt5quickparticles5 libtracker-extract-0.16-0 libtracker-miner-0.16-0 libtracker-sparql-0.16-0 python-distlib python-wxversion qml-module-ubuntu-connectivity qml-module-ubuntu-onlineaccounts2 qt5-default qtdeclarative5-ubuntu-settings-components shared-desktop-ontologies ubuntu-core-launcher ubuntustudio-sounds
Utilice «sudo apt autoremove» para eliminarlos.
Se instalarán los siguientes paquetes NUEVOS:
 checkinstall
0 actualizados, 1 nuevos se instalarán, 0 para eliminar y 196 no actualizados.
Se necesita descargar 121 kB de archivos.
Se utilizarán 516 kB de espacio de disco adicional después de esta operación.
Desea continuar? [S/n] S
Des::1 http://es.archive.ubuntu.com/ubuntu xenial/universe amd64 checkinstall amd64 1.6.2-4ubuntu1 [121 kB]
Descargados 121 kB en 0s (442 kB/s)
N: Omitiendo el fichero «50unattended-upgrades.ucf-old» del directorio «/etc/apt/apt.conf.d/», ya que tiene una extensión de nombre de fichero no válida
seleccionando el paquete checkinstall previamente no seleccionado.
(Leyendo la base de datos ... 362902 ficheros o directorios instalados actualmente.)
Preparando para desempaquetar .../checkinstall_1.6.2-4ubuntu1_amd64.deb ...
Desempaquetando checkinstall (1.6.2-4ubuntu1) ...
Procesando disparadores para man-db (2.7.5-1) ...
[]
```

Una vez que termina, introducimos el siguiente comando: “apt-get install libreadline-gplv2-dev libncursesw5-dev libssl-dev libsqlite3-dev tk-dev libgdbm-dev libc6-dev libbz2-dev” para instalar librerías necesarias.

```
jue 09:18
root@gregorio-Aspire-5755G: /home/gregorio
Archivo Editar Ver Buscar Terminal Ayuda
gregorio@gregorio-Aspire-5755G:~$ sudo su
[sudo] password for gregorio:
root@gregorio-Aspire-5755G:/home/gregorio# sudo apt-get install build-essential checkinstall
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
build-essential ya está en su versión más reciente (12.ubuntu2).
Fijado build-essential como instalado manualmente.
Los paquetes indicados a continuación se instalaron de forma automática y ya no son necesarios.
 account-plugin-tools evolution-indicator gcc-4.8-base:i386 gcc-4.9-base:i386 gir1.2-tracker-0.16 gnome-nettool gnome-shell-extension-weather libband2.3.1 libbonobout2-common libcamd2.3.1 libconnectivity-qt1 libdirac-encoder0 libenc2a libhybris-utils libiptcdata0 libonlines-accounts-qt1 libjpeg2 libpan-freerdp libpoptler-qt5-1 libqofono-qt5-0 libqt5quickparticles5 libtracker-extract-0.16-0 libtracker-miner-0.16-0 libtracker-sparql-0.16-0 python-distlib python-wxversion qml-module-ubuntu-connectivity qml-module-ubuntu-onlineaccounts2 qt5-default qtdeclarative5-ubuntu-settings-components shared-desktop-ontologies ubuntu-core-launcher ubuntustudio-sounds
Utilice «sudo apt autoremove» para eliminarlos.
Se instalarán los siguientes paquetes NUEVOS:
 checkinstall
0 actualizados, 1 nuevos se instalarán, 0 para eliminar y 196 no actualizados.
Se necesita descargar 121 kB de archivos.
Se utilizarán 516 kB de espacio de disco adicional después de esta operación.
Desea continuar? [S/n] S
Des::1 http://es.archive.ubuntu.com/ubuntu xenial/universe amd64 checkinstall amd64 1.6.2-4ubuntu1 [121 kB]
Descargados 121 kB en 0s (442 kB/s)
N: Omitiendo el fichero «50unattended-upgrades.ucf-old» del directorio «/etc/apt/apt.conf.d/», ya que tiene una extensión de nombre de fichero no válida
seleccionando el paquete checkinstall previamente no seleccionado.
(Leyendo la base de datos ... 362902 ficheros o directorios instalados actualmente.)
Preparando para desempaquetar .../checkinstall_1.6.2-4ubuntu1_amd64.deb ...
Desempaquetando checkinstall (1.6.2-4ubuntu1) ...
Procesando disparadores para man-db (2.7.5-1) ...
Configurando checkinstall (1.6.2-4ubuntu1) ...
N: Omitiendo el fichero «50unattended-upgrades.ucf-old» del directorio «/etc/apt/apt.conf.d/», ya que tiene una extensión de nombre de fichero no válida
root@gregorio-Aspire-5755G:/home/gregorio# apt-get install libreadline-gplv2-dev libncursesw5-dev libssl-dev libsqlite3-dev tk-dev libgdbm-dev dev libbz2-dev[]
```

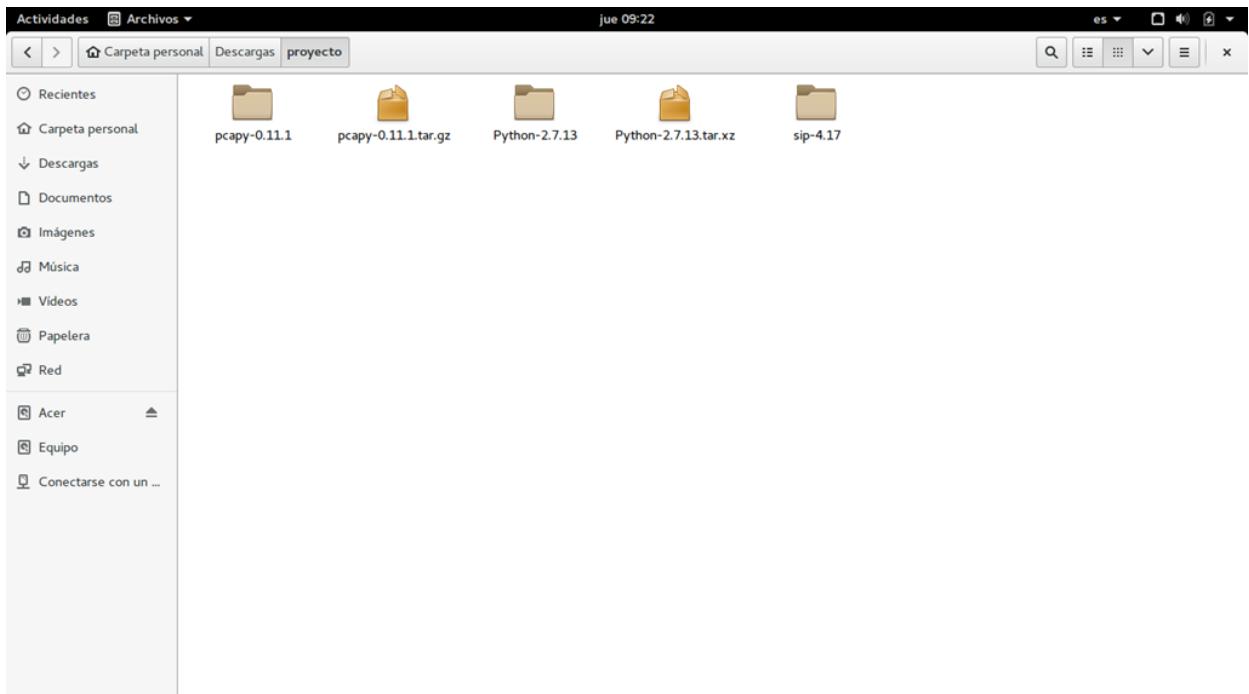
Volvemos a introducir “S” para continuar.



```
jue 09:18
root@gregorio-Aspire-5755G: /home/gregorio

Archivo Editar Ver Buscar Terminal Ayuda
Descargados 121 kB en 0s (442 kB/s)
N: Omitiendo el fichero «5unattended-upgrades.ucf-old» del directorio «/etc/apt/apt.conf.d/», ya que tiene una extensión de nombre de fichero no válida
Seleccionando el paquete checkinstall previamente no seleccionado.
(Leyendo la base de datos ... 362902 ficheros o directorios instalados actualmente.)
Preparando para desempaquetar .../checkinstall_1.6.2-4ubuntu1_amd64.deb ...
Desempaquetando checkinstall (1.6.2-4ubuntu1) ...
Procesando disparadores para man-db (2.7.5-1) ...
Configurando checkinstall (1.6.2-4ubuntu1) ...
N: Omitiendo el fichero «5unattended-upgrades.ucf-old» del directorio «/etc/apt/apt.conf.d/», ya que tiene una extensión de nombre de fichero no válida
root@gregorio-Aspire-5755G: /home/gregorio# apt-get install libreadline-gplv2-dev libncursesw5-dev libssl-dev libsqlite3-dev tk-dev libgdbm-dev libc-dev ev libbz2-dev
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
libc-dev ya está en su versión más reciente (2.23-0ubuntu5).
Los paquetes indicados a continuación se instalaron de forma automática y ya no son necesarios.
account-plugin-tools evolution-indicator gcc-4.8-base:i386 gir1.2-tracker-0.16 gnome-nettool gnome-shell-extension-weather libband2.3.1 libbonobout2-common libcamd2.3.1 libconnectivity-qt1 libdirac-encoder0 libencad libhybris-utils libiptcdtad0 libonline-accounts-qt1 libjpeg libpam-freerdp libpoppler-qt5-1 libqofono-qt5-0 libqt5quickparticles5 libtracker-extract-0.16-0 libtracker-miner-0.16-0 libtracker-spargl-0.16-0 python-distlib python-wxversion qml-module-ubuntu-connectivity qml-module-ubuntu-onlineaccounts2 qt5-default qtdeclarative5-ubuntu-settings-components shared-desktop-ontologies ubuntu-core-launcher ubuntustudio-sounds
Utilice «sudo apt autoremove» para eliminarlos.
Se instalarán los siguientes paquetes adicionales:
 bzip2-doc libfontconfig1-dev libfreetype6-dev libice-dev libpng12-dev libsm-dev libssl1.0.0 libtinfo-dev libxft-dev libxrender-dev libxss-dev libxt-dev tcl-dev tcl8.6-dev tk8.6-dev xproto-render-dev xproto-scnsaver-dev
Paquetes sugeridos:
 libice-doc ncurses-doc libsm-doc sqlite3-doc libxt-doc tcl-doc tcl8.6-doc tk-doc tk8.6-doc
Se instalarán los siguientes paquetes NUEVOS:
 bzip2-doc libbz2-dev libfontconfig1-dev libfreetype6-dev libgdbm-dev libice-dev libncursesw5-dev libpng12-dev libreadline-gplv2-dev libsm-dev libsqlite3-dev libssl-dev libtinfo-dev libxft-dev libxrender-dev libxss-dev libxt-dev tcl-dev tcl8.6-dev tk-dev tk8.6-dev xproto-render-dev xproto-scnsaver-dev
Se actualizarán los siguientes paquetes:
 libssl1.0.0
1 actualizados, 24 nuevos se instalarán, 0 para eliminar y 195 no actualizados.
Se necesita descargar 8.724 kB de archivos.
Se utilizarán 31,7 MB de espacio de disco adicional después de esta operación.
¿Desea continuar? [S/n] 
```

Una vez que termine, nos descargamos y descomprimimos *python* del siguiente enlace: <https://www.python.org/downloads/>, importante descargarse la versión 2.7.



Nos colocamos en la ruta de la carpeta *python* descomprimida e introducimos el siguiente comando: “./configure”.

```

Actividades Terminal jue 09:26
root@gregorio-Aspire-5755G:/home/gregorio/Descargas/proyecto/Python-2.7.13# ./configure
checking build system type... x86_64-pc-linux-gnu
checking host system type... x86_64-pc-linux-gnu
checking for --enable-universalsdk... no
checking for --with-universal-archs... 32-bit
checking MACHDEP... linux2
checking EXTRAPLATDIR...
checking for --without-gcc... no
checking for --with-icc... no
checking for gcc... gcc
checking whether the C compiler works... yes
checking for C compiler default output file name... a.out
checking for suffix of executables...
checking whether we are cross compiling... no
checking for suffix of object files... o
checking whether we are using the GNU C compiler... yes
checking whether gcc accepts -g... yes
checking for gcc option to accept ISO C89... none needed
checking for --with-cxx-main=<compiler>... no
checking for g++... no
configure: WARNING:
  By default, distutils will build C++ extension modules with "g++".
  If this is not intended, then set CXX on the configure command line.

checking how to run the C preprocessor... gcc -E
checking for grep that handles long lines and -e... /bin/grep
checking for egrep... /bin/grep -E
checking for ANSI C header files... [REDACTED]

```

Una vez ha terminado escribimos: “make”.

```

Actividades Terminal jue 09:27
root@gregorio-Aspire-5755G:/home/gregorio/Descargas/proyecto/Python-2.7.13# make
gcc -pthread -c -fno-strict-aliasing -g -O2 -DNDEBUG -g -fwrapv -O3 -Wall -Wstrict-prototypes -I. -I./Include -DPy_BUILD_CORE -o Modules/python.o ./Modules/python.c
gcc -pthread -c -fno-strict-aliasing -g -O2 -DNDEBUG -g -fwrapv -O3 -Wall -Wstrict-prototypes -I. -I./Include -DPy_BUILD_CORE -o Parser/acceler.o Parser/acceler.c
gcc -pthread -c -fno-strict-aliasing -g -O2 -DNDEBUG -g -fwrapv -O3 -Wall -Wstrict-prototypes -I. -I./Include -DPy_BUILD_CORE -o Parser/grammar.o Parser/grammar.c
gcc -pthread -c -fno-strict-aliasing -g -O2 -DNDEBUG -g -fwrapv -O3 -Wall -Wstrict-prototypes -I. -I./Include -DPy_BUILD_CORE -o Parser/listnode.o Parser/listnode.c
gcc -pthread -c -fno-strict-aliasing -g -O2 -DNDEBUG -g -fwrapv -O3 -Wall -Wstrict-prototypes -I. -I./Include -DPy_BUILD_CORE -o Parser/node.o Parser/node.c
gcc -pthread -c -fno-strict-aliasing -g -O2 -DNDEBUG -g -fwrapv -O3 -Wall -Wstrict-prototypes -I. -I./Include -DPy_BUILD_CORE -o Parser/parser.o Parser/parser.c
[REDACTED]

```

Una vez terminada ya tenemos instalado *python*.

3.2. Instalación de PyQt4

3.2.1 Instalación en Windows

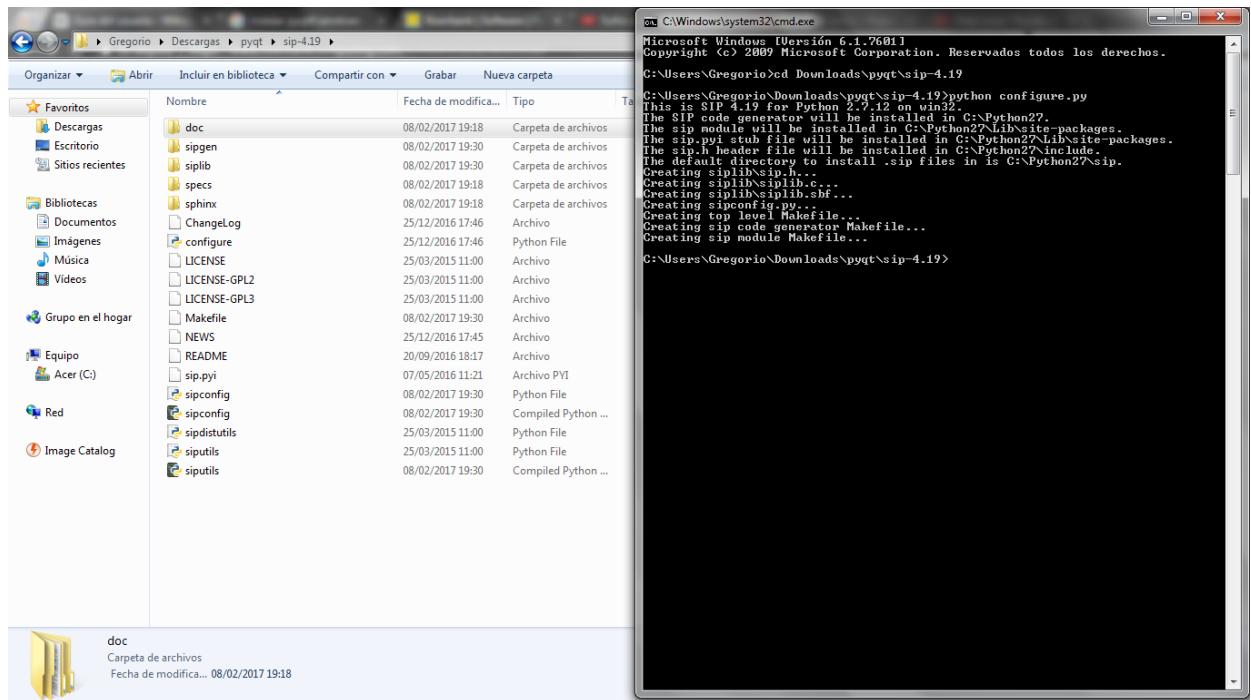
Para que la interfaz de la aplicación funcione debemos tener esta librería instalada en nuestro sistema operativo.

Antes de instalar la librería debemos instalar la librería *sip* la cual la podemos encontrar en el siguiente enlace:

<https://www.riverbankcomputing.com/software/sip/download>.

Una vez descargado y descomprimido el archivo abrimos un terminal de Windows, y nos colocamos en la ruta donde se encuentra el fichero, en mi caso sería: “C:\Users\Gregorio\Downloads\pyqt\sip-4.19”.

Una vez hecho introducimos la siguiente instrucción en nuestra terminal: “python configure.py”, también podemos indicarle con que versión de *python* debe hacer esto: “C:\python35\python configure.py”.

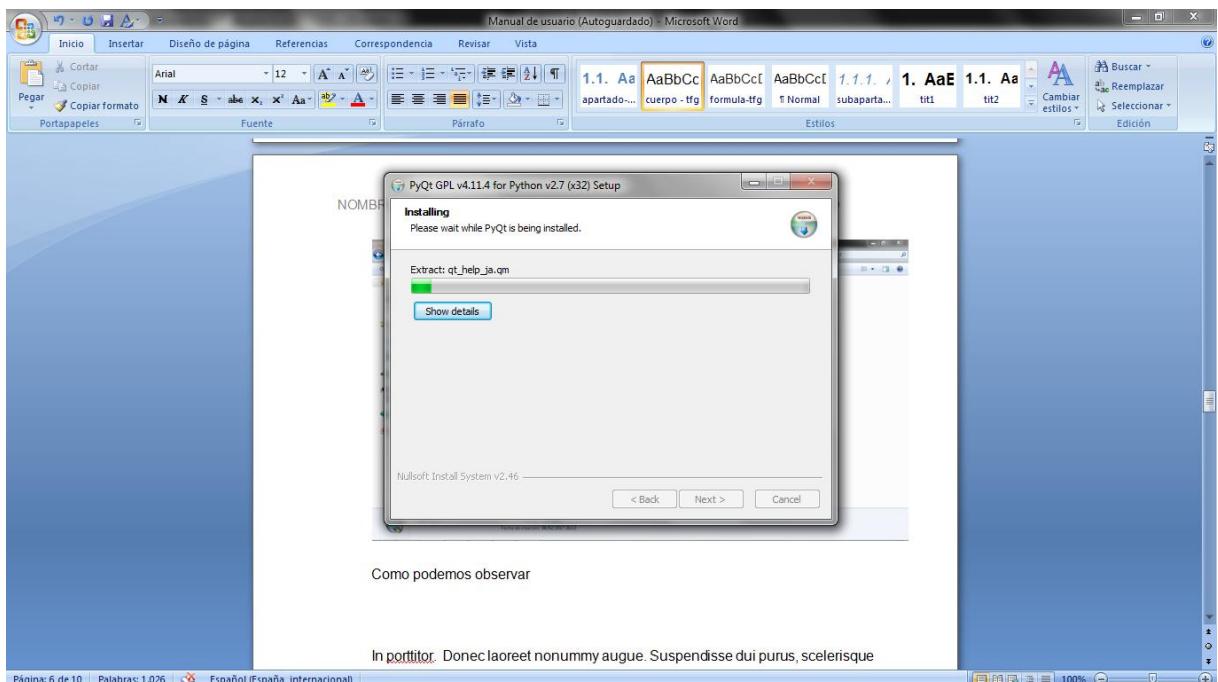


Ahora introducimos el siguiente comando: “make”, debemos esperar a que termine (en mi caso ya está instalado). Una vez que acabe dicho proceso debemos ejecutar “make install”.

También podemos seguir la guía de instalación la cual se encuentra en la carpeta “doc/installation”.

Cuando acabe este proceso, tendremos instalado el SIP, que es un componente necesario para la instalación de PyQt4. Para instalar PyQt4 lo descargaremos del siguiente enlace:

<https://sourceforge.net/projects/pyqt/files/PyQt4/PyQt-4.11.4/>.



Únicamente debemos seguir los pasos de la instalación.

3.2.2 Instalación en Linux

Para instalar PyQt4 debemos abrimos una terminal con permisos de super usuario (si venimos del paso anterior nos serviría la misma terminal). Una vez hecho esto introducimos el siguiente comando: “apt-cache search qt4”.

```
jue 09:40
root@gregorio-Aspire-5755G:/home/gregorio/Descargas/proyecto/Python-2.7.13# apt-cache search qt4
fcitx - Flexible Input Method Framework
fcitx-bin - Flexible Input Method Framework - essential binaries
fcitx-data - Flexible Input Method Framework - essential data files
fcitx-dbg - Flexible Input Method Framework - debugging symbols
fcitx-frontend-all - Flexible Input Method Framework - frontends metapackage
fcitx-frontend-gtk2 - Flexible Input Method Framework - GTK+ 2 IM Module frontend
fcitx-frontend-gtk3 - Flexible Input Method Framework - GTK+ 3 IM Module frontend
fcitx-frontend-qt4 - Flexible Input Method Framework - Qt4 IM Module frontend
fcitx-libs-dev - Flexible Input Method Framework - library development files
fcitx-module-dbus - Flexible Input Method Framework - D-Bus module and IPC frontend
fcitx-module-kimpanel - Flexible Input Method Framework - KimPanel protocol module
fcitx-module-lua - Flexible Input Method Framework - Lua module
fcitx-module-x11 - Flexible Input Method Framework - X11 module and XIM frontend
fcitx-modules - Flexible Input Method Framework - core modules
fcitx-nozc - Mozc engine for fcitx - Client of the Mozc input method
fcitx-pinyin - Flexible Input Method Framework - classic Pinyin engine
fcitx-table - Flexible Input Method Framework - table engine
fcitx-ui-classic - Flexible Input Method Framework - Classic user interface
gir1.2-fcitx-1.0 - Flexible Input Method Framework - GObject Introspection
libavahi-q4-1 - Avahi Qt 4 Integration library
libavahi-q4-dev - Development headers for the Avahi Qt 4 integration library
libdee-q4-3 - Qt4 binding for Dee - shared library
libdee-q4-dev - Qt binding for Dee - development files
libfcitx-config4 - Flexible Input Method Framework - configuration support library
libfcitx-core0 - Flexible Input Method Framework - library of core funtions
libfcitx-gclient0 - Flexible Input Method Framework - D-Bus client library for Glib
libfcitx-qt0 - Flexible Input Method Framework - Meta package for Qt library
libfcitx-utils0 - Flexible Input Method Framework - utility support library
libpoppler-qt4-4 - PDF rendering library (Qt 4 based shared library)
libpoppler-qt4-dev - PDF rendering library -- development files (Qt 4 interface)
libqt4-dbg - Qt 4 library debugging symbols
libqt4-dbus - Qt 4 D-Bus module
libqt4-declarative - Qt 4 Declarative module
libqt4-declarative-gestures - Qt 4 gestures QML plugin
libqt4-declarative-particles - Qt 4 particles QML plugin
libqt4-designer - Qt 4 designer module
libqt4-designer-dbg - Qt 4 designer library debugging symbols
libqt4-dev - Qt 4 development files
```

Cuando haya acabado introducimos: “apt-get install python-qt4”.

```
jue 09:43
root@gregorio-Aspire-5755G:/home/gregorio/Descargas/proyecto/Python-2.7.13#
root@gregorio-Aspire-5755G:/home/gregorio/Descargas/proyecto/Python-2.7.13# apt-get install python-qt4
[scribus-data - Open Source Desktop Page Layout - stable branch (data files)
scribus-dev - Open Source Desktop Page Layout - stable branch (development files)
smplayer - complete front-end for MPlayer, MPlayer2 and MPV
smplayer-l10n - complete front-end for MPlayer and MPlayer2 - translation files
source-highlight-ide - Qt4 IDE for GNU Source-highlight
spyder-common - Python IDE for scientists (common files)
strigi-client - Qt4 client for Strigi Desktop Search
strigi-daemon - fast indexing and searching tool for your personal data (daemon)
sutil-dbg - Debugging symbols for sutil
tora - graphical toolkit for database developers and administrators
tora-dbg - graphical toolkit for database developers and administrators - debugging symbols
ulatency - scriptable latency regulator using cgroups (client)
vtk-examples - C++, Tcl and Python example programs/scripts for VTK
wally - Qt4 wallpaper changer
x2goclient - X2Go Client application (Qt4)
x2goclient-dbg - X2Go Client application (Qt4), debug symbols (client)
x2goplugin - X2Go Client application (Qt4) as browser plugin
x2goplugin-dbg - X2Go Client application (Qt4), debug symbols (plugin)
xca - X509 Certification Authority management tool based on QT4
xcwcp - Morse code tutor - graphical user interface
yate-qt4 - YATE-based universal telephony client
N: Omitiendo el fichero «5unattended-upgrades.ucf-old» del directorio «/etc/apt/apt.conf.d/», ya que tiene una extensión de nombre de fichero no válida
da
root@gregorio-Aspire-5755G:/home/gregorio/Descargas/proyecto/Python-2.7.13# sudo apt-get install python-qt4
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
python-qt4 ya está en su versión más reciente (4.11.4+dfsg-1build4).
Los paquetes indicados a continuación se instalaron de forma automática y ya no son necesarios.
account-plugin-tools evolution-indicator gcc-4.8-base:i386 gcc-4.9-base:i386 gir1.2-tracker-0.16 gnome-nettool gnome-shell-extension-weather
libband2.3.1 libbonoboui2-common libcmnd2.3.1 libconnectivity-qt4 libdirac-encoder0 libenca libhybris-utils libiptcdatab libonlines-accounts-qt1
libopenjpeg2 libpam-freerdp libpoppler-qt5-1 libqffono-qt5-0 libqt5quickparticles5 libtracker-extract-0.16-0 libtracker-miner-0.16-0
libtracker-sparql-0.16-0 python-distlib python-wxversion qml-module-ubuntu-connectivity qml-module-ubuntu-onlineaccounts2 qt5-default
qtdeclarative5-ubuntu-settings-components shared-desktop-ontologies ubuntu-core-launcher ubuntustudio-sounds
Utilice «sudo apt autoremove» para eliminarlos.
0 actualizados, 0 nuevos se instalarán, 0 para eliminar y 195 no actualizados.
N: Omitiendo el fichero «5unattended-upgrades.ucf-old» del directorio «/etc/apt/apt.conf.d/», ya que tiene una extensión de nombre de fichero no válida
da
root@gregorio-Aspire-5755G:/home/gregorio/Descargas/proyecto/Python-2.7.13# ]
```

En nuestro caso no instalo nada debido a que ya lo está. Por último introducimos el comando: “apt-get install libqt4-designer”

```

Actividades Terminal jue 09:45
root@gregorio-Aspire-5755G: /home/gregorio/Descargas/proyecto/Python-2.7.13
Archivo Editar Ver Buscar Terminal Ayuda
x2goclient-dbg - X2Go Client application (Qt4), debug symbols (client)
x2goplugin - X2Go Client (Qt4) as browser plugin
x2goplugin-dbg - X2Go Client application (Qt4), debug symbols (plugin)
xca - x509 Certification Authority management tool based on Qt4
xcwcp - Morse code tutor - graphical user interface
yate-qt4 - YATE-based universal telephony client
N: Omitiendo el fichero «50unattended-upgrades.ucf-old» del directorio «/etc/apt/apt.conf.d/», ya que tiene una extensión de nombre de fichero no válida
root@gregorio-Aspire-5755G: /home/gregorio/Descargas/proyecto/Python-2.7.13# sudo apt-get install python-qt4
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
python-qt4 ya está en su versión más reciente (4.11.4+dfsg-1build4).
Los paquetes indicados a continuación se instalaron de forma automática y ya no son necesarios.
  account-plugin-tools evolution-indicator gcc-4.8-base:i386 gcc-4.9-base:i386 gir1.2-tracker-0.16 gnome-nettool gnome-shell-extension-weather
  libband2.3.1 libbonoboui2-common libcamd2.3.1 libconnectivity-qt1 libdirac-encoder0 libenc0 libhybris-utils libiptcdat0 libonline-accounts-qt1
  libopenjpeg2 libpam-freerdp libpoppler-qt5-1 libqofono-qt5-0 libqt5quickparticles5 libtracker-extract-0.16-0 libtracker-miner-0.16-0
  libtracker-sparql-0.16-0 python-distilb python-wxversion qml-module-ubuntu-connectivity qml-module-ubuntu-onlineaccounts2 qt5-default
  qtdeclarative5-ubuntu-settings-components shared-desktop-ontologies ubuntu-core-launcher ubuntustudio-sounds
Utilice «sudo apt autoremove» para eliminarlos.
0 actualizados, 0 nuevos se instalarán, 0 para eliminar y 195 no actualizados.
N: Omitiendo el fichero «50unattended-upgrades.ucf-old» del directorio «/etc/apt/apt.conf.d/», ya que tiene una extensión de nombre de fichero no válida
root@gregorio-Aspire-5755G: /home/gregorio/Descargas/proyecto/Python-2.7.13# apt-get install libqt4-designer
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
libqt4-designer ya está en su versión más reciente (4:4.8.7+dfsg-5ubuntu2).
Los paquetes indicados a continuación se instalaron de forma automática y ya no son necesarios.
  account-plugin-tools evolution-indicator gcc-4.8-base:i386 gcc-4.9-base:i386 gir1.2-tracker-0.16 gnome-nettool gnome-shell-extension-weather
  libband2.3.1 libbonoboui2-common libcamd2.3.1 libconnectivity-qt1 libdirac-encoder0 libenc0 libhybris-utils libiptcdat0 libonline-accounts-qt1
  libopenjpeg2 libpam-freerdp libpoppler-qt5-1 libqofono-qt5-0 libqt5quickparticles5 libtracker-extract-0.16-0 libtracker-miner-0.16-0
  libtracker-sparql-0.16-0 python-distilb python-wxversion qml-module-ubuntu-connectivity qml-module-ubuntu-onlineaccounts2 qt5-default
  qtdeclarative5-ubuntu-settings-components shared-desktop-ontologies ubuntu-core-launcher ubuntustudio-sounds
Utilice «sudo apt autoremove» para eliminarlos.
0 actualizados, 0 nuevos se instalarán, 0 para eliminar y 195 no actualizados.
N: Omitiendo el fichero «50unattended-upgrades.ucf-old» del directorio «/etc/apt/apt.conf.d/», ya que tiene una extensión de nombre de fichero no válida
root@gregorio-Aspire-5755G: /home/gregorio/Descargas/proyecto/Python-2.7.13# 
```

Nosotros no instalamos nada porque ya tenemos instalado el QtDesigner, ahora para comprobar que tenemos instalado PyQt4 introducimos el comando: “python”, el cual transforma nuestra terminal en un intérprete de *python*, e introducimos: “import PyQt4” y pulsamos la tecla enter.

```

Actividades Terminal jue 09:48
root@gregorio-Aspire-5755G: /home/gregorio/Descargas/proyecto/Python-2.7.13
Archivo Editar Ver Buscar Terminal Ayuda
root@gregorio-Aspire-5755G: /home/gregorio/Descargas/proyecto/Python-2.7.13# python
Python 2.7.12 (default, Nov 19 2016, 06:48:10)
[GCC 5.4.0 20160609] on linux2
Type "help", "copyright", "credits" or "license" for more information.
>>> import PyQt4
>>> 
```

Como apreciamos no ha dado ningún error, y por tanto ya tenemos instalado PyQt4.

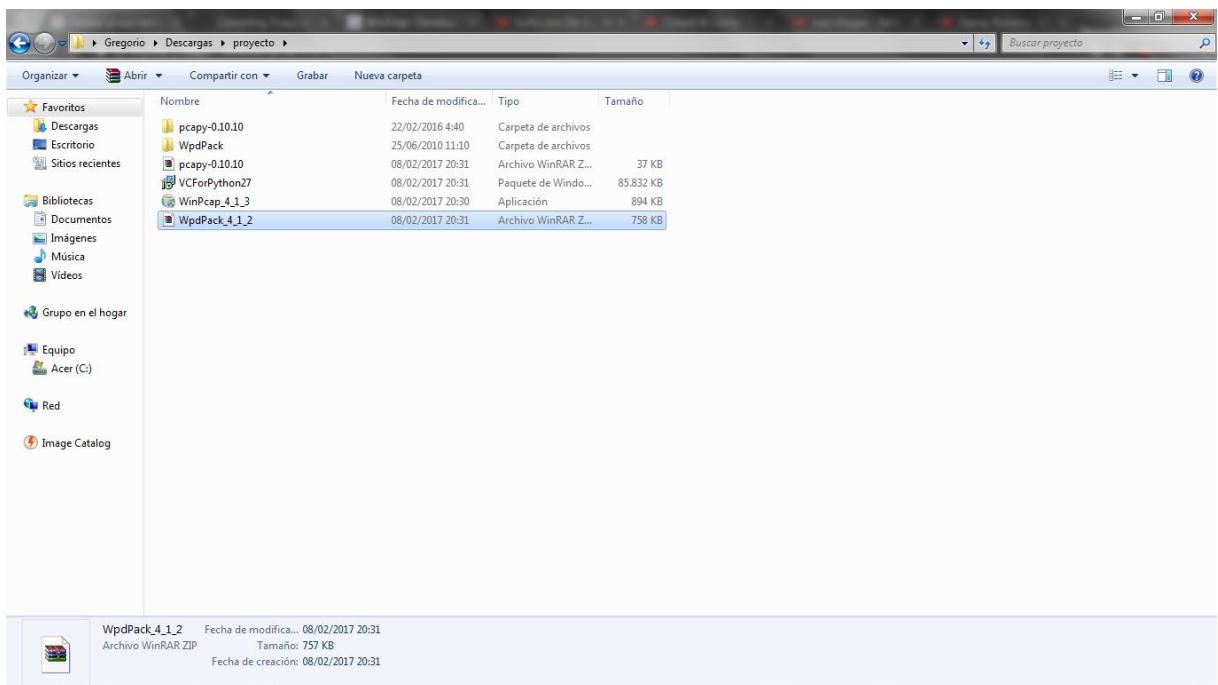
3.3. Instalación de Pcap

3.3.1 Instalación en Windows

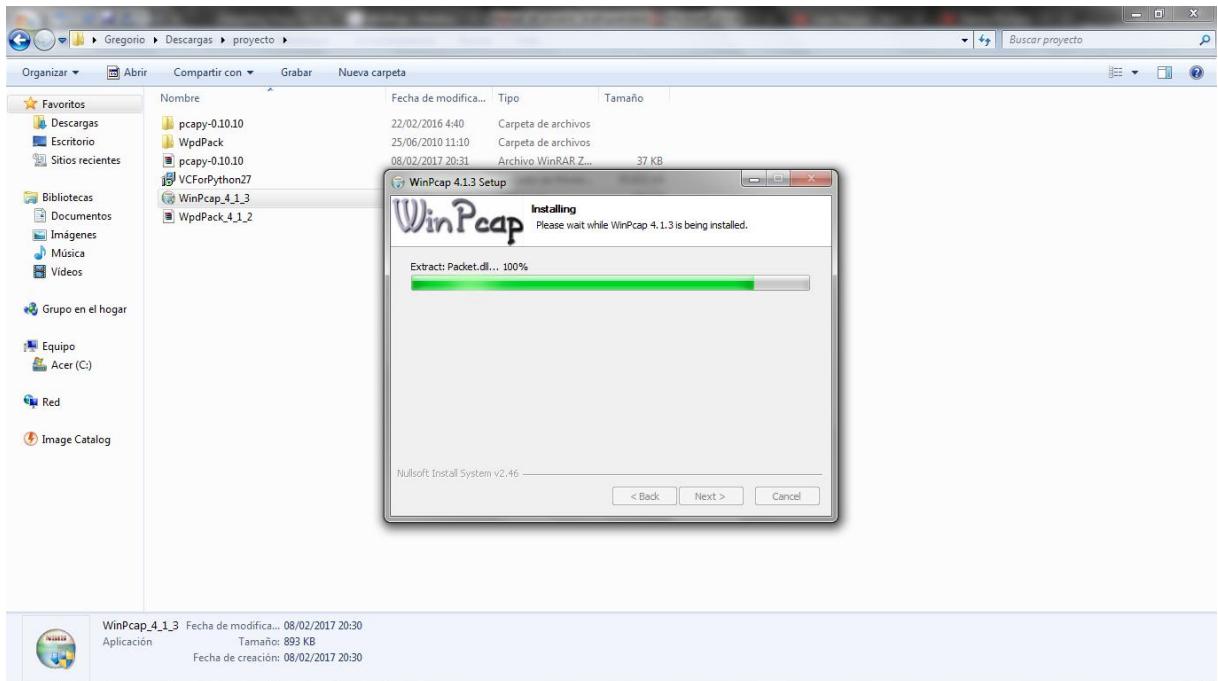
Para instalar esta librería debemos descargar los siguientes elementos (estos elementos son para Windows, en Linux se realizaría otra instalación, aunque si debemos instalar la librería Pcapy):

- WinPcap: <https://www.winpcap.org/install/default.htm>.
- Microsoft Visual C++ Compiler for Python 2.7: <http://aka.ms/vcpython27>.
- Pcapy: <https://github.com/CoreSecurity/pypcap/releases>.
- WinPcap Developer's Pack: <https://www.winpcap.org/devel.htm>.

Una vez descargados estos elementos procedemos a su instalación.



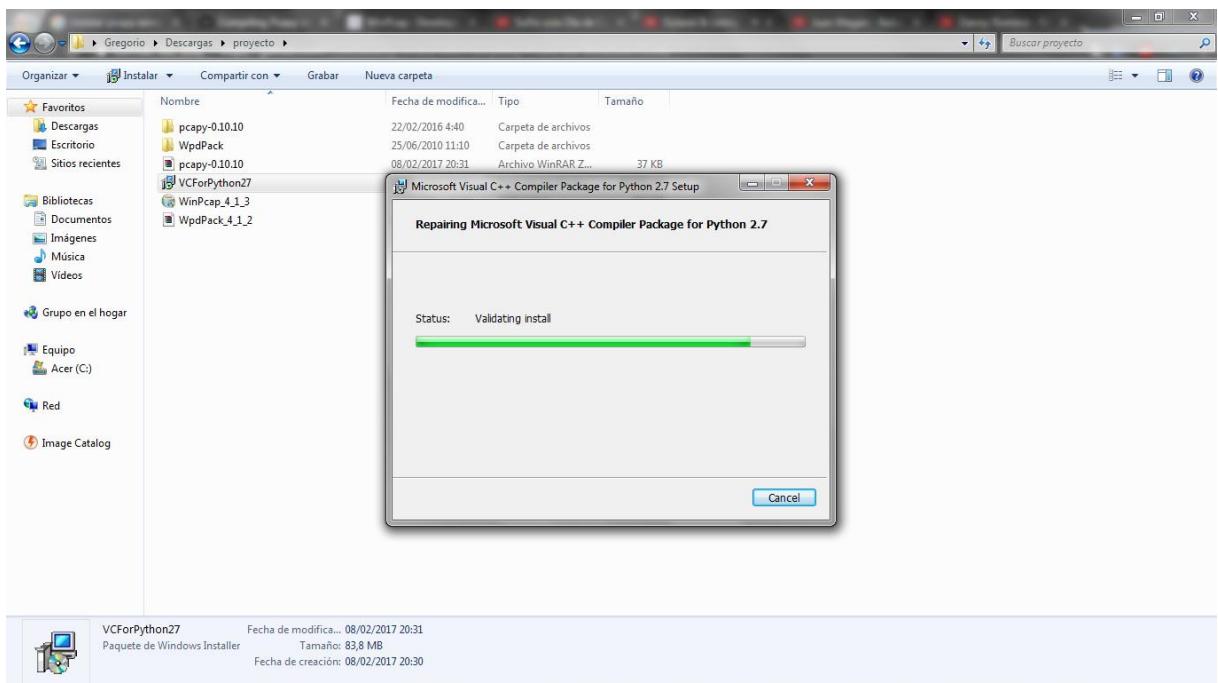
3.3.1.1 WinPcap



Iniciamos el ejecutable y seguimos los pasos, cuando finalice pasamos al siguiente.

3.3.1.2 Microsoft Visual C++ Compiler for Python 2.7

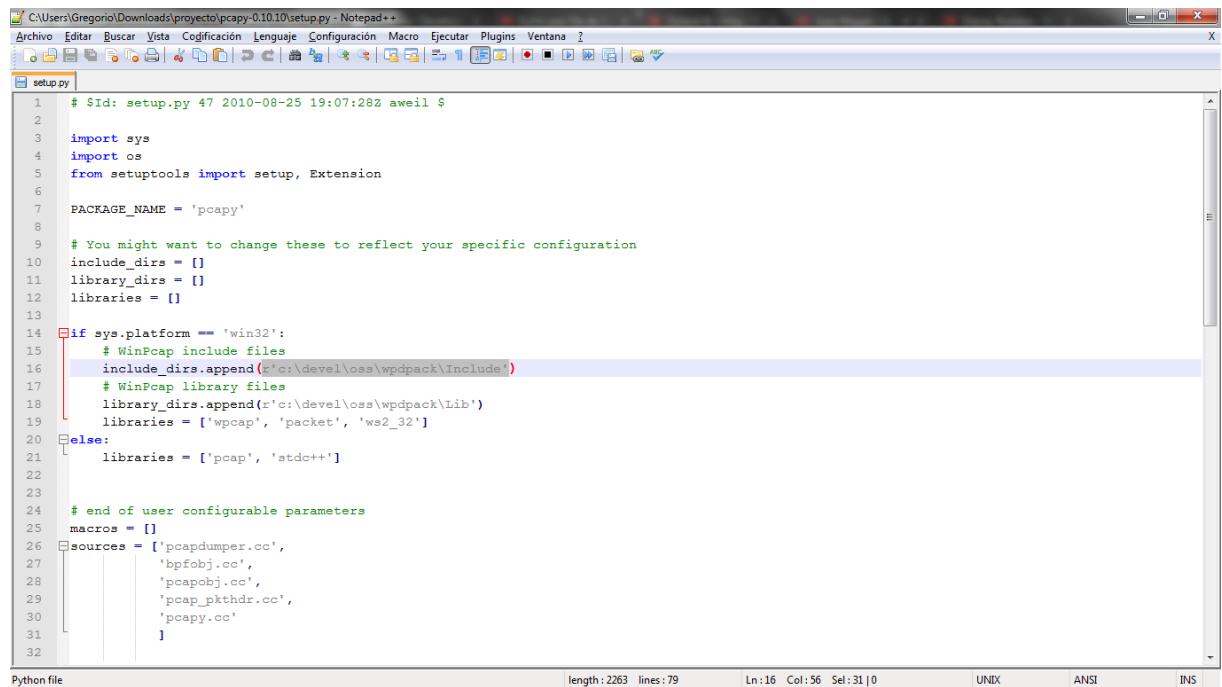
En mi caso ya lo tenía instalado y me ha pedido que lo repare.



Una vez finalizado pasamos al siguiente elemento.

3.3.1.3 Pcapy

En la instalación debemos realizar unos cambios en el fichero setup.py



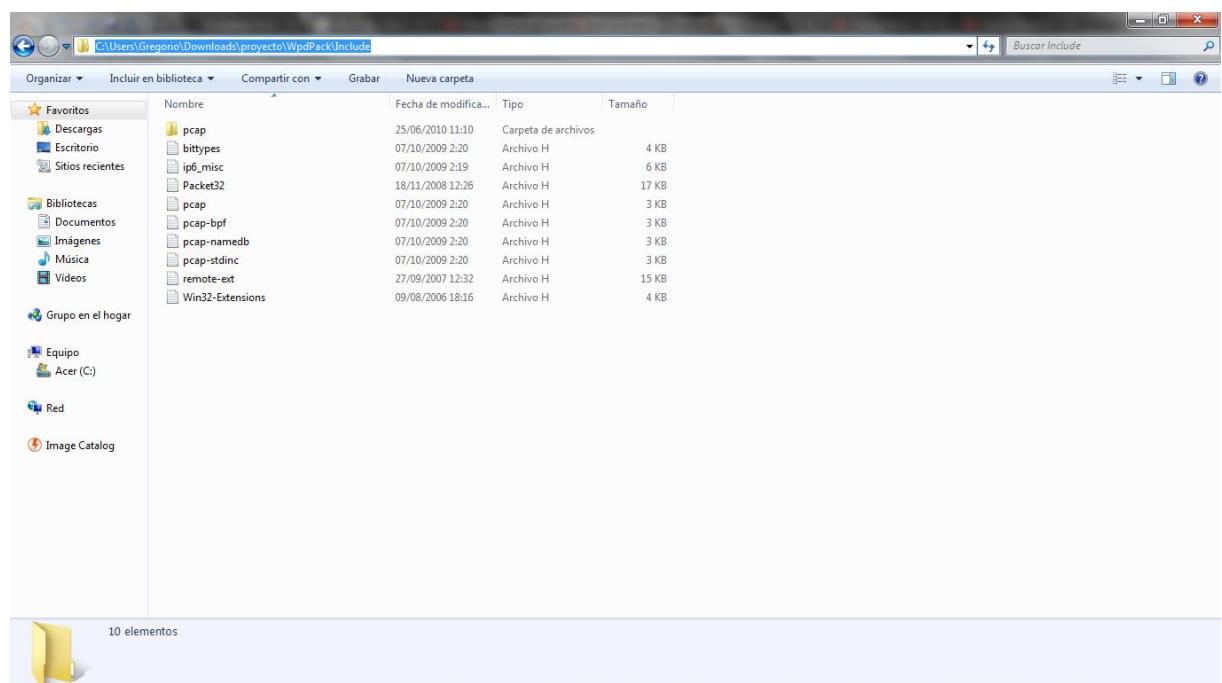
```

C:\Users\Gregorio\Downloads\proyecto\pcapy-0.10.10\setup.py - Notepad++
Archivo Editar Buscar Vista Codificación Lenguaje Configuración Macro Ejecutar Plugins Ventana ?
setup.py
1 # $Id: setup.py 47 2010-08-25 19:07:28Z awei $
2
3 import sys
4 import os
5 from setuptools import setup, Extension
6
7 PACKAGE_NAME = 'pcapy'
8
9 # You might want to change these to reflect your specific configuration
10 include_dirs = []
11 library_dirs = []
12 libraries = []
13
14 if sys.platform == 'win32':
15     # WinPcap include files
16     include_dirs.append('c:\devel\oss\wpdpack\Include')
17     # WinPcap library files
18     library_dirs.append('c:\devel\oss\wpdpack\Lib')
19     libraries = ['wpcap', 'packet', 'ws2_32']
20 else:
21     libraries = ['wpcap', 'stdc++']
22
23
24 # end of user configurable parameters
25 macros = []
26 sources = ['pcapdumper.cc',
27             'bpffobj.cc',
28             'pcapobj.cc',
29             'pcap_pkthdr.cc',
30             'pcapy.cc'
31         ]
32

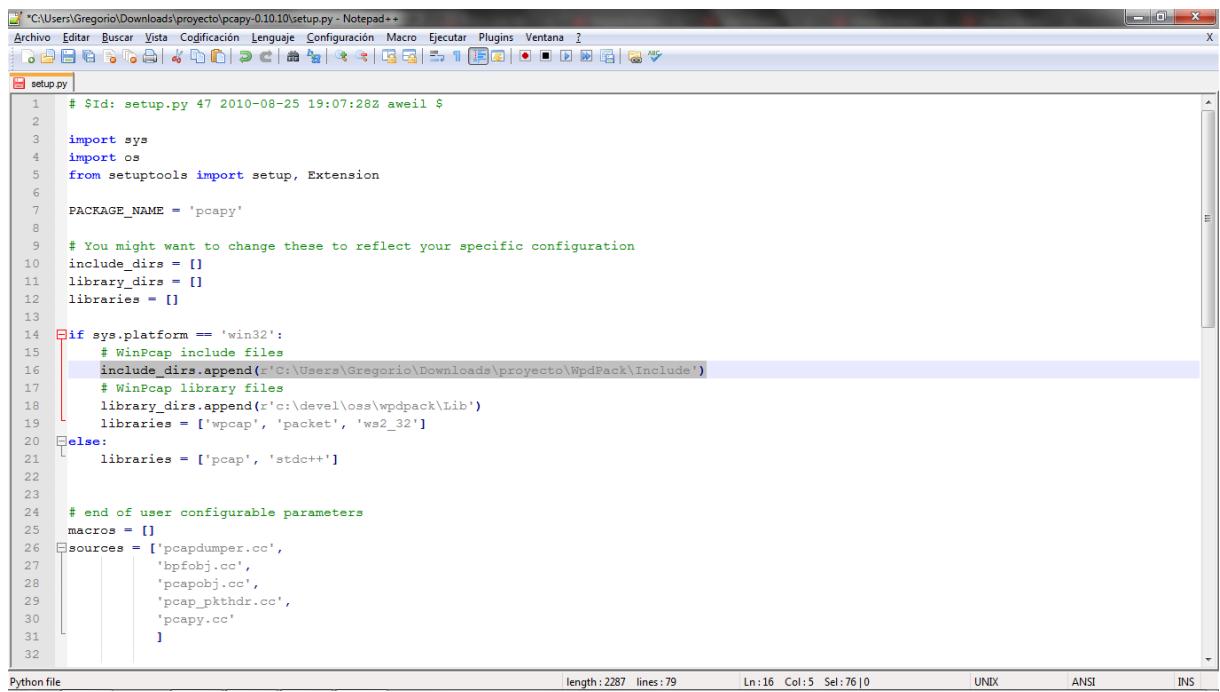
```

Python file length: 2263 lines: 79 Ln:16 Col:56 Sel:31|0 UNIX ANSI INS

La parte seleccionada la cambiamos por la ruta a la carpeta *include* del elemento descargado llamado *WpdPack*.



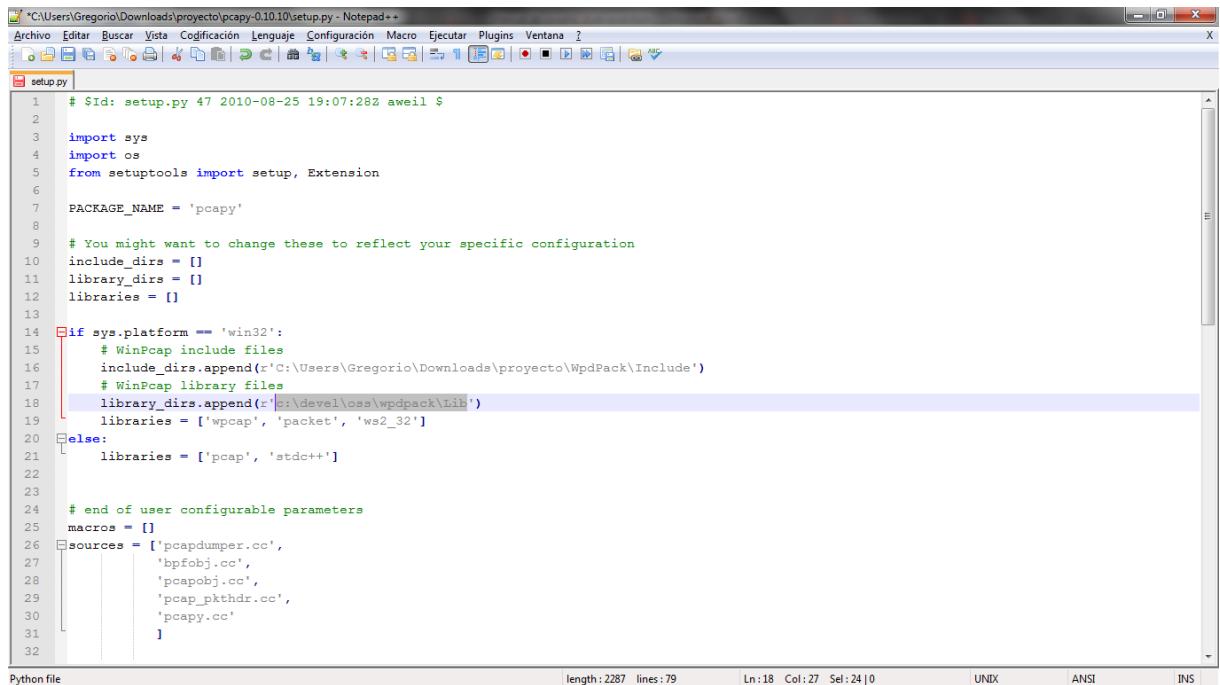
En vuestro ordenador saldrá una ruta diferente, pero esta es la que debemos insertar en ese elemento, de tal forma que queda:



```
*C:\Users\Gregorio\Downloads\proyecto\pcapy-0.10.10\setup.py - Notepad++
Archivo Editar Buscar Vista Codificación Lenguaje Configuración Macro Ejecutar Plugins Ventana ?
setup.py
1 # $Id: setup.py 47 2010-08-25 19:07:28Z aweiil $
2
3 import sys
4 import os
5 from setuptools import setup, Extension
6
7 PACKAGE_NAME = 'pcapy'
8
9 # You might want to change these to reflect your specific configuration
10 include_dirs = []
11 library_dirs = []
12 libraries = []
13
14 if sys.platform == 'win32':
15     # WinPcap include files
16     include_dirs.append(r'C:\Users\Gregorio\Downloads\proyecto\WpdPack\Include')
17     # WinPcap library files
18     library_dirs.append(r'C:\devel\oss\wpdpack\Lib')
19     libraries = ['wpcap', 'packet', 'ws2_32']
20 else:
21     libraries = ['pcap', 'stdc++']
22
23
24 # end of user configurable parameters
25 macros = []
26 sources = [
27     'bpfobj.cc',
28     'pcapobj.cc',
29     'pcap_pkthdr.cc',
30     'pcapy.cc'
31 ]
32

Python file length: 2287 lines: 79 Ln:16 Col:5 Sel:76|0 UNIX ANSI INS
```

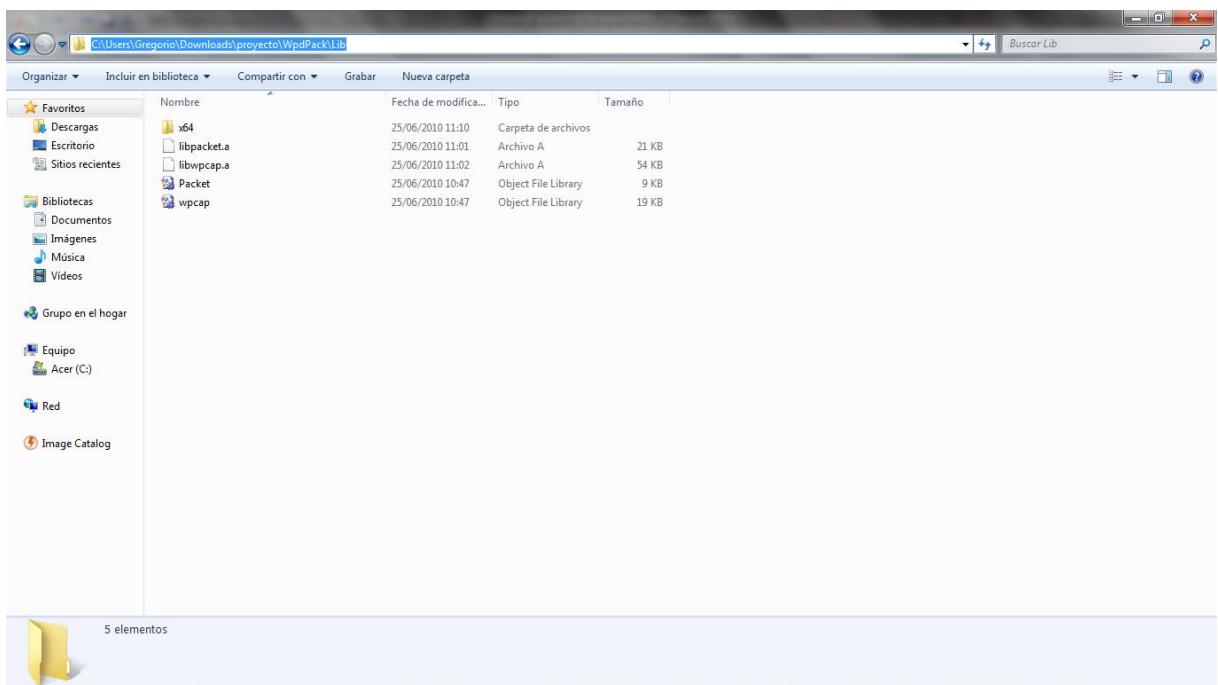
Ahora procedemos a cambiar la siguiente ruta:



```
*C:\Users\Gregorio\Downloads\proyecto\pcapy-0.10.10\setup.py - Notepad++
Archivo Editar Buscar Vista Codificación Lenguaje Configuración Macro Ejecutar Plugins Ventana ?
setup.py
1 # $Id: setup.py 47 2010-08-25 19:07:28Z aweiil $
2
3 import sys
4 import os
5 from setuptools import setup, Extension
6
7 PACKAGE_NAME = 'pcapy'
8
9 # You might want to change these to reflect your specific configuration
10 include_dirs = []
11 library_dirs = []
12 libraries = []
13
14 if sys.platform == 'win32':
15     # WinPcap include files
16     include_dirs.append(r'C:\Users\Gregorio\Downloads\proyecto\WpdPack\Include')
17     # WinPcap library files
18     library_dirs.append(r'C:\devel\oss\wpdpack\Lib')
19     libraries = ['wpcap', 'packet', 'ws2_32']
20 else:
21     libraries = ['pcap', 'stdc++']
22
23
24 # end of user configurable parameters
25 macros = []
26 sources = [
27     'bpfobj.cc',
28     'pcapobj.cc',
29     'pcap_pkthdr.cc',
30     'pcapy.cc'
31 ]
32

Python file length: 2287 lines: 79 Ln:18 Col:27 Sel:24|0 UNIX ANSI INS
```

La cual hace referencia a la carpeta *Lib*, como puede observarse.



Y lo sustituimos en el fichero. Tenemos que tener en cuenta que si descargamos la versión x64 de *python* la ruta debe ser a la carpeta x64, es decir, sería: C:\Users\Gregorio\Downloads\proyecto\WpdPack\Lib\x64.

En nuestro caso hemos descargado la versión x86 de *python*.

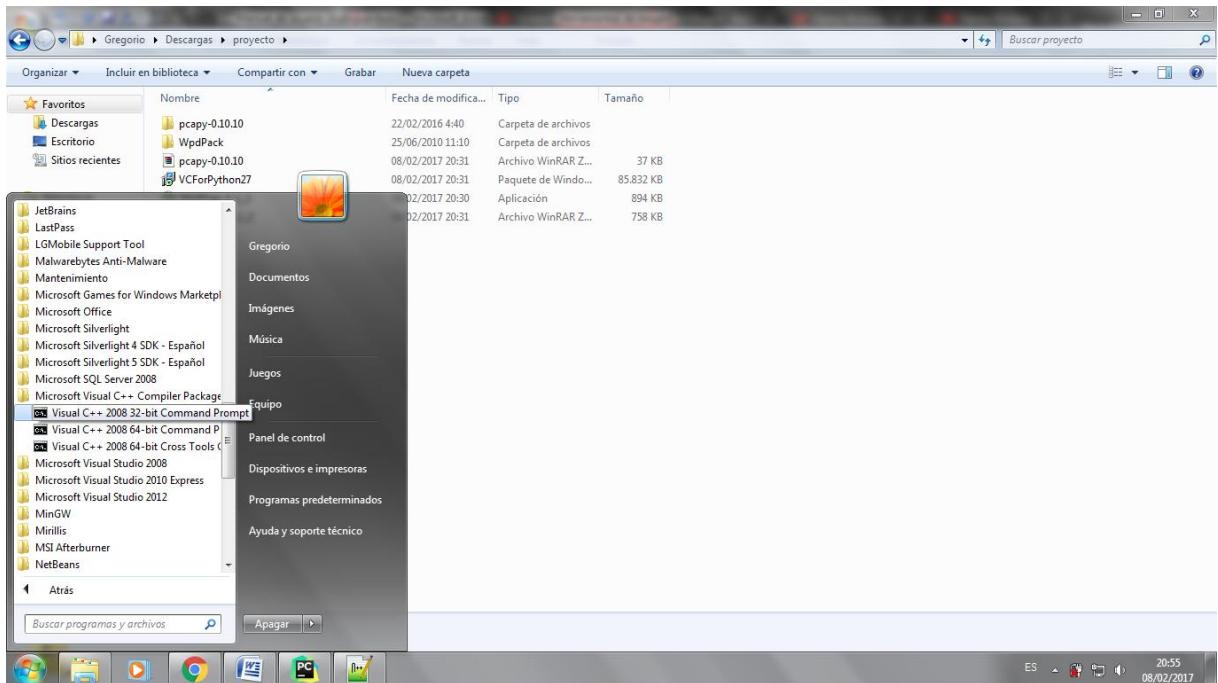
```

1 # $Id: setup.py 47 2010-08-25 19:07:28Z aweiil $
2
3 import sys
4 import os
5 from setuptools import setup, Extension
6
7 PACKAGE_NAME = 'pcapy'
8
9 # You might want to change these to reflect your specific configuration
10 include_dirs = []
11 library_dirs = []
12 libraries = []
13
14 if sys.platform == 'win32':
15     # WinCap include files
16     include_dirs.append(r'C:\Users\Gregorio\Downloads\proyecto\WpdPack\Include')
17     # WinCap library files
18     library_dirs.append(r'C:\Users\Gregorio\Downloads\proyecto\WpdPack\Lib')
19     libraries = ['wpcap', 'packet', 'ws2_32']
20 else:
21     libraries = ['pcap', 'stdc++']
22
23
24 # end of user configurable parameters
25 macros = []
26 sources = ['pcapdumper.cc',
27             'bpfobj.cc',
28             'pcapobj.cc',
29             'pcap_pkthdr.cc',
30             'pcapy.cc'
31         ]
32

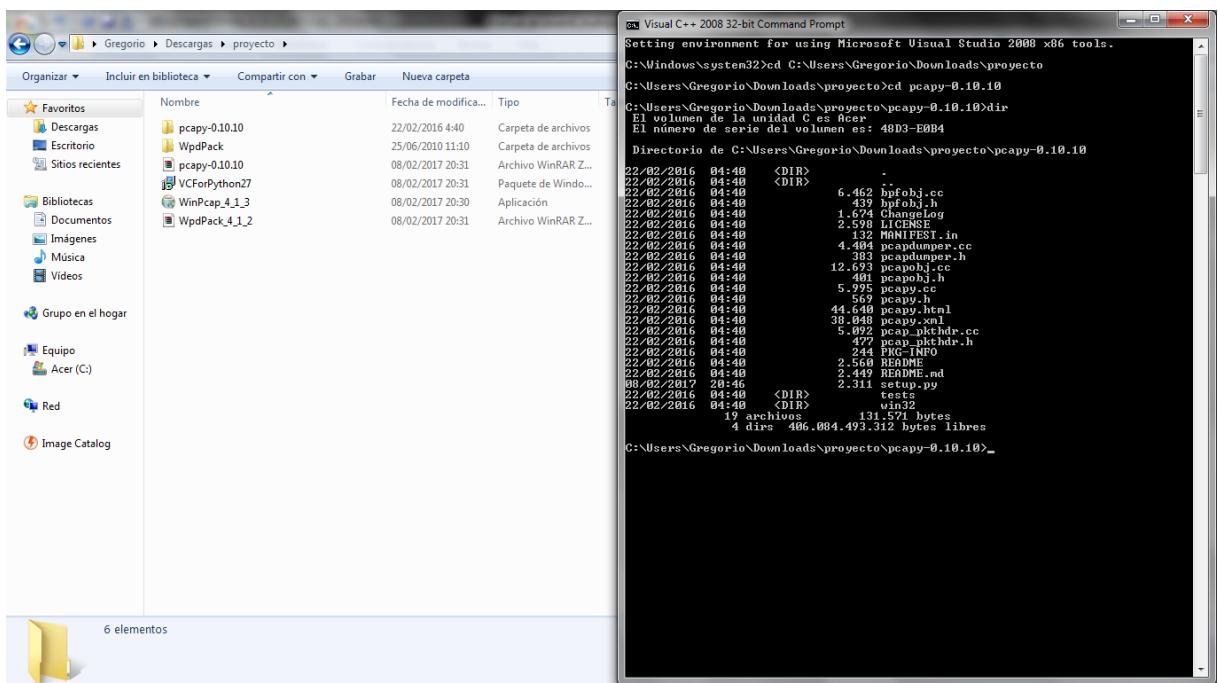
```

Una vez realizados estos cambios guardamos el fichero.

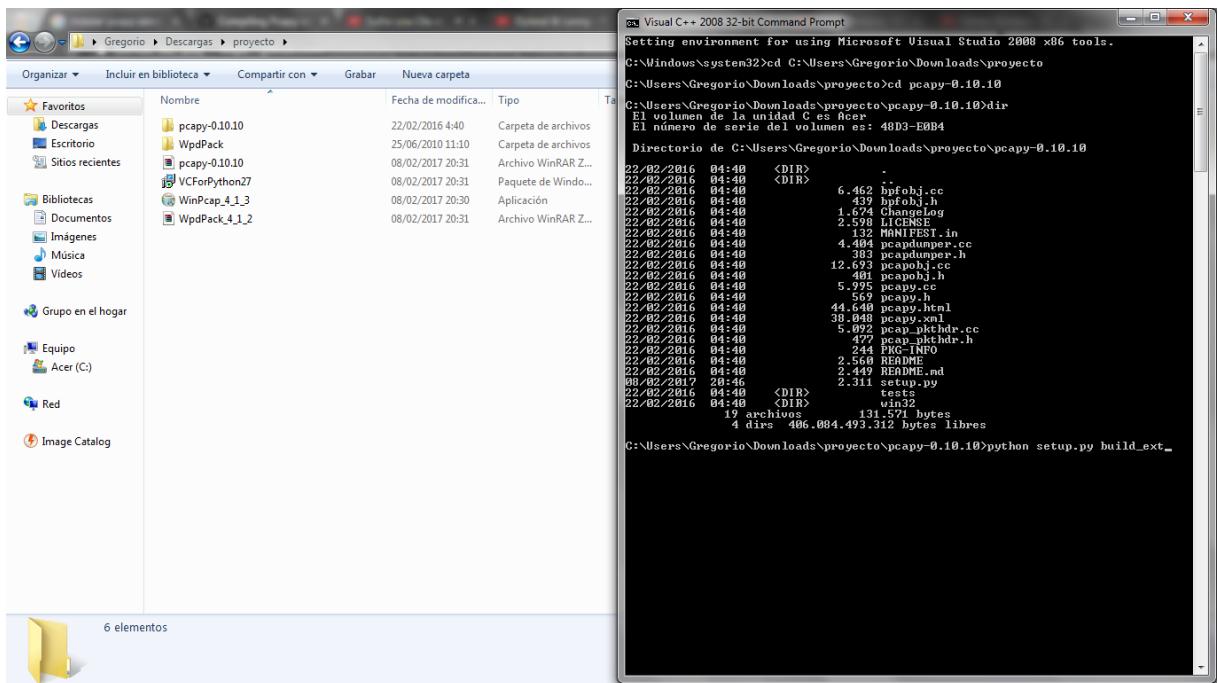
En la segunda parte de la instalación debemos abrir el terminal Microsoft Visual C++ Compiler for Python 2.7 Command Prompt (dependiendo de las versiones instaladas).



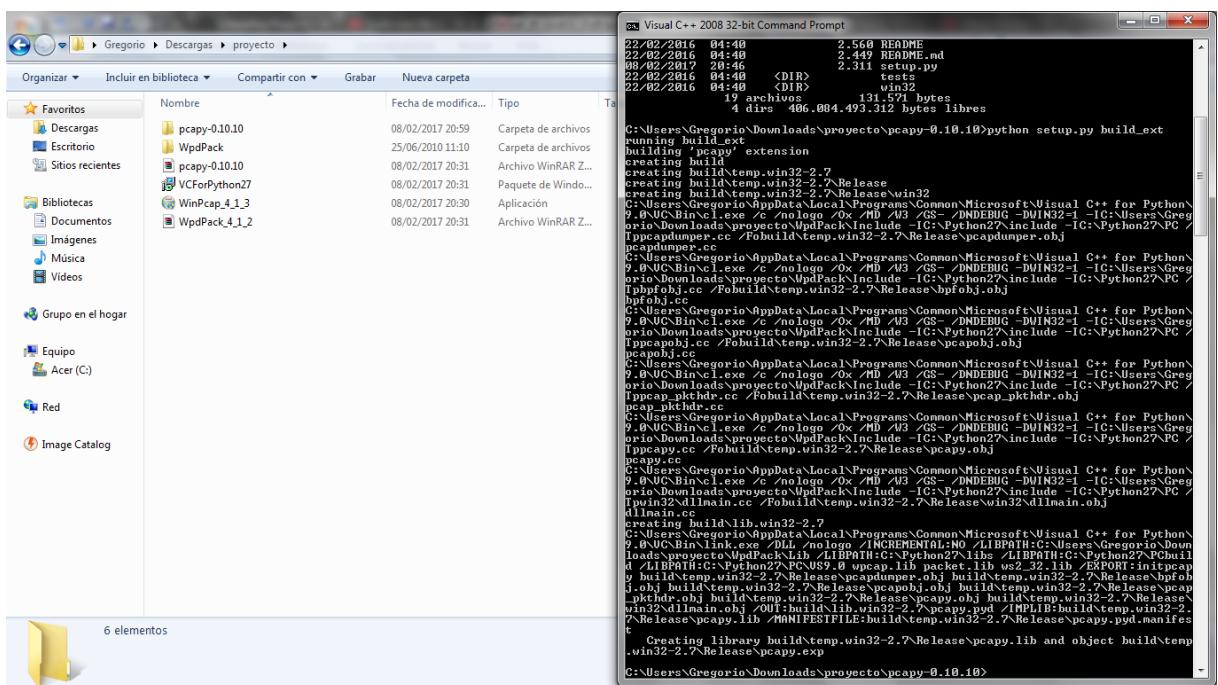
Nos posicionamos en la ruta de la carpeta Pcap.



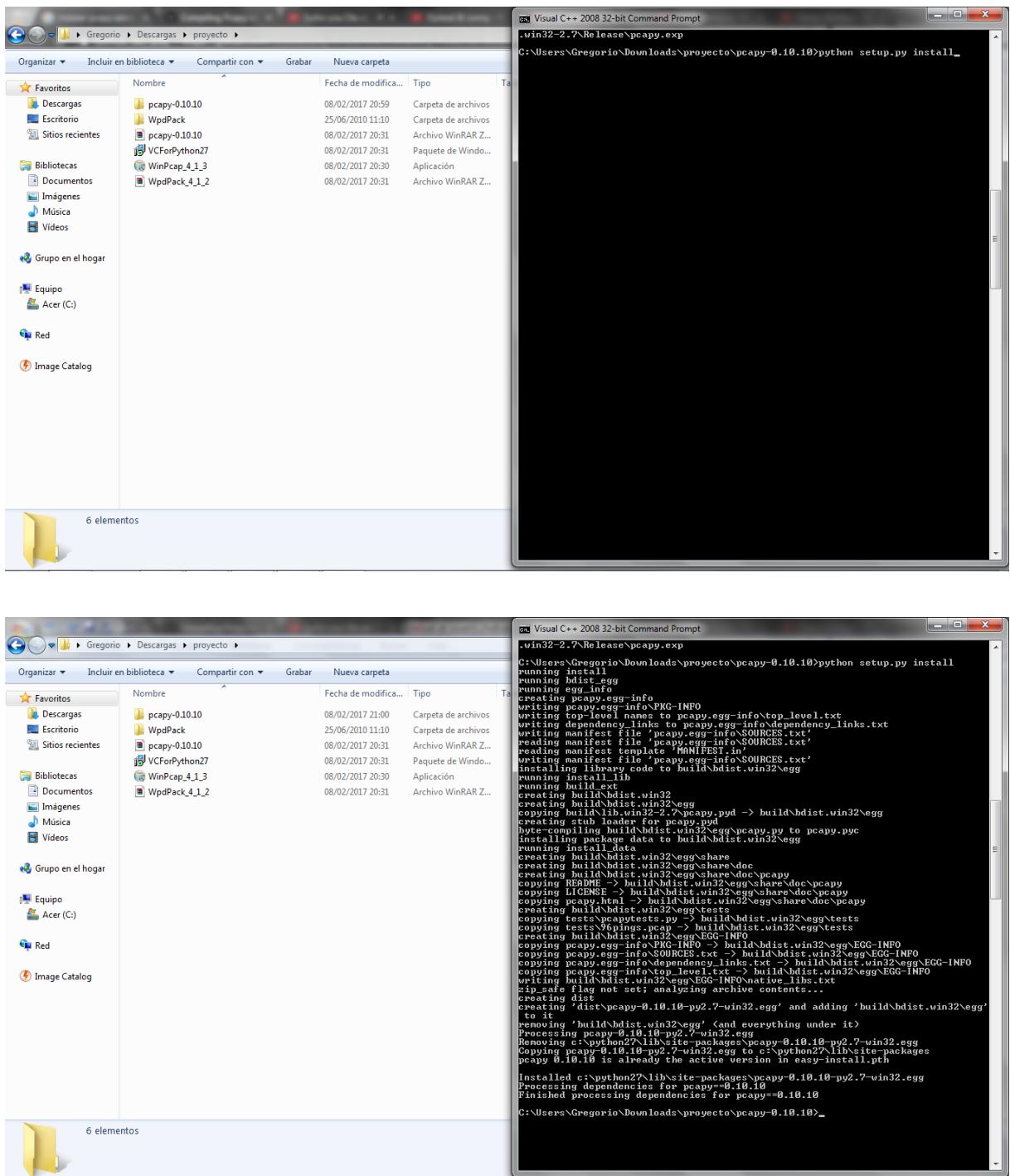
Introducimos el siguiente comando: "python setup.py build_ext"



Esperamos a que termine.



Una vez ha acabado introducimos el siguiente comando: "python setup.py install"



Cuando haya terminado tenemos instalada la librería Pcap.

3.2.3 Instalación en Linux

Primero instalamos las actualizaciones del sistema, por si hubiera alguna no instalada, con el comando: "apt-get update". Una vez que acabe este comando introducimos: "apt-get install python-pcap".

```
root@gregorio-Aspire-5755G:/home/gregorio/Descargas/proyecto/Python-2.7.13# apt-get install python-pcap
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
python-pcap ya está en su versión más reciente (0.10.8-1build1).
[
```

Cuando termine ya estará instalado Pcap. Para comprobar que está, podemos introducir el comando: “python” y, al igual que en la librería anterior, cuando se abra el interprete de *python* introducir: “import pcap”.

```
root@gregorio-Aspire-5755G:/home/gregorio/Descargas/proyecto/Python-2.7.13# apt-get install python-pcap
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
python-pcap ya está en su versión más reciente (0.10.8-1build1).
Los paquetes indicados a continuación se instalaron de forma automática y ya no son necesarios.
account-plugin-tools evolution-indicator gcc-4.8-base:i386 gcc-4.9-base:i386 gir1.2-tracker-0.16 gnome-nettool gnome-shell-extension-weather
libbam2.3.1 libbonoboui2-common libcamd2.3.1 libconnectivity-qt4 libdirac-encoder0 libenc2a libhybris-utils libiptcdata0 libonlines-accounts-qt4
libopenjpeg2 libpan-freerdp libpoppler-qt5-1 libqofono-qt5-0 libqt5quickparticles5 libtracker-extract-0.16-0 libtracker-miner-0.16-0
libtracker-sparql-0.16-0 linux-headers-4.4.0-51 linux-headers-4.4.0-51-generic linux-image-4.4.0-51-generic linux-image-extra-4.4.0-51-generic
python-distro python-wxversion qml-module-ubuntu-connectivity qml-module-ubuntu-onlineaccounts2 qt5-default
qtdeclarative5-ubuntu-settings-components shared-desktop-ontologies ubuntu-studio-sounds
Utilice «sudo apt autoremove» para eliminarlos.
0 actualizados, 0 nuevos se instalarán, 0 para eliminar y 35 no actualizados.
N: Omitiendo el fichero «5ounattended-upgrades.ucf-old» del directorio «/etc/apt/apt.conf.d/», ya que tiene una extensión de nombre de fichero no válida
root@gregorio-Aspire-5755G:/home/gregorio/Descargas/proyecto/Python-2.7.13# python
Python 2.7.12 (default, Nov 19 2016, 06:48:10)
[GCC 5.4.0 20160609] on linux2
Type "help", "copyright", "credits" or "license" for more information.
>>> import pcap
>>> [
```

Como vemos no se ha producido ningún error, en consecuencia se instaló correctamente.

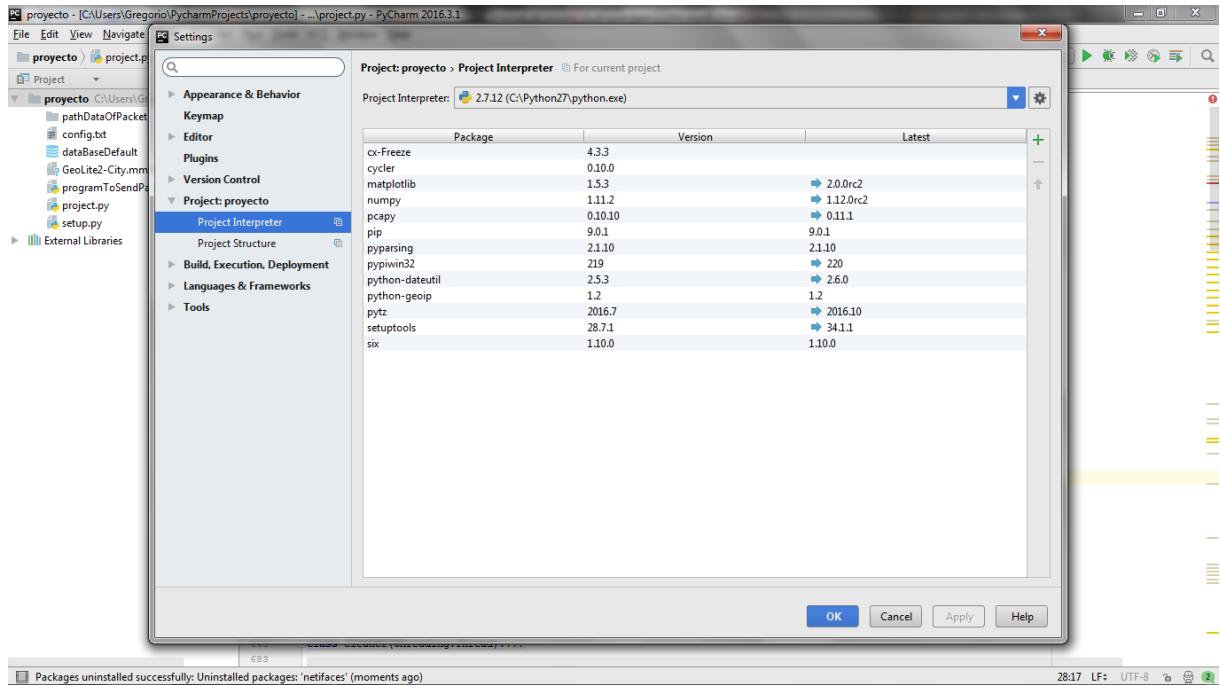
3.4. Instalación del resto de las librerías

Ahora debemos instalar el resto de librerías, ya que las más importantes están instaladas. Las librerías que vamos a instalar son:

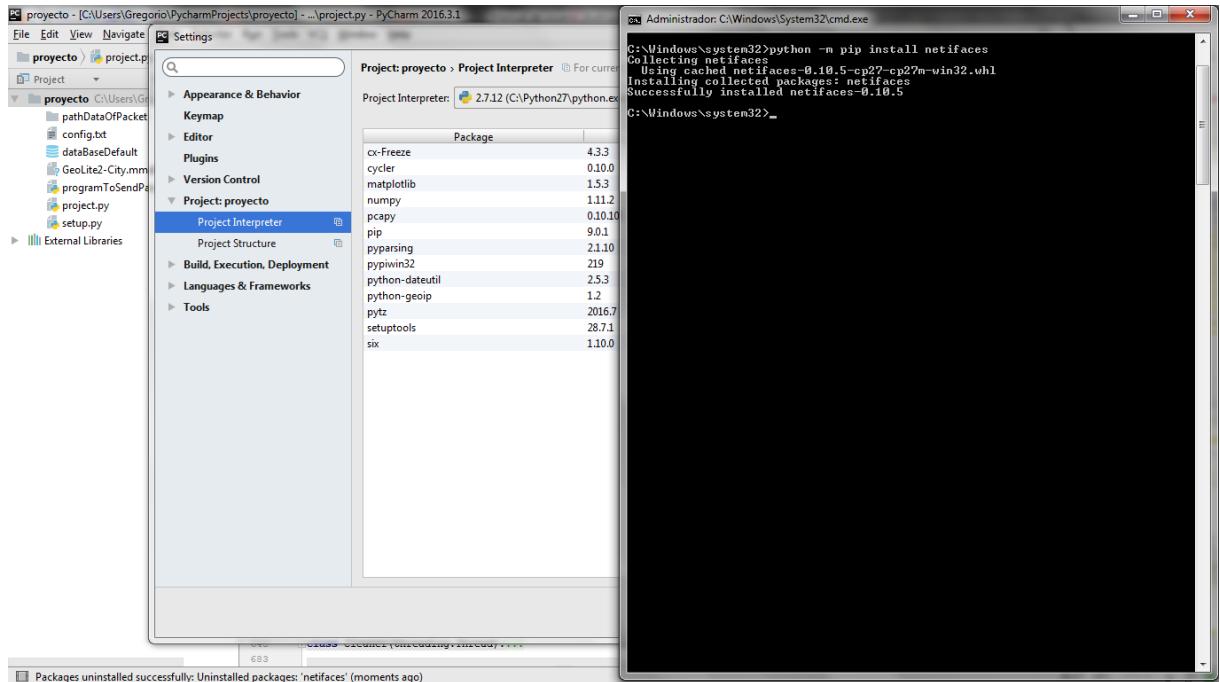
- Matplotlib
- Netifaces
- Numpy
- Python-geoip
- Setuptools
- Pyparsing
- Pyiwin32
- Python-dateutil
- Pytz
- Six

La instalación tanto en Windows como en Linux de todas las librerías es de la misma forma, para instalar abrimos una terminal con permisos de administrador, y tecleamos la instrucción pip de *python*, la cual sirve para la instalación de paquetes. Vamos a poner un ejemplo con una librería de las citadas.

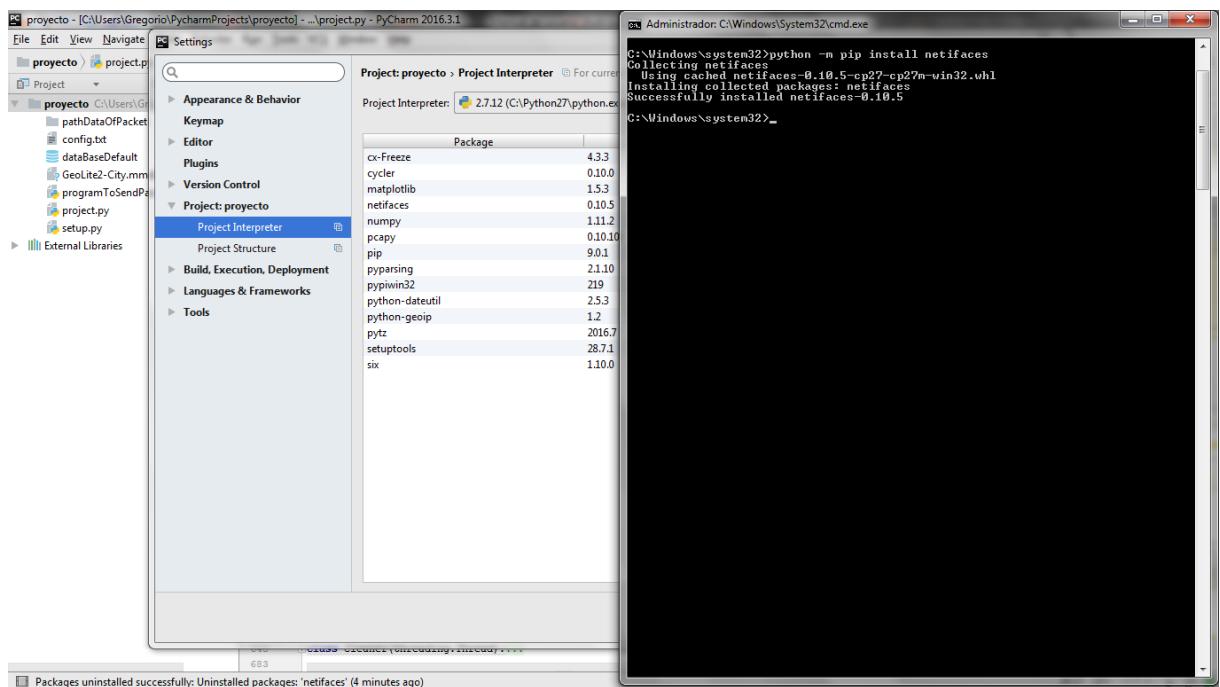
Nosotros no tenemos instalada la librería *netifaces*, como puede verse.



Ahora ejecutamos este comando en una terminal: “python -m pip install netifaces”, si tuviésemos varias versiones de python deberíamos escoger la que quisiéramos.



Si refrescamos la lista de librerías veremos que ahora si aparece.

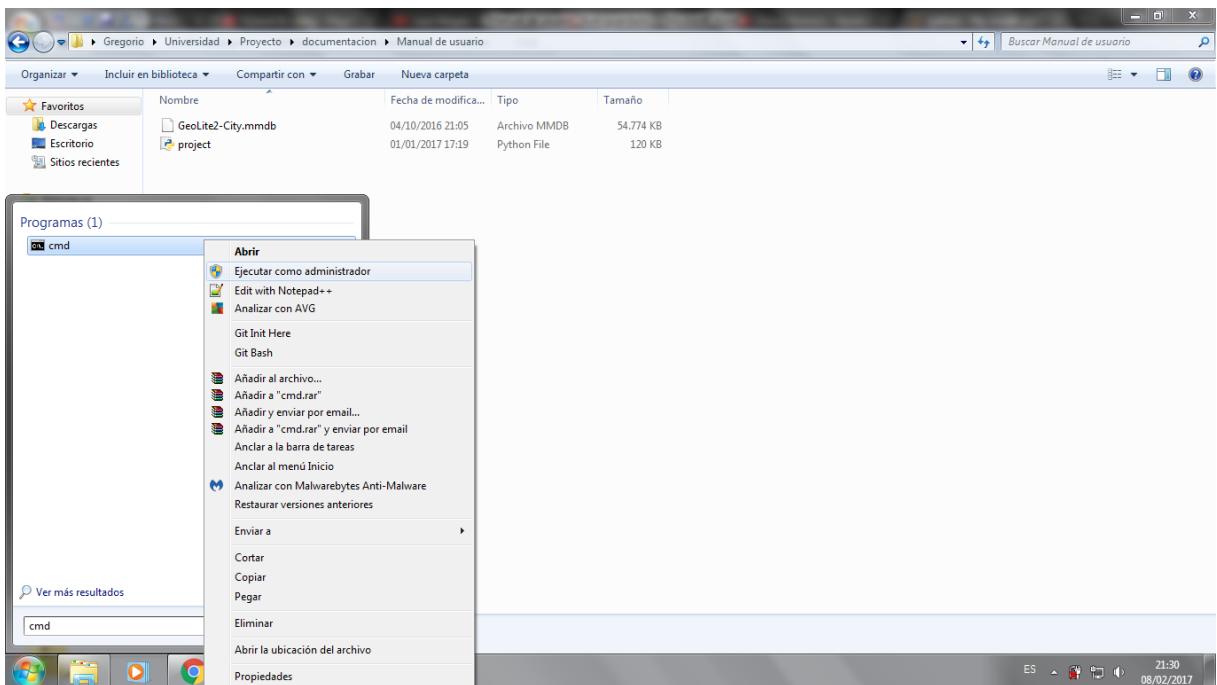


Este método lo aplicaríamos a todas las librerías.

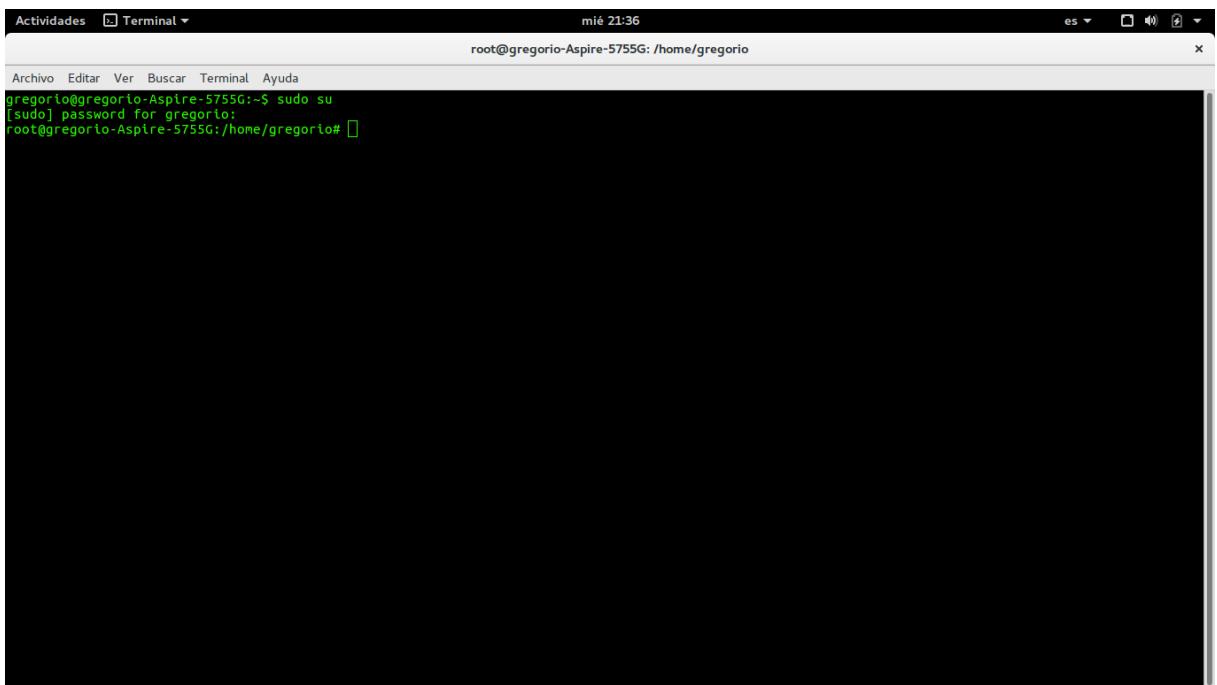
3.5. Configuración de la aplicación.

Una vez hemos instalado todas las librerías que requiere nuestro sistema, debemos configurarlo para que este pueda funcionar de forma correcta. Lo primero que debemos hacer es ejecutar la aplicación abriendo un terminal de Windows.

Importante: Debemos ejecutarlo con permisos de administrador para que funcione correctamente. En Windows se realizaría como muestra la siguiente imagen:

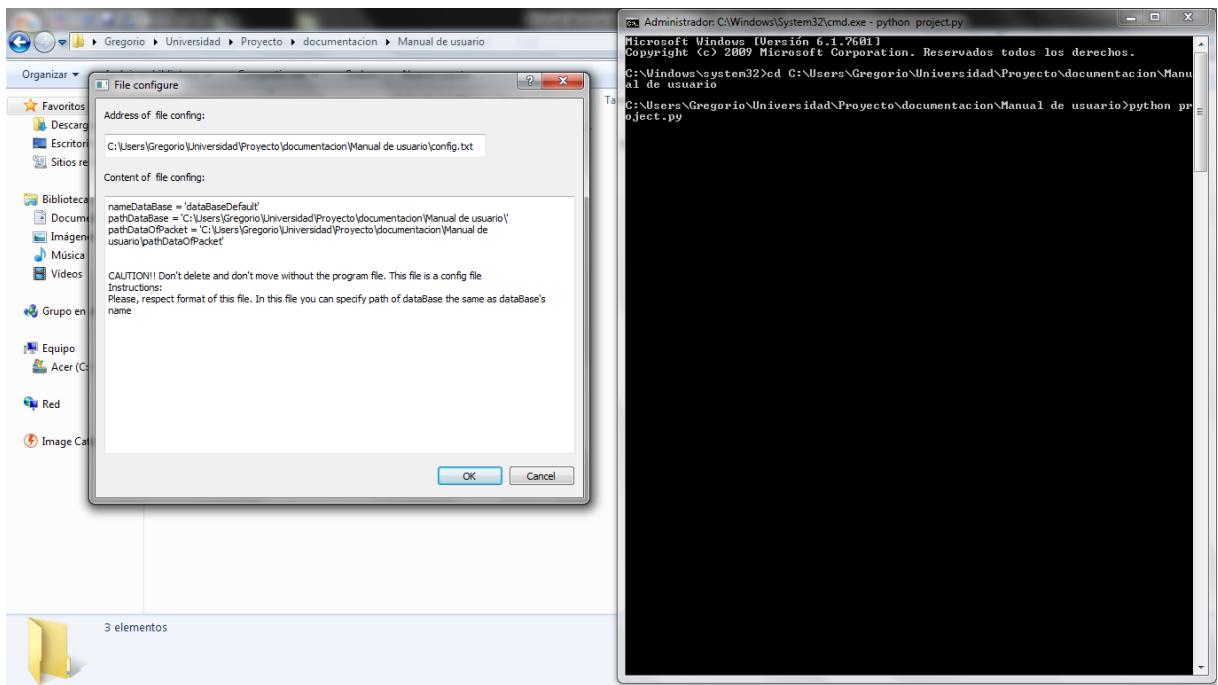


Mientras que en Linux se realiza de esta otra forma:

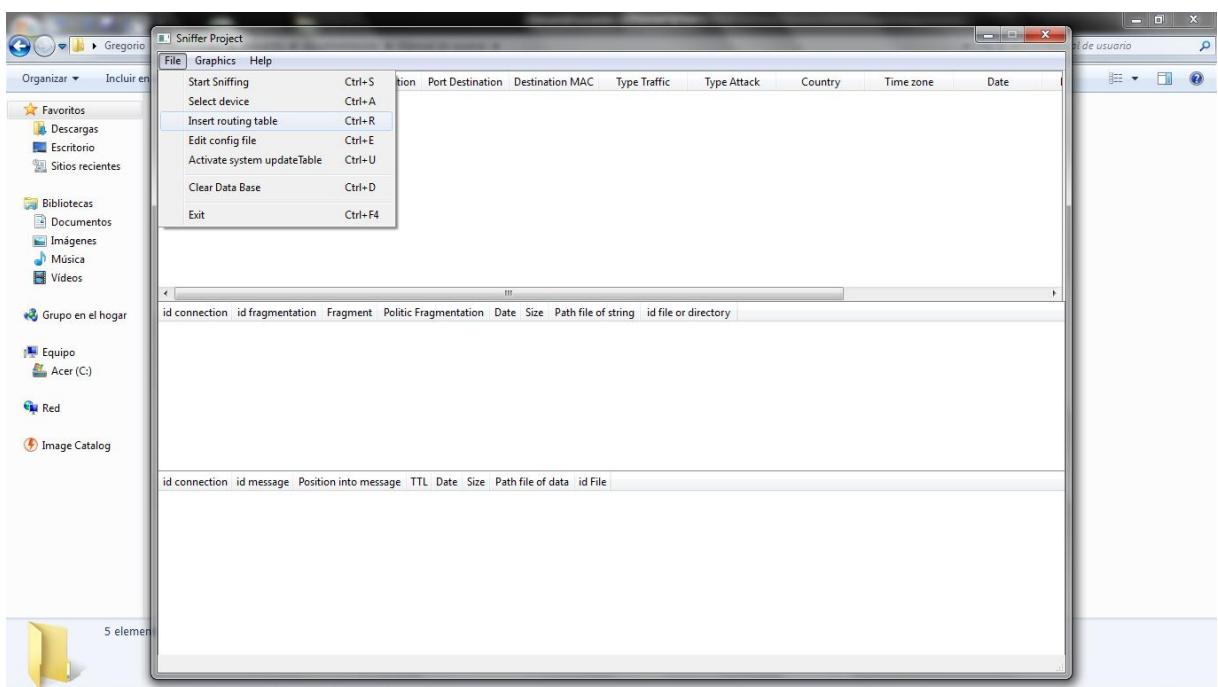


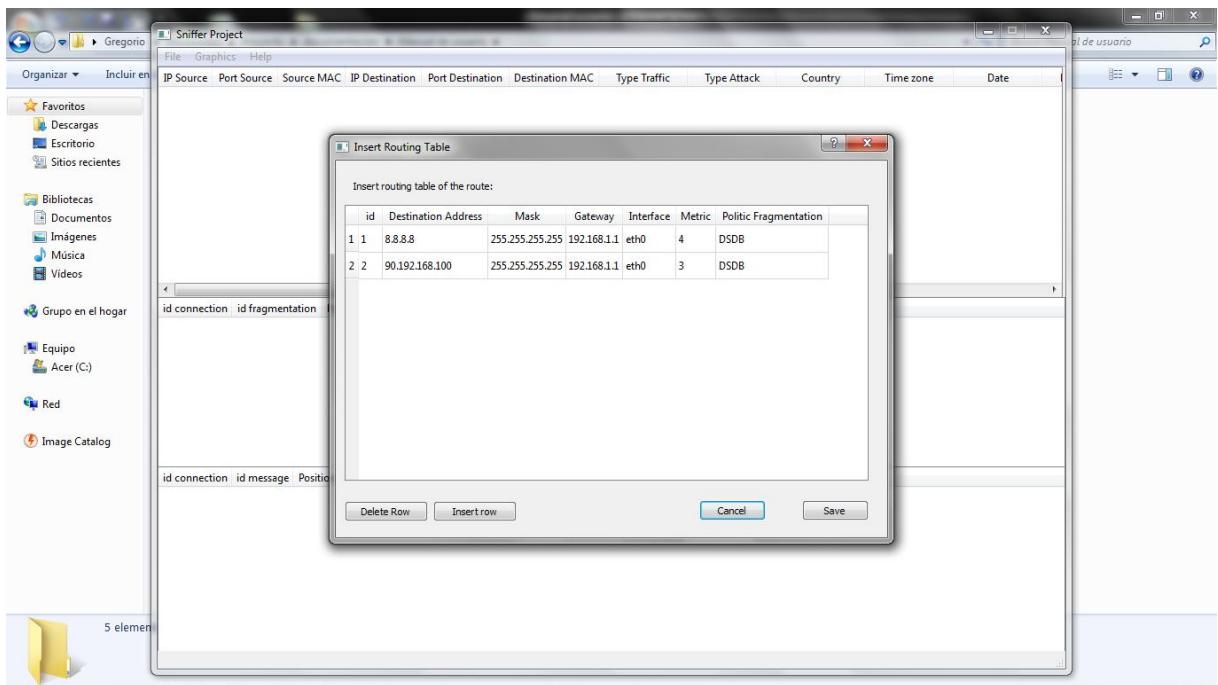
Nota: A partir de este punto todos los pasos son comunes en ambos sistemas operativos.

Una vez iniciado como administrador pasamos a ejecutar el software.

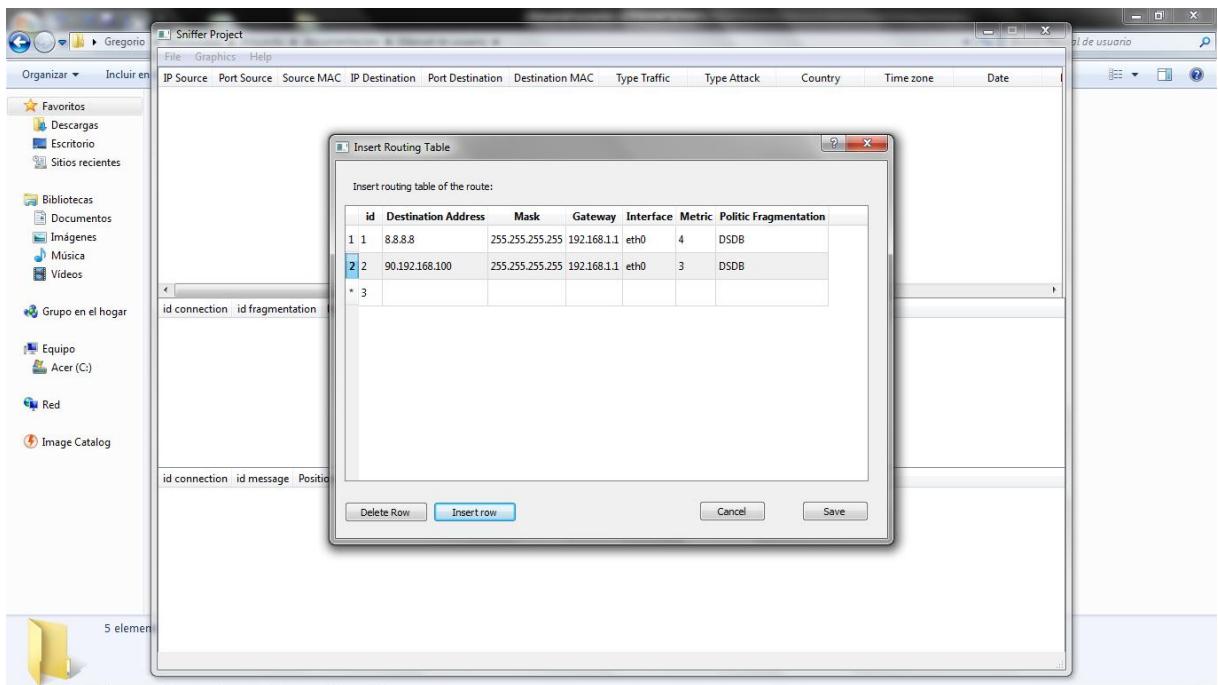


Como vemos la primera pantalla nos pide que configuremos las direcciones de la base de datos, y del sistema de ficheros que ayuda a la base de datos a guardar los contenidos de los paquetes. Cuando pulsamos aceptar se nos abre interfaz de la aplicación, lo que necesitamos configurar es la tabla de rutas de la aplicación, para que el sistema sea capaz de detectar ataques basados en TTL.

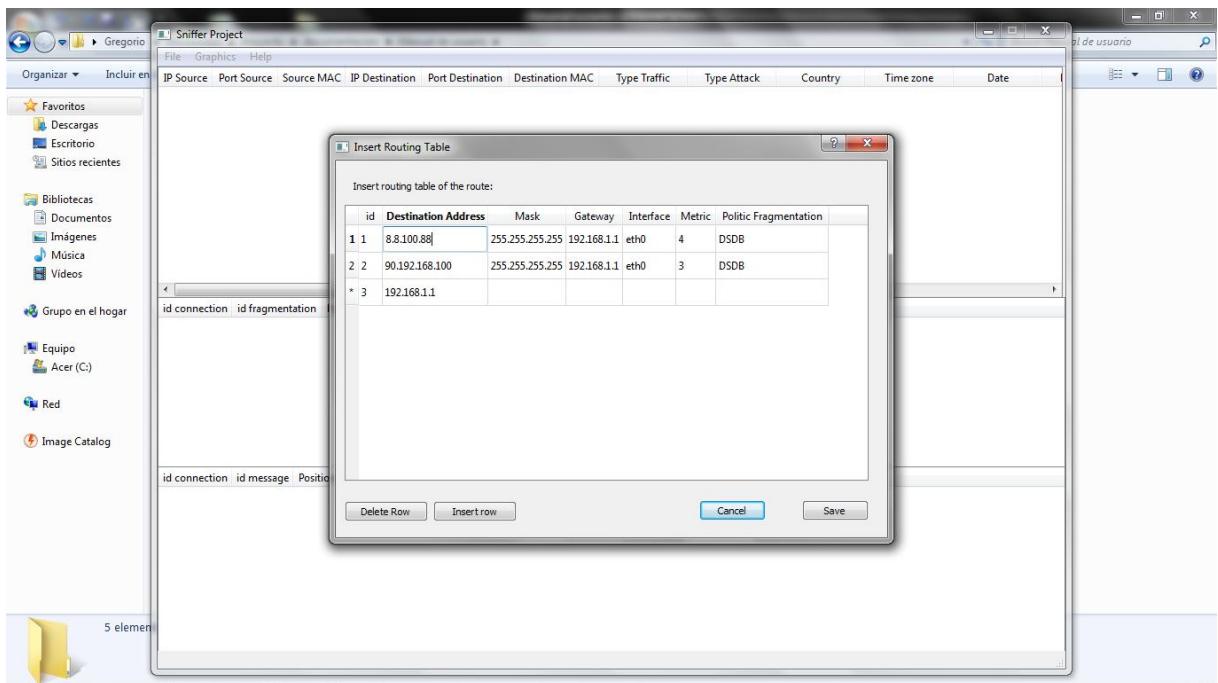




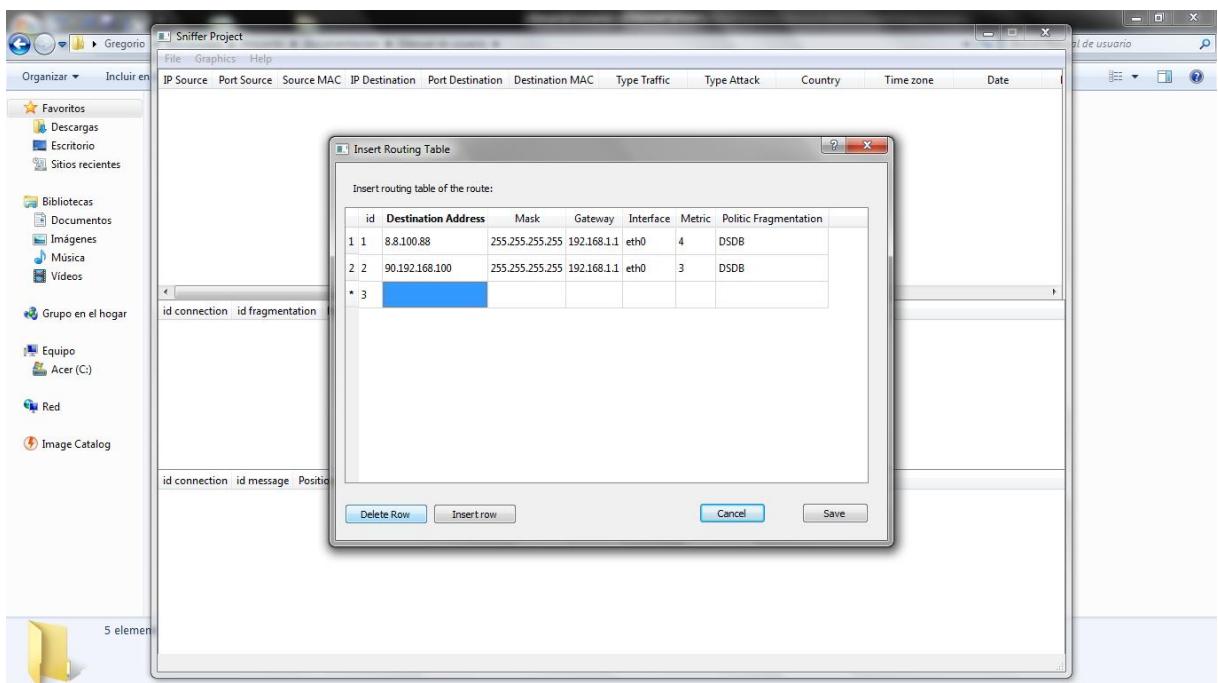
Una vez aquí podemos observar que se pueden introducir nuevas filas, pulsando sobre “Insert row”.

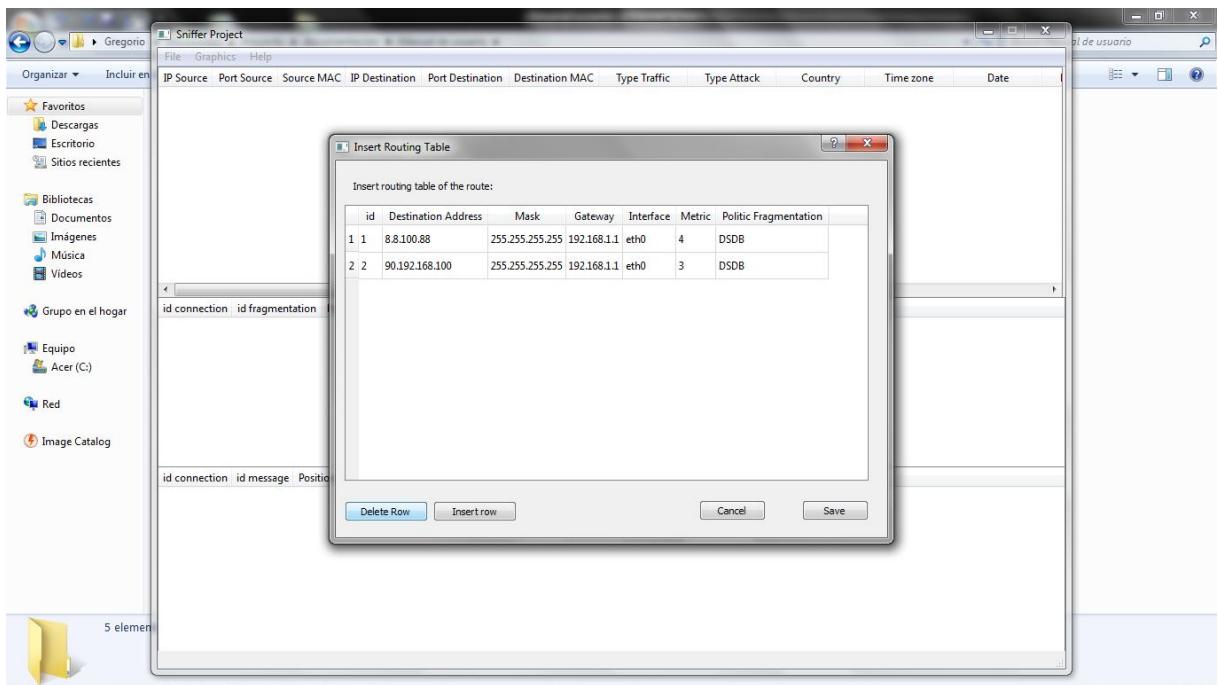


También se puede modificar cualquier fila haciendo doble click sobre el campo que se quiere modificar.



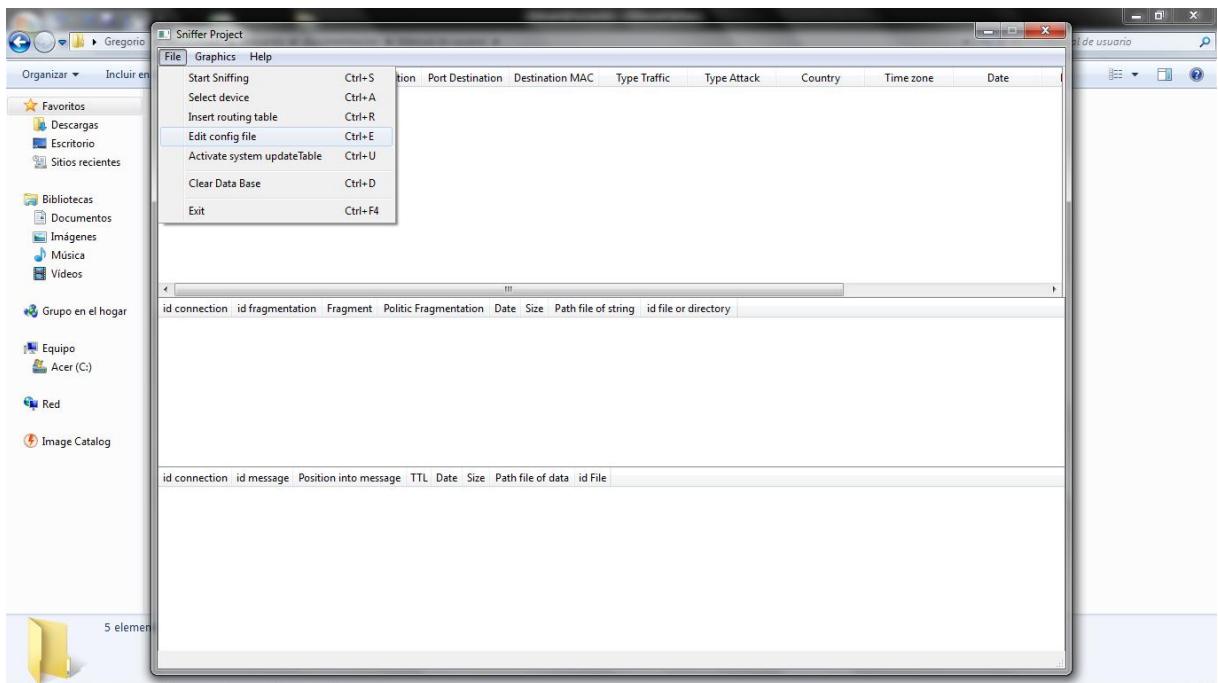
Si queremos borrar una fila solo tenemos que pulsar sobre ella y el botón “Delete row”.

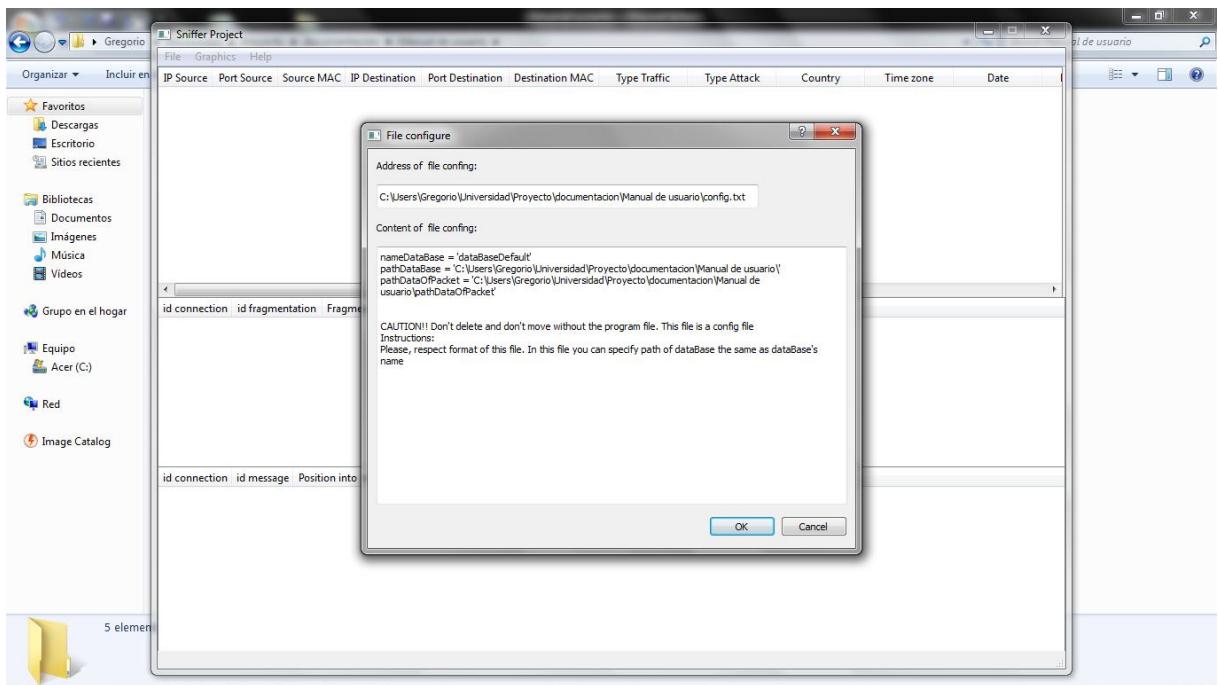




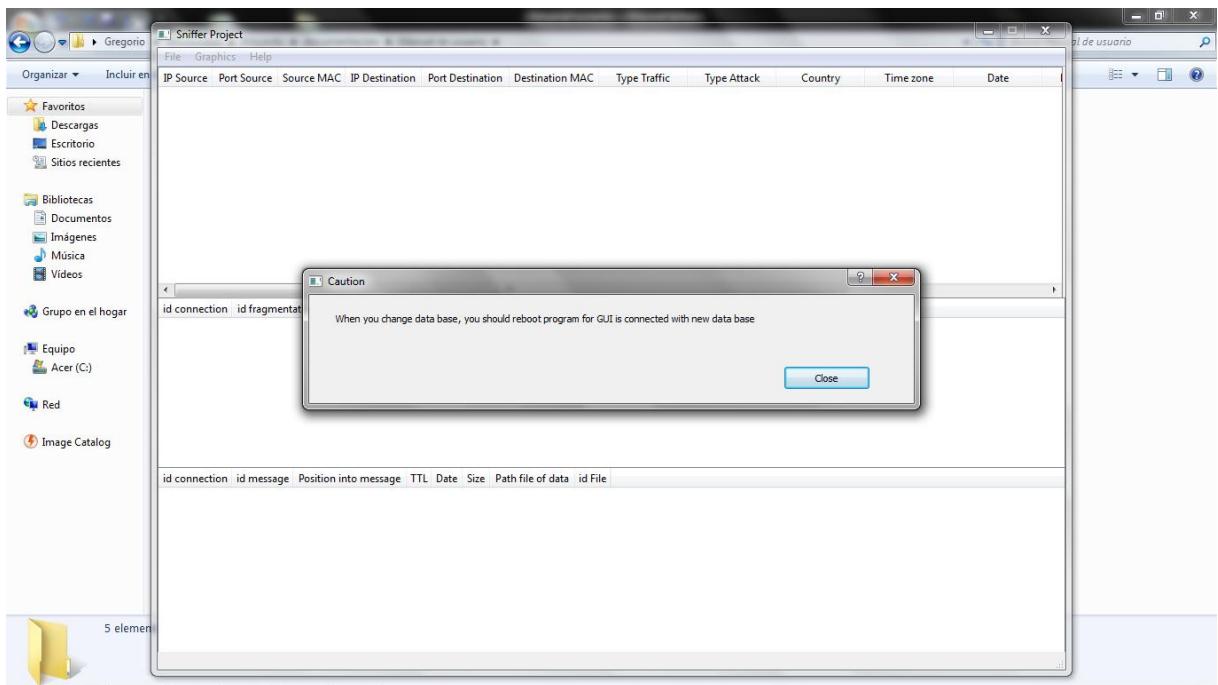
En cualquier momento podemos pulsar “cancel” o “save”. El primero solo cierra la ventana sin alterar el contenido, y el segundo guarda la tabla para su utilización.

En cualquier momento podemos cambiar la configuración del sistema pulsando en la correspondiente opción.





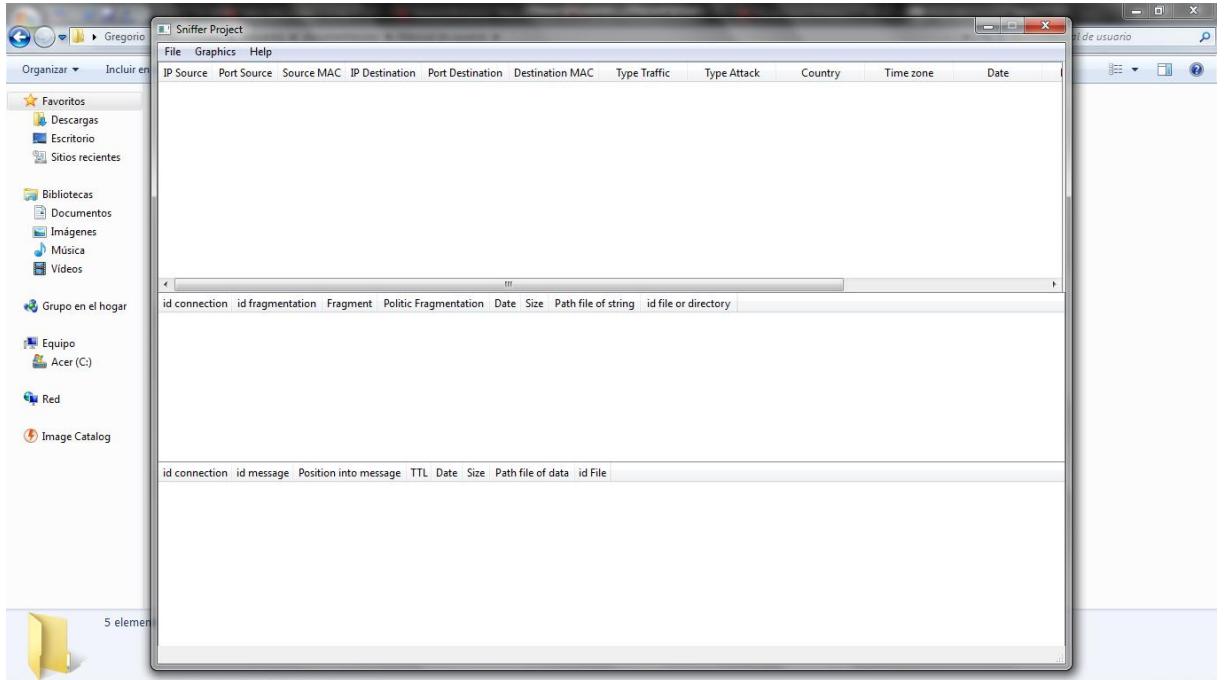
Si realizásemos una modificación sobre cualquier elemento de la configuración nos saldría un mensaje.



Nos indica que debe reiniciar la aplicación para que esos cambios tengan lugar y se utilice la base de datos que el usuario escogió.

Interfaz de la aplicación

En este apartado vamos a hablar sobre la interfaz de la aplicación, aunque ya se ha mostrado dicha interfaz. La interfaz es la siguiente:

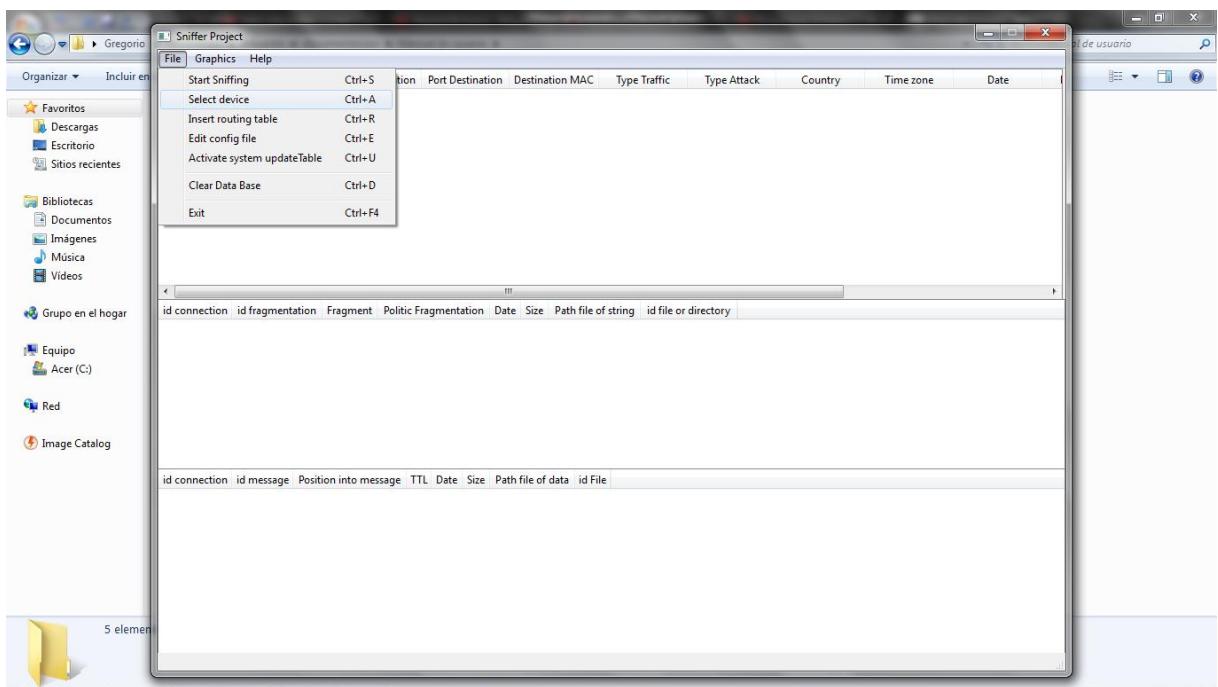


Como podemos ver en la barra del menú tenemos tres opciones que son:

- File
- Graphics
- Help

4.1 File

Esta es la primera opción la cual contiene la mayor parte de la funcionalidad del sistema.

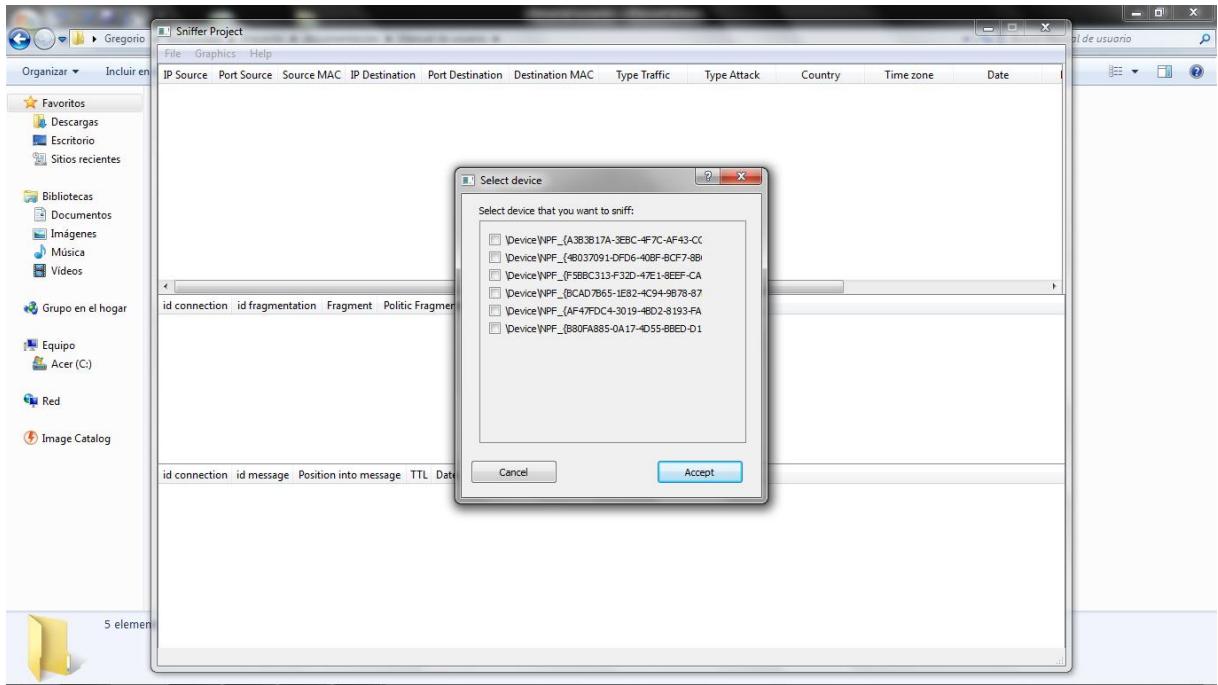


Desde esta opción se pueden lanzar las siguientes pantallas (todas las opciones pueden activarse pulsando la combinación de teclas expuestas al lado de los nombres):

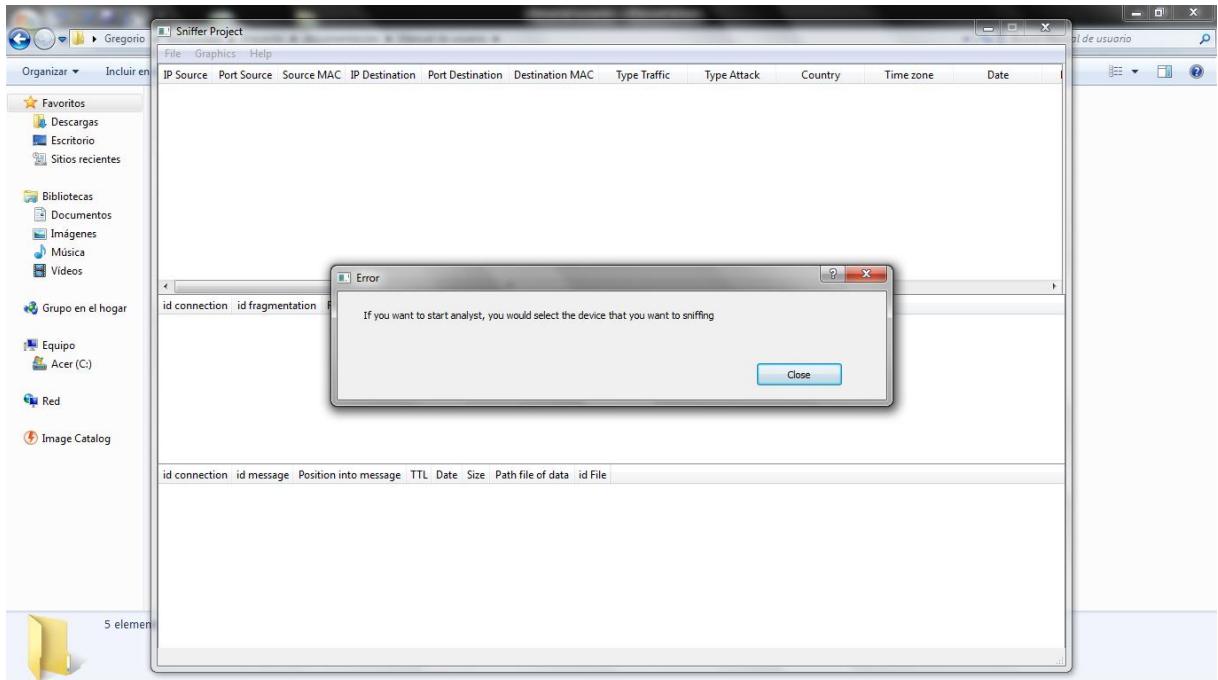
- Start Sniffing.
- Select device.
- Insert routing table.
- Edit Config file.
- Activate system update table.
- Clear data base.
- Exit.

4.1.1 Start Sniffing

En esta opción lanzamos el *sniffer* y el *analyst*, para analizar el tráfico en busca de posibles ataques, antes de lanzarlo debemos indicar que servicio de red tiene que supervisar.

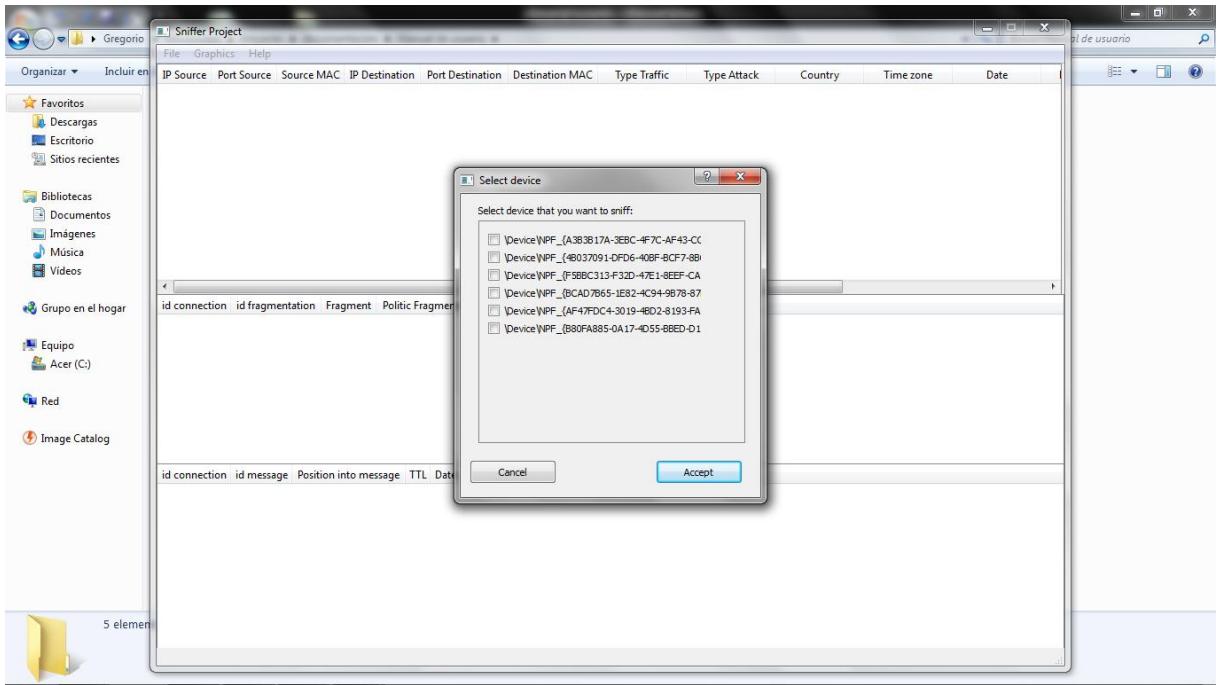


Una vez se seleccione uno se iniciara el *sniffer* y el *analyst*. Si pulsamos sobre cancelar saldría un mensaje para indicarnos que la selección del servicio de red, es una condición sine qua non para iniciar el *sniffing* de la red.



4.1.2 Select device

Esta opción nos permite elegir el servicio de red que queremos escanear.



4.1.3 Insert routing table

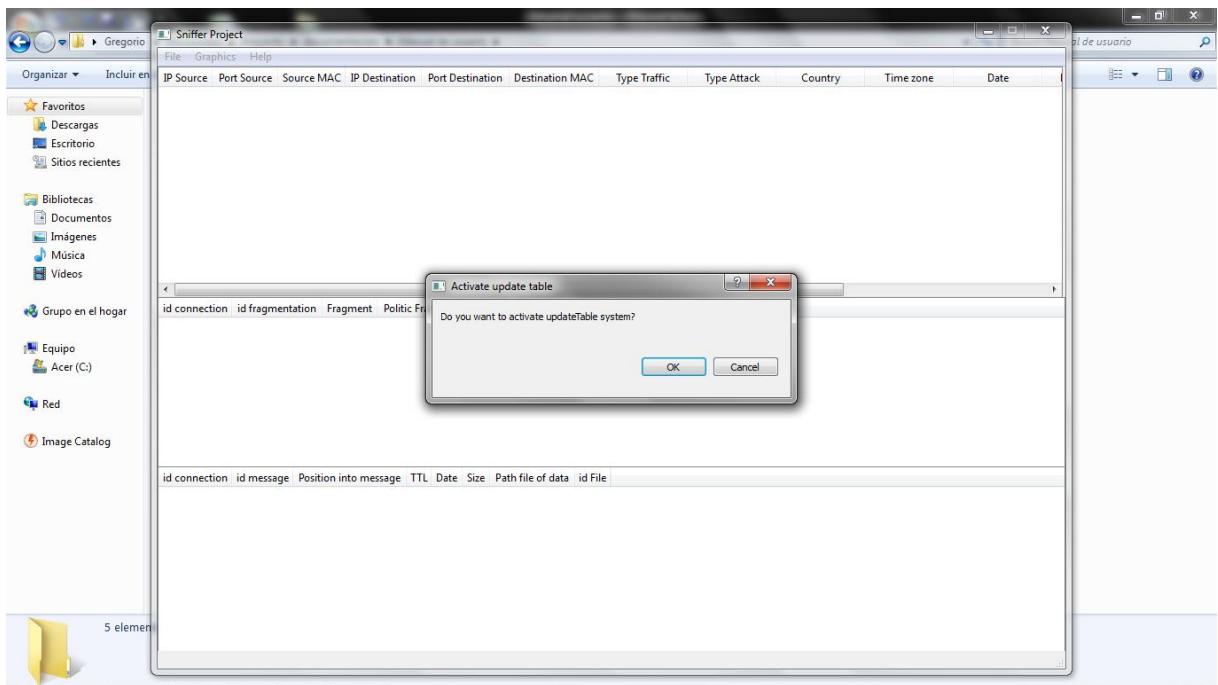
Nos permite insertar la tabla de rutas. Su funcionamiento ha sido expuesto en el apartado de [configuración](#).

4.1.4 Edit config file

Al igual que la opción anterior, esta ha sido mostrada en el apartado de [configuración](#). Su funcionalidad es la de proporcionar la capacidad a usuario de cambiar la base de datos o su ubicación.

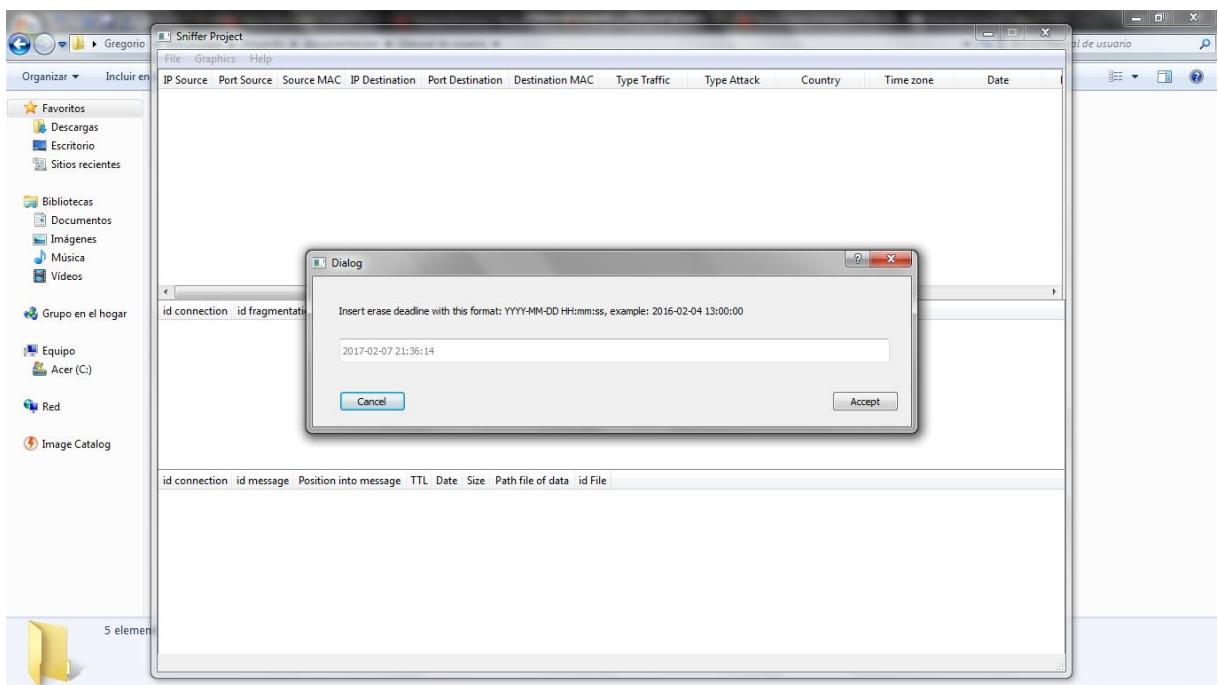
4.1.5 Activate system update table

Esta acción activa un algoritmo para que el refresco de las tablas no se realice en función de un número estático, sino que este vaya alterándose en función del número de elementos que se introducen en la base de datos, en una franja de tiempo. Pudiéndose así evitar que el refresco de las tablas impida su correcto funcionamiento.



4.1.6 Clear data base

Con la finalidad de evitar que la base de datos se sature por una gran cantidad de registros (es difícil alcanzar el límite ya que este es superior a los 100 TB), ofrecemos la capacidad de limpiarla a partir de una marca de tiempo. Cuando se pulse sobre ella, se mostrará un mensaje para la introducción de una fecha a partir de la cual se borrarán los registros anteriores a la fecha introducida.



Se muestran dos ejemplos para indicar al usuario el formato en el que debe introducir la fecha.

4.1.7 Exit

Esta opción cierra la aplicación, cerrando todos los procesos que hayan sido iniciados por el usuario.

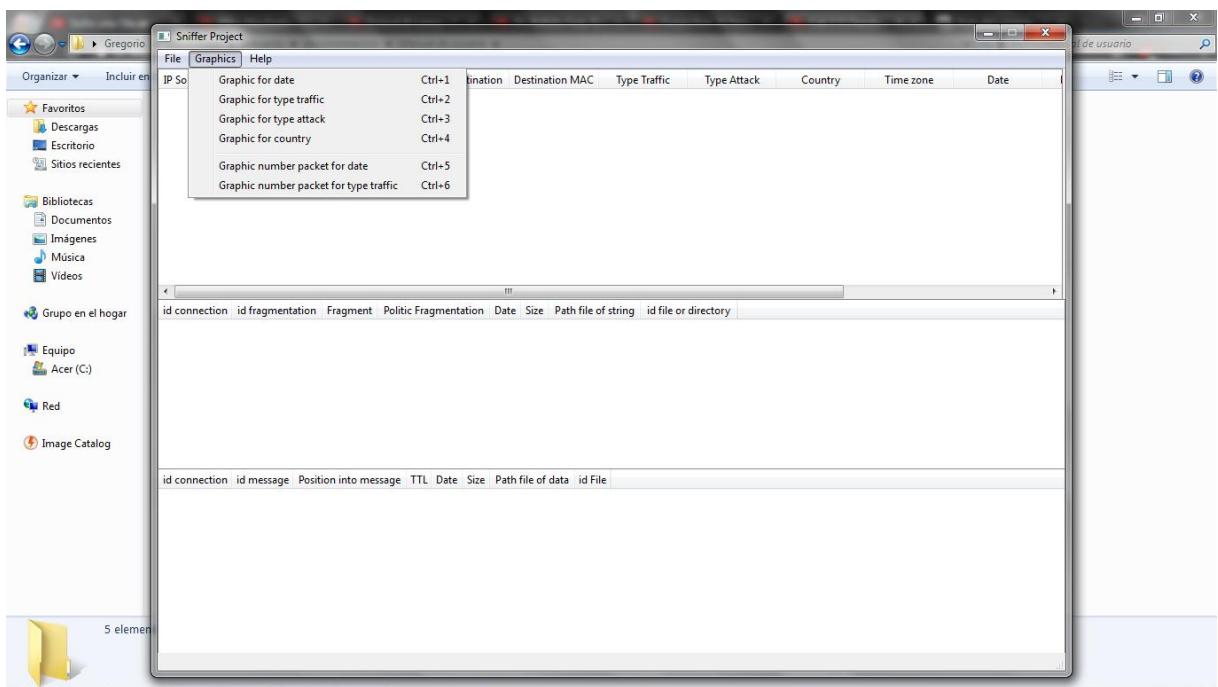
4.2 Graphics

Estas acciones nos permiten mostrar varias representaciones gráficas de los datos almacenados en la base de datos seleccionada.

Tenemos seis gráficas que son:

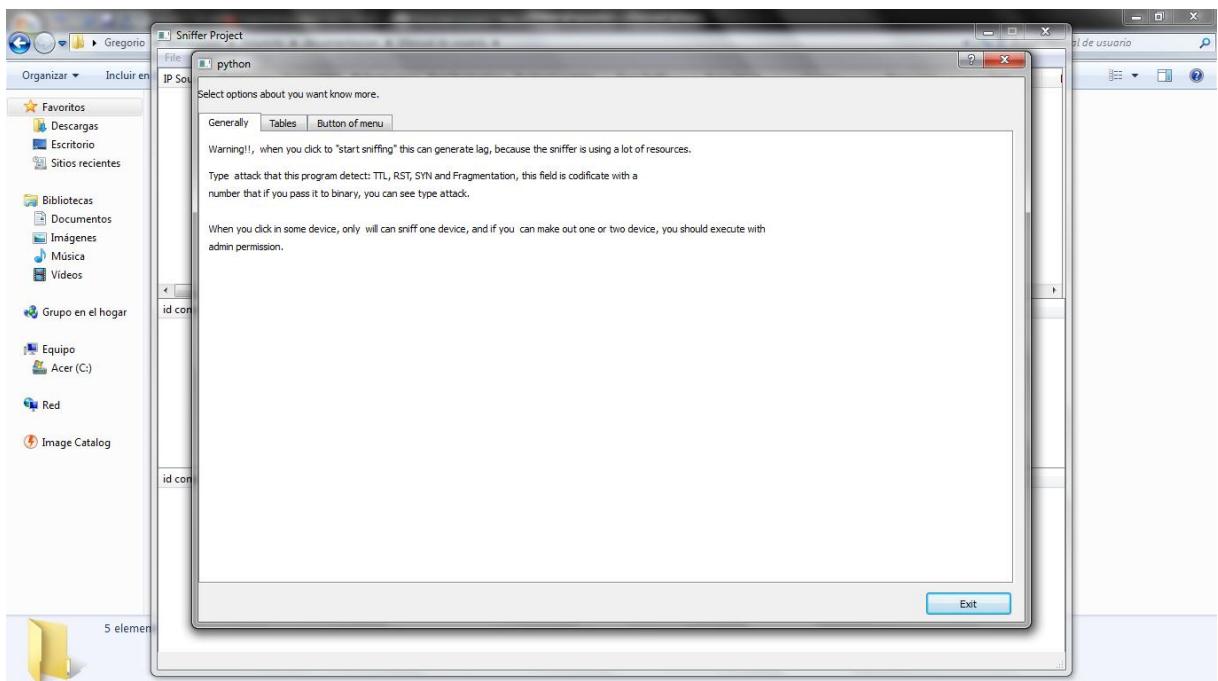
- Graphics for Date: por fecha
- Graphics for type traffic: por tipo de tráfico
- Graphics for type attack: por tipo de ataque
- Graphics for country: por país de origen
- Graphics number packet for date: número de paquetes por fecha
- Graphics number packet for type traffic: número de paquetes por tipo de tráfico.

Tal y como se muestra en la siguiente imagen:



4.3 Help

En esta nos muestra una ayuda sobre las distintas opciones que tenemos en nuestra aplicación.



Problemas

1. Cuando pulsa sobre un servicio para escanearlo y no aparenta que registre ningún tipo de ataque. Esto se puede producir por dos causas:
 - a. Que el servicio no tenga salida de red.
 - b. Que se haya iniciado pero todavía no se ha detectado ningún ataque.

Nota: Se recomienda utilizar la aplicación sobre un sistema Linux.

Contacto

Para más ayuda puede contactar al siguiente correo electrónico:

- Nombre: Gregorio Carazo Maza
- Email: gcm00014@gmail.com