**How to run**

Pay attention:

\* The functions we used are compatible with Windows only.

If you wish to run this program on other operating systems you need to use other libraries (for example pyWinHook will need to be changed to pyxhook).

\* If you are running Python version 3.7 and above, you need to use pyWinHook in keylogger.py script. Else, use: pyhook. The default in our program is: pyWinHook

Run: pip install pyWinhook -q

**Steps**

**Pre-requisites:**

\* Installation: PyCharm and python3.6 or above

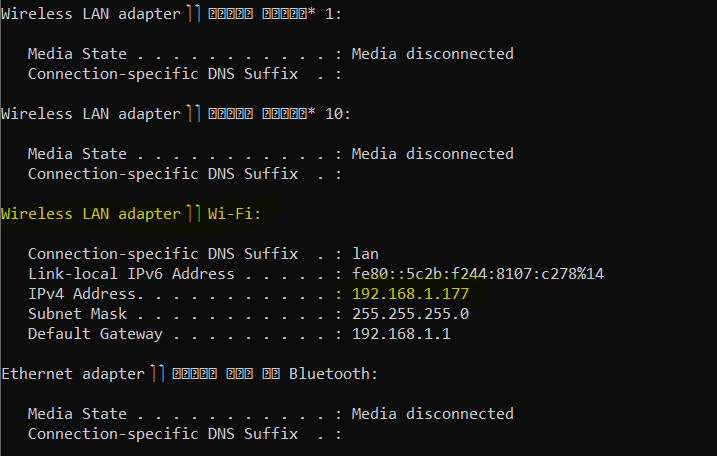
\* operating system: Windows

\* Packages required: pyWinHook, Listener, pythoncom,

os, threading, base64, socket

\* Make sure you run the scripts from the src directory, since there are dependencies (such as hidden the file in the directory, and use **newServer.py** and **base64cipher.py**)

\* IP address: please disable your firewall and prepare the attacked IP (to put in the keylogger.py and newClient.py), from your current WIFI:

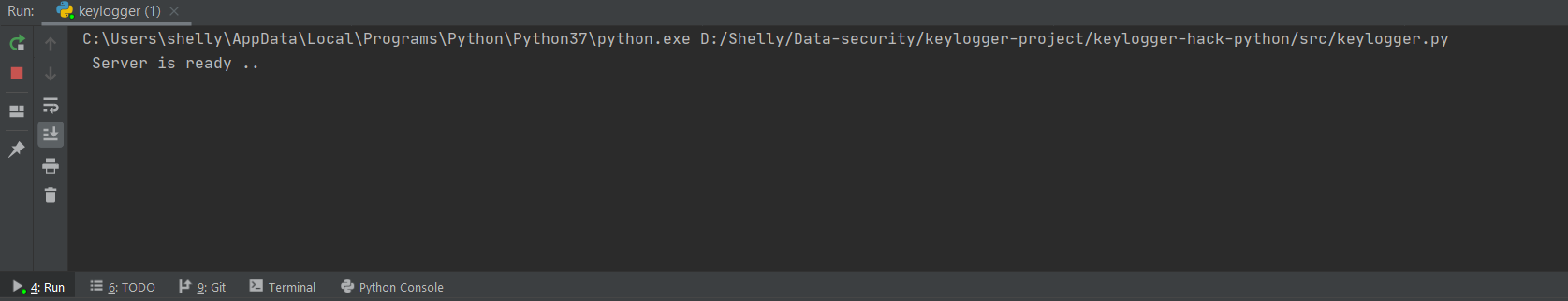


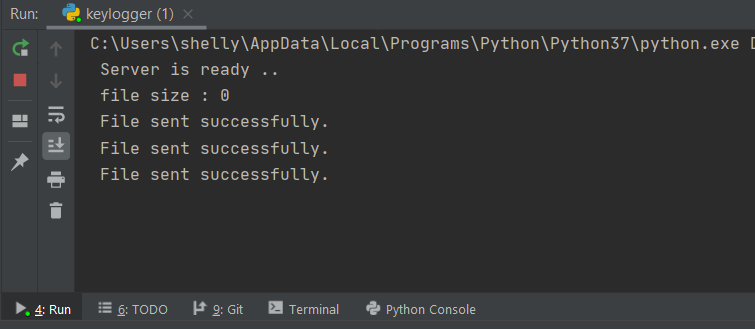
**From PyCharm:**

Getting started:

1. In src directory, there are 2 files that relevant for us to run:
   1. keylogger.py
   2. newCilent.py

the program that the "attacked" host runs is keylogger.py, and the "attacker" host is running the newClient.py script. First, we **will run the keylogger.py** and that is what should be appear in the console:

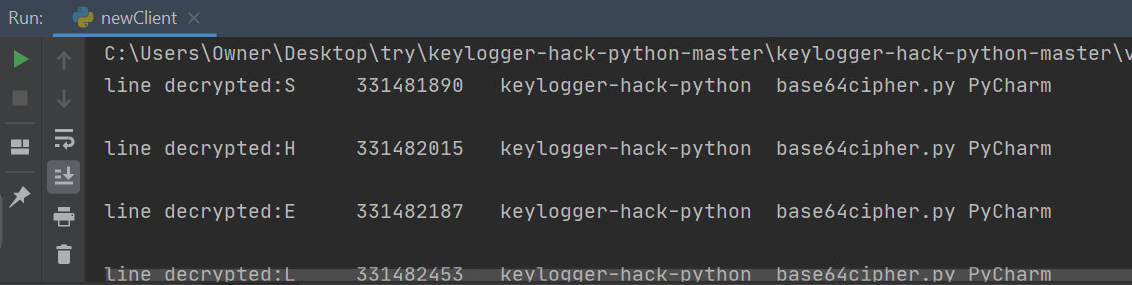


1. Then, the attacker (newClient.py) needs to listen to this host. **Run newClient.py**. then on the keylogger.py, this will show up:

Each 5 seconds, the keylogger sent an encrypted file, using base64 cipher, called log.txt to the client (the attacker), with all the keys typed. Then the client decrypt it back and put it in his own log.txt file:

For example, The encrypted file inside the attacked side:

And in the attacker side:



**Known issues**

* Currently the executable files we created for the keylogger.py and the newClient.py, are closing immediately each time we click on them, we tried to put an input field, so that the window will stay, but didn’t succeed. So, we are running the code from PyCharm.
* The keys sent to the client, are correct, and by their order.

However, the keys are sometimes duplicated and therefore close themselves because of incorrect padding error. We didn’t succeed to figure why.