**PYPEN**

**By:**

**Mohamad Hachem**

**Mohamad Hussam Hodaifa**

**Oussama Kheireldine**

**Advisor:**

**Dr. CharbelBoustany**

**Senior Project Submitted to the Faculty of Arts and Sciences**

**Department of Computer Science**

**AMERICAN UNIVERSITY OF SCIENCE & TECHNOLOGY**

**In Partial Fulfillment of the Requirements for the Degree of**

**Bachelor of Science in**

**Information Communication Technology**

**Achrafieh**

**31/05/2021**

We certify that we have read and tested this project and that, in our opinion, it is satisfactory in scope and quality as a senior year project for the degree of Bachelor of Science in Information Communication Technology.

**Project Committee**

**ElieNasr, Ph.D.**

*Assistant Professor -Chairperson of CSI and ICT Departments*

**Charbel Boustany, M.E.**

*Lecturer-Director of Zahleh Campus*

**Antoine Aouad. M.S.**

*Lecturer-Director of Sidon Branch*

**Giuseppe Boschiero**

*Lecturer and Game Design/Developer Expert*

**Acknowledgments**

We wish to convey our sincere gratitude and appreciation to each and every person who helped us through this project.

Special thanks for Dr.Charbel Boustany for his immense help and assistance in the computer and technical work and for guiding us throughout our project.

We thank god every day and every second for his grace on us, to be here reaching one of our goals, without forgetting our parents that they always have been there for us for days and nights.

**ABSTRACT**

**Purpose -**this paper presents an approach to assist in further more user friendly penetration testing, using python scripts, which will work on different platforms windows, Linux, and mobile. Now a days creating user friendly, simplified, and easily approached app/software with GUI interactions is not yet seen as much for penetration testing.

**Design/methodology/approach/Specifications**- Using python function scripts. These scripts will run from GUI application that displays the scripts you decide to create, once clicked, the script will execute and output its results inside the GUI application. An interpreter is used to enable the function to work on mobile platforms.

**Findings**- Based on the proposed approach of this software; it explores and evaluates a specific network status and highlights the findings such as weaknesses and vulnerabilities of certain factors in the targeted network, therefore later the user may assess his findings and implement the suitable fix to those weaknesses and vulnerabilities

**Practical implications -**The paper provides very useful insights, and tools for developers and specifically white hat users. In order to demonstrate this, we will be running a windows machine on virtualBox and executing the scripts. This research also has a great value in identifying the vulnerabilities/gaps which are critical to insure a secure environment.

**Originality/value**- The research methodology is a unique approach in making this app/software user friendly, easy to use, and of course suitable for every platform. The GUI feature combined with different automated possible penetration testing attacks makes it a really good, and innovative tool for users who seek for a fully stacked, automated penetration testing tool.

**TABLE OF CONTENTS**

**Page**

Chapter 1 - Introduction and Background 1

1.1 - Introduction 2

Chapter 2 -Problem Statement 3

2.1 - The Problem 4  
2.2 - Software Limitation 4

**Chapter 3 -Existing Solutions** 5

3.1 - The Lazy Script 6

3.1.1 - Advantages of Lazy script 6

3.1.2 - Disadvantages of Lazy script 7

**Chapter 4 - Proposed Solution** 8

4.1 - PyPen 9

**Chapter 5 -** **Design Specifications** 10

5.1 - System Architecture 11

5.2 - Data Flow Diagram 12

**Chapter 6 - Implementation** 13

6.1 - Programming Languages Comparison PYTHON VS Ruby 14

6.1.1 - Ruby 14

6.1.2 - Python 14

6.2 - Implementing by QT 16

6.3 - Implementation Languages and Platforms 16

**Chapter 7 - Testing and Verification** 22

7.1 - Main Menu 23

7.2 - Functions 24

7.2.1 - Key logger 24

7.2.2 - Network scanner 25

7.2.3 - WIFI Passwords 26

7.2.4 - Python.exe 27

7.2.5 - HTTP Sniffer 28

**Chapter 8 - Conclusions and Future Work** 29

8.1 - Conclusions 30

8.2 - Future Work 30

**References**……………………………………………………………………………………….31

**LIST OF FIGURES**

**Page**

Figure 5.1 - System Architecture 11

Figure 5.2 - Data Flow Diagram 12

Figure 6.1 - Language Growth Graph Comparison 15

Figure 6.2 - Key Logger Script 19

Figure 6.3 - Network Scanner Script 20

Figure 6.4 - Wi-Fi Passwords Script 21

Figure 7.1 - PyPen Menu 23

Figure 7.2 - Key Logger 24

Figure 7.3 - Network scanner 25

Figure 7.4 - Wifi-Passwords 26

Figure 7.5 - Python to exe 27

Figure 7.6 - HTTP Sniffer 28

**CHAPTER 1**

**INTRODUCTION AND BACKGROUND**

1.1 - Introduction2

* 1. **Introduction and Background**

Penetration testing is widely used now a day to test the protection level of companies, and environment. Discovering the faults, and vulnerabilities in it. Companies assign penetration testers which they have to test the company’s security, and to shine a light on the vulnerabilities, and weak spots, that endanger the company. After a tester finish his/her job. A well-constructed, and detailed report should be presented. Stating the weaknesses, and the vulnerabilities discovered, in addition to a proper solution, suggestions to fix this system. The tester should also state how he/she was able to exploit, and attack those vulnerabilities.

Many tools are used in penetration testing to discover vulnerabilities, and weaknesses. It’s known these tools vary in performance, and not all tools are user-friendly, and easy to master. Some tools are hard to use. Some tools are free open-source, some are free, and some are paid for. The tools for sure must be up to date, and modern to ensure the best performance, and precise results, and to ensure secure environment for the company and for yourself as well. Keeping in mind that penetration testers are exposed, and able to reach valuable information, that are to be protected, and taken care of. A trusted, and safe tools are very crucial thing in the work field of penetration testers.

This software is combined to work on many platforms like Windows, Linux, and Android.

The GUI is an added value for our software that distinguish from any other software. It can breaks the

Rules Of limitation, at the same time simplify the objective needed.

This project is not like any project, our goal letting the user go to the extreme that he needs at the same

Time be comfortable with the easy structure and organized software that target all his needs with a click Button.

The problem statement people are facing will be discussed in details alongside with some of the existing solutions such as existing penetration testing tools. Our proposed solution will tackle the limitation of such existing tools, and why our proposed application (PyPen) will eliminate these limitations, and excel in the cyber security field. Design specification and implementation of PyPen will be precisely discussed in this report, testing and verification will also be available through screen shots of the application itself. Finally, we’ll be addressing what to do for the future work to improve PyPen and in further more delivering the essentials and more features into penetration testing world.

**CHAPTER 2**

**Problem Statement**

2.1 - The Problem4

2.2 - Software Limitation4

**2.1 -The Problem**

Penetration testing is one important and major step/method in establishing a successful, and secure environment/business.

When it comes to security testing, it can be really messy, and tricky. A lot of scripts need to be done, mistake may happen which leads to loosing valuable assets, and most important thing its time consuming. Security team/IT will need to perform loads of attacks that require huge amount time scripting, and coding every attack one by one.

Companies and penetration testers are missing on a way to get things done fast, and the easiest way possible. They are missing on a tool that could deliver all their needs. From attacks to available vulnerability in an automated, simplifies, and less time consuming, yet with precise, professional way.

Cyber security students and teachers are in need of a tool to pen test their new cyber security projects, networks, and assignments. Of course the tool needed must be automated, and GUI instructed to deliver the simplest, easiest way to get their work done. That what we as cyber security students are missing out on, and we are here to make that tool available for way better learning.

**2.2 - Software Limitation**

The penetration testers face many issues:

* Cannot run on many platforms
* Not readable for beginners
* No graphics
* Not easy to be modified
* Hard access for log files

Those limitations create a big void in the penetration testing world. As those limitations can be seen in many penetration testing tools. The void must be filled by eliminating these limitations, and by creating and implementing new beneficial system/ideas. Many of the penetration testing applications act as an existing solution, but still facing issues as part of the limitations the world is facing.

**CHAPTER 3**

**Existing Solution**

3.1 - The Lazy Script6

3.1.1 - Advantages6

3.1.2 - Disadvantages7

**3. - Existing Solution**

**3.1 - The lazy script**

The LAZY script is a framework of serious penetration tools that can be explored easily from within it. This powerful and simple tool can be used for everything from installing new add-ons to grabbing a WPA handshake in a matter of seconds. Plus, it's easy to install, set up, and utilize.

Lazy Script automates process involved in Wifi Penetration Testing and Hacking. And script is for Kali Linux operating system.

The LAZY script starts by merely typing the letter l into a terminal window, then it asks for the name of your network interfaces after the first run. It uses the names you supply to connect to the tools needed to execute any attacks you select. Aside from that initial input, the majority of the possible attacks can be performed merely by choosing the option number from the menu. This means you can grab a network handshake or download a new hacking tool like Pupy by just selecting from one of the menu options.

**3.1.1 - Advantages**

The Lazy script attempts to create a friendlier way for beginners to start using some of the best and most reliable hacking techniques.

-It pulls together the necessary tools to execute the attack with a minimum of interaction from the user. In this way, the user interface and experience become the primary goal, and the script has the purpose of anticipating the tools and tactics a penetration tester would need quick access to in the field.

- The benefit of the LAZY script is that it was built with penetration testers in mind. This means it's essentially a guided tour through some of the best and most potent scripts available today. Some of the most easily accessible menu options include quick access to networking information like the gateway IP (usually the router), your IP address, MAC address, and a scan function that executes an ARP scan to reveal all other devices on the network.

**3.1.2 - Disadvantages**

-Though lazy script may seem very powerful tool for hacking but it has the disadvantage of preventing beginners from understanding what's happening "under the hood", it takes a very skilled person to understand thoroughly what each command or option the user launches does and operates.

-Another big disadvantage is that the Lazy script only runs on Linux based operating system “Kali”

The lazy script is a significant existing tool, but with having all these disadvantages, it lacks the performance the cyber security people needs. Our proposed solution took into consideration that those problems causing issues in the penetration testing fields. Our team decided to solve them, thus improving the performance in security.

**CHAPTER 4**

**Proposed Solution**

4.1 - PyPen 9

**4.1 - PyPen**

The application (...) runs on Linux, or Mac using python’s PyQt framework. PyQt is a combination of python version 2 and python version 3 bindings from Qt company, and it runs across different platform that are supported by Qt such as Windows, MacOS, Linux, IOS, and Android. PyQt features platform-independent abstractions for graphical user interfaces (GUI), and many other powerful features. Qt includes Qt designer a graphical user interface designer. PyQt is able to generate python code through the designer, and is able to add new GUI controls written in python as well.

The GUI application (...) will contain a set of predefined scripts pre-written, that assist the user in his journey. The scripts that (…) application can perform are many. Here are some examples:

* Discover devices on same network
* Spoof the local Wi-Fi network I am connected to
* Sniff packets on the local network
* Create a key logger
* Etc...

The user can pick any of these scripts. Once the user clicks on any script a new window will appear, that display the output of the script selected.

The user will have the ability to save the script output in a text file for later use. The program also has a log file in case the user wants to check his logs of commands.

For safe and authentic activity, credentials will be required in order for the program to be used.

PyPen will include features that other tools are missing, with a lot of attacks to satisfy the user's need.

In the upcoming chapter we'll talk about the design specification of PyPen. Visual representations of how the PyPen works and how its build, throughout system architecture and data flow diagram.

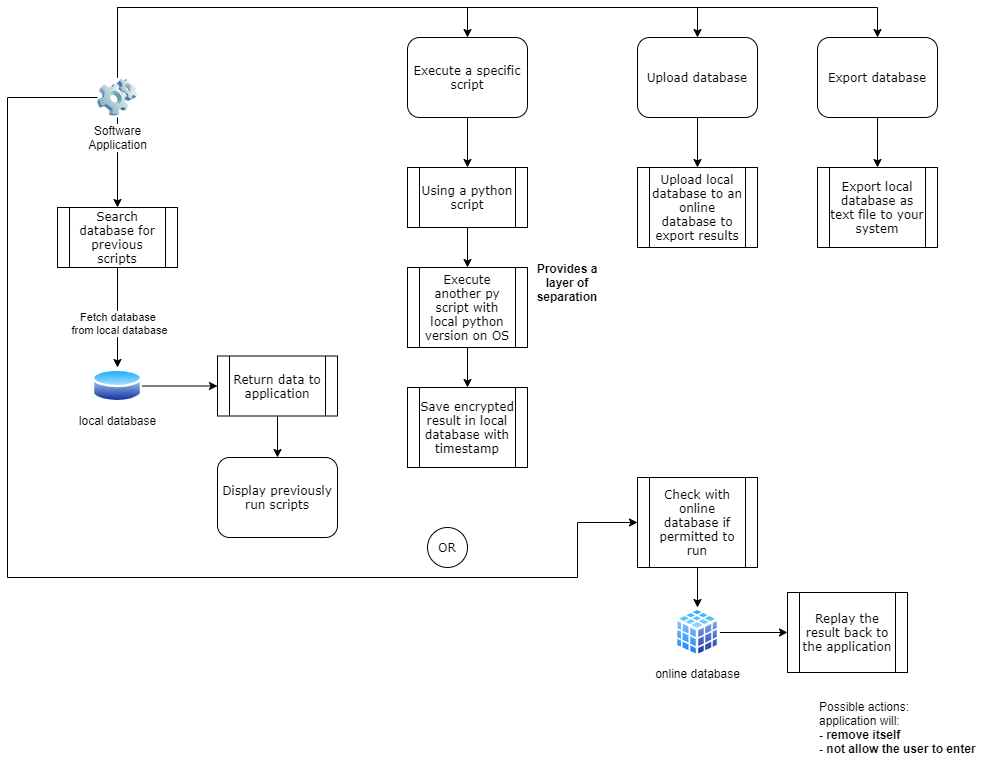
**CHAPTER 5**

**Design Specifications**

5.1 - System Architecture 11

5.2 - Data Flow Diagram 12

**5.1 - System Architecture**

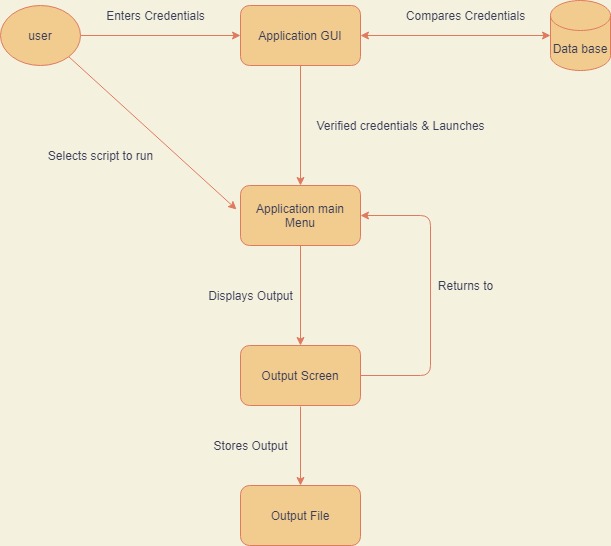


**(Figure 5.1 - System Architecture)**

**5.2 - Data Flow Diagram**

Data flow diagrams (DFD) are visual representations that show the components that relate various parts of a data or information system together. They are used by organizations to visually depict the flow of data and system requirements in an entire system. DFD’s are useful in a number of business activities, especially in mapping the data necessary for a process to run; this includes:

* Notation styles
* Physical requirements
* System automation



**(Figure 5.2 - Data Flow Diagram)**

These diagrams illustrate in details the Papen’s architecture and how it works. The next chapter will continue with the implementation part. Algorithms, the language used and why this language is used Also what software used to successfully build PyPen.

**CHAPTER 6**

**Implementation**

6.1 - Programming Languages Comparison PYTHON VS Ruby 14

6.1.1 - Ruby 14

6.1.2 - Python 14

6.2 - Implementing by PYQT 16

6.3 - Implementation Languages and Platforms 16

**6.1 - Programming Languages Comparison PYTHON (Python vs Ruby)**

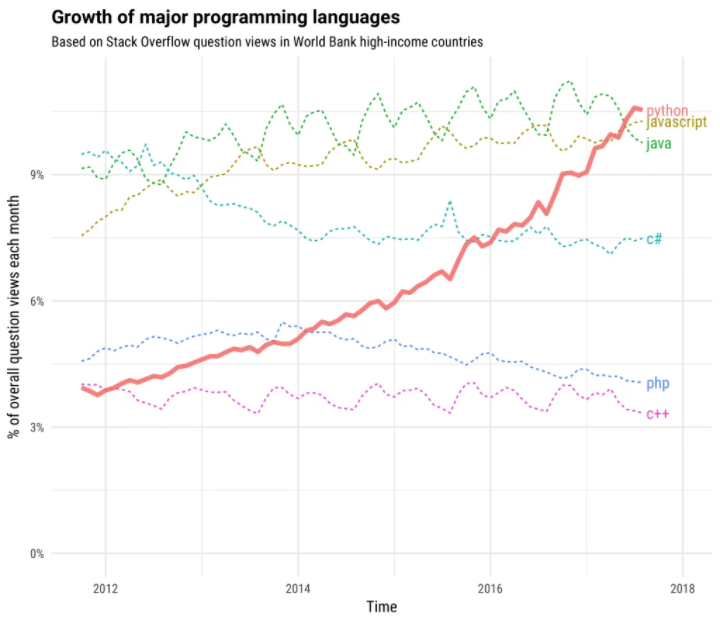
**6.1.1. Ruby**

Ruby is a pure object-oriented programming language. It is a dynamic open-source language and it runs on all kinds of platforms like Mac OS, Windows, and all versions of UNIX. Ruby as python may deliver high-level scripting, which helps in functionality. It is a general-purpose language similar to Python, so it has many other applications like data analysis, prototyping, and proof of concepts.

**6.1.2. Python**

There are many languages that can perform tasks related to cybersecurity, but Python is demolishing in the cyber security field now a day.

Python has been used widely in cyber security field because of its easy to learn syntax, and wide range of libraries, which helps in giving it a lot of functionality. As we can see in this diagram



**(Figure 6.1 - Language Growth Graph Comparison)**

**6.2 - Implementing by PyQT**

In order to achieve a very friendly user interaction, with GUI buttons, PyQT framework is one of many good ways to do so. With PyQT we were able to increase the amount of interaction between the user, and the app. There are many software modules that can generate GUI interfaces for example TKinter. We decided to choose PyQT over anything else, since PyQT is used for achieving advanced programming results.

* Some of PyQT advantages:

Qt offers several widgets, such as buttons or menus, all designed with a basic appearance across all supported platforms. Qt also offers Qt designer which is less time consuming, and much easier.

Qt uses a wide array of native platform APIs for the purpose of networking, database creation, and many more. It offers primary access to them via a unique API.

Very flexible code GUI programming with Qt is created around signals and slots for communication amongst objects.

**6.3 - Implementation Languages and Platforms**

Our team members worked and developed PyPen using Python language, and used multiple algorithms and scripts with the help of PyQT.

* Python language:

Python advantages are many. Here are the most useful advantages of python:

Easier debugging: Python use minimal code, which makes it way easier to spot errors, and reduce the risk of complexity.

Productivity: Python language provides advanced control capabilities to the user. It also has text processing capabilities and secure integration, which improve functionality.

Automatic memory management: Python has a built in memory management, called python memory management, this is basically to help the user in caching, partitioning, and memory allocation.

Python excellence in security: Python is used by cyber security professionals, in analyzing malwares, port scanning, network scanning, accessing ports, decoding packets, and many more.

Vast libraries: Python contains many libraries, which make the work way easier, and less time consuming. That’s why Python is widely used by professional cyber security people.

Cross -platformer language: Most operating systems come with python pre-installed on it, however some platform may need python interpreter in order for python to work. Python works on Linux, Microsoft, MacOS, and many more.

Although Python is considered one of the best programming languages considering security, it still has some disadvantages compared to other languages.

Some of Disadvantages of python:

Limited speed: Python has lower speed of execution compared to other languages; this is because python code is executed line by line

Error detection: Error can’t be detected during compilation, since Python is executed through interpreter not a compiler

Not suitable for mobiles: Since Python has a low speed of execution, and due to its memory consumption, it’s not that good programing language concerning mobiles.

Access to Database: Python Database layer is underdeveloped, compared to other technologies such as Open Database Connectivity (ODBC) or Java Database Connectivity (JDBC)

* Algorithms used in creation of PyPen

Algorithm is a step-by-step procedure, to execute a certain instruction in a defined order to achieve the desirable outcome.

Not all procedures commands/scripts are called algorithm.

Characteristics of an algorithm:

Clear: algorithm should clear. Steps, inputs, outputs must be clear and only lead to one meaning.

Input: An algorithm should contain minimum 0, and maximum many well-structured input.

Output: An algorithm should contain minimum 1, and maximum many well-structured output.

Note: the output should be as desired

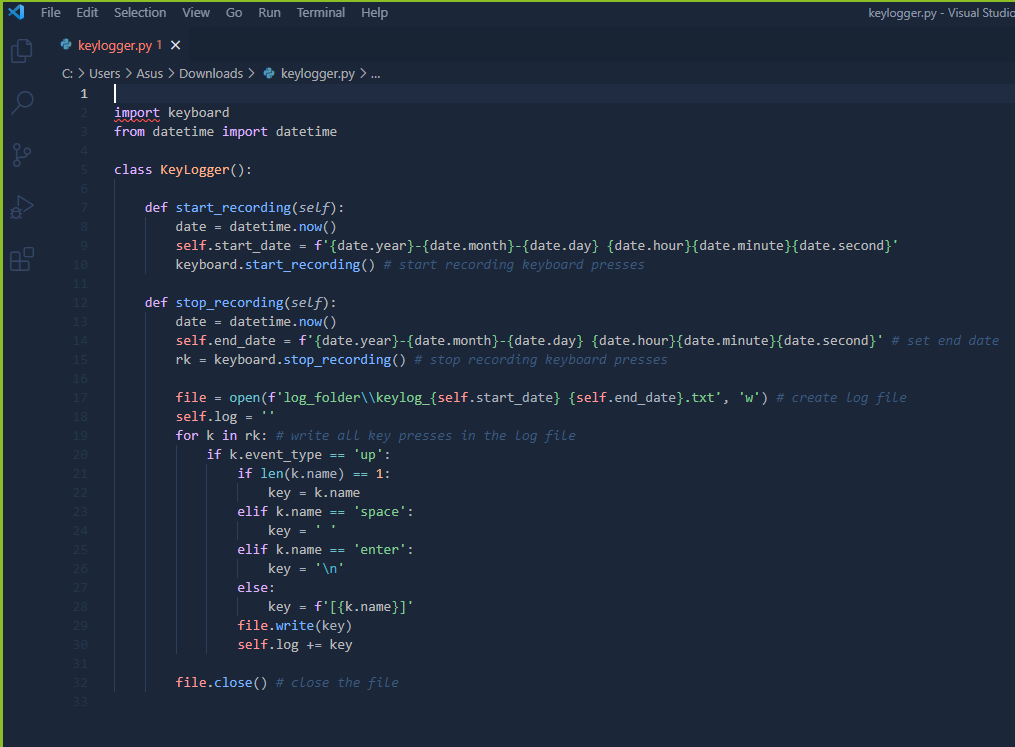
Feasible: should be feasible with the available resources.

Used algorithms:

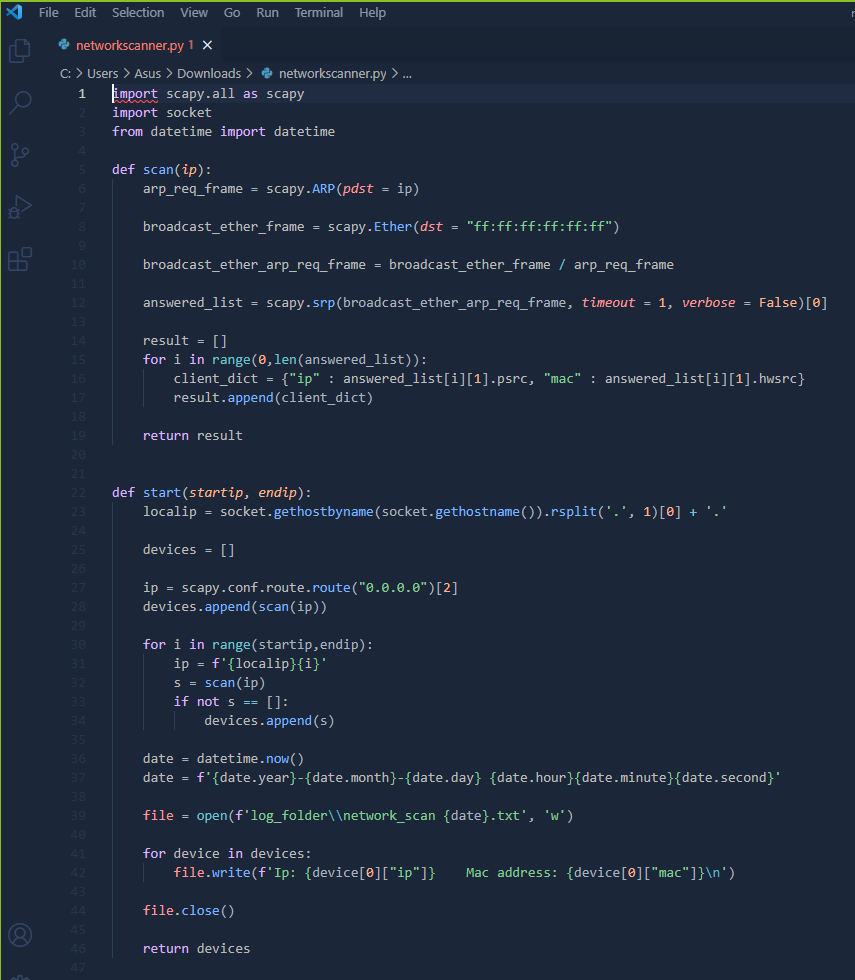
* Search: search algorithm is to search for a specific item in a specific data structure
* Insert: insert algorithm is to insert a specific item in a specific data structure
* Delete: delete algorithm is to delete a specific item in a specific data structure

These algorithms have been used numerous times in the predefined scripts while making PyPen.

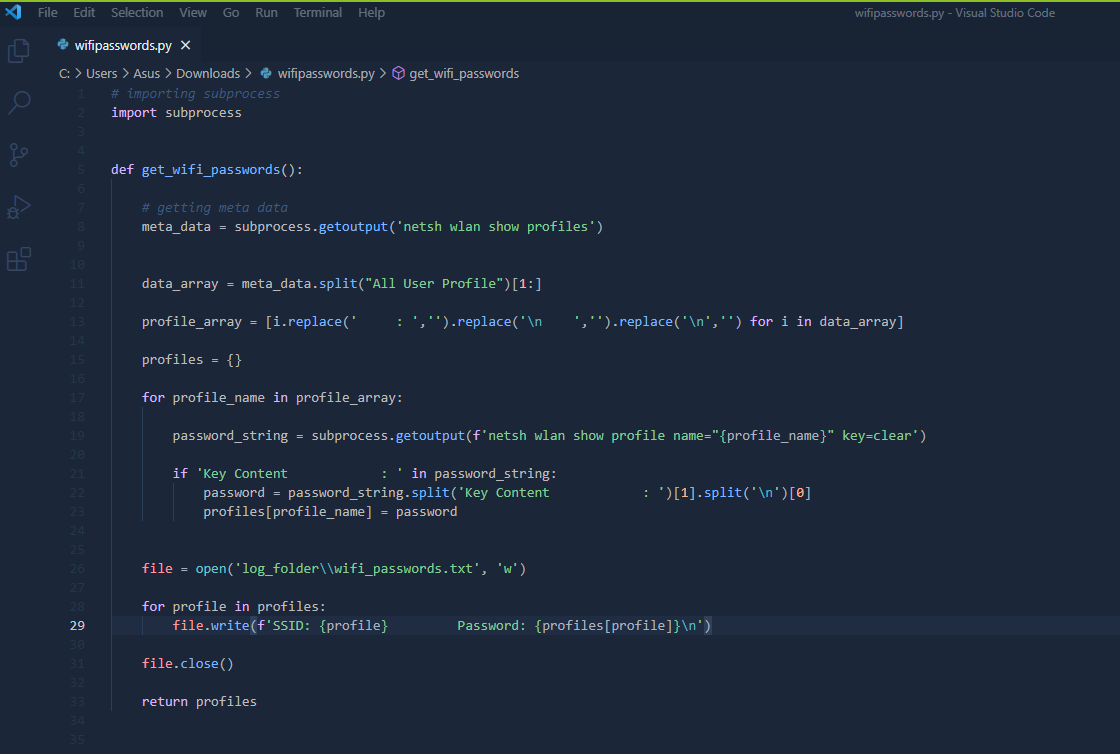
* Scripts have been used



(**Figure 6.2 - Key Logger Script)**



**(Figure 6.3 - Network Scanner Script)**



**(Figure 6.4 - Wi-Fi Passwords Script)**

After building the application with this specific implementation design, testing and verification will be in the next chapter, where we provide a screenshots of the application itself.

**CHAPTER 7**

**Testing and Verification**

7.1 - Main Menu 23

7.2 - Functions 24

7.2.1 - Key Logger 24

7.2.2 - Network scanner 25

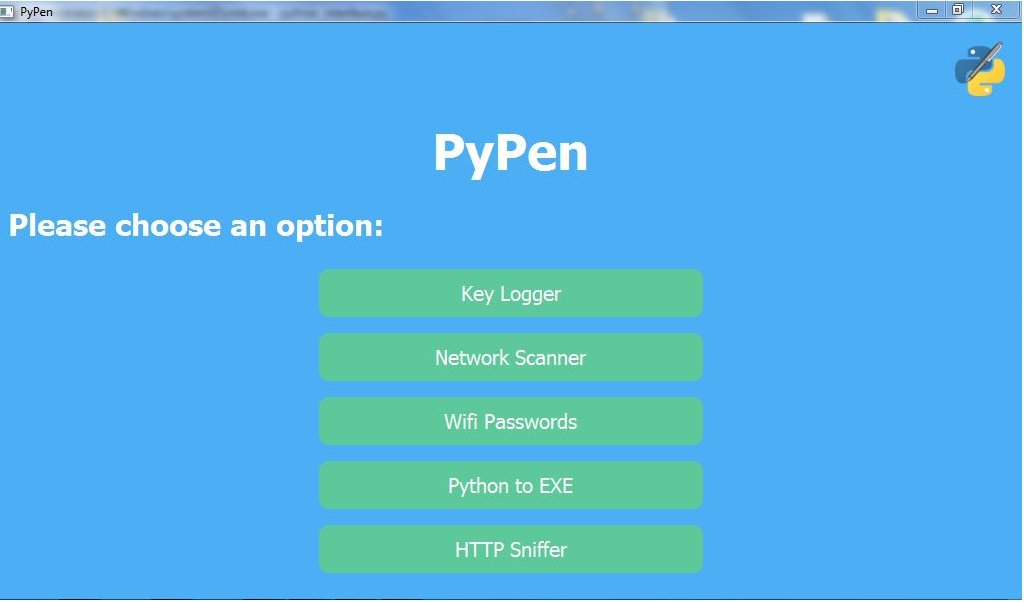
7.2.3 - Wi-Fi Passwords 26

7.2.4 - Python to Exe 27

7.2.5 - HTTP Sniffer 28

**7.1 - Menu**

This is the screen which welcomes the user offering him/her a variety of five functions to perform as shown in the screen shot below

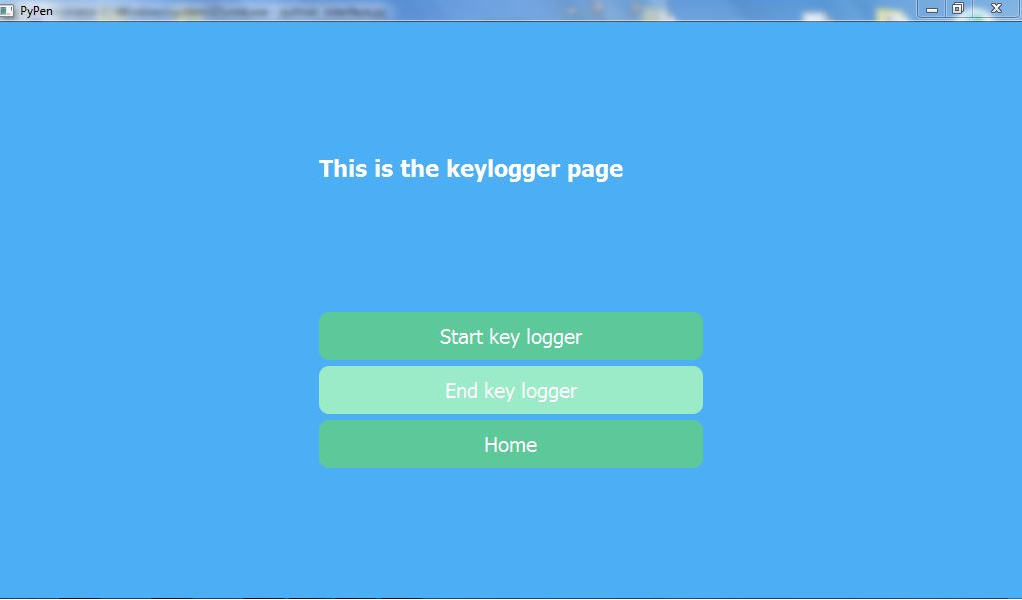


**(Figure 7.1 - PyPen Menu)**

**7.2 - Functions**

**7.2.1 - Key logger**

This is the first function from the top of the main menu. The user launches this function by clicking "Start key logger" then everything the user types of clicks is being recorded and saved for when the user ends logging**.** The recorded typing the user did will be shown on the screen and stored in a log file.

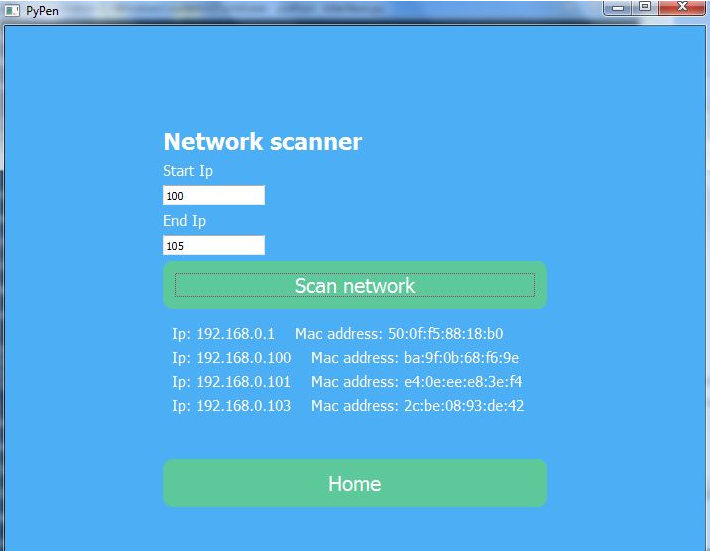


**(Figure 7.2 - Key Logger)**

**7.2.2 - Network scanner**

In this function the user needs to enters the Start of an IP(only the fourth octet) and the end ip (also the forth octet only), then the user presses "scan network" button to see all the active hosts on the same network as the user detailed with the IP and MAC addresses.

The output of the network scan then is displayed on the screen and stored in a separated log file as shown in the figure below.

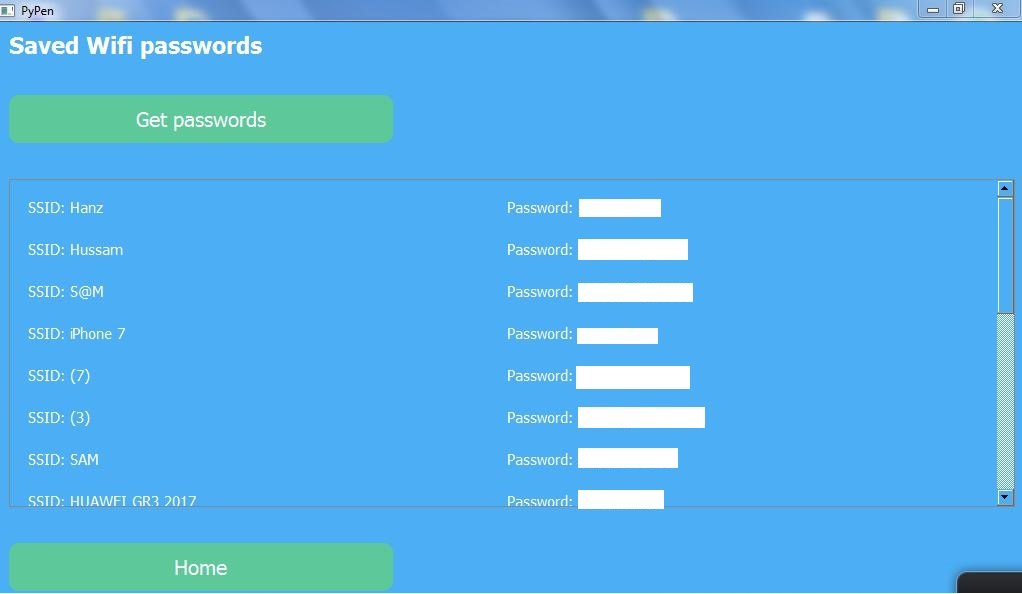


**(Figure 7.3 - Network scanner)**

**7.2.3 - Wi-Fi Passwords**

In this function, all what the user needs to do here to get all of the previously stored SSID and their passwords is to click on "Get passwords".

All of this information are then outputted on the screen and stored in the log file as well as shown in the figure below:

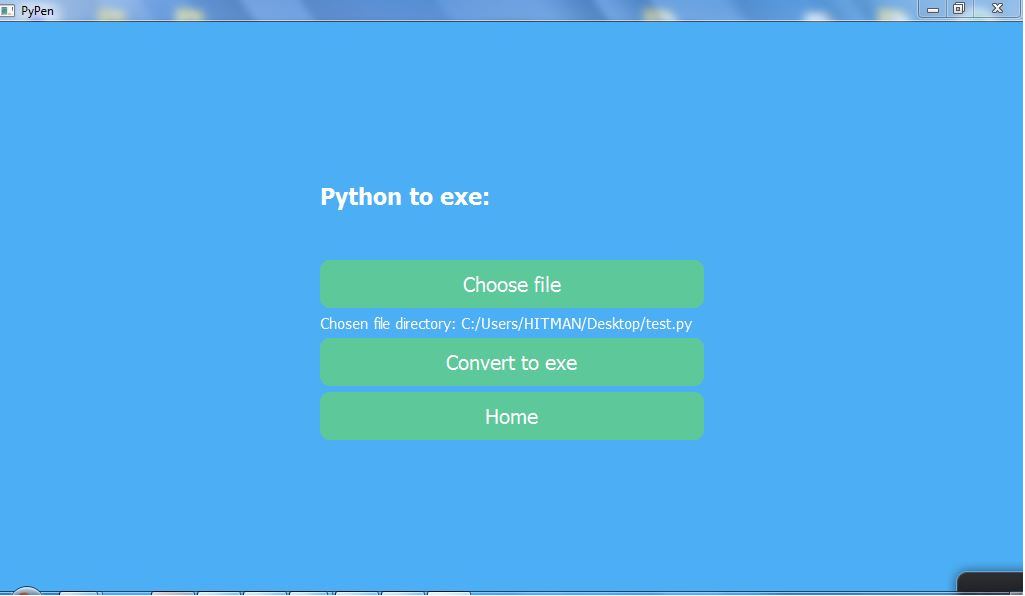


**(Figure 7.4 - Wifi-Passwords)**

**7.2.4 - Python to Exe**

This particular function requires the user to navigate his directory to select his python script file he wants to convert to executable fille by clicking "Choose File" and once the file is chosen for converting the user must click on "Convert to exe".

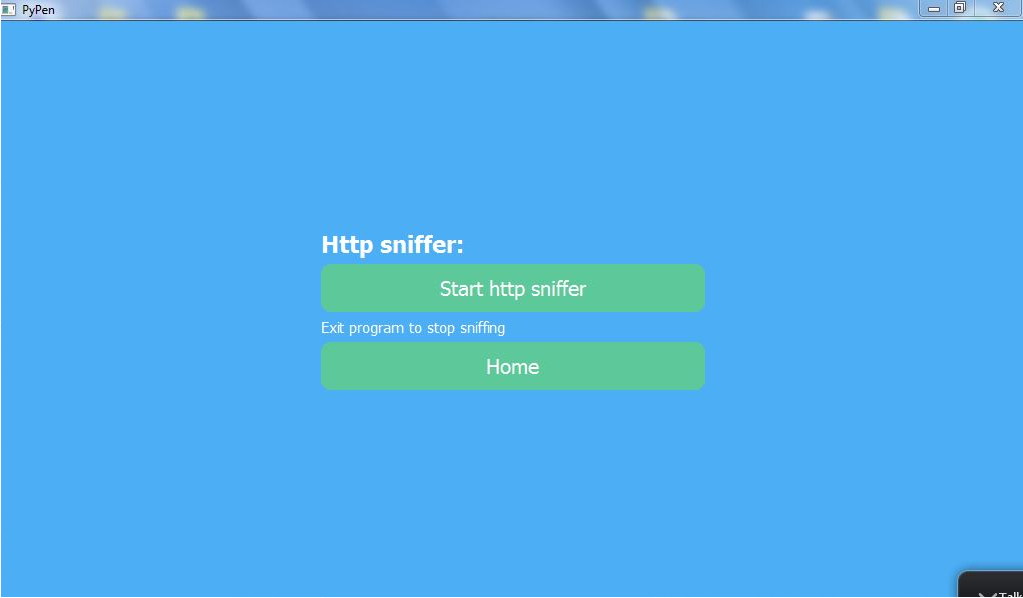
The output or the conversion result is stored in the log file in a folder named "dist".



**(Figure 7.5 - Python to exe)**

**7.2.5 - HTTP Sniffer**

In this function the user can intercept and sniff all unsecure HTTP traffic like credentials simply by starting and launching this function, this traffic will be displayed on the screen as well as stored in a separated log file for later use**.**



**(Figure 7.6 -HTTP Sniffer)**

Finally in the upcoming last chapter, we conclude the aim of our project, what is the future work of PyPen, and how are we going to further more improve its functionality; to overcome the obstacles and limitation users are facing in penetration testing.

**CHAPTER 8**

**Conclusion and future work**

8.1 - Conclusion 30

8.2 - Future Work 30

**8.1 - Conclusion**

We aimed to build PyPen application considering many important reasons to do so. We wanted to give the user the ability to perform countless pen testing functions with most comfortable and easy way possible, through-out GUI interfaces. Using Python language to build this project was the right decision, as Python supports wide range of libraries and provides advanced control capabilities to the user. It also has text processing capabilities and secure integration, which improve functionality. Also for making the application a cross-platform as most operating systems come with python pre-installed on it. Python works on Linux, Microsoft, MacOS, and many more.

**8.2 - Future Work**

Our objective is to build a cross-platform penetration testing application, with wide range of attacks. Unfortunately, due to time limitation at the present we have a fully functional well coded application that runs on windows platform only. For sure the application will be available for multi-platform in the near future. More functions will be added to the list of attacks. More graphical interfaces will be implemented to increase the interaction between the user and the application.

**References:**

[1] C. J. Benitez, “Find the WiFi Password in Windows 10 Using CMD,” Help Desk Geek, 30-Mar-2021. [Online]. Available: https://helpdeskgeek.com/how-to/find-the-wifi-password-in-windows-10-using-cmd/. [Accessed: 27-May-2021]

[2] pyinstallerpyinstallerpyinstaller, “pyinstaller,” PyPI, 17-Apr-2021. [Online]. Available: https://pypi.org/project/pyinstaller/. [Accessed: 27-May-2021]

[3] subprocess subprocesssubprocess , “subprocess - Subprocess management¶,” subprocess - Subprocess management - Python 3.9.5 documentation, 01-Jan-2021. [Online]. Available: https://docs.python.org/3/library/subprocess.html. [Accessed: 27-May-2021]

[4] A. A. Rockikz, “How to Make a Network Scanner using Scapy in Python,” Python Code, 28-Jul-2019. [Online]. Available: https://www.thepythoncode.com/article/building-network-scanner-using-scapy. [Accessed: 27-May-2021]

[5] P. P. I. PyPI, “scapy,” PyPI, 19-Apr-2021. [Online]. Available: https://pypi.org/project/scapy/. [Accessed: 27-May-2021]

[6] P. P. I. PyPI, “keyboard,” PyPI, 23-Mar-2020. [Online]. Available: https://pypi.org/project/keyboard/. [Accessed: 27-May-2021]

[7] A. Rockikz, “How to Sniff HTTP Packets in the Network using Scapy in Python,” Python Code, 25-Aug-2019. [Online]. Available: https://www.thepythoncode.com/article/sniff-http-packets-scapy-python. [Accessed: 27-May-2021]

[8] python.org python.org, “re - Regular expression operations¶,” re - Regular expression operations - Python 3.9.5 documentation, 27-Jul-2009. [Online]. Available: https://docs.python.org/3/library/re.html. [Accessed: 27-May-2021]

[9] “Find, install and publish Python packages with the Python Package Index,” PyPI, 01-Mar-2020. [Online]. Available: https://pypi.org/. [Accessed: 27-May-2021]

[10] P. Q. PyQt5, “PyQt5 Reference Guide¶,” PyQt5 Reference Guide -PyQt 5.7 Reference Guide. [Online]. Available: https://doc.bccnsoft.com/docs/PyQt5/. [Accessed: 27-May-2021]