SpyD: An Automated Testing Tool

BACHELOR OF ENGINEERING

IN INFORMATION TECHNOLOGY

By

Abhishek Pandey

Harshvardhan Dubey

Pratik Varma

Sumit Awinash

Under The Guidance Of

Prof. Reena Kothari

Assistant Professor, Department of Information Technology

Shree Rahul Education Society's (Regd.)

SHREE L.R. TIWARI College of Engineering

(Approved by AICTE, Government of Maharashtra and Affiliated to University of Mumbai)

ISO 9001:2008 Certified.

Near Commissioner's Bungalow, Kanakia Park, Mira Road (E), Thane-401107, Maharashtra.

Department of Information Technology

2018-19

A Project Report On

SpyD: An Automated Testing Tool

Submitted to Mumbai University

In partial fulfillment for the award of the degree of

BACHELOR OF ENGINEERING

IN INFORMATION TECHNOLOGY

By

Abhishek Pandey

Harshvardhan Dubey

Pratik Varma

Sumit Awinash

Under The Guidance Of

Prof. Reena Kothari

Assistant Professor, Department of Information Technology

Shree Rahul Education Society's (Regd.)

SHREE L.R. TIWARI College of Engineering

(Approved by AICTE, Government of Maharashtra and Affiliated to University of Mumbai)

ISO 9001:2008 Certified.

Near Commissioner's Bungalow, Kanakia Park, Mira Road (E), Thane-401107, Maharashtra.

Department of Information Technology

2018-19

Shree Rahul Education Society's (Regd.)
SHREE L.R. TIWARI College of Engineering
(Approved by AICTE, Government of Maharashtra and Affiliated to University of Mumbai)
ISO 9001:2008 Certified.

Near Commissioner's Bungalow, Kanakia Park, Mira Road (E), Thane-401107, Maharashtra.

UNIVERSITY OF MUMBAI

CERTIFICATE

This is to certify that the project titled "**SpyD: An Automated Testing Tool**" has been completed under our supervision and guidance by the following students:

Abhishek Pandey

Harshvardhan Dubey

Pratik Varma

Sumit Awinash

In the partial fulfillment of degree of Bachelor of Engineering in Information Technology branch as prescribed by the University of Mumbai during the academic year 2018-2019. The said work has been assessed and is found to be satisfactory.

Signature of the Internal Examiner

Name: Prof. Reena Kothari
Date: _____

Signature of the External Examiner

Name: _____
Date: _____

Signature of the H.O.D.

Name: Prof. Deepali Patil
Date: _____

Signature of the Coordinator

Name: Prof. Aarti Puthran
Date: _____

Signature of the Principal

Name: Dr. S. Ram Reddy

Date: _____

DECLARATION

We do hereby declare that the work embodied in the project entitled "**SpyD: An Automated Testing Tool"** is the outcome of our original work under the guidance and supervision of **Prof. Reena Kothari**. This piece of work or any part of it has not been submitted previously for the award of any other degree, diploma, or other title to any other institution.

We also declare that this written submission represents our ideas in our own words and where others ideas or words have been included. We have adequately cited and referenced the original sources. We also declare that we have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in our submission. We understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

Abhishek Pandey Date: The _____ April, 2019
Roll No.: 33
Exam Seat No.:

Harshvardhan Dubey
Roll No.: 07
Exam Seat No.:

Pratik Varma
Roll No.: 67
Exam Seat No.:

Sumit Awinash
Roll No.: 02
Exam Seat No.:

ACKNOWLEDGEMENT

A few sublime human experiences defy expressions of any kind, and a feeling of true gratitude is one of them. I, therefore, find words quite inadequate to express my indebtedness to my Guide **Prof. Reena Kothari** her virtuous guidance, encouragement and help throughout this work. Their deep insight into the problem and the ability to provide solutions has been immense value in improving the quality of project at all stages. This experience of working with them shall ever remain a source of inspiration and encouragement for me.

I express my thanks to **Prof. Deepali Patil**, HOD (IT), SLRTCE, Mira Road , for extending her support that she gave truly help the progression of the project work. I express my thanks to **Prof. Aarti Puthran**, Asst. Prof., SLRTCE, Mira Road for extending his support.

My sincere thanks to **Dr. S. B. Singh**, Director and Dr**. S. Ram Reddy**, Principal, SLRTCE, Mira Road for providing me the necessary administrative assistance in the completion of the work.

I am extremely grateful to the celebrated authors whose precious works have been consulted and referred in my project work. I also wish to convey my appreciation to my friends who provided encouragement and timely support in the hour of need.

Special thanks to my Parents whose love and affectionate blessings have been a constant source of inspiration in making this a reality.

All the thanks are, however, only fraction of what is due to Almighty for granting me an opportunity and the divine grace to successfully accomplish this assignment.

Abhishek Pandey

Harshvardhan Dubey

Pratik Varma

Sumit Awinash

TABLE OF CONTENTS

ABSTRACT

As people are realizing the importance of cyber security, the price for hacking tool such as Pineapple is increasing rapidly. One Pineapple device can cost up to Rs. 21000+Tax+Shipping charges. Due to high cost people do not prefer buying such a product which hamper their security as an organization. By building SpyD we can provide a similarly effective product at a very considerably lower price which will help the organization to manage the security. SpyD is a automated network penetration device/bot which will be able to perform all major operations through the means of an automated script with built in failsafe which can be performed be any professional ethical hacker. This will ensure that the routine jobs are performed more precisely as compared to a hired professional, thus, eliminating the threat of any man-made errors. Additionally, SpyD will automate the work of softwares like Nmap, Aircrack-NG, Karma, Tcpdump etc.

LIST OF FIGURES

Chapter 1

Introduction

Chapter 1

Introduction

This chapter will introduce the reader with SpyD, which is used as a penetration testing tool and a hacker device. It will light up the topics like the description of the project and the former formulation of the problem behind it as well as what motivated the makers of the project to take a decision to make this project and its related problem solutions and thus covering up the scope of the project.

About

As people are realizing the importance of cyber security, the price for a testing and hacking tool such as Pineapple is increasing rapidly. One Pineapple device can cost a lot in Indian markets. Due to high cost people do not prefer buying such a product which hamper their security as an organization. By building SpyD we can provide a similarly effective and completely automated product at a very considerably lower price which will help the organization to manage the security and reduce human efforts.

By building SpyD we aim to provide a product similar to current pen test tools like PineAP at a very considerably lower price along with packet tracing which will help the organization to manage their security.

1.1 Description

SpyD is a follow up of the established Wi-Fi Pineapple and adds various functionalities like complete automation, packet monitoring, etc. The secondary objective is to lower the cost of the device to target smaller organizations and start-ups.

In this project we aim to overcome the shortcomings of conventional testing tools like rutabaga or PineAP that are need for a professional, high cost, manual work etc.

There are number of product available in the market with different vendors and it creates a bit confusion in the mind of consumer that which component to buy. Therefore it becomes necessary to design effective system to summarize the pros and cons of product characteristics so that consumer can quickly find their favorable product so as to develop an efficient and cost effective security testing tool. We have developed a system which performs

tests and automates the working, thus, removing a middle-man which was required to be a testing professional.

The proposed system will help the users or organizations to perform their tests locally and improve the network's efficiency. From a hacking point of view, they can perform hacks with the help of SpyD in a much better and feasible way.

Finally, a fully automated router/modem which is the SpyD will be available at the remote locations to access the internet and perform the desired tests to improve the network and eventually the system security.

1.2 Problem Formulation

Nowadays, the current penetration testing and hacking tools such as conventional Wi-Fi PineAP are being modified to be made more advanced due to current security needs and policies. This modification has led to a significant increase in the cost of such tools and more man power is required to carry out such tasks by professionals.

With our automated network penetration device we'll be able to perform major operations through the means of automated scripts with built-in failsafe which can be performed by any professional ethical hacker. This ensures that routine jobs are performed more precisely as compared to any hired professional, thus, eliminating the threat of man-made errors.

1.3 Motivation

The overwhelming demand of security in local and small organizations due to daily security attacks has motivated us to develop this project. In addition to bring the usability of testing tools has made us to think to develop this project so as to simplify the process of testing to make the efficient penetration testing tool for the end user.

So we wished to design a system that could help the user to perform automated tests and monitor packets with only sitting on a PC. Also we wish to give the user the features that will enable them to be at ease when attempting to gain access to networks with performing tests.

1.4 Proposed Solution

We aim to provide a system that is an automated network penetration device which will perform all major operations through the means of an automated script with built-in failsafe which can be performed by any professional hacker, thus, providing automation at a lower cost and eliminating any man made errors at the same time.

SpyD aims at providing an automated solution towards the previously used network scanners and packet sniffers like Nmap, Aircrack-NG, and Tcpdump.

1.5 Scope of the project

The project aims to ease the process of Network security testing, so it will be easy for the users to perform tests to check for vulnerabilities as per the requirement. In many organizations, the testing tools which are used are the conventional ones which are usually operated and worked upon by testing professionals. This isn't a feasible option for end users and the organizations that are relatively smaller than those organizations. Through this proposed system, the end users as well as small scale organizations can perform tests and attacks to check for any vulnerability in their system and remove those flaws. The project uses various firmwares from the significant domain Network security like OpenWRT, LuCI, and operating systems like Python, etc to perform attacks like Man-in-the-middle, Framereplay, and fake Frame generation. All these functions are clustered in one chipset connected to the network to perform tests or attacks based on user's requirements.

.

Chapter 2

Review of Literature

Chapter 2
Review of Literature

Here we will elaborate the aspects like the literature survey of the project and what all projects are existing and been actually used in the market which the makers of this project took the inspiration from and thus decided to go ahead with the project covering with the problem statement.

2.1 Literature Survey

2.1.1 Cybergrenade: Automated Exploitation of Local Network machines via Single Board Computers by Anurag Akkiraju, David Gabay, Halim Burak Yesilyurt, Hidayet Aksu, Selcuk Uluagac

In the reference paper, a defensive cyber security framework device called Cyber grenade automating various penetration testing tools to sequentially exploit machines connected to a Single local network, all underneath a single application running on a Single-Board Computer (SBC). The paper further describes the advantage of the SBC's unique capabilities in a way that manual exploitation simply cannot match. Currently, while many SBCs are being used in research as exploitation tool-kits, the current state of automation of the processes associated with exploitation leaves much to be desired. While the reference paper describes the device Framework, it can be used as a guideline for future research automating the exploitation process. The Device allows tools such as Nmap, OpenVAS, and Metasploit tools to be automatically utilized under one framework. The Paper is divided into five sub parts

I. Introduction
II. Related Work
III. System Design and Implementation
IV. Experiments
V. Conclusion

2.1.2 Computer Network Security and Technology Research by Fan Yan, Yang Jian-wen and Cheng Lin

The rapid development of computer network system brings both a great convenience and new security threats for users. Network security problem generally includes network system security and data security. Specifically, it refers to the reliability of the network

system, confidentiality, integrity, and availability of data information in the system. Network security problem exists through all the layers of the computer network, and the network security objective is to maintain the confidentiality, authenticity, integrity, dependability, availability, and audit-ability of the network. This paper introduces the network security technologies mainly in detail, including authentication, data encryption technology, firewall technology, intrusion detection system (IDS), antivirus technology and virtual private network (VPN). Network security problem is related to every network user, so we should put a high value on network security, try to prevent hostile attacks and ensure the network security.

The penetration testing devices available in foreign countries and Multinational companies are PineAP or rutabaga. These aren't feasible because of the necessity of a professional user/tester and the high cost and maintenance. This leads to the need of an alternative in the local market feasible for all users.

2.2 Problem Statement:

Nowadays, the current penetration testing and hacking tools such as conventional Wi-Fi PineAP are being modified to be made more advanced due to current security needs and policies. This modification has led to a significant increase in the cost of such tools and more man power is required to carry out such tasks by professionals.

With our automated network penetration device we'll be able to perform major operations through the means of an automated script with built-in failsafe which can be performed by any professional ethical hacker. This ensures that routine jobs are performed more precisely as compared to any hired professional, thus, eliminating the threat of man-made errors.

Chapter 3

System Analysis

Chapter 3
System Analysis

System Analysis will cover the topics like the Functional, Non-Functional and the specific requirements of the project and touching all the software and the hardware requirements as well.

3.1 Requirements

3.1.1 Functionality

3.1.1.1 Provide comprehensive network details.

SpyD will display detailed information of the available access network.

3.1.1.2 Provide Network Access

SpyD will provide detail network access to the user.

3.1.1.3 Provide packet monitoring

SpyD will monitor the incoming/outgoing packets through the network.

3.1.1.4 Provide creating of pseudo access point

SpyD will create a pseudo access point that will exploit the vulnerability of the victim.

3.1.1.5 Provide packet injection.

SpyD allows pseudo frame generation that is injected into the network.\

3.1.1.6 Provide penetration testing.

SpyD allows thorough self-evaluation of its current network to find vulnerabilities.

3.1.1.7 Provide various attacks.

SpyD allows attacks like Man in the Middle (MITM), Arp Spoofing, etc.

3.1.1.8 Generate Reports Related to Packets.

SpyD will generate detailed reports for specific packets.

3.1.2 Usability

3.1.2.1 Graphical User Interface

SpyD provides the Luci web interface for further development

Luci Web Interface is a Graphical User Interface provided by SpyD

3.1.3 Reliability & Availability

SpyD is reliable since it monitors packets flowing in the network actively and perform operations as specified by the user by using standardized algorithm used by professional ethical hackers to test the network.

3.1.4 Performance

SpyD is a device that has to be physically connected or be within proximity of the network.
The performance will depend on the hardware / network of the device.

3.1.5 Security

3.1.5.1 Data Transfer

The device performs data transfer between files present at the multiple nodes within the connected network in an encrypted manner. So, no eavesdropper can intercept the data transfer between the nodes.

3.1.5.2 Data Storage

SpyD uses extRoot to store data on the device. The device storage capability depends on the model and version of SpyD, where the minimum capacity will be 16 Giga Bytes.

3.1.6 Interfaces

There are many types of interfaces as such supported by our SpyD namely; User Interface, Software Interface and Hardware Interface.

The protocol used shall be HTTP, FTP, SMTP, and TCP/IP.

The Port number used will be varied as per the requirement.

3.1.6.1 User Interfaces

The device is accessible via a command line interface that is either via “ssh” or “telnet”. In the latest version access via telnet is not possible due to encryption hence ssh is preferred. On linux ssh is used and on windows puTTy is used.

3.1.6.2 Hardware Interfaces

CPU	Ram	Flash	Network	USB	Serial	JTag
Atheros AR9331@400MHz	32MiB	4MiB	1 x 100MBit	1 x 2.0	Yes	No

- SoC: Atheros AR9330 rev 1
- 802.11 b/g/n 150 Mbps
- Powered via USB B-Mini (5 Volts)
- Tiny form factor
- 5.7 cm x 5.7 cm PCB
- 6.7 cm x 7.4 cm x 2.2 cm case
- RJ45 I/O port

3.1.6.3 Software Interfaces

LuCI is installed as a 'meta package' which installs several other packages by having these defined as a dependency. Notably, it installs the uHTTPd web server, configured for use with LuCI. The dependent packages are the following (see the LuCI technical reference for more information):

- uhttpd
- uhttpd-mod-ubus
- luci-mod-admin-full
- luci-theme-bootstrap
- luci-app-firewall

- luci-proto-core
- luci-proto-ppp
- libiwinfo-lua

In case you want to use uHTTPd for the web interface there is little configuration necessary as uHTTPd is configured with CGI to make LuCI work with the Lua interpreter. By default this is organized as follows. By default /www is the standard document root. Thus, by requesting this docroot (by pointing your browser to the devices IP address) an index file such as index.html is searched for (per uHTTPdsettings). The file /www/index.html (installed with LuCI) is prepared such that when requested, it redirects you to /cgi-bin/luci, which is the default CGI gateway for LuCI. This is just a script, which basically calls Lua at /usr/bin/lua. uhttpd is configured by default to load pages as CGI in the /cgi-bin path, and thus starts serving these pages with the /cgi-bin/luci script.

It is also possible to run LuCI with Lua as an embedded process. uhttpd supports this; see the corresponding section of the uHTTPd Web Server Configuration article on the UCI configuration of uhttpd.

3.1.6.4 Communications Interfaces

LuCI web interface is used for the same.

3.2 Use case diagram of the proposed system

A use case is a methodology used in system analysis to identify, clarify, and organize system requirements. There are three actors which is customer, system, and the authentication system. The customer can sign up and login which will be authenticated by authentication system and can add items and will get final base rating for assembled System.

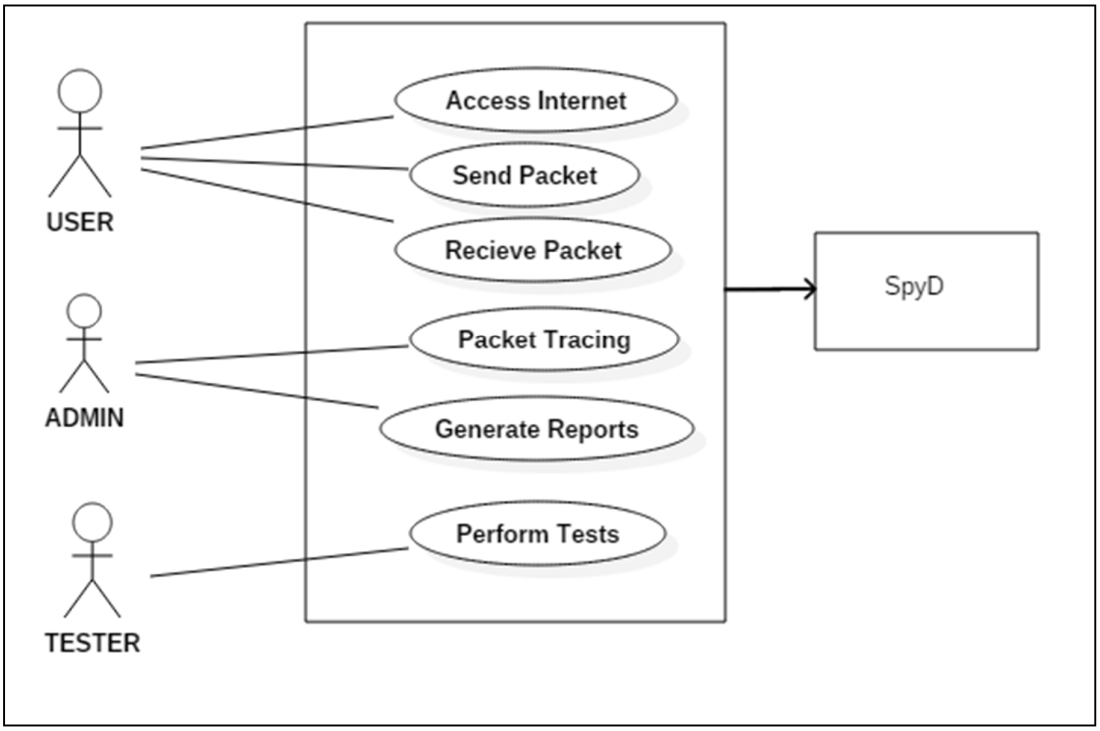


Figure 3.2: Use Case diagram for SpyD

Chapter 4

Analysis Modeling

Chapter 4
Analysis Modeling

All the aspects of the proposed system will be covered in this chapter in the diagrammatic manner and provides the detailed manner of the system.

4.1 Class Diagram

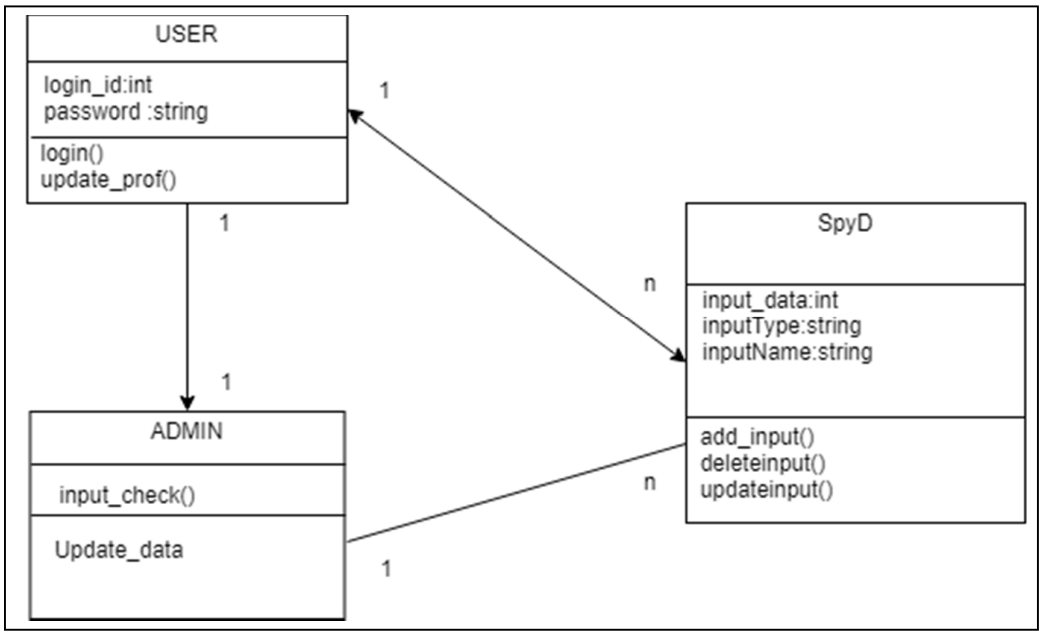
The class diagram is a static diagram. It represents the static view of an application. Class diagram is not only used for visualizing, describing and documenting different aspects of a system but also for constructing executable code of the software application. The class diagram shows a collection of classes, interfaces, associations, collaborations and constraints. It is also known as a *structural diagram.*

The class diagram describes the attributes and operations of a class and also the constraints imposed on the system. The class diagrams are widely used in the modelling of object oriented systems because they are the only UML diagrams which can be mapped directly with object oriented languages.

So the purpose of the class diagram can be summarized as:

- Analysis and design of the static view of an application.
- Describe responsibilities of a system.
- Base for component and deployment diagrams.
- Forward and reverse engineering.

Here we have three classes viz. user , Admin and SpyD device .The user will have login details and would be able to search for the operation he wants to perform .Each operation have its unique ID, name, and functionality. The task of administrator is to keep on monitoring the system and updating the system catalog. Finally the SpyD will return the desired output.



4.2 Sequence Diagram

The Sequence Diagram models the collaboration of objects based on a time sequence. It shows how the objects interact with others in a particular scenario of a usecase. With the advanced visual modeling capability, we can create complex sequence diagram in few clicks. Besides, VP-UML can generate sequence diagram from the flow of events which you have defined in the use case description.

Sequence diagram below demonstrates the interaction between various entities of SpyD system like user, SpyD, OpenWRT, and Tester in a sequential manner.

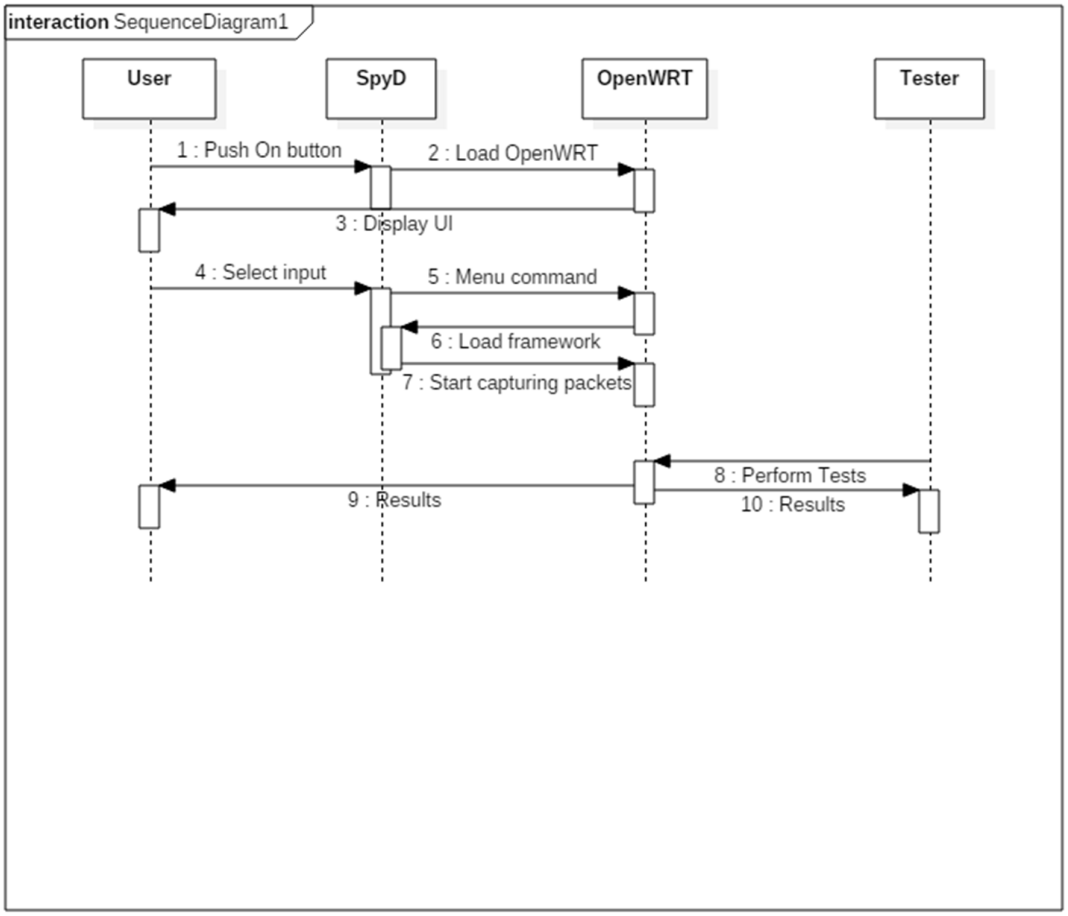


Figure 4.2: Sequence diagram for SpyD

to describe the static deployment view of a system. People diagrams consist of nodes and their relationships.

The name *Deployment* itself describes the purpose of the diagram. Deployment diagrams are used for describing the hardware components where software components are deployed. Component diagrams and deployment diagrams are closely related.

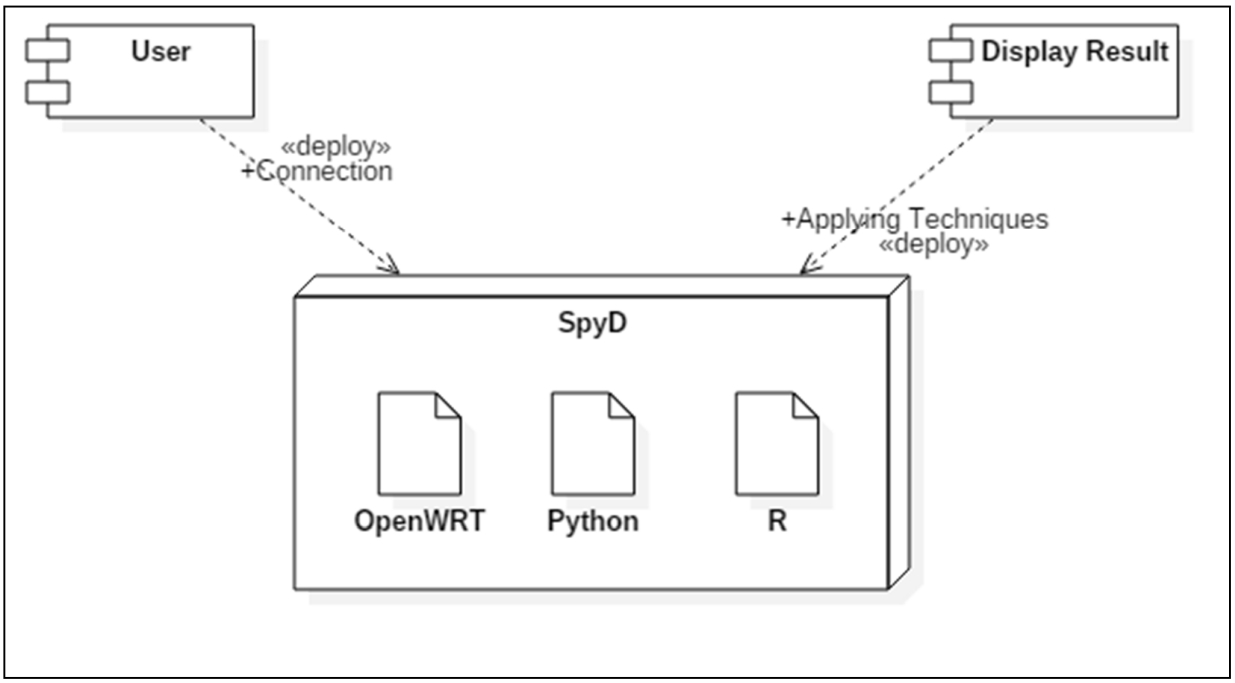


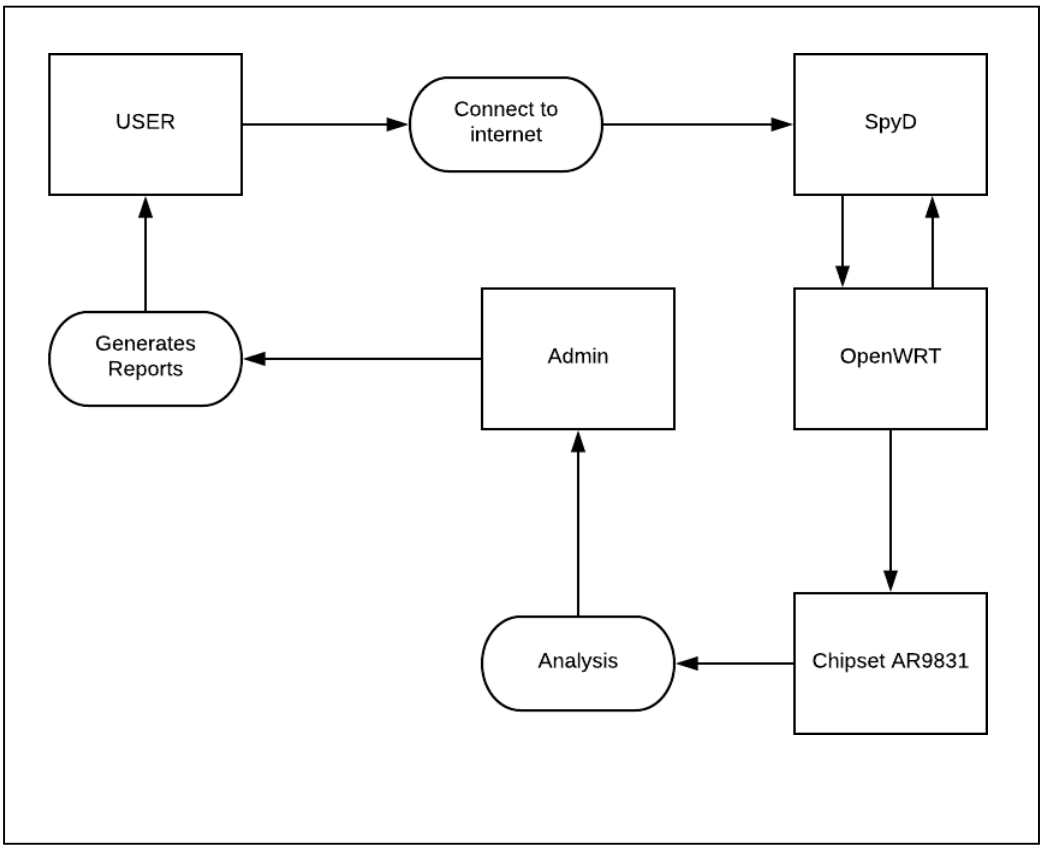
Figure 4.3: Deployment diagram for SpyD

4.4 Data flow diagram of the system

A data flow diagram (DFD) is a graphical representation of the "flow" of data through an information system, modelling its process aspects. A DFD is often used as a preliminary step to create an overview of the system, which can later be elaborated.

A two-dimensional diagram that explains how data is processed and transferred in a system. The graphical depiction identifies each source of data and how it interacts with other data sources to reach a common output.

Individuals seeking to draft a data flow diagram must (1) identify external inputs and outputs, (2) determine how the inputs and outputs relate to each other, and (3) explain with graphics how these connections relate and what they result in. This type of diagram helps business development and design teams visualize how data is processed and identify or improve certain aspect.


USER
Connect to internet
SpyD
Generates Reports
Admin
OpenWRT
Analysis
Chipset AR9831

Chapter 5

Design

Chapter 5
Design

Design will elaborate the step by step flow of SpyD based on user Benchmark thus giving up the detailed information as to the basic flow of the system.

5.1 Flowchart

Flowcharts are used in designing and documenting complex processes or programs. Like other types of diagrams, they help visualize what is going on and thereby help the people to understand a process, and perhaps also find flaws, bottlenecks, and other less-obvious features within it. There are many different types of flowcharts, and each type has its own repertoire of boxes and notational conventions. The two most common types of boxes in a flowchart are:

- A processing step, usually called *activity*, and denoted as a rectangular box
- A decision usually denoted as a diamond.

A flowchart is described as "cross-functional" when the page is divided into different swim lanes describing the control of different organizational units. A symbol appearing in a particular "lane" is within the control of that organizational unit. This technique allows the author to locate the responsibility for performing an action or making a decision correctly, showing the responsibility of each organizational unit for different parts of a single process.

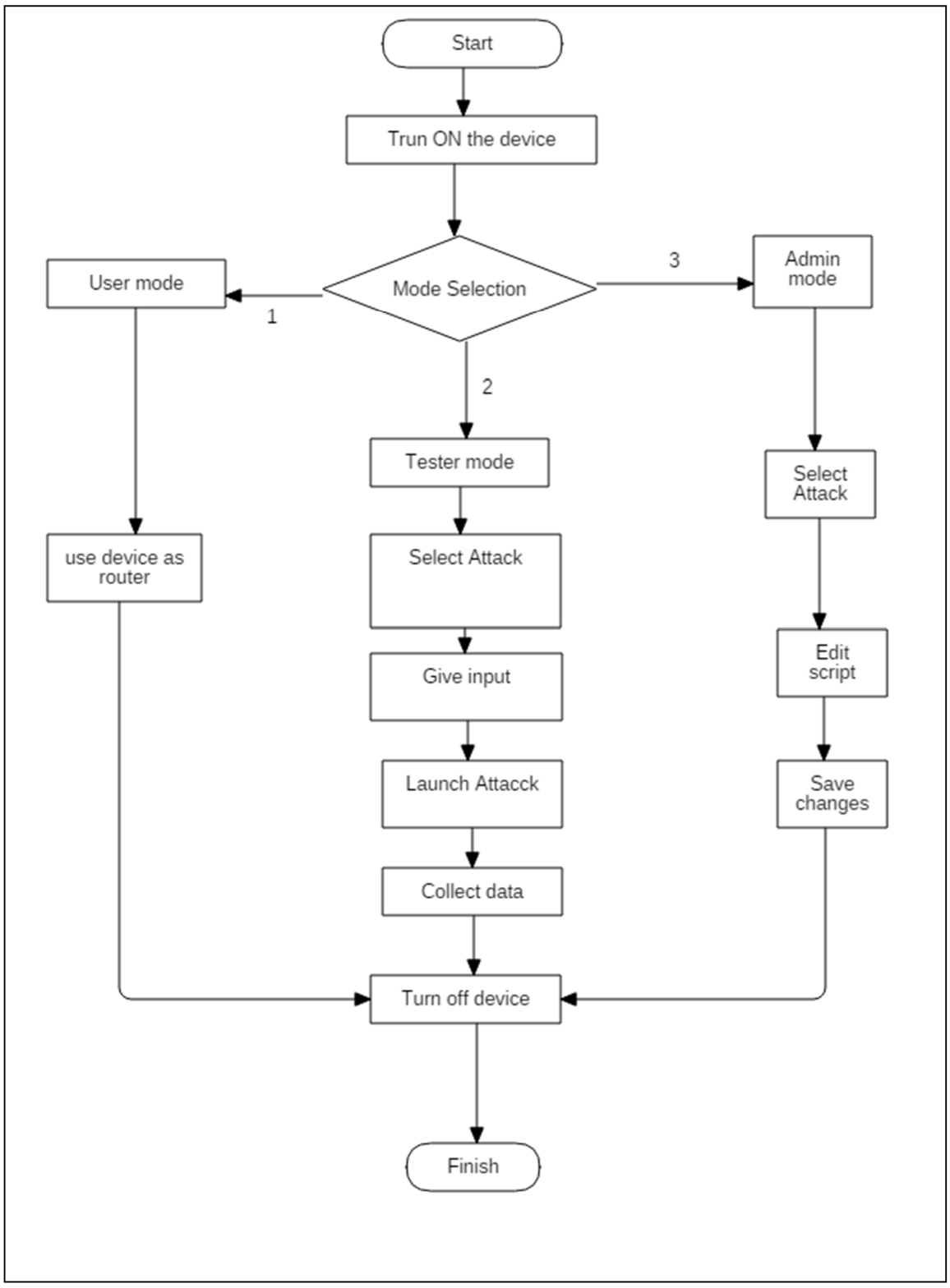


Figure 5.1: Flowchar for SpD

Chapter 6

Implementation Details

Chapter 6

Implementation Details

This chapter will give all the Implementation details and methods used to implement the Application in brief.

6.1 Implementation

We have designed following algorithm which would be followed throughout the implementation phase of this system

PenSec Algorithm

1. Start
2. Turn on the device
3. Select Mode of operation
    a: User mode
        Use device as a router
    b: Tester mode
        i: Select Attack
            -Man in the middle Attack
            -Frame replay attack
            -Fake frame generation attack
            -Create fake access point
            -More attacks will be added
        ii: Give appropriate input with respect to attack
        iii: launch attack
        iv: Collect appropriate data
        v: Finish or go back to step i.

    c: Admin mode
        i: Select Attack
            -Man in the middle Attack
            -Frame replay attack
            -Fake frame generation attack
            -Create fake access point
            -More attacks will be added
        ii: Make changes in script as per requirement
        iii: Finish or go back to step i.
4. Exit selected mode
5. Turn off the device or Change user mode
6. Stop

6.2 Implementation Screenshots

Flashing the Custom Firmware on SpyD

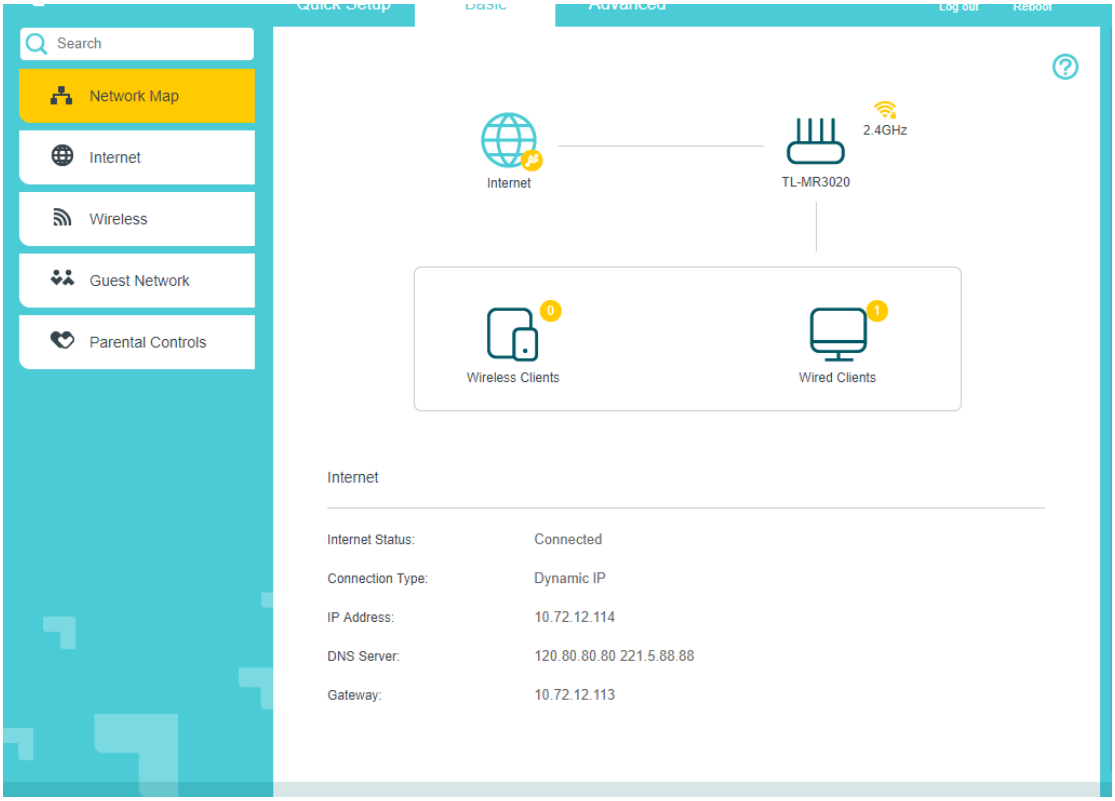


Figure 6.2.1: SpyD Stock User Interface

Figure 6.2.2: Custom Firmware for SpyD

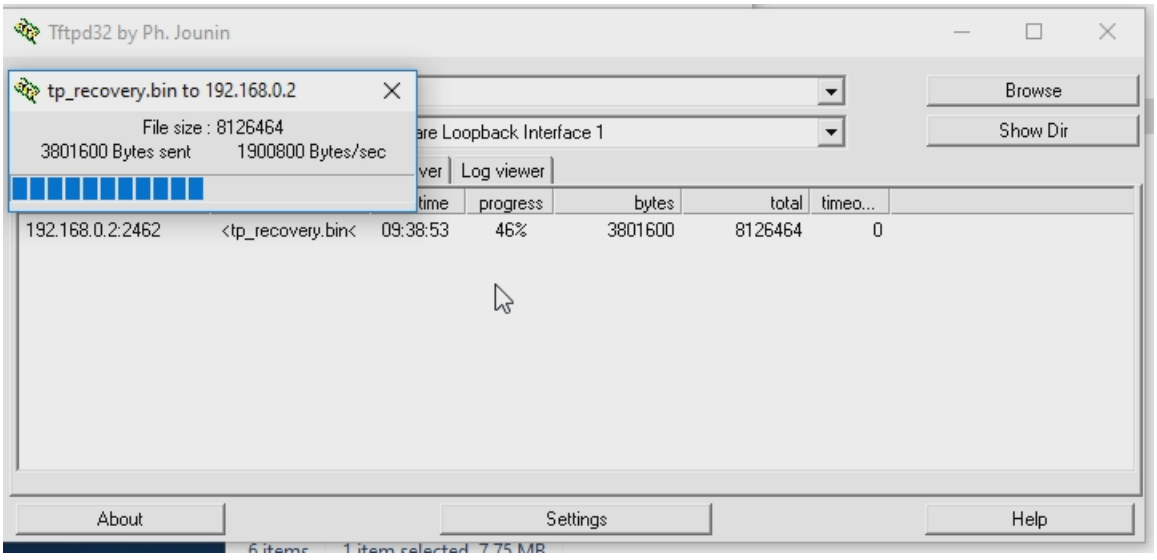


Figure 6.2.3: Custom Firmware Being Flashed On SpyD

The following screenshot shows all the network interfaces available by using the command ifconfig -a

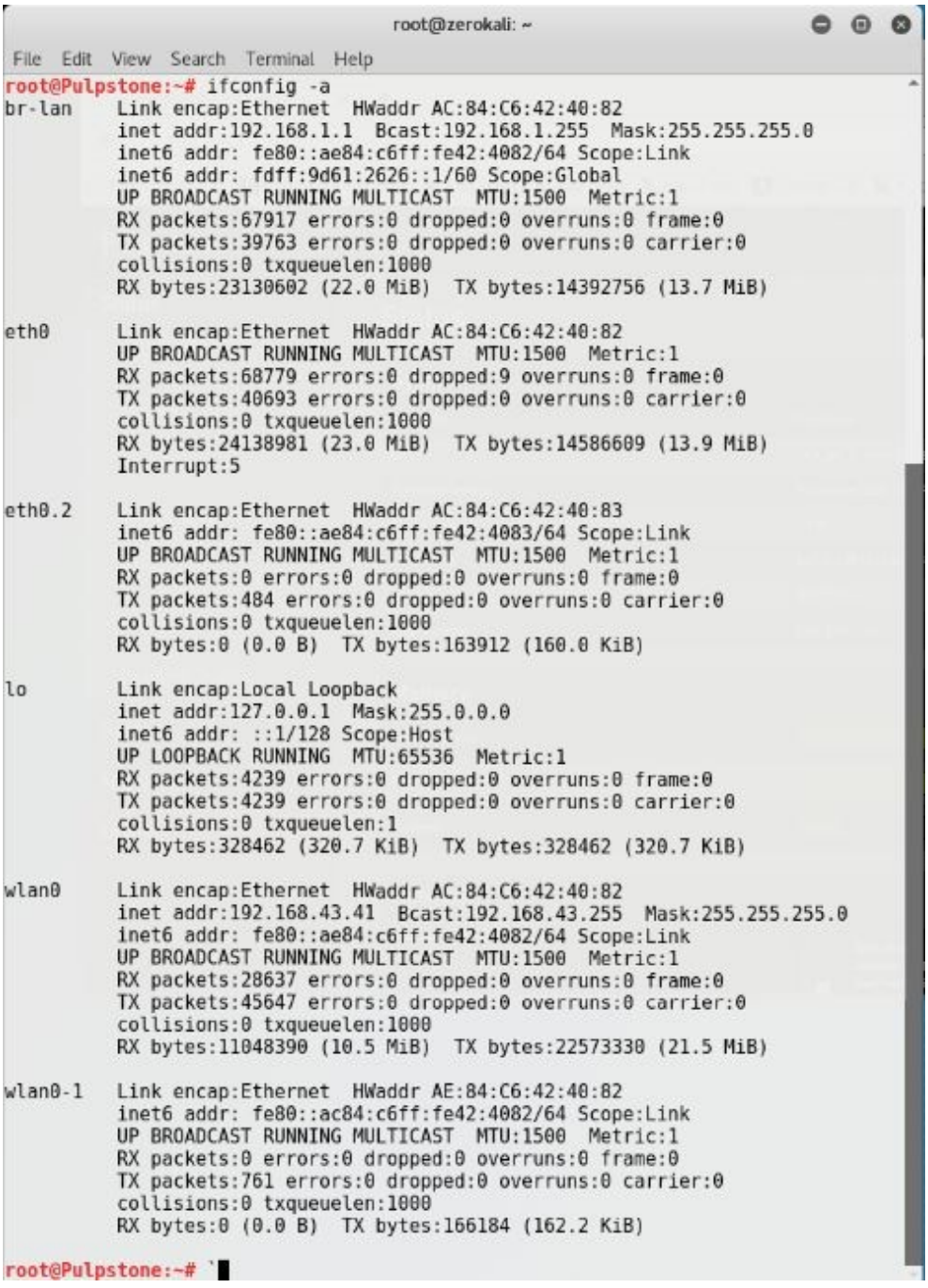


Figure 6.2.4: SpyD Network Interfaces

Updating SpyD repositories by using the command ‘opkg update’

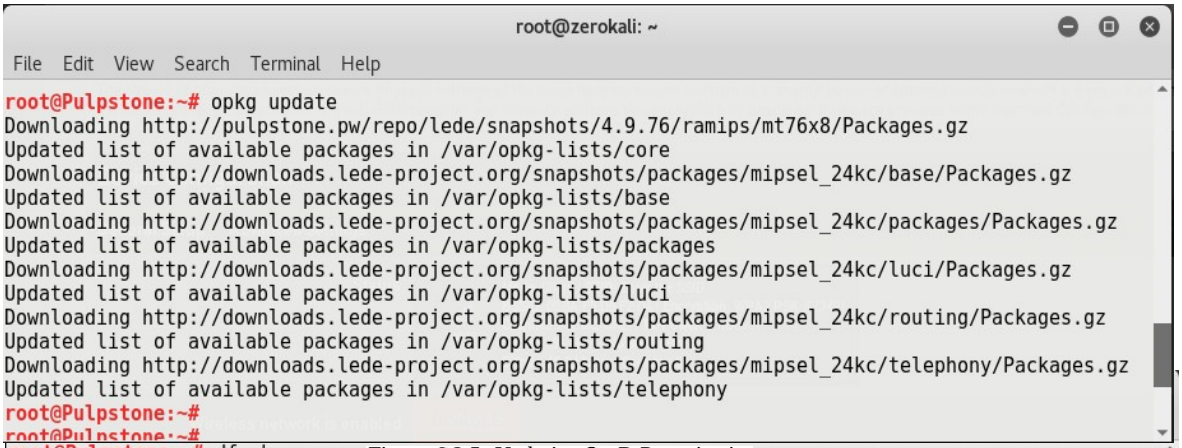


Figure 6.2.5: Updating SpyD Repositories

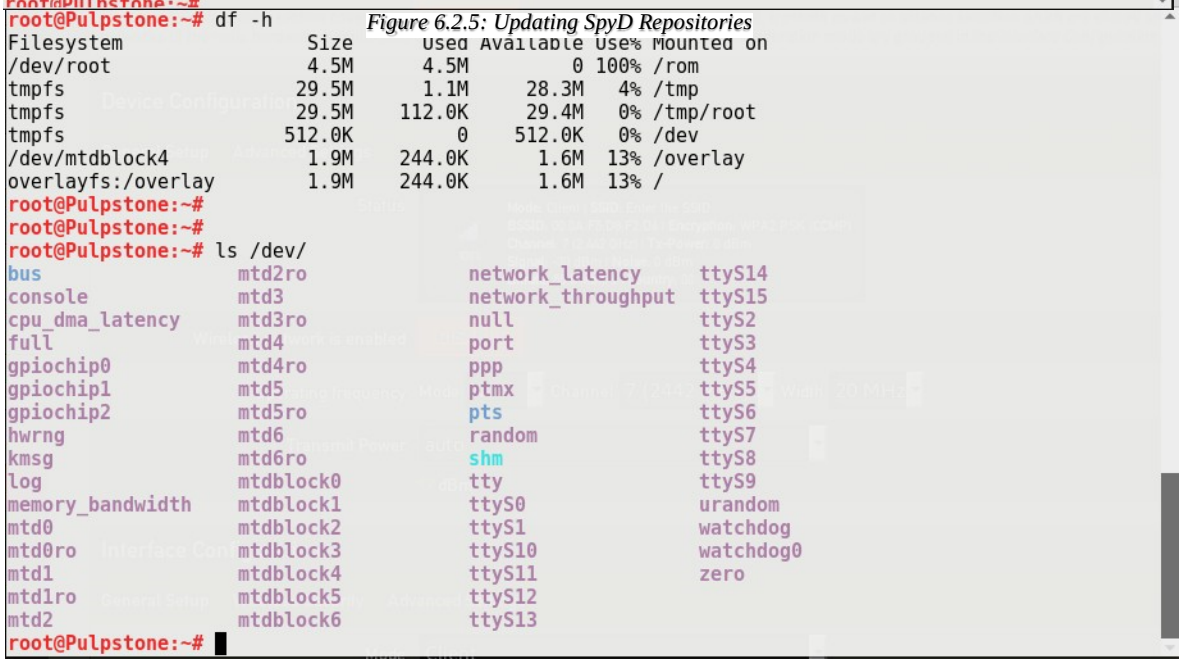


Figure 6.2.6: Browsing /dev/ directory of SpyD

We see the device directory. ‘df’ command displays the amount of disk space available on the file system containing each file name argument. Here the ‘-h’ option prints the size in in powers of 1024 that is in the form of megabytes.

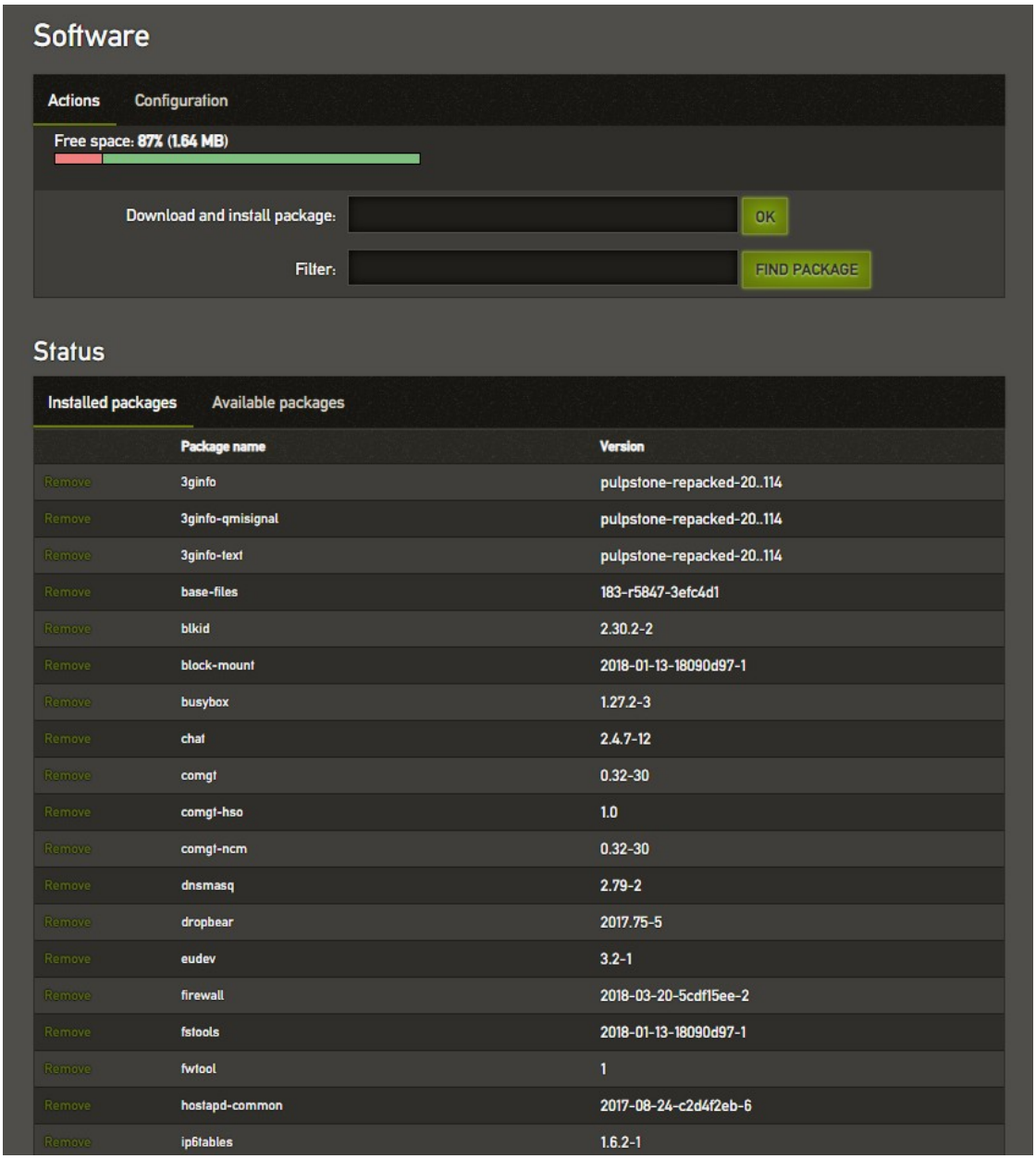


Figure 6.2.7: Installed Packagess on SpyD

LuCI web interface is used to check the storage available as well a list down all the installed packages.
It also gives us the option to search for packages and install them. Other packages are available at ‘Available Packages’

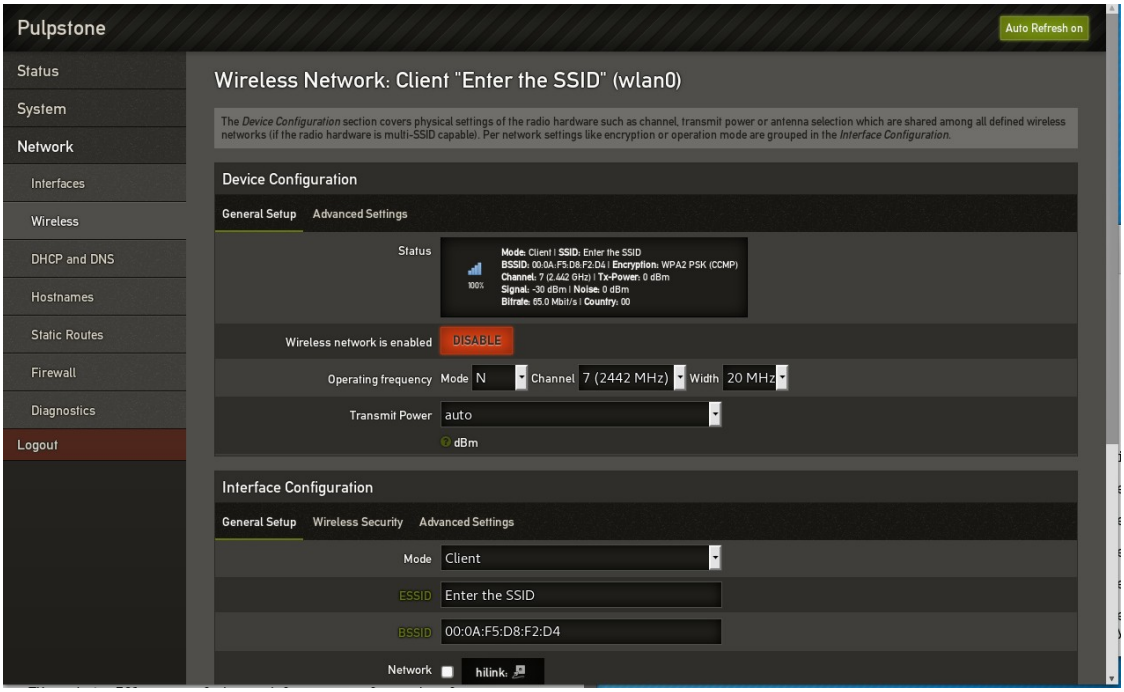


Figure 6.2.8: LuCI Wireless Network Interface of SpyD

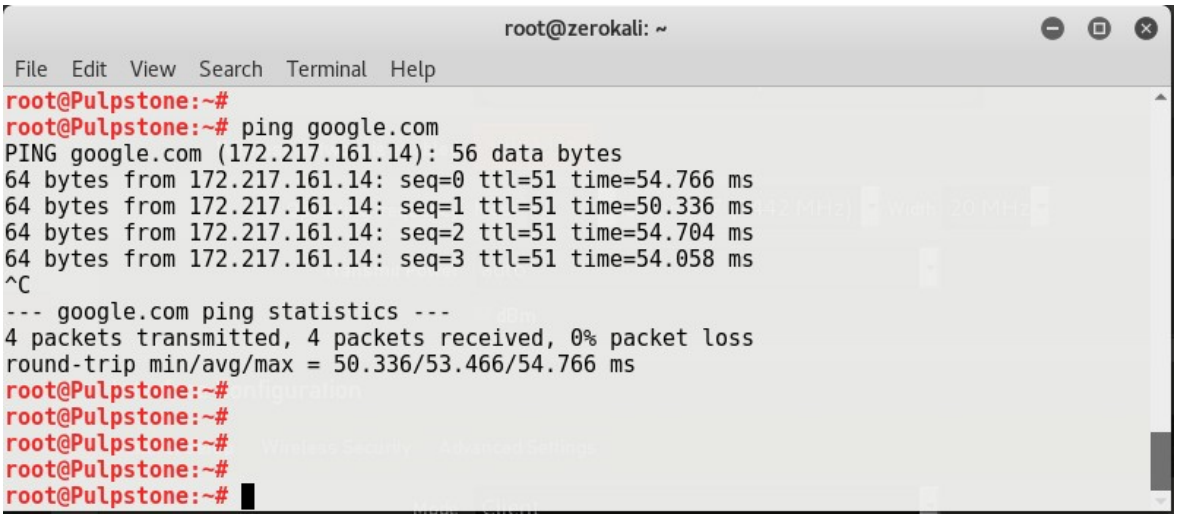


Figure 6.2.9: Testing SpyD Network Connection using ping

SpyD is successfully connected to “Enter the SSID”. It works in dual channel that is spyD is connected to a network as well as providing one. This is possible due to its multiple interfaces.

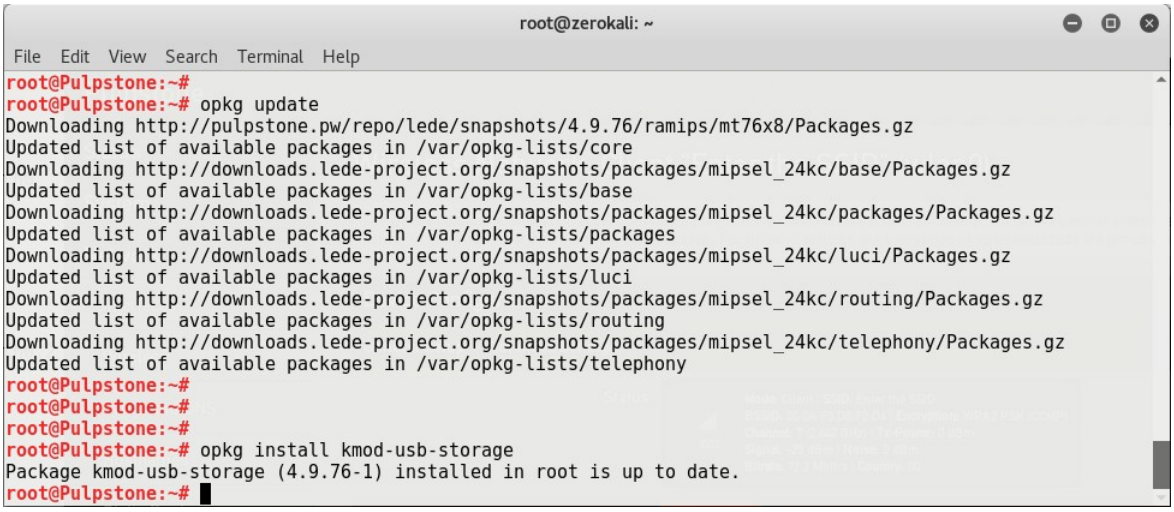


Figure 6.2.10: Installing kmod-usb-storage package on SpyD

Figure 6.2.11: Installing kmod-fs-vfat on SpyD

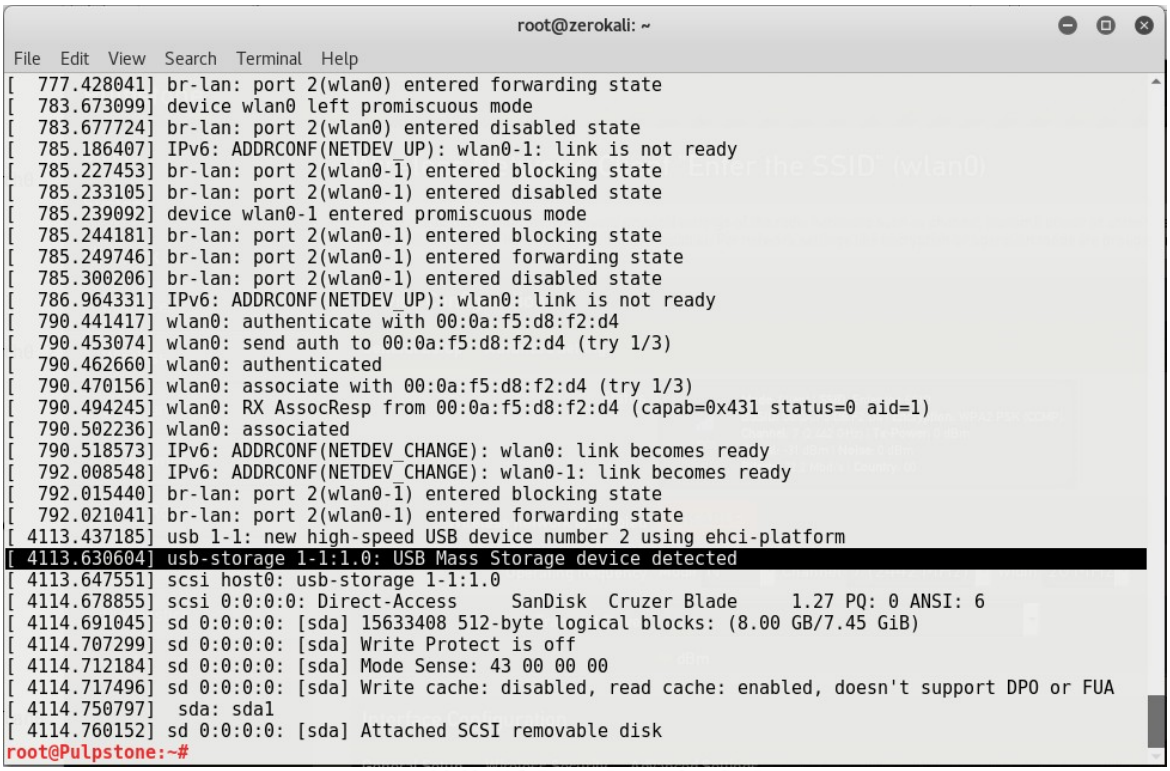


Figure 6.2.12: Usb Device Detected on SpyD

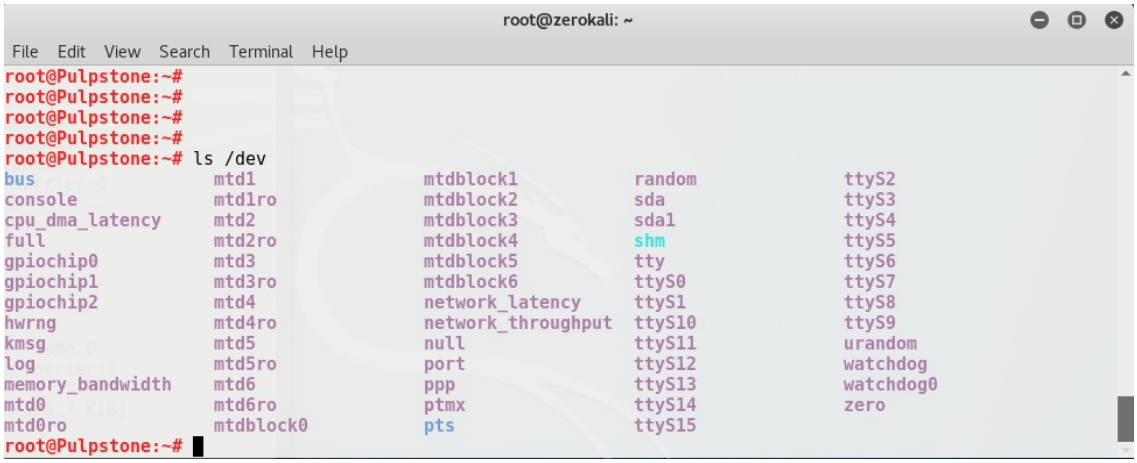


Figure 6.2.13: sda & sda1 in /dev/ on SpyD

USB storage is installed on SpyD. This will provide the storage needed for attacks, scripts, files, etc.

The main script (home) of the device interface is as follows:

Figure 6.2.14: SpyD: Homepage

Code:

```
#!/bin/bash
clear
echo "
-------------------------------------------------
|          #####                ######           |
|         #     # #####  #   # #     #           |
|         #       #    #  # #  #     #           |
|   #####  #####  #    #   #   #     # #####     |
|               # #####    #   #     #           |
|         #     # #        #   #     #           |
|          #####  #        #   ######            |
-------------------------------------------------"
echo ""
echo "Please select the SpyD Operating Mode
1: User Mode
2: Tester Mode
3: Admin Mode
4: Restart Device
5: Exit"
echo ""
read -p "Enter your Choice : " mode
case $mode in
1)sh ./UserMode.sh
;;
2)sh ./TesterMode.sh
;;
3)sh ./AdminMode.sh
;;
4)reboot
;;
5)exit
;;
*)echo "Please enter the correct option"
;;
esac
```

User mode:

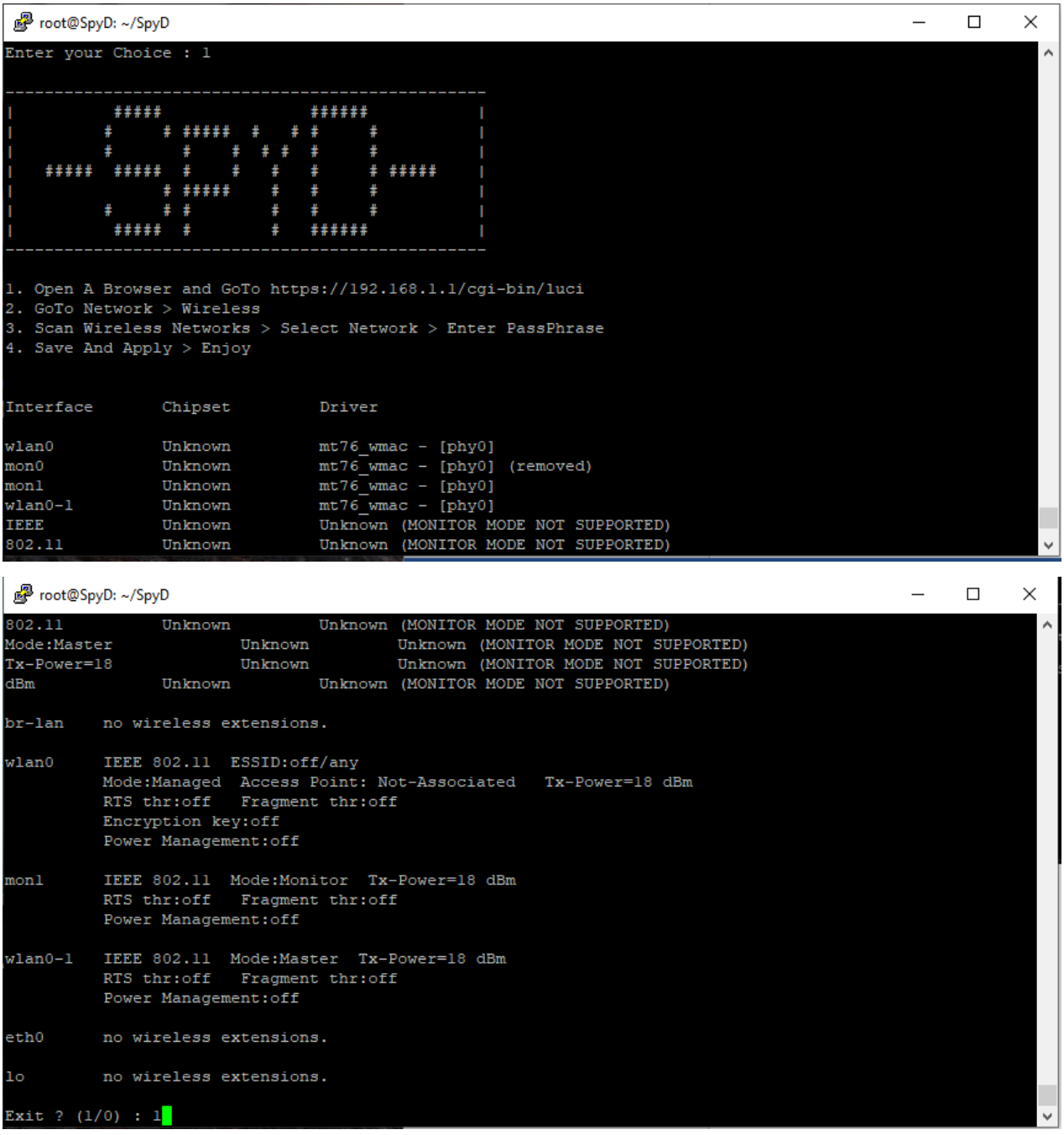


Figure 6.2.15: SpyD: User Mode

Tester mode:

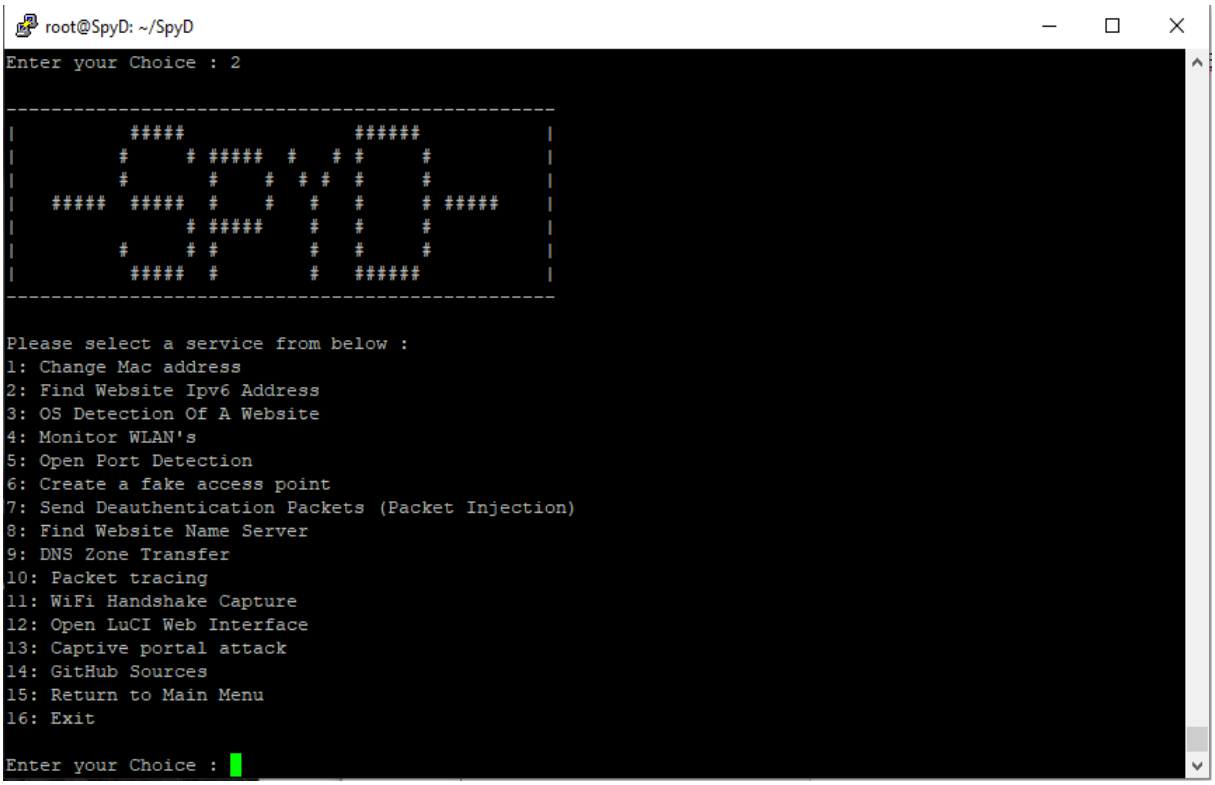


Figure 6.2.16: SpyD: Tester Mode

Admin mode:

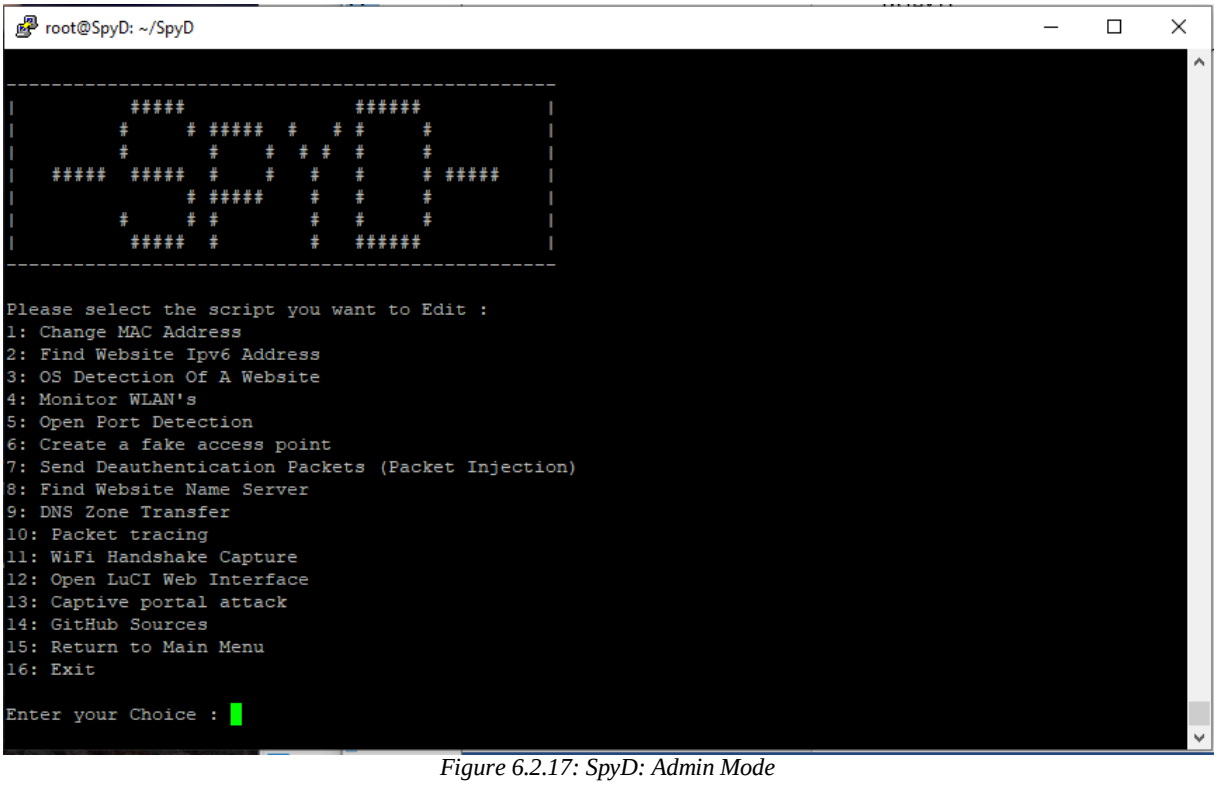


Figure 6.2.17: SpyD: Admin Mode

Implementation files and the main module directory:

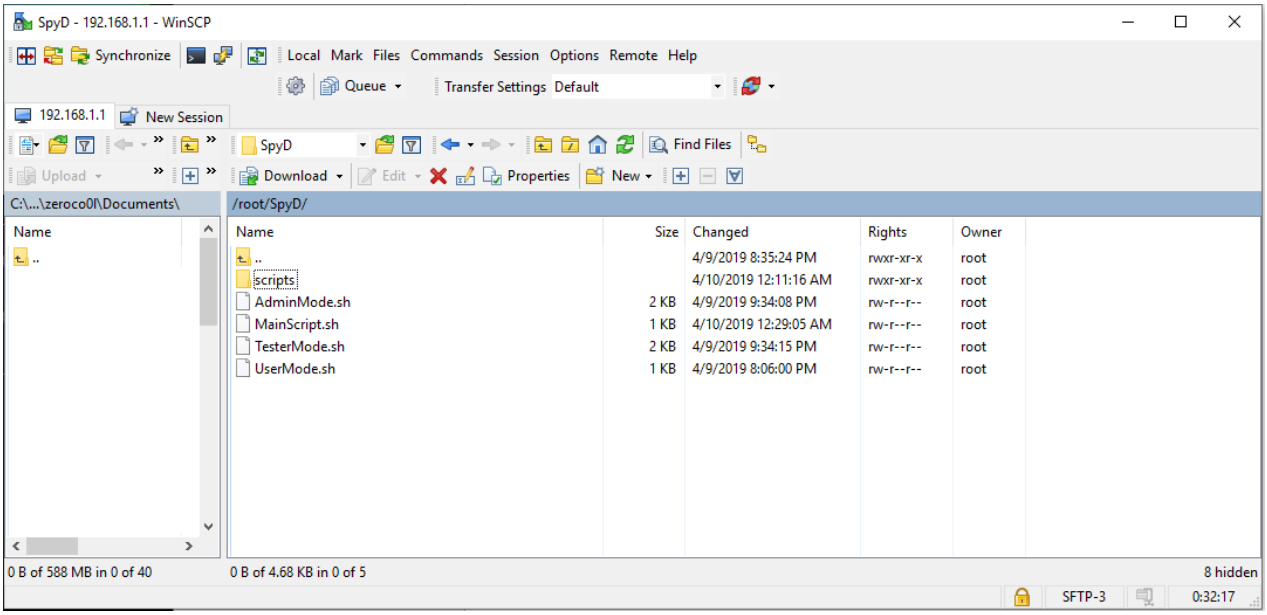


Figure 6.2.18: Directory

Service Scripts:

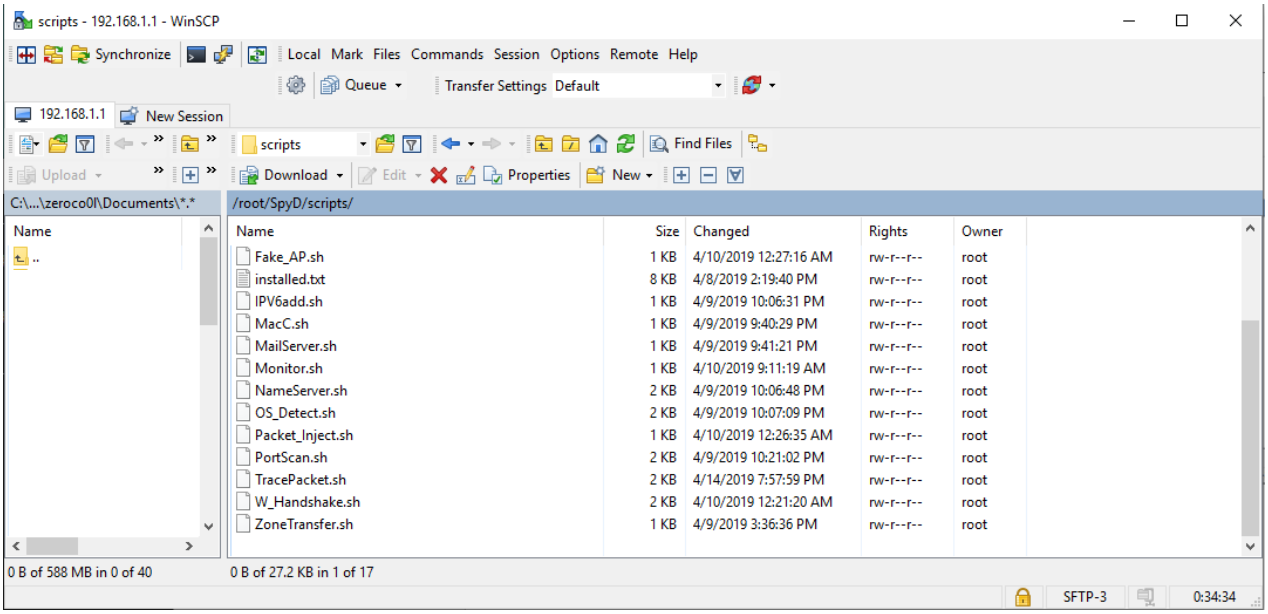


Figure 6.2.19: Service Scripts

Some of the Services Provided

1. MAC Changer

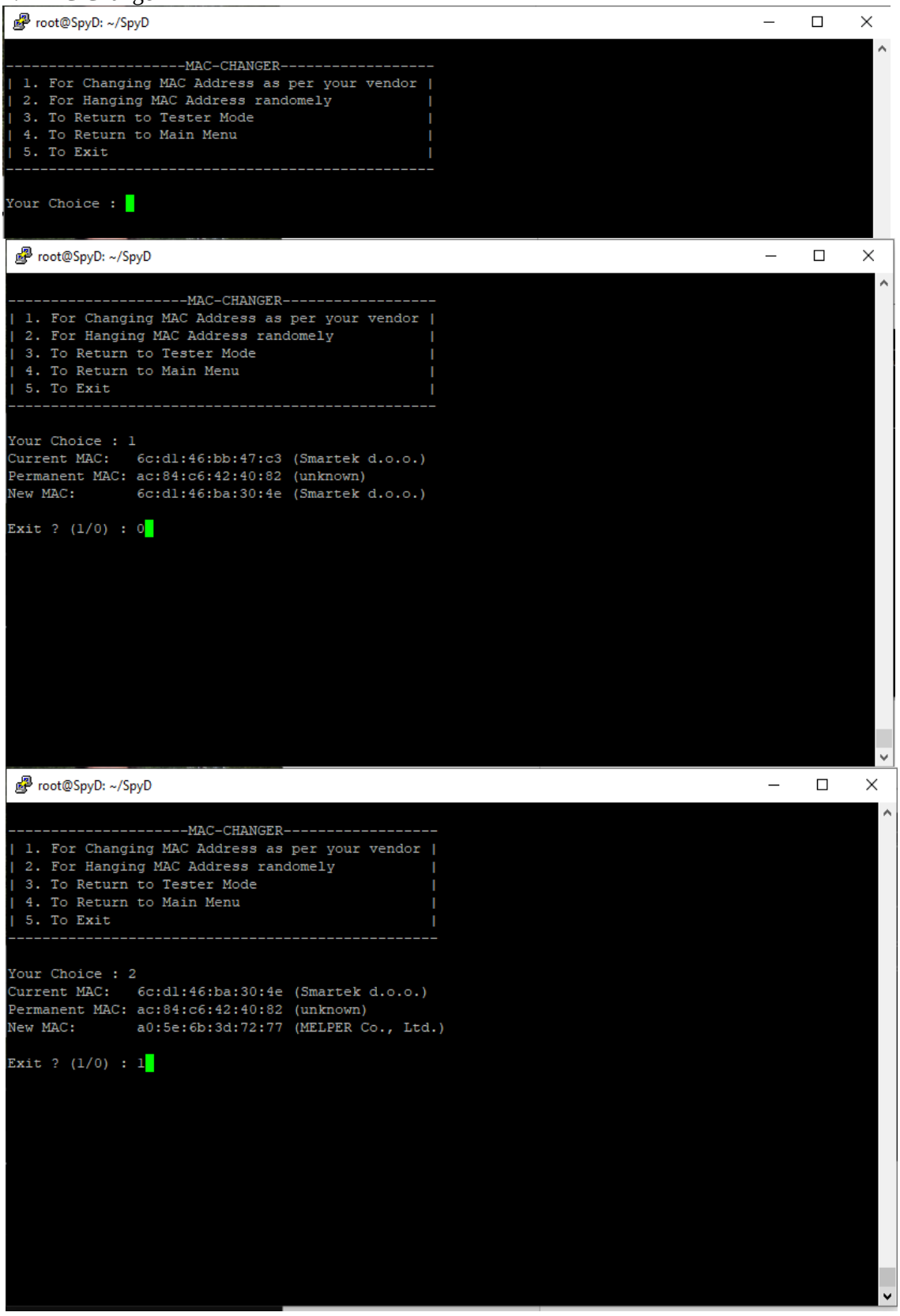


Figure 6.2.20: MAC Changer

2. Finding IPV6 Address of a Website

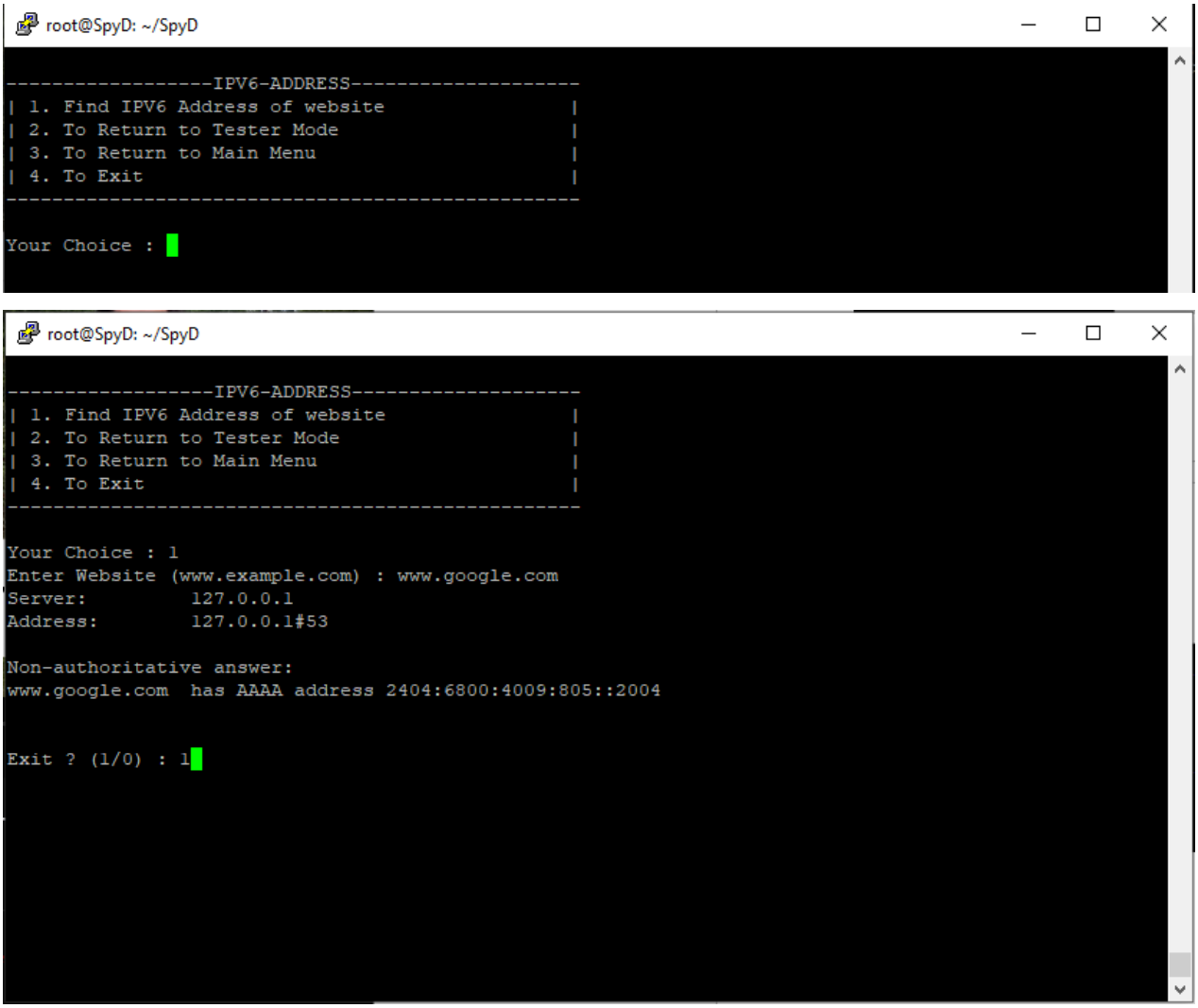


Figure 6.2.21: IPV6 Changer

3. OS Detection of a Website

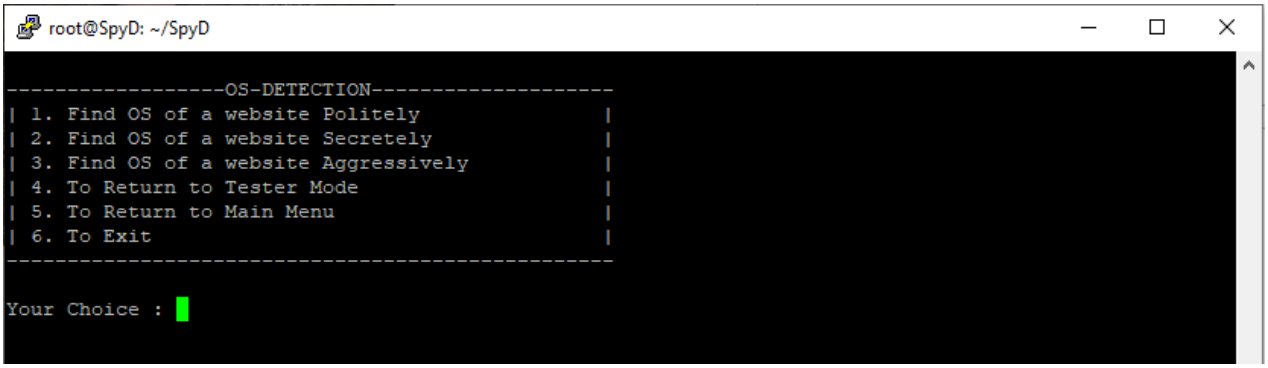


Figure 6.2.22: OS Detection

4. Monitor WLANs

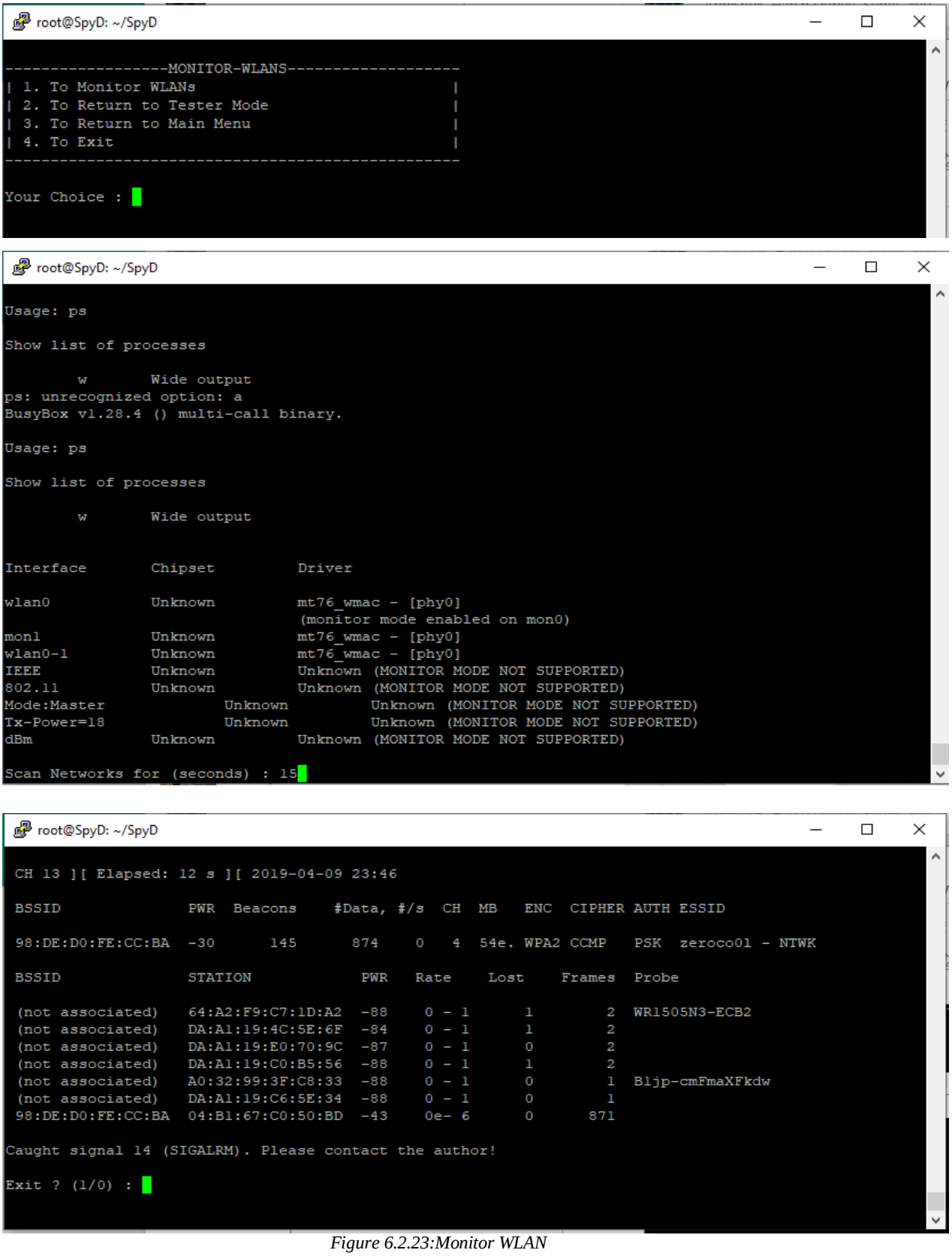


Figure 6.2.23:Monitor WLAN

5. Open Port Detection

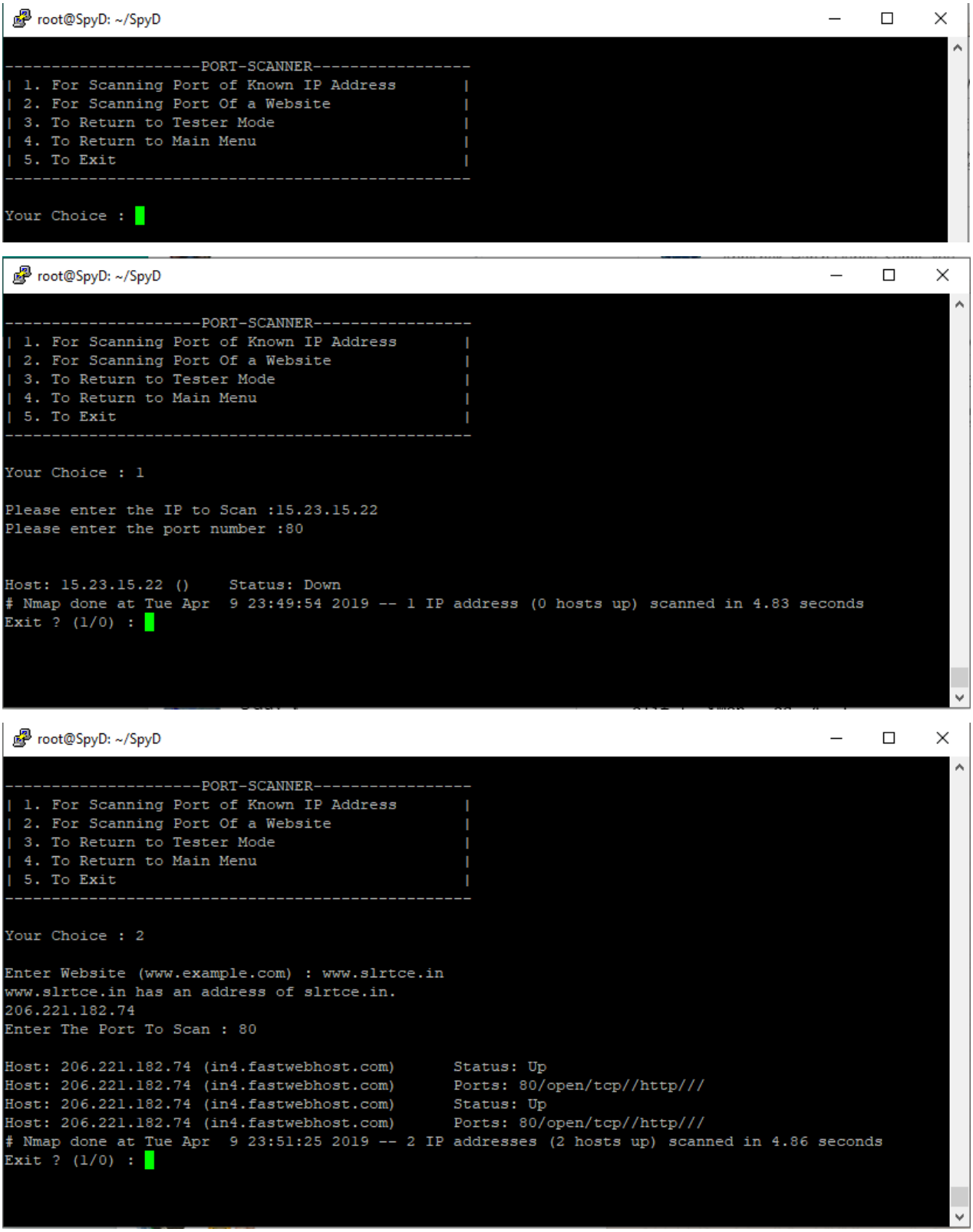


Figure 6.2.24: Open Port Detection

6. Create a Fake Access Point

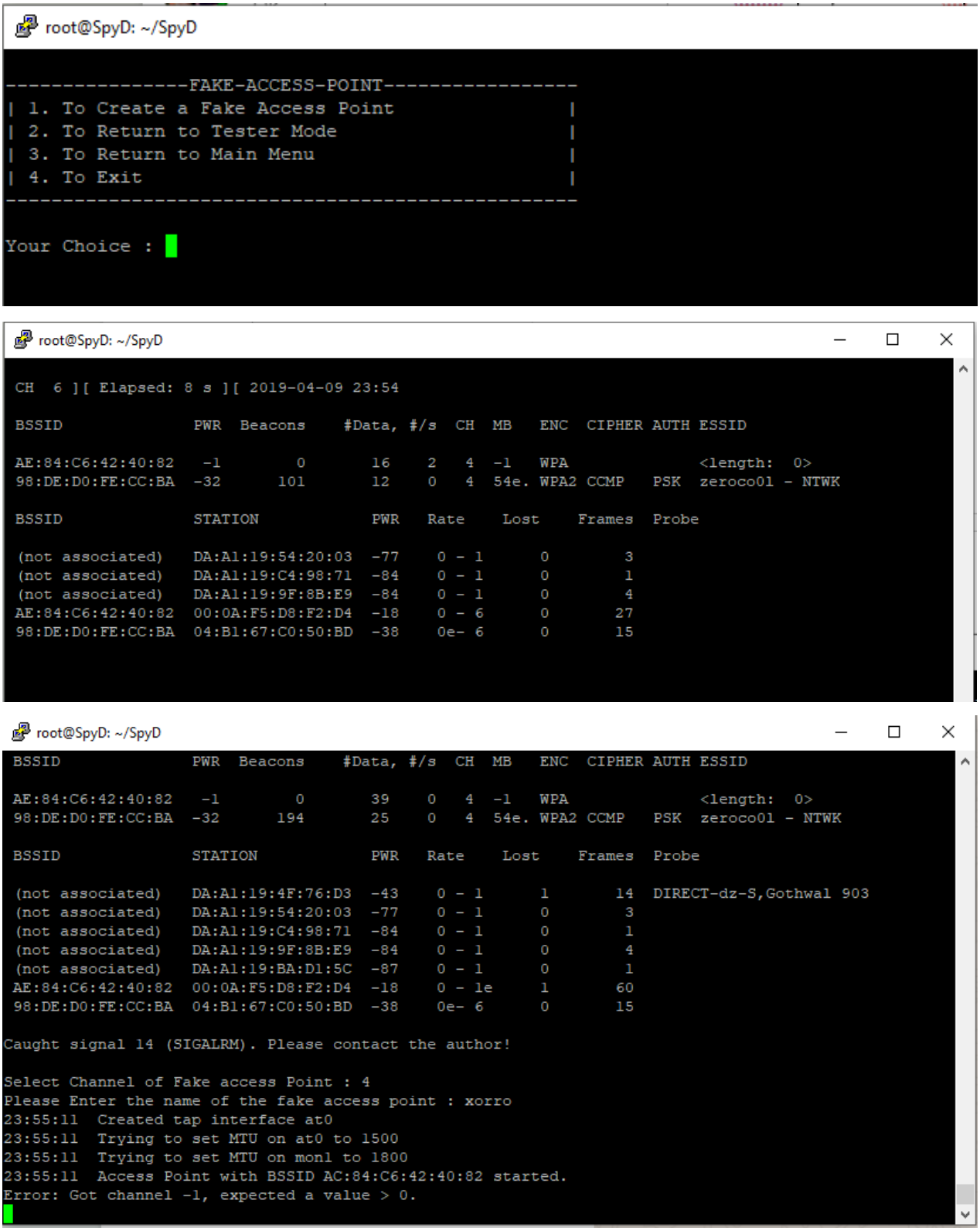
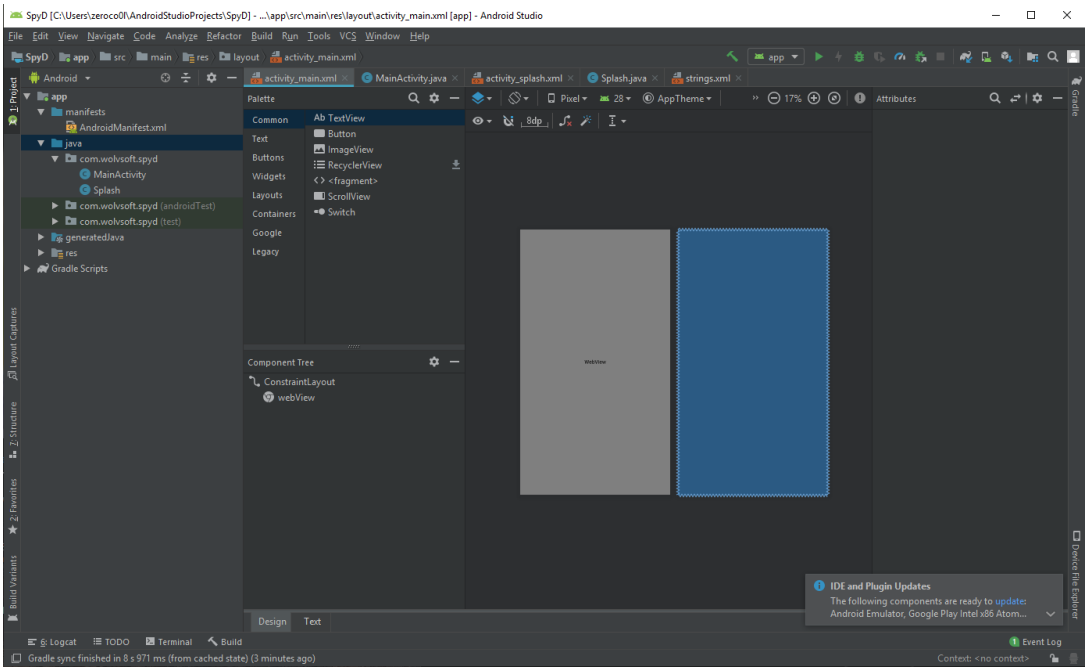


Figure 6.2.25: Fake access point creation

Android application

1. activity_main.xml



2. activity_splash.xml

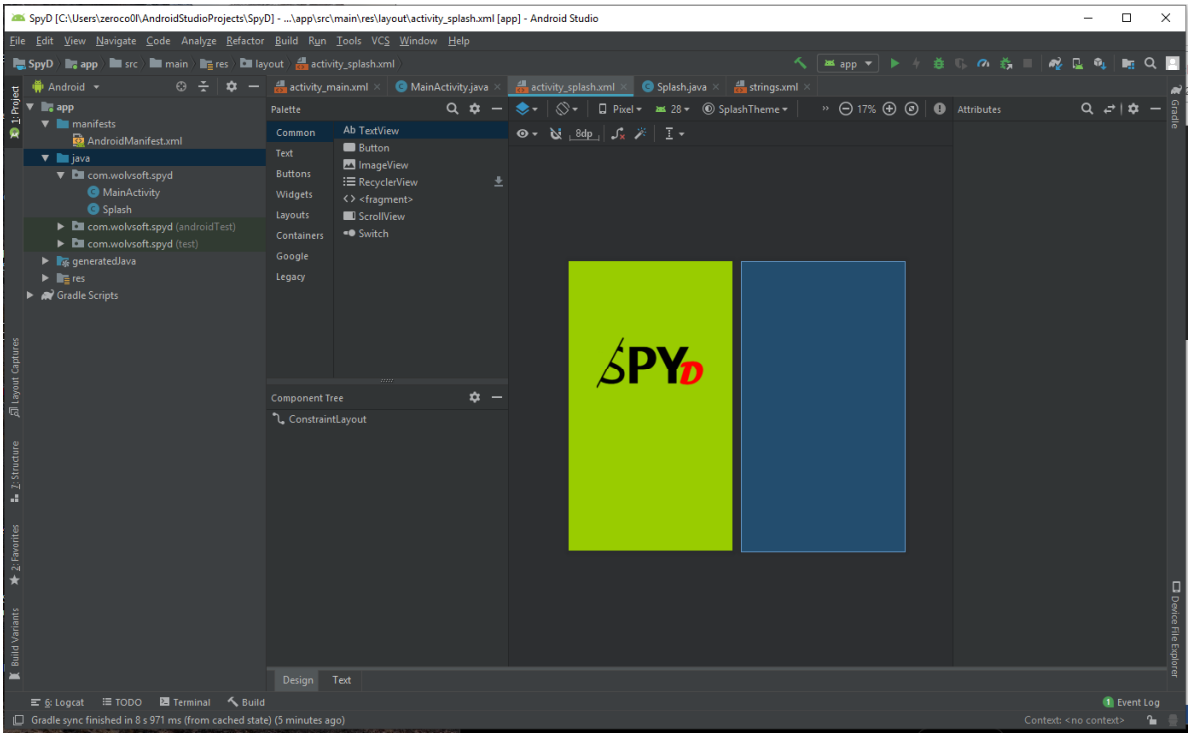
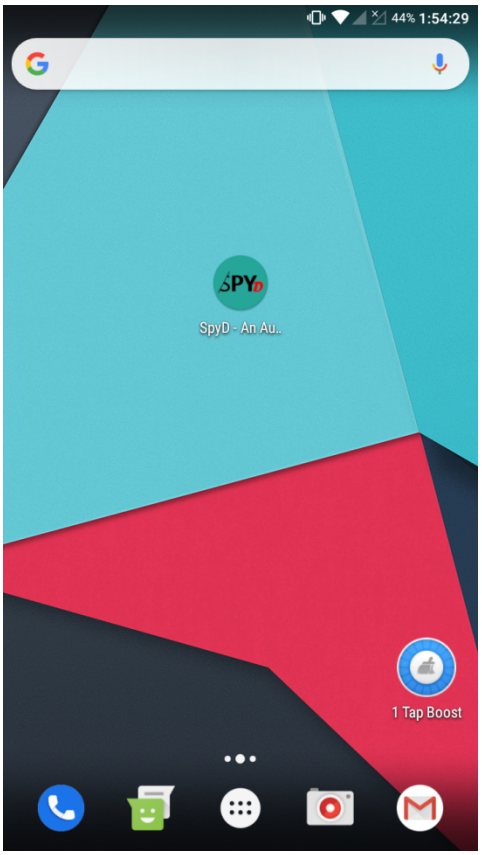
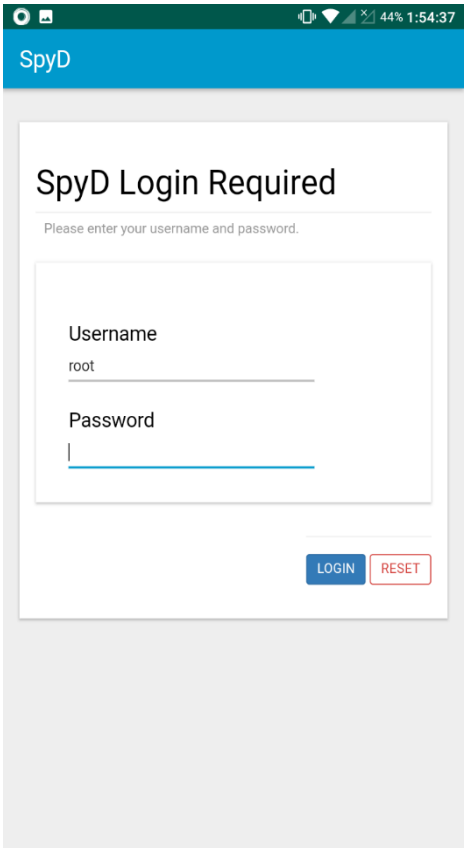
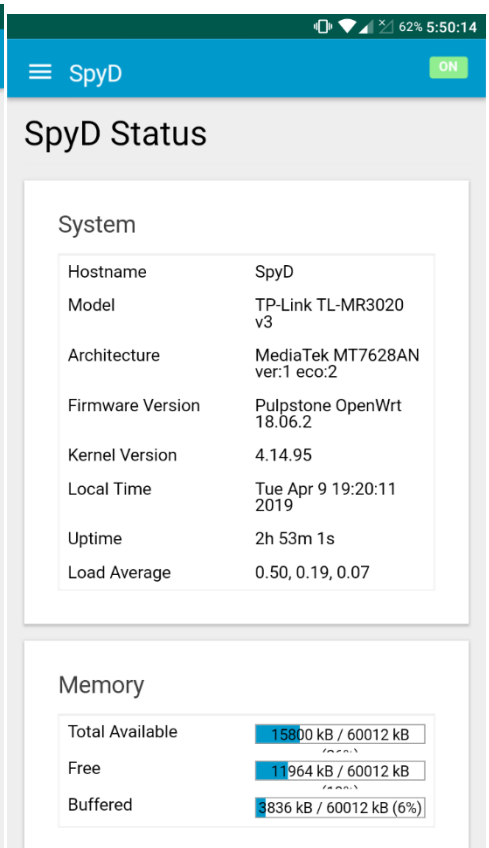


Figure 6.2.31:Android Environment

4. Application







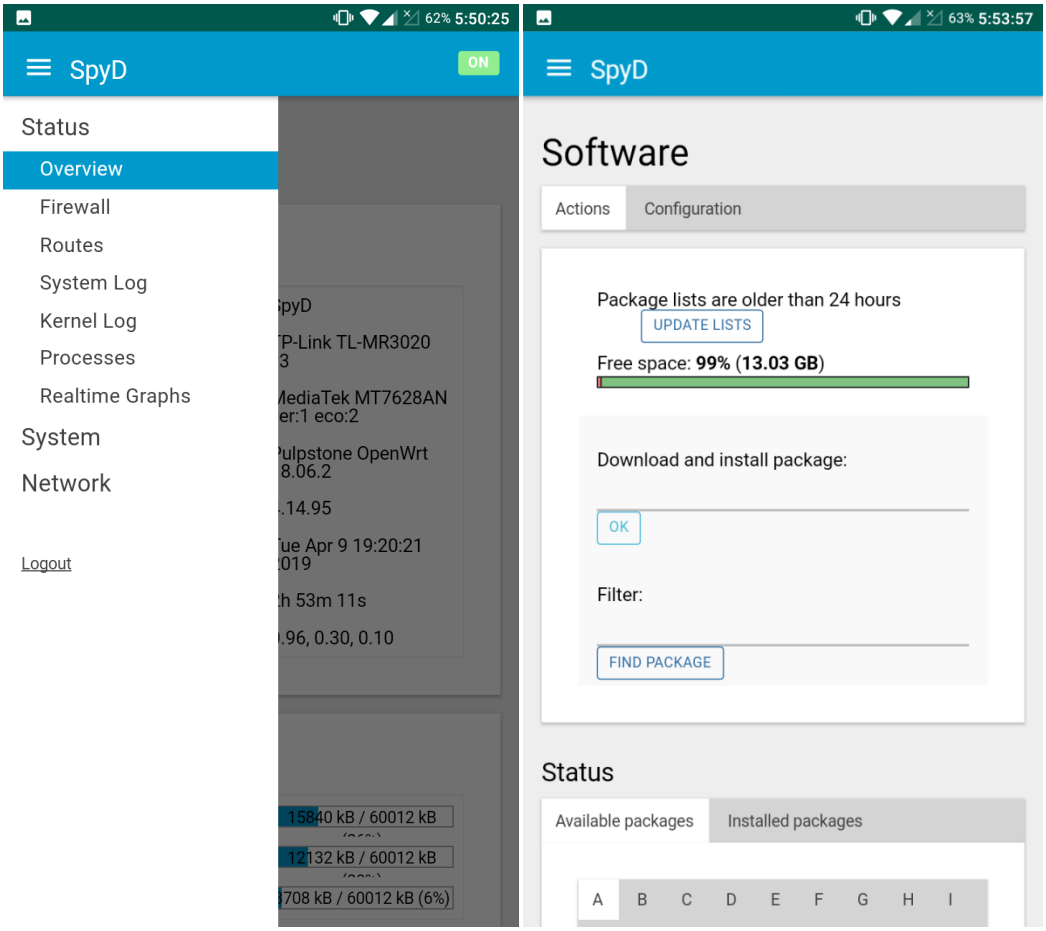


Figure 6.2.33:Android Interface

Chapter 7

Conclusion

Chapter 7
Conclusions

7.1 Conclusion

Internet usage is increasingly becoming important as more and more users access the Internet, and many users are using the Internet to express and share their opinions. Thus our Goal is to find the favorable or interesting way to provide security at a lower level as to provide security to each user.

In this project we propose a Penetration testing system where users/organizations can use various attacks to check whether their network is vulnerable against those. If the network is vulnerable, they can undertake necessary actions to avoid and remove those loopholes from the network. We use an OpenWRT based framework to add the functionalities to a chipset which can be employed locally at any site, and then perform tests or attacks using the provided predefined attacks. The experimental results show that the system is practical and the attacks are feasible.

This tool will ultimately help the users or small organizations to manage network security without hassles of going and researching the whole market thereby wasting a lot of money and time. Overall our project will give a way to perform tests at a local platform without any need of professionals, thus, improving the efficiency of organizations that are small scaled.

7.2 Future Scope

- Intrusion detection

  By monitoring system activity through examining vulnerabilities in the system, the integrity of files and conducting an analysis of patterns based on already known attacks, SpyD, along with multiple additions like a session generator and tracker, will be able to detect intruders. It will also automatically monitor the Internet to search for any of the latest threats which could result in a future attack.

- Support a wide range of users with the use of a chipset with a higher computational speed.

Chapter 8

References

References

[1] Cybergrenade: Automated Exploitation of Local Network Machines via Single Board Computers. Available: https://ieeexplore.ieee.org/document/8108803

[2] Computer Network Security and Technology Research by Fan Yan, Yang Jian-wen2, Cheng Lin1 Available: https://ieeexplore.ieee.org/document/7263569

[3] TP-Link TL-MR3020 OpenWRT flashing. Available: https://wiki.openwrt.org/toh/tp-link/tl-mr3020

[4] K. Scarfone, M. Souppaya, A. Cody, and A. Orebaugh, "Technical guide to information security testing andassessment", National Institute of Standards and Technology, Tech. Rep., 2008.

[5] F. Palmieri, U. Fiore, and A. Castiglione, "Automatic security assessment for next generation wireless mobile networks," Mobile Information Systems, vol. 7, no. 3,pp. 217–239, 2011.

[6] F. Holik, J. Horalek, O. Marik, S. Neradova, and S. Zitta, "Effective penetration testing with metasploit framework and methodologies," in 2014 IEEE 15thInternationalSymposium on Computational Intelligence and Informatics (CINTI), Nov. 2014, pp. 237–242. DOI:10.1109/CINTI.2014.7028682.

[7] Y. Hu, D. Sulek, A. Carella, J. Cox, A. Frame, and K. Cipriano, "Employing miniaturized computers for distributed vulnerability assessment," in 11th International Conference for Internet Technology and Secured Transactions (ICITST), Dec. 2016, pp. 57–61. DOI: 10. 1109/ICITST.2016.7856666.

Chapter 9

Paper Published

© 2019 JETIR April 2019, Volume 6, Issue 4 www.jetir.org (ISSN-2349-5162)

SpyD: An automated security tool

[1]Abhishek Pandey, [2]Harshvardhan Dubey, [3]Pratik Varma, [4]Sumit Awinash, [5]Prof. Reena Kothari
[1]B.E., [2] B.E., [3] B.E., [4] B.E., [5]M. Tech
[1]Information Technology,
[1]Mumbai University, Mumbai, India

Abstract : As people are realizing the importance of cyber security, the price for hacking tool such as Pineapple is increasing rapidly. One Pineapple device can cost up to Rs. 13000+Tax+Shipping charges. Due to high cost people do not prefer buying such a product which hamper their security as an organization. By building SpyD we can provide a similarly effective product at a very considerably lower price which will help the organization to manage the security. SpyD is a automated network penetration device/bot which will be able to perform all major operations through the means of an automated script with built in failsafe which can be performed be any professional ethical hacker. This will ensure that the routine jobs are performed more precisely as compared to a hired professional, thus, eliminating the threat of any man-made errors. Additionally, SpyD will automate the work of softwares like Nmap, Aircrack-NG, Karma, Tcpdump etc.

Index Terms **- Automation, Cyber security, Penetration testing.**

I. INTRODUCTION

This chapter will introduce the reader with SpyD, which is used as a penetration testing tool and a hacker device. It will light up the topics like the description of the project and the former formulation of the problem behind it as well as what motivated the makers of the project to take a decision to make this project and its related problem solutions and thus covering up the scope of the project. SpyD is a follow up of the established Wi-Fi Pineapple and adds various functionalities like complete automation, packet monitoring, etc. The secondary objective is to lower the cost of the device to target smaller organizations and start-ups. In this project we aim to overcome the shortcomings of conventional testing tools like rutabaga or PineAP that are need for a professional, high cost, manual work etc.

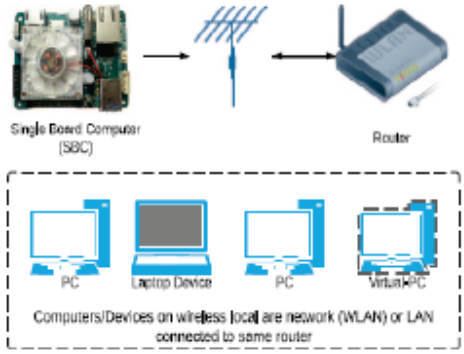


Fig. 1. SpyD system flow

II. RELATED WORK

Great deals of researchers have worked on using SBCs as potential pentesting tool kits. However, there are very few researchers who have expounded on a way to automate the pentesting tools within a single SBC to take full advantage of the SBC's capabilities. Another work automates MSFconsole successfully in a similar way we end up automating it, by manipulating resource scripts. However, the caveat is that for their paper, it would only work in the case that they knew ahead of time which exploits to run, thereby not being as flexible in terms of our solution in which the resource script file sent with MSFconsole is truly procedurally generated each time we run the script. The system also does not focus on using these tools in tandem with SBCs, which we are focusing on with this paper. Other than these, various penetration testing tools are being used currently in the industries that provide a compelling level of security to the firm's infrastructure, data and other details. An example of such a system is the Wifi PineApple system that is a penetration testing device just like the SpyD to provide security to the systems by aiming at the vulnerabilities present and alerting the organizations about them.

© 2019 JETIR April 2019, Volume 6, Issue 4 www.jetir.org (ISSN-2349-5162)

III. SYSTEM DESIGN AND IMPLEMENTATION

A. Architecture and Flow of SpyD

In SpyD, the Pulpstone framework is flashed over the stock MR 3020 chipset. Various tools and base modules are installed over the chipset's framework to provide further support. An extroot firmware is added to provide storage support to the Chipset that can be used to add modules of larger storage and processing requirements.

Design will elaborate the step by step flow of SpyD based on user Benchmark thus giving up the detailed information as to the basic flow of the system.

Flowcharts are used in designing and documenting complex processes or programs. Like other types of diagrams, they help visualize what is going on and thereby help the people to understand a process, and perhaps also find flaws, bottlenecks, and other less-obvious features within it. There are many different types of flowcharts, and each type has its own repertoire of boxes and notational conventions.

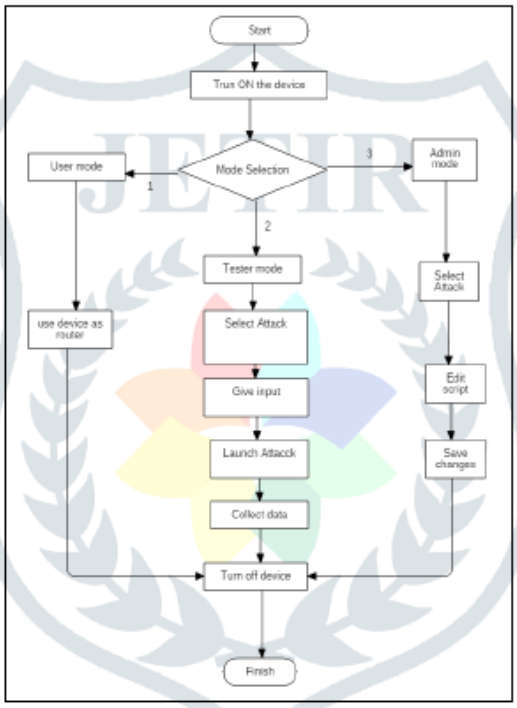


Fig. 2. Architecture of SpyD

As Kali Linux came with the 'Metasploit Penetration Testing Framework' built in, it was chosen to find and deliver exploits to the local machines using information given to it by the other two tools. All of the penetration-testing tools used were made generally to be manually used, with little, if any, support for automation, which presented a serious problem and demanded some unique solutions as we progressed.

B. Algorithm

1. Start
2. Turn on the device
3. Select Mode of operation
    a: User mode
        Use device as a router
    b: Tester mode
        i: Select Attack
            -Man in the middle Attack
            -Frame replay attack
            -Fake frame generation attack

© 2019 JETIR April 2019, Volume 6, Issue 4 www.jetir.org (ISSN-2349-5162)

-Create fake access point
-More attacks will be added

ii: Give appropriate input with respect to attack
iii: launch attack
iv: Collect appropriate data
v: Finish or go back to step i.

c: Admin mode
i: Select Attack
-Man in the middle Attack
-Frame replay attack
-Fake frame generation attack
-Create fake access point
-More attacks will be added

ii: Make changes in script as per requirement
iii: Finish or go back to step i.

4. Exit selected mode
5. Turn off the device or Change user mode
6. Stop

C. Implementation

SpyD has a home interface that allows the user to select the mode they wish to operate in. The user may select the User mode or the Admin mode. The admin mode has various controls and privileges over normal users.

Following this, the user may select any one of the many available tests for the system. Listing 1, the user has selected to inject packets into a channel to test the vulnerability of the channel.

```
#!/bin/bash
#PACKET_INJECTIO
ifconfig wlan0 down
airmon-ng start wlan0
airodump-ng mon0
echo "Please select the channel in which you wish to inject the packets."
read CHANNEL
echo "Please enter the name of the fake a access point"
read AP_NAME
airbase-ng --essid $AP_NAME -c $CHANNEL mon0
```

Listing 1: Fake access point and packet injection

- With the use of the airbase –ng command, the $CHANNEL channel is attacked.
- The $AP_NAME is a fake access point that is created to launch the attack from without getting traced to the original attacker.

If the user wants to test the vulnerability of the system against web port scanners, they can launch the Listing 2 script to attack the available web ports.

```
nmap -oG - $IP -p $PORT -vv > /Home/$IP.txt
```

Listing 2: Web port scanning

- The IP that is specified is the URL of the website.
- The PORT that is specified in the script is the ports that are to be checked for any vulnerability that may be present in the system.

To go into monitoring mode, the system executes another script that enables SpyD to monitor all the incoming and outgoing packets in the whole network, to and fro from all the access points and devices.

```
#!/bin/bash
#monitor the network
ifconfig wlan0 down
airmon-ng start wlan0
airodump-ng mon0
```

Listing 3: Packet monitoring

- The airmon-ng and airodump commands are used to assist the system in packet monitoring.

© 2019 JETIR April 2019, Volume 6, Issue 4 www.jetir.org (ISSN-2349-5162)

D. Technologies used

1. OpenWrt

OpenWrt is an open source project for embedded operating system based on Linux, primarily used on embedded devices to route network traffic. The main components are Linux, util-linux, musl, and BusyBox.

E. Programming languages

1. Python

Python is an interpreted, high-level, general-purpose programming language. Created by Guido van Rossum and first released in 1991, Python has a design philosophy that emphasizes code readability, notably using significant whitespace. It provides constructs that enable clear programming on both small and large scales.

IV. APPLICATIONS

The project aims to ease the process of Network security testing, so it will be easy for the users to perform tests to check for vulnerabilities as per the requirement. In many organizations, the testing tools which are used are the conventional ones which are usually operated and worked upon by testing professionals. This isn't a feasible option for end users and the organizations that are relatively smaller than those organizations. Through this proposed system, the end users as well as small scale organizations can perform tests and attacks to check for any vulnerability in their system and remove those flaws. The project uses various firmwares from the significant domain Network security like OpenWRT, LuCI, and operating systems like Python, etc to perform attacks like Man-in-the-middle, Framereplay, and fake Frame generation. All these functions are clustered in one chipset connected to the network to perform tests or attacks based on user's requirements.

V. CONCLUSION

Internet usage is increasingly becoming important as more and more users access the Internet, and many users are using the Internet to express and share their opinions. Thus our Goal is to find the favorable or interesting way to provide security at a lower level as to provide security to each user.

In SpyD we propose a Penetration testing system where users/organizations can use various attacks to check whether their network is vulnerable against those. If the network is vulnerable, they can undertake necessary actions to avoid and remove those loopholes from the network. We use an OpenWRT based framework to add the functionalities to a chipset which can be employed locally at any site, and then perform tests or attacks using the provided predefined attacks. The experimental results show that the system is practical and the attacks are feasible.

This tool will ultimately help the users or small organizations to manage network security without hassles of going and researching the whole market thereby wasting a lot of money and time. Overall our project will give a way to perform tests at a local platform without any need of professionals, thus, improving the efficiency of organizations that are small scaled.

ACKNOWLEDGMENT

We would like to express our sincere gratitude to the teaching faculty at SLRTCE whose timely inputs and suggestions, helped in the completion of the project. We would also like to thank the Library of our college for allowing us to carry out our research. Finally, we are thankful for having been given this opportunity to learn something new about the world of technology.

REFERENCES

[1] K. Scarfone, M. Souppaya, A. Cody, and A. Orebaugh, "Technical guide to information security testing andassessment", National Institute of Standards and Technology, Tech. Rep., 2008.

[2] F. Palmieri, U. Fiore, and A. Castiglione, "Automatic security assessment for next generation wireless mobile networks," Mobile Information Systems, vol. 7, no. 3,pp. 217–239, 2011.

[3] F. Holik, J. Horalek, O. Marik, S. Neradova, and S. Zitta, "Effective penetration testing with metasploit framework and methodologies," in 2014 IEEE 15thInternationalSymposium on Computational Intelligence and Informatics (CINTI), Nov. 2014, pp. 237–242. DOI:10.1109/CINTI.2014.7028682.

[4] J. Muniz, Penetration testing with raspberry pi. Packet, 2015.

© 2019 JETIR April 2019, Volume 6, Issue 4 www.jetir.org (ISSN-2349-5162)

[5] Y. Hu, D. Sulek, A. Carella, J. Cox, A. Frame, and K. Cipriano, "Employing miniaturized computers for distributed vulnerability assessment," in 11th International Conference for Internet Technology and Secured Transactions (ICITST), Dec. 2016, pp. 57–61. DOI: 10. 1109/ICITST.2016.7856666.

[6] TP-Link TL-MR3020 OpenWRT flashing. Available: https://wiki.openwrt.org/toh/tp-link/tl-mr3020