Tarea 2

Dante Chavez Dante.chavez@alumnos.uv.cl,

1. Introducción

Se requiere crear una herramienta basada en línea de comandos para consultar el fabricante de una tarjeta de red dada su dirección MAC o su IP, en este informe se explicará cómo se creó esa herramienta, las funciones de esta, diagramas de flujo y documentación de Código.

2. Materiales y Métodos

Esta herramienta está basada en el sistema operativo **Windows** y esta desarrollada en Python, además se usará un archivo disponible en el repositorio del programa Wireshark[1] como base de datos, este archivo consiste en direcciones Mac conocidas, posee alrededor de 23000 direcciones Mac que se implementaran en el programa.

3. Resultados

En esta sección se describirá la documentación del software **OUILookup**, cada función principal, lo que hace y su diagrama de flujo correspondiente.

En la **Figura1** se explica que significa cada figura de los diagramas de flujo.

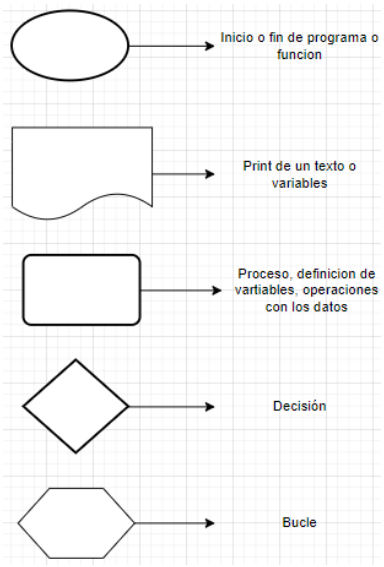

Figura 1. Explicación figura de diagrama de flujo

3.1. Función para leer base de datos

La función para leer la base de datos llamada “leer_base_datos” funciona como su nombre lo indica, leer la base de datos, junto con el código principal se encuentra la base de datos llamada “manuf.txt”, este texto está dividido en Mac, nombre vendedor, nombre empresa. La función elimina las líneas que empiezan con #, agrega a un diccionario la dirección Mac como llaves y los nombres de las empresas como valor de esas llaves, si no existe el nombre de la empresa se agrega el del vendedor.

Diagrama de flujo se muestra en **Figura2**

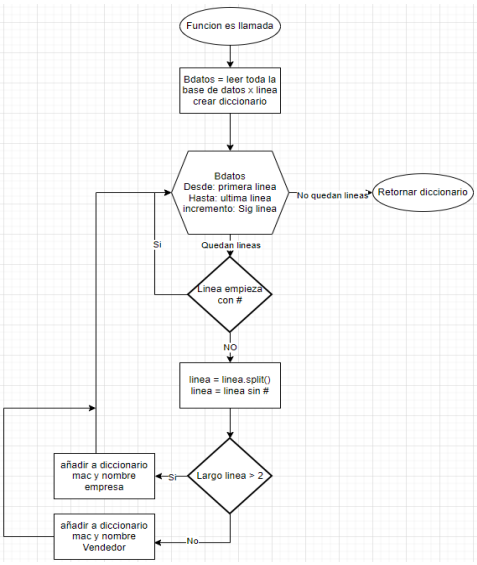

Figura 2. Diagrama de flujo función leer_base_datos

3.2. Función obtener datos por IP

La función para obtener datos por IP llamada en código “obtener_datos_por_ip(ip)” obtiene los datos por la IP que le llega como parámetro, importa la tabla ARP del computador a través e otra función, lee si la IP esta dentro de nuestra red, la cual es 192.168.1.0/24 si la IP que le llega como parámetro esta dentro de nuestra red la busca en la tabla ARP, al encontrarla obtiene su Mac usa la el diccionario “diccionario_mac” para obtener información del vendedor o fabricante para luego imprimirla en pantalla, si la IP a buscar no esta en el rango o no esta en la tabla ARP se mostrara un mensaje indicando ello.

Diagrama de flujo se muestra en **Figura3**

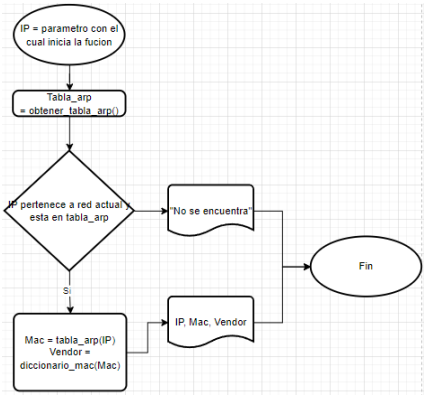

Figura 3. Diagrama de flujo función leer_base_datos

3.3. Función obtener datos por Mac

La función obtener datos por Mac llamada en el Código "obtener_datos_por_mac(mac)" se le ingresa un parámetro el cual es la Mac, la función "corta" la Mac a la mitad para obtener los primeros 24 bits y así obtener el identificador único de organización y ese identificador buscarlo en la base de datos para devolver el vendedor de esa Mac.

Diagrama de flujo se muestra en **Figura4**

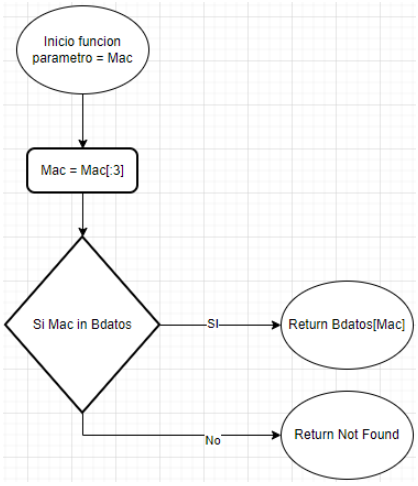

Figura 4. Diagrama de flujo función obtener_datos_por_mac

3.4. Función Obtener tabla ARP

La función para obtener la tabla ARP llamada en Código "obtener_tabla_arp" no usa parámetros, usa la librería "subprocess" para obtener la tabla ARP, ignora las 2 primeras líneas porque es texto y guarda las IP y las Mac en una librería para luego retornarla al Código que llamo esta función.

Diagrama de flujo se muestra en **Figura5**

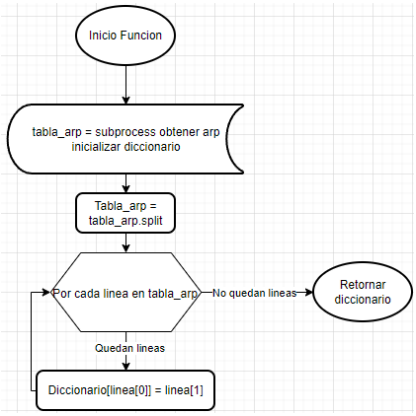

Figura 5. Diagrama de flujo función codigo_obtener_tabla_arp

3.5. Función Main

La función Main llamada en el Código "main(argv)" usa como parámetro los argumentos con el que se ejecuto el programa, permite elegir entre 4 opciones las cuales son: --help, --arp, --ip <argv> y --mac <argv> cada una ejecuta una de las funciones vistas anteriormente.

Diagrama de flujo se muestra en **Figura6**

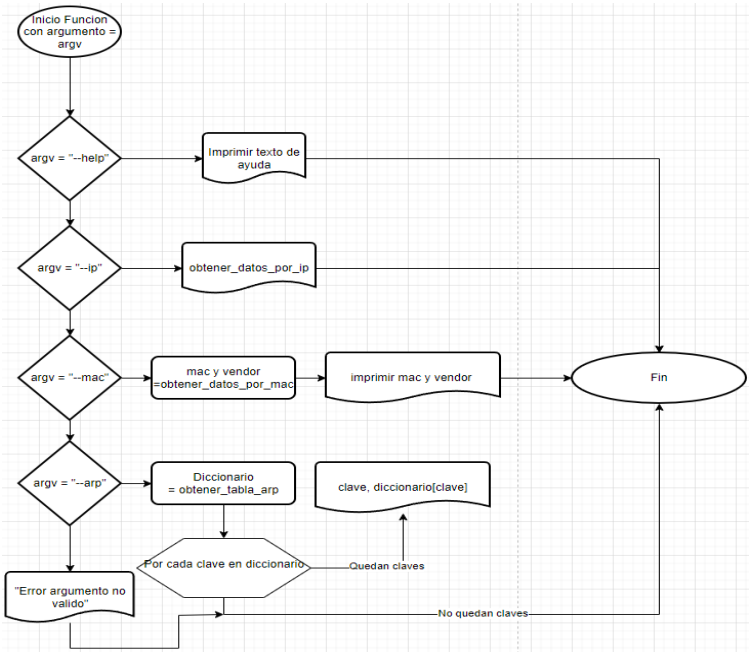

Figura 6. Diagrama de flujo función main

4. Discusión y conclusiones

Creando esta herramienta se puede aprender que es bastante simple el funcionamiento de buscar el vendedor, empresa de un dispositivo por su dirección Mac y automatizar el proceso, además de poder hacer lo anterior con la tabla ARP del computador para saber quién fabrico o vendió los dispositivos que tenemos alrededor de nosotros.

5. Referencias

[1] https://github.com/boundary/wireshark/blob/master/manuf