Universidad De Valparaíso

Redes de Computadores - ICI 224

OUILookup

Tarea N°1

Integrantes: Gustavo Espinoza.
Jorge Hernández.
Lukas Pinto.

Profesor: Gabriel Astudillo.

20 de Noviembre de 2020

Índice

1. Introducción :

Se busca implementar una herramienta tecnológica basada en línea de comandos, para consultar el fabricante de una tarjeta de red según su dirección MAC o para consultar si un dispositivo se encuentra conectado en la propia red según su dirección IP.

Dirección MAC(Media Access Control), se conoce también como dirección física, la cuál es única para cada dispositivo(tarjeta de red).

Dirección IP(Internet Protocol) es un conjunto de números que identifica de manera lógica y jerárquica a una interfaz en la red(elemento de conexión o comunicación) de un dispositivo, que utilice el protocolo IP o que corresponda al nivel de red del modelo TCP/IP.

2. Requisitos :

- La aplicación se deberá llamar **OUILookup**
- Sólo se puede ocupar *bash*, *perl*, **python**,*c* o *c++*, como lenguaje de programación para implementar la herramienta descrita.
- El archivo **manuf.txt**, base de la herramienta, debe estar parametrizado como variable global.
- Al ser una herramienta basada en líneas de comando, **debe** implementar los siguientes comandos con sus respectivas respuestas:
  1. *–help* considerando el motivo .
  2. *–ip* **dirección IP**,considerando los casos de que pertenezca o no a la misma red.
  3. *–mac* **dirección MAC**, considerando los casos de que esté o no en la base de datos.
- El software se debe implementar para **un** sistema operativo, en éste caso, el software está enfocado para sistemas operativos **Linux**.
- Los parámetros ingresados deben ser procesados por la función **getopt**

3. Restricciones :

El uso y funcionamiento de ésa herramienta tecnológica, dada su implementación, se encuentra restringido por los siguientes parámentros:

- Utilizar sistema operativo Linux.
- Tener instalado el package *net-tools* para el sistema operativo anteriormente dicho. Dicho package trae implementados los comandos necesarios para hacer funcionar la herramienta.
- Tener instalado el package *libcurl* para poder ejecutar los comando 'curl', para la transferencia de datos no supervisados en la red.
- Para conocer el fabricante de la tarjeta de red de un host, es necesario ingresar directamente la dirección MAC del dispositivo.

4. Desarrollo :

4.1. Cómo abordar el problema :

Para abordar la implementación del software descrito en la descripción, se tuvieron en mente los siguientes aspectos:

- Manejo de datos:

  Para hacer el manejo de datos, primero se pensó en el algoritmo que lea el archivo *manuf.txt*, almacene en una matriz y permita separar el archivo por cada línea que se encuentra en éste(matriz 1xN). Los datos necesarios y útiles para la función **–mac** del software, comienzan desde la línea 56, por lo tanto, todo el contenido que no sirve debe ser desechado del la variable parametrizada como global(la matriz).

  Una vez se cargaron y manipularon los datos adecuadamente, el siguiente paso fue obtener los datos de manera remota, para la implementación del algoritmo se utilizó el archivo de manera local, para posteriormente, obtener el archivo y su contenido directo del repositorio git del cuál fue extraído.

- Definir las líneas de comando **(main)**:

  Dada la necesidad de pensar en la experiencia del usuario, es que se implementan distintas formas para ingresar los comandos solicitados. Cada comando especificado debe ir acompañado de la correspondiente dirección(*IP o MAC*), por lo que en caso de no ser ingresada, se imprimirá el mensaje correspondiente a éste tipo de error. Lo mismo si el comando no se ejecuta de manera completa, es decir, si el usuario no ingresa –help pero ingresa –h, entonces se entenderá que el usuario desea utilizar la función help, lo mismo aplica para –ip y –mac. En caso de que no se cumpliera ninguna de las situaciones descritas anteriormente, se imprime otro mensaje que indica la falta de argumentos válidos para la utilización de la herramienta.

- Método comprobador de dirección MAC **(checkMAC)**:

  Cada vez que se ejecute la función –mac, se debe comprobar si ésta dirección es válida, antes de buscar una similitud con los fabricantes dispuestos en la base de datos, de ésta manera, no sólo se puede indicar cuando una dirección está o no identificada dentro del *manuf.txt*, sino que se le puede notificar al usuario si la dirección ingresada como parámetro corresponde a una dirección mac real o no.

- Método comprobador de dirección IP **(checkIP)**:

  Con la misma intención que el método anterior, se debe comprobar que la dirección ip ingresada corresponda a una dirección válida, antes de comprobar si el dispositivo está conectado dentro o fuera de la red. Si la dirección IP ingresada, no se encuentra en la tabla **arp** no se encontrará nunca el fabricante de su tarjeta de red, ya que no se conocerá la dirección MAC del dispositivo en cuestión.

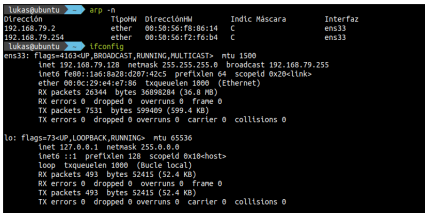

Figura 1: Dispositivos conectados arp

5. Vistas lógicas :

A continuación se presentan las vistas lógicas de la herramienta presentada en este informe :

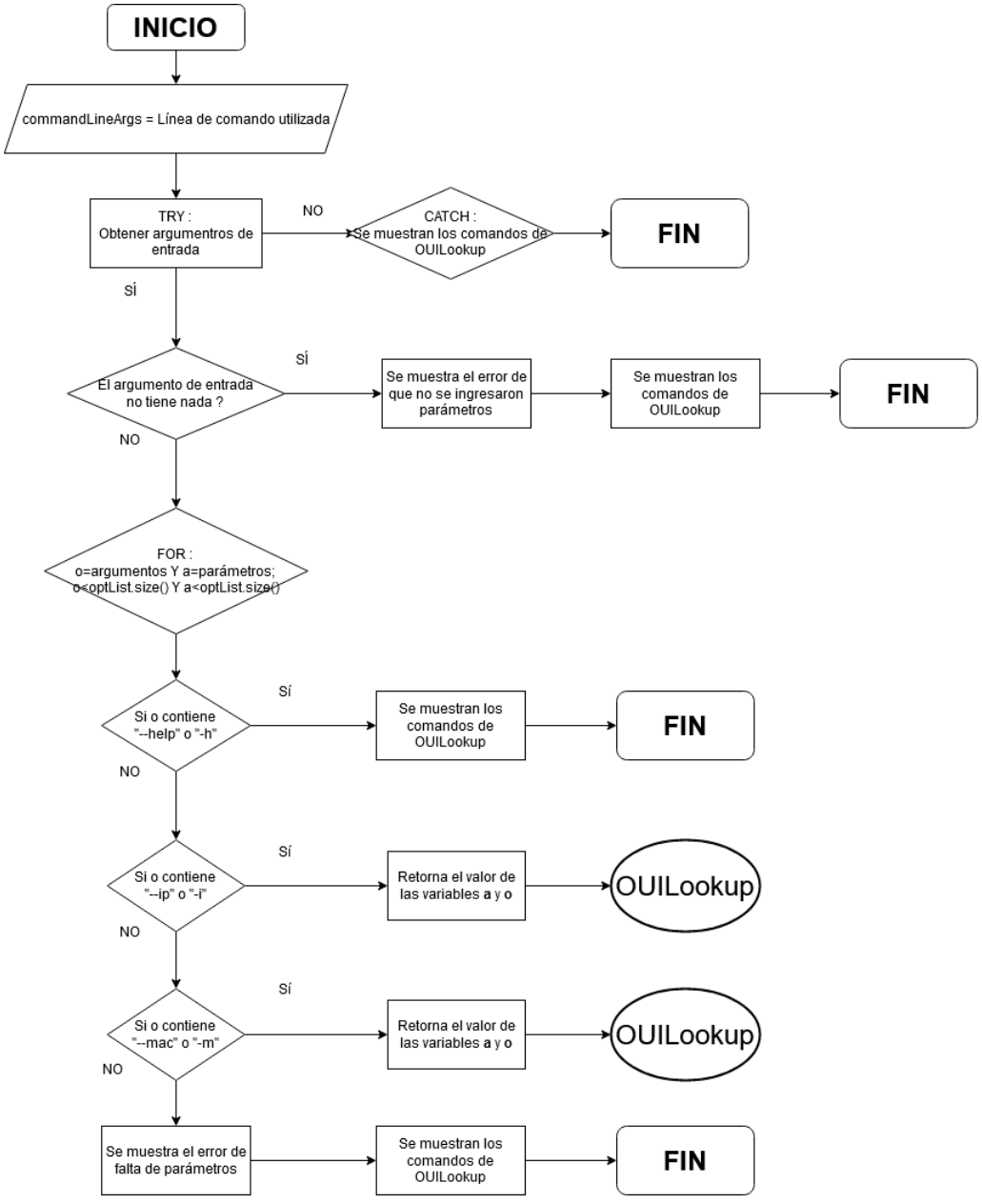

Figura 2: Diagrama de flujo, de función main()

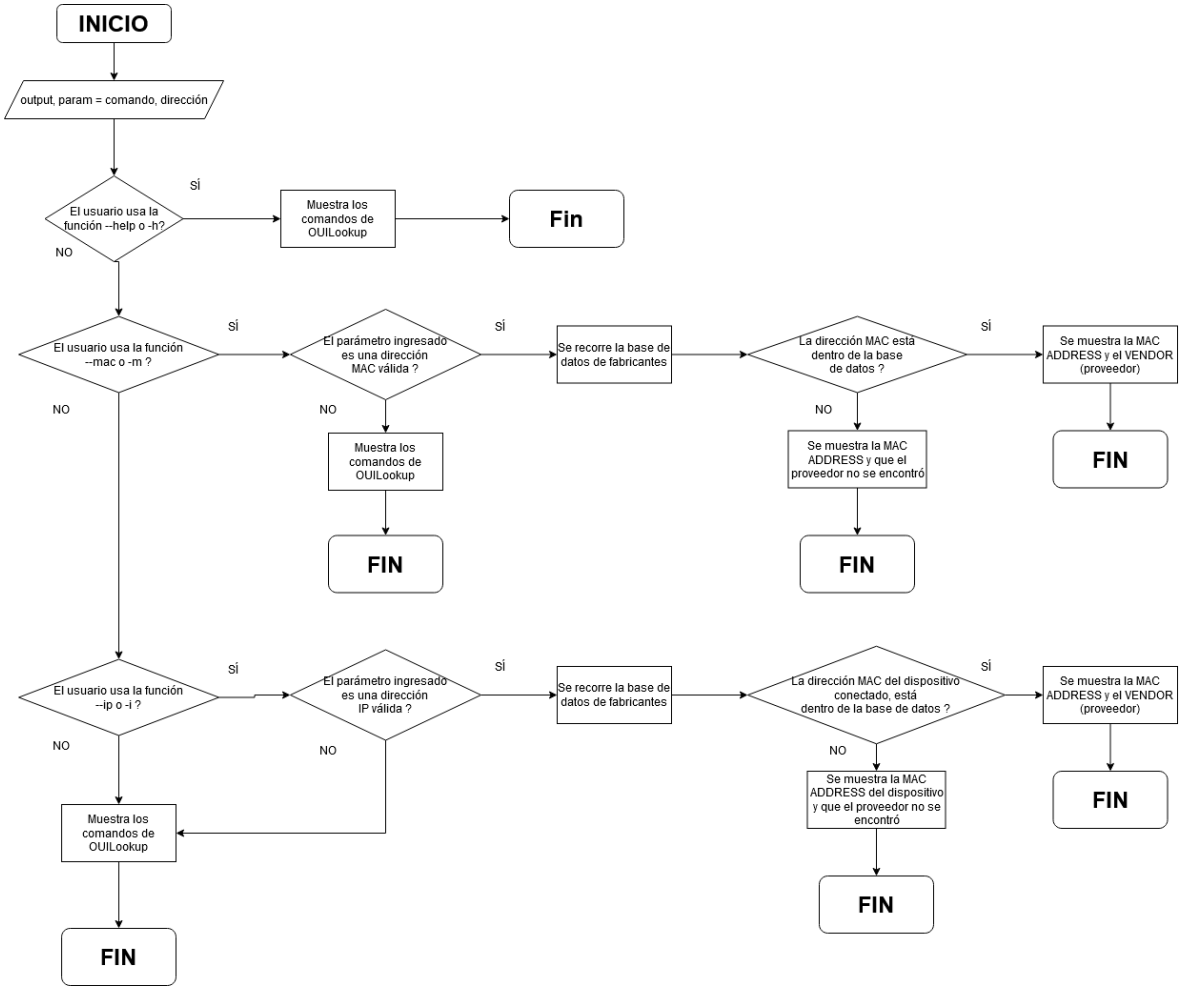

Figura 3: Diagrama de flujo, de la ejecución de **OUILookup**

6. Pruebas del producto y pruebas de calidad :

Una vez finalizada la implementación de la herramienta, se probaron si cada una de las funciones propuestas cumplían su propósito.

1. Se comprueba utilizando direcciones MAC y direcciones IP falsas, para probar los métodos de validación implementados. Arroja los resultados esperados.

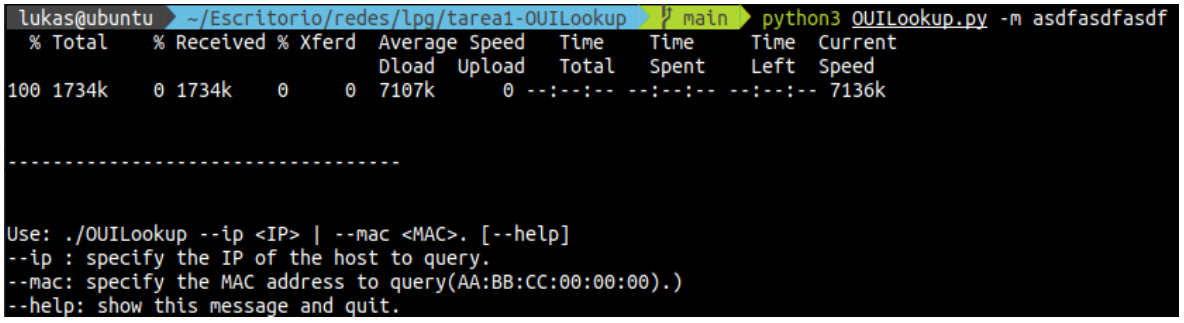

Figura 4: Diagrama de flujo, de la ejecución de **OUILookup**

2. Se comprueba utilizando una dirección MAC cualquiera, que se encuentre dentro de la base de datos. Arroja los resultados esperados.

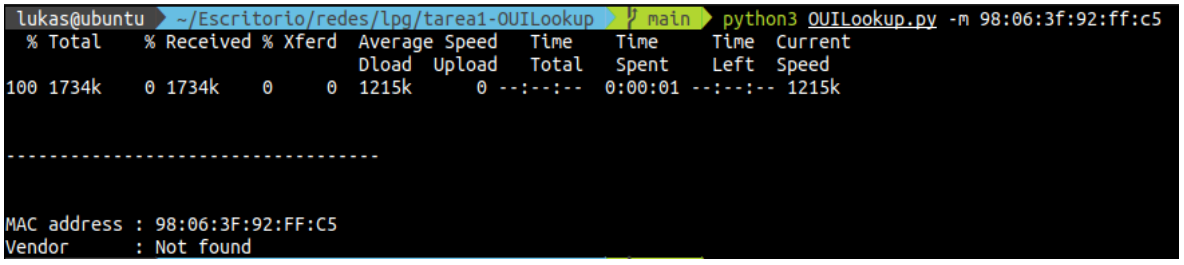

Figura 5: Diagrama de flujo, de la ejecución de **OUILookup**

3. Se comprueba utilizando una dirección IP que se encuentre dentro. Entrega resultados esperados.

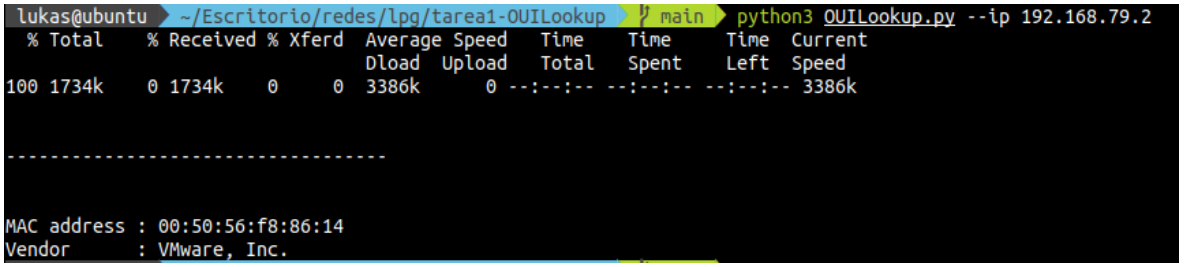

Figura 6: Diagrama de flujo, de la ejecución de **OUILookup**

7. Repositorio

Puede encontrar la herramienta implementada en el siguiente link:

https://github.com/LukasPinto/lpg/tree/main/tarea1-OUILookup

8. Conclusión :

Para implementar una herramienta tecnológica de éste tipo, el lenguaje de programación sólo entrega diferencias en tiempos de ejecución y formas de compilación, pero el sistema operativo al que está enfocado tiene bastante importancia. Si no se identifica el sistema operativo objetivo, y se intenta implementar una solución tecnológica para que funcione sin importar el ambiente en el que se ejecuta, hay demasiadas variable a considerar, importantes para el desarrollo del software.