

Redes de computadores: Tarea 2

Diego-Valenzuela-Parada, diego.valenzuelap@alumnos.uv.cl

Introducción

El presente informe aborda el desarrollo de OUILookup, una herramienta diseñada para consultar el fabricante de dispositivos a partir de direcciones MAC. Las direcciones MAC son identificadores únicos asignados a las interfaces de red, utilizados para la identificación de dispositivos en redes locales. La herramienta permite a los usuarios consultar el fabricante de una dirección MAC específica mediante una API pública y obtener los fabricantes de los dispositivos conectados en la red local a través de la tabla ARP.

Problemática y diseño de la solución

El problema que se aborda en esta tarea consiste en desarrollar una herramienta de línea de comandos llamada OUILookup, la cual debe permitir a los usuarios consultar el fabricante de una tarjeta de red a partir de su dirección MAC. El objetivo es proporcionar una solución que utilice una API REST pública para obtener la información de los fabricantes.

Para resolver el problema, se implementa OUILookup en Python utilizando getopt para procesar los argumentos de la línea de comandos, permitiendo consultar una dirección MAC específica o los fabricantes disponibles en la tabla ARP.

La herramienta utiliza la API de maclookup.app para obtener el fabricante de la dirección MAC ingresada o de las MACs extraídas mediante el comando arp -a. La salida se muestra en formato MAC/Vendor.

El diseño contempla varios casos de uso, como la consulta de una MAC individual o la obtención de múltiples fabricantes desde la tabla ARP, manejando eficientemente errores de red y formato, mostrando mensajes claros cuando no se encuentran datos.

Implementación

La implementación de **OUILookup** se llevó a cabo utilizando Python, aprovechando bibliotecas como **getopt** para gestionar los argumentos de línea de comandos y **requests** para realizar consultas a la API de maclookup.app. La herramienta permite a los usuarios consultar el fabricante de direcciones MAC específicas y obtener información sobre los dispositivos conectados a la red local mediante la tabla ARP.

Estructura del código:

print_help (): Muestra un mensaje de ayuda con las opciones de uso del programa, proporcionando detalles sobre los argumentos que acepta (--mac, --arp, --help).

lookup_mac(mac_address):

Consulta el fabricante de una dirección MAC específica utilizando la API de maclookup.app. Realiza una solicitud HTTP, analiza la respuesta JSON y muestra el nombre del fabricante. Si la dirección MAC no está en la base de datos, se muestra el mensaje "Not found". Además, mide y muestra el tiempo de respuesta de la API en milisegundos.

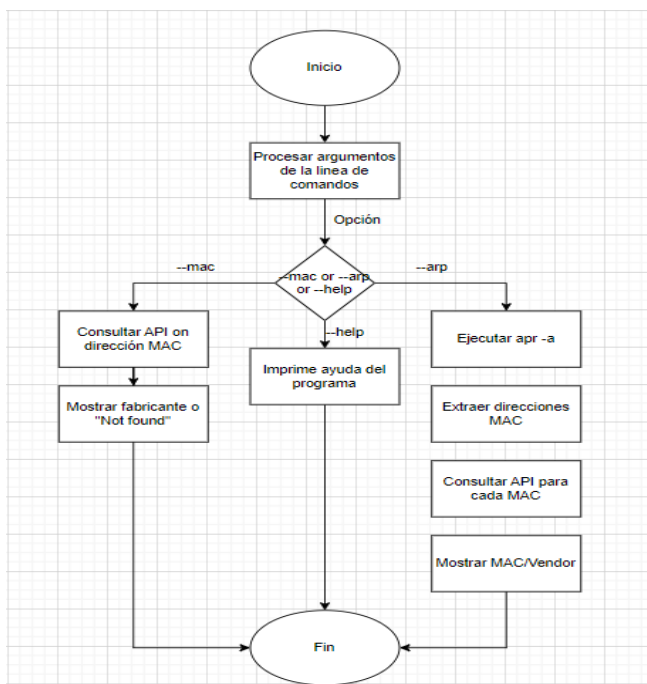
get_mac_vendor(mac_address):

Similar a lookup_mac(), esta función consulta la API, pero retorna solo el fabricante o "Not found" en lugar de mostrar directamente la información. Se utiliza en situaciones donde los resultados se combinan con otras informaciones, como en el caso de la tabla ARP.

lookup_arp():

Esta función ejecuta el comando arp -a en el sistema para capturar la tabla ARP, que contiene las direcciones MAC de los dispositivos conectados a la red local. Luego, procesa la salida línea por línea, extrae las direcciones MAC válidas utilizando una expresión regular y consulta la API para obtener el fabricante de cada dirección. Finalmente, muestra los resultados en el formato MAC/Vendor.

Diagrama de flujo:



Pruebas:

```
PS C:\Users\diego\OneDrive\Escritorio\Tarea 2> py OUILookup.py --mac 98:06:3c:92:ff:c5
MAC address: 98:06:3c:92:ff:c5
Fabricante: Samsung Electronics Co.,Ltd
Tiempo de respuesta: 603.96 ms
```

```
PS C:\Users\diego\OneDrive\Escritorio\Tarea 2> py OUILookup.py --mac 9c:a5:13
MAC address: 9c:a5:13
Fabricante: Samsung Electronics Co.,Ltd
Tiempo de respuesta: 605.33 ms
```

```
PS C:\Users\diego\OneDrive\Escritorio\Tarea 2> py OUILookup.py --mac 48-E7-DA
MAC address: 48-E7-DA
Fabricante: AzureWave Technology Inc.
Tiempo de respuesta: 591.74 ms
```

Para asegurar que el código funciona de manera correcta, las pruebas realizadas confirmaron que el programa OUILookup cumple con los requisitos establecidos y ofrece una funcionalidad fiable para la consulta de fabricantes a partir de direcciones MAC y la obtención de datos de la tabla ARP.

Funcionamiento de las direcciones MAC aleatorias en dispositivos electrónicos:

Las direcciones MAC aleatorias son identificadores generados por dispositivos para mejorar la privacidad y seguridad al conectarse a redes. En lugar de utilizar una dirección MAC fija, los dispositivos pueden generar direcciones aleatorias al escanear redes Wi-Fi, lo que dificulta el rastreo de actividades por parte de terceros. Este método es especialmente útil en dispositivos móviles y es soportado por sistemas operativos modernos como iOS y Android. Aunque las direcciones MAC aleatorias aumentan la privacidad, también pueden ocasionar problemas de conexión en redes que requieren autenticación basada en direcciones MAC fijas.

Referencias:

<https://www.tarlogic.com/bsam/es/controles/mac-aleatoria-bluetooth>

<https://telefonicatech.com/blog/mac-aleatorias-y-privacidad-parte2>

Conclusión

El desarrollo del programa OUILookup ha permitido explorar de manera efectiva el uso de direcciones MAC en redes de comunicación. Asimismo, el análisis de las direcciones MAC aleatorias revela su importancia creciente en el contexto actual de privacidad y seguridad digital. Estos identificadores, al ser generados de forma dinámica, contribuyen a la protección de la identidad del usuario y dificultan el seguimiento de actividades por parte de terceros. Sin embargo, también se han identificado desafíos, como posibles problemas de conexión en redes que requieren autenticación basada en direcciones MAC fijas.

Referencias:

Requests: HTTP for Humans - Documentación oficial sobre cómo realizar solicitudes HTTP en Python.

- [Requests Documentation](#)

Python getopt module to parse command line options - Explicación de cómo usar getopt para procesar argumentos en la línea de comandos.

- [GeeksforGeeks - Getopt Module](#)

Subprocess Management in Python - Documentación sobre cómo utilizar el módulo subprocess para ejecutar comandos del sistema.

- [Python Subprocess Documentation](#)

How to Use an API in Python - Tutorial sobre cómo realizar solicitudes a APIs en Python, incluyendo ejemplos y manejo de respuestas.

- [Real Python - APIs in Python](#)