

UNIVERSIDAD PRIVADA “FRANZ TAMAYO”

FACULTAD DE INGENIERÍA

CARRERA DE INGENIERÍA DE SISTEMAS



Seguridad Informática

**“MANUAL DE USUARIO DEL DISEÑO DE UN
CÓDIGO EN PYTHON PARA EL ESCANEADO DE
PUERTOS SYN Y OBTENCIÓN DE PUERTOS IP
USANDO SCAPY”**

AUTORES:

Víctor Hugo Chambi Cáceres

Santa Cruz, Bolivia

ÍNDICE GENERAL

1	Introducción Manual de usuario Seguridad informática.....	1
1.1	¿Scapy que es?	2
1.2	Kali Linux	2
2	Instalación del sistema	3
2.1	Kali Linux	3
2.2	Scapy para Windows	3
2.3	VirtualBox	4
2.4	Python o sublime text	5
3	Diagrama general del sistema	7
4	Manual de referencia	7
4.1	Situaciones de errores	11

1 Introducción Manual de usuario Seguridad informática

Un manual es una publicación que incluye los aspectos fundamentales de una materia. Se trata de una guía que ayuda a entender el funcionamiento de algo, o bien que educa a sus lectores acerca de un tema de forma ordenada y concisa. Un usuario es, por otra parte, la persona que usa ordinariamente algo o que es destinataria de un producto o de un servicio. (Gardey, 2010)

Un manual es una publicación que incluye los aspectos fundamentales de una materia, explicación a detalle del software o del sistema, Se trata de una guía que ayuda a entender el funcionamiento de algo, o bien que educa a sus lectores acerca de un tema de forma ordenada y concisa. Un usuario es, por otra parte, la persona que usa ordinariamente algo o que es destinataria de un producto o de un servicio.

1.1 ¿Scapy que es?

Aplicado al ámbito de la seguridad informática, esta herramienta nos permite realizar escaneos y/o ataques de red. La principal ventaja de Scapy es que, a diferencia de otras herramientas, nos proporciona la capacidad de modificar los paquetes de red a bajo nivel, permitiéndonos utilizar los protocolos de red existentes y parametrizarlos en base a nuestras necesidades. Y en el caso de que decidamos utilizar sus librerías en Python, podremos desarrollar nuestras propias herramientas, y de esta forma, podríamos realizar otros desarrollos de más alto nivel e integrarlos todos en función de nuestras necesidades. (Gómez, 2020)

Scapy es una de las herramientas más potentes para el análisis y hacking en redes, escrita en Python. Con ella se puede generar paquetes de red de prácticamente todos los protocolos conocidos (no solo TCP/IP) pero en este caso lo estaremos usando para un escaneo de puertos TCP Syn.

1.2 Kali Linux

Kali Linux es una distribución de Linux basada en Debian, específicamente diseñada para temas de seguridad muy variados, como análisis de redes, ataques inalámbricos, análisis forenses y otros que más adelante citaremos. Contiene herramientas para llevar a cabo todas estas pruebas de seguridad y análisis. Fue desarrollado en base a la reescritura de BackTrap, otra distribución de Linux para semejantes usos, por Mati Aharoni y Devon Kearns de Offensive Security. Kali Linux se encuentra entre las distribuciones de seguridad de Linux más usadas, ya que es una de las mejores, tanto para uso personal como profesional, proporcionando a los usuarios paquetes de herramientas como Foremost, Wireshark, Maltigo as-Aircrack-ng, Kismet y más. (Altube, 2018)

Su principal objetivo es poner a disposición del usuario, las mejores herramientas para trabajar la auditoría en internet y contar con un potente sistema de seguridad informática ante los peligros que puedan existir. permite adaptarlo para conseguir las herramientas necesarias adaptadas a las tareas específicas que queremos realizar.

2 Instalación del sistema

2.1 Kali Linux

Para comenzar el procedimiento, dos cosas será necesarias, en primero lugar el Hipervisor, en este caso será VirtualBox y una imagen ISO de Kali Linux para proceder a su instalación. Utilizaremos la versión de VirtualBox 6.0. También utilizaremos la última versión de Kali Linux disponible que es la 2018.4 en su versión de 64 bits. Con todo listo, abriremos VirtualBox y, situados en la pantalla principal, pulsaremos sobre “Máquina -> Nueva” para comenzar la creación de esta VM. Recomendamos pulsar sobre el botón inferior “Modo experto” para obtener la totalidad de las opciones de configuración de la máquina virtual.

2.2 Scapy para Windows

Ejecute directamente el comando: `Python pip install scapy`

```

C:\Users\Administrator>scapy
INFO: Can't import matplotlib. Won't be able to plot.
INFO: Can't import PyX. Won't be able to use psdump() or pdfdump().
WARNING: No pcap provider available ! pcap won't be used
Traceback (most recent call last):
  File "D:\install\Python\Python36-32\Scripts\scapy", line 25, in <module>
    interact()
  File "D:\install\Python\Python36-32\lib\site-packages\scapy\main.py", line 461, in interact
    init_session(session_name, mydict)
  File "D:\install\Python\Python36-32\lib\site-packages\scapy\main.py", line 329, in init_session
    scapy_builtins = {k: v for k, v in six.iteritems(importlib.import_module(".all", "scapy").__dict__)} if _validate_loc
al(k)} # noqa: E501
  File "D:\install\Python\Python36-32\lib\importlib\__init__.py", line 126, in import_module
    return _bootstrap._gcd_import(name[level:], package, level)
  File "<frozen importlib._bootstrap>", line 994, in _gcd_import
  File "<frozen importlib._bootstrap>", line 971, in _find_and_load
  File "<frozen importlib._bootstrap>", line 955, in _find_and_load_unlocked
  File "<frozen importlib._bootstrap>", line 665, in _load_unlocked
  File "<frozen importlib._bootstrap_external>", line 678, in exec_module
  File "<frozen importlib._bootstrap>", line 219, in _call_with_frames_removed
  File "D:\install\Python\Python36-32\lib\site-packages\scapy\all.py", line 18, in <module>
    from scapy.arch import *
  File "D:\install\Python\Python36-32\lib\site-packages\scapy\arch\__init__.py", line 67, in <module>
    from scapy.arch.windows import * # noqa F403
  File "D:\install\Python\Python36-32\lib\site-packages\scapy\arch\windows\__init__.py", line 56, in <module>
    from scapy.arch.pcapdnet import NPCAP_PATH, get_if_raw_addr, \
ImportError: cannot import name 'NPCAP_PATH'
https://blog.csdn.net/Amdy_amdy

```

Figura 1. Instalación de scapy Windows

2.3 VirtualBox

Para aprovechar todas las características de virtualización que ofrecen estos programas, es necesario que el software acceda a ciertas características de uso del procesador avanzadas, las cuales es posible que en tu equipo no estén activadas, como la aceleración por hardware VT-x/AMD-V, la paginación anidada y la Para virtualización Híper-V.

El proceso de instalación es muy sencillo y no varía en nada a los típicos next, next, next... cuando instalamos una aplicación en Windows. Si ya eres todo un pro en instalar aplicaciones en Windows (lo más probable) “puedes saltarte esta parte”. Sitúa el archivo de instalación que descargaste en el paso anterior y ejecútalo, de inmediato comenzará el proceso de instalación y en esta primera ventana haz click sobre Next.



Figura 2. Instalación de VirtualBox

2.4 Python o sublime text

Primero comprueba si tu ordenador ejecuta la versión 32 bits de Windows o la de 64, en "Tipo de sistema" en la página de "Acerca de". Para llegar a esta página, intenta uno de estos métodos:

Si tu ordenador ejecuta la versión de 64 bits de Windows, descarga Windows x86-64 executable installer. De lo contrario, descarga Windows x86 executable installer. Después de descargar el instalador, deberías ejecutarlo (dándole doble click) y seguir las instrucciones. Una cosa para tener en cuenta: Durante la instalación, verás una ventana de "Setup". Asegúrate de

marcar las casillas "Add Python 3.6 to PATH" o "Add Python to your environment variables" y hacer click en "Install Now".



Figura 3. Instalación de Python

3 Diagrama general del sistema

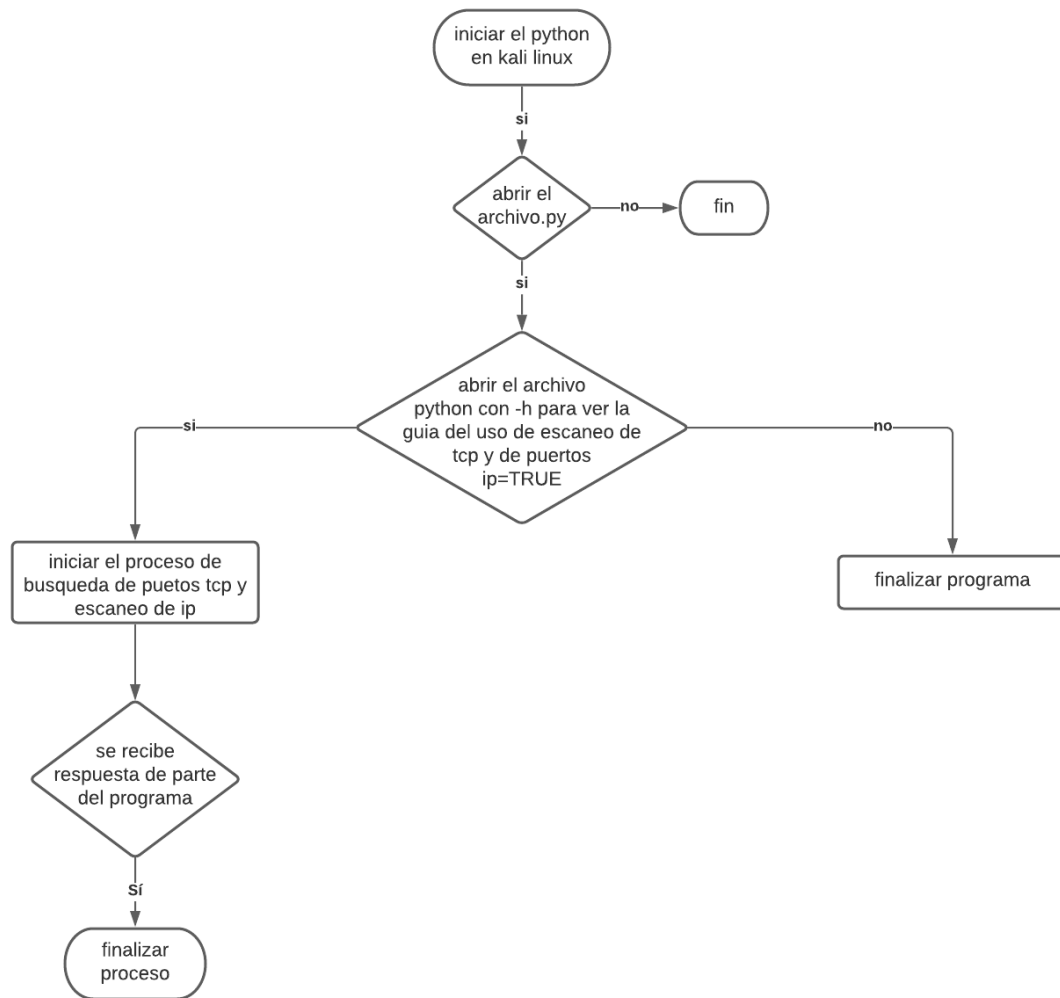


Figura 4. Diagrama general del sistema

4 Manual de referencia

- Como primer paso se debe iniciar a la terminal de Kali Linux

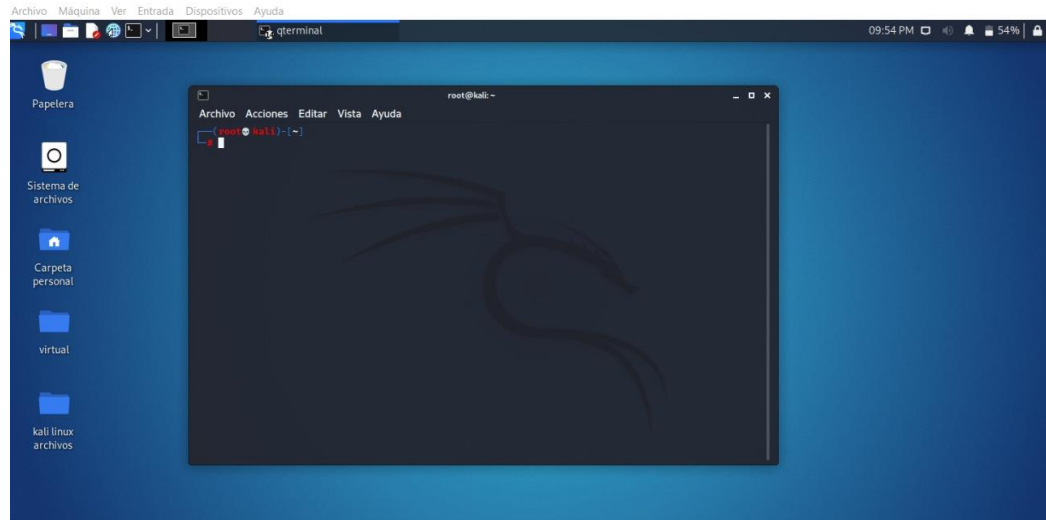


Figura 5. Terminal root Kali linux

- Lo siguiente es iniciar el programa que esta con extensión en Python con .py(para abrir el archivo de Python se debe abrir con sudo seguido del nombre del archivo” ” sudo permite ejecutar un comando como otro usuario, pero respetando una serie de restricciones sobre qué usuarios pueden ejecutar qué comandos en nombre de qué otros usuarios)

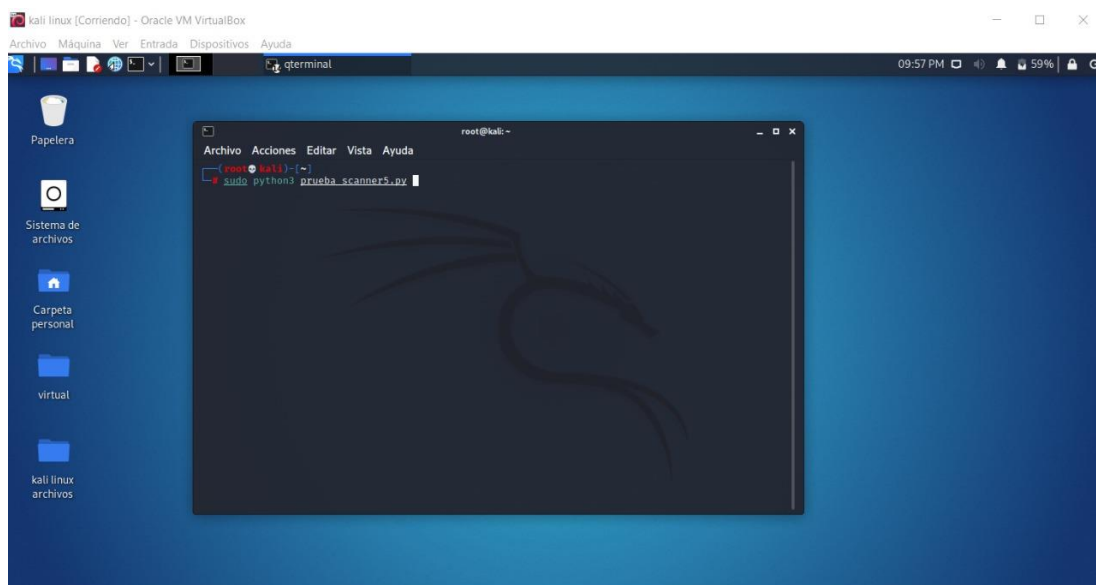
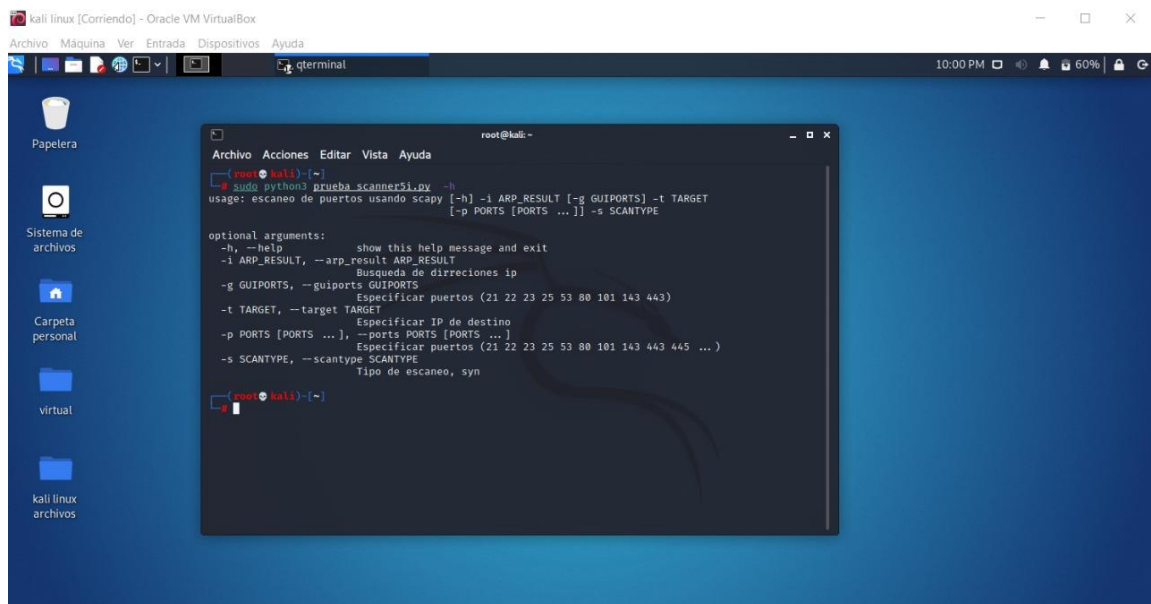


Figura 6. Abrir el archivo con extensión .py

- Lo siguiente será acceder a la guía que se encuentra dentro del programa gracias al argparser escribiendo lo siguiente ‘nombre del archivo Python’ -h esto mostrará al usuario las diferentes funciones para que pueda realizar el escaneo de puertos tcp y el escaneo de direcciones ip



```

root@kali: ~
└─$ sudo python3 prueba_scanner5i.py -h
usage: escaneo de puertos usando scapy [-h] -i ARP_RESULT [-g GUIPORTS] -t TARGET
                                         [-p PORTS [PORTS ...]] -s SCANTYPE

optional arguments:
  -h, --help            show this help message and exit
  -i ARP_RESULT, --arp_result ARP_RESULT
                        Busqueda de direcciones ip
  -g GUIPORTS, --guiports GUIPORTS
                        Especificar puertos (21 22 23 25 53 80 101 143 443)
  -t TARGET, --target TARGET
                        Especificar IP de destino
  -p PORTS [PORTS ...], --ports PORTS [PORTS ...]
                        Especificar puertos (21 22 23 25 53 80 101 143 443 445 ...)
  -s SCANTYPE, --scantype SCANTYPE
                        Tipo de escaneo, syn
  
```

Figura 7. Manual de guía del programa

- Dentro de la guía encontrara que con -i podrá realizar un escaneo de puertos ip, con -g podrá ver la guía del puerto que se está buscando, -t tendrá que especificar su puerto de destino en este caso seria 192.168.x.xx, -p podrá buscar el puerto que quiere encontrar sin la necesidad de realizar la búsqueda de los 1024 puertos, -s permite realizar el escaneo de tipo “SYN”

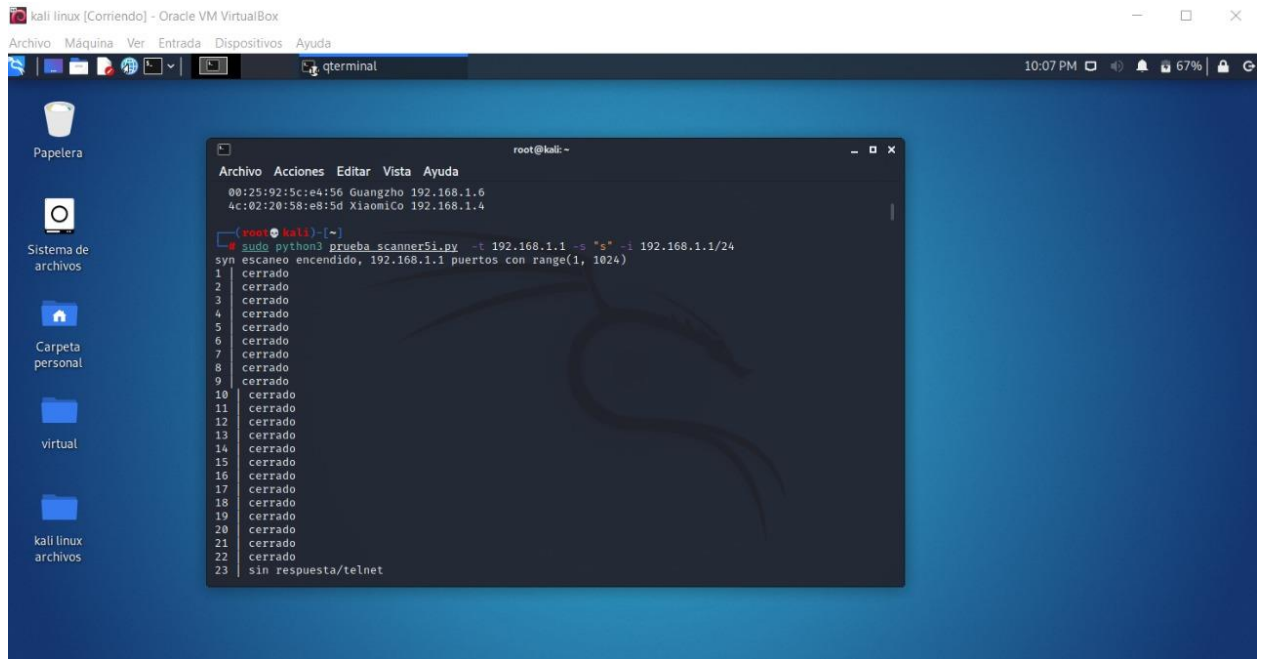


Figura 8. Escaneo de puertos tcp

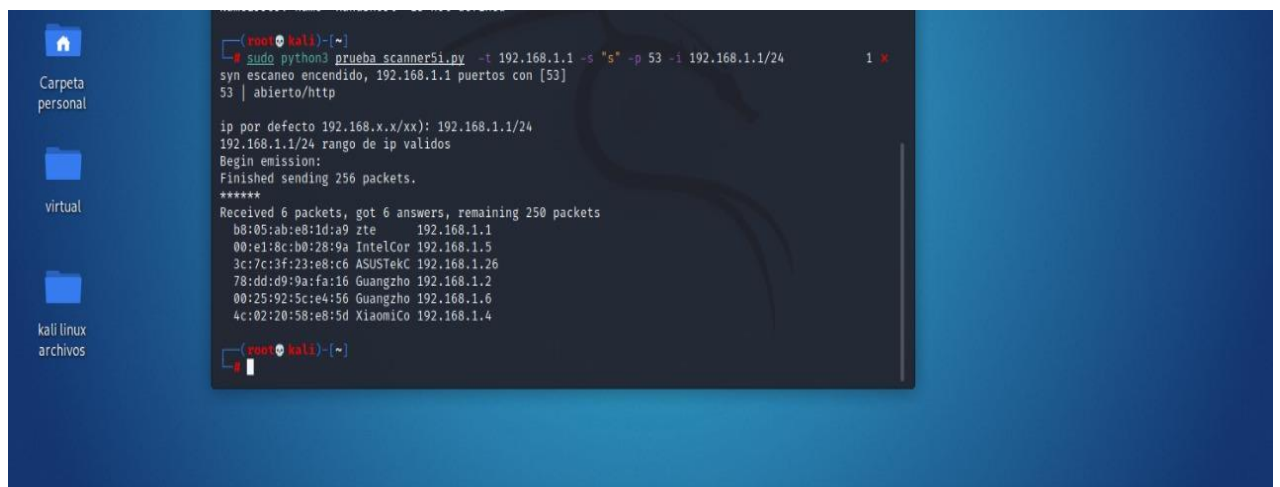


Figura 9. Escaneo de un puerto tcp seleccionado junto con un escaneo de puertos ip

- Luego de realizar el escaneo tcp syn se encontrara que estará solicitando la ip por defecto que en este caso seria 192.168.x.xx junto con eso se tendrá una guía de lo que seria los distintos puertos tcp y su significado.

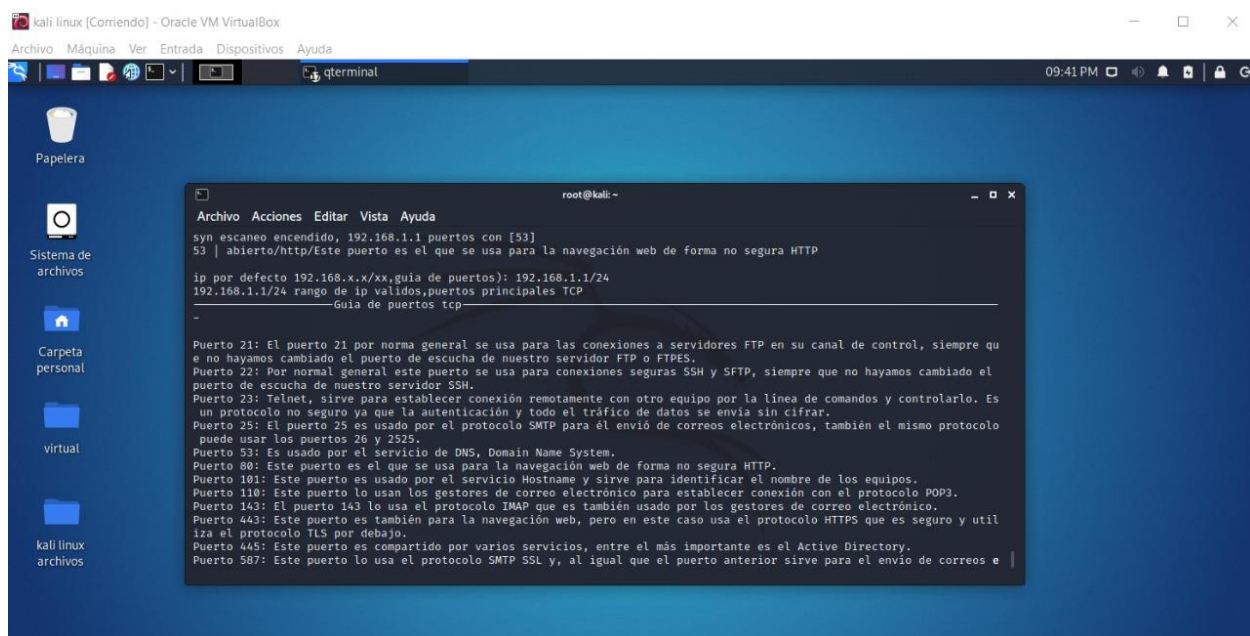


Figura 10. Guía de puertos TCP y su significado

4.1 Situaciones de errores

- es importante que el usuario si desea buscar un el puerto ip ponga los siguientes comandos de -t y -s luego el -i para el escaneo de puertos ip

```
(root@kali)-[~]
# sudo python3 prueba_scanner5i.py -i 192.168.1.1/24
usage: escaneo de puertos usando scapy [-h] -i ARP_RESULT [-g GUIPORTS] -t TARGET
      [-p PORTS [PORTS ...]] -s SCANTYPE
escaneo de puertos usando scapy: error: the following arguments are required: -t/--target, -s/--scantype
```

Figura 10. Error en la búsqueda de direcciones ip

para la búsqueda de puertos tcp igualmente deberá poner los siguientes comando -t(búsqueda de dirección) -s(tipo de escaneo) -i(escaneo de puertos ip)

```
(root@kali)-[~]
# sudo python3 prueba_scanner5i.py -t 192.168.1.1 -s "s"
usage: escaneo de puertos usando scapy [-h] -i ARP_RESULT [-g GUIPORTS] -t TARGET
      [-p PORTS [PORTS ...]] -s SCANTYPE
escaneo de puertos usando scapy: error: the following arguments are required: -i/--arp_result
```

Figura 11. Error en la búsqueda de escaneo tcp syn

- el usuario deberá poner dentro del escaneo de tipo tcp (“s”) tal como esta para realizar el escaneo de tipo SYN caso contrario no podrá realizar el escaneo y le Sandra el siguiente mensaje de “escaneo no permitido”



The screenshot shows a Kali Linux desktop environment with a terminal window open. The terminal prompt is root@kali:~#. The user has entered the command `sudo python3 prueba_scanner5i.py -t 192.168.1.1 -s "k" -i 192.168.1.1/24`. The output of the command is `escaneo no permitido`. Below the output, there is a line `ip por defecto 192.168.x.x/xx):` followed by a cursor. On the left side of the terminal window, the text "kali linux" and "archivos" is visible.

Figura 12. Escaneo no permitido