

Cybersecurity tool box

CIBER MONKEY

Miguel Ángel Roldán de Haro

21 de Abril del 2024

Índice

Índice	1
1. Objetivo	2
2. Instalación	3
● Linux	3
Opción 1 (Larga)	3
Opción 2 (Corta)	4
● Windows	4
3. Interfaz Gráfica	5
4. Herramientas	6
● Opción 1 : Calculadora CIDR	6
● Opción 2 : Wifi Scanner	7
● Opción 3 : Descubrir IP activa	8
● Opción 4 : Escucha de puertos	9
● Opción 5 : Listar subdominios	9
● Opción 6 : Ataque de DDoS	10
● Opción 7 : Crear contraseñas	11
● Opción 8 : Encriptar o desencriptar archivos	13
● Opción 9 : Generar datos	16
● Opción 10: Inyección SQL	17
● Opción 11 : Auditar Bases de Datos SQL	18
● Opción 99 :Salir del programa	18
5. Conclusión	19

1. Objetivo

Mi proyecto personal conocido como Ciber Monkey tiene como objetivo principal profundizar en el funcionamiento de herramientas de pentesting a través de su desarrollo propio, al mismo tiempo que fortalece mis habilidades como programador. **Es importante destacar que no me responsabilizo por un uso inapropiado de esta herramienta con fines ilegales, ya que fue concebida con propósitos lúdicos y educativos.**



2. Instalación

● Linux

Opción 1 (Larga)

1. Actualizar/Instalar Librerías con apt:

```
sudo apt update && sudo apt upgrade -y
```

2. Instalar Git:

```
sudo apt install git -y
```

3. Clonar el repositorio Ciber Monkey:

```
git clone git://github.com/Mayky23/Ciber\_Monkey.git
```

4. Entrar al proyecto:

```
cd Ciber_Monkey
```

5. Instalar librerías de Python con pip una vez dentro del directorio de tu proyecto:

```
pip install -r requirements.txt
```

5.1 En caso de error...

```
pip requests pywifi comtypes python-nmap pytz anytree cryptography dnspython  
ipy netifaces scapy libpcap pypcap pymysql colorama GitPython sublist3r pyfigle
```

6. Ejecutar Ciber Monkey:

```
python3 CiberMonkey.py
```

Opción 2 (Corta)

Comando Único

```
sudo apt update && sudo apt upgrade -y && sudo apt install git -y && git clone  
git://github.com/Mayky23/Ciber_Monkey.git && cd Ciber_Monkey && pip install -r  
requirements.txt && python3 CiberMonkey.py
```

● Windows

Para que la herramienta también funcione en sistemas Windows, asegúrate de tener Python instalado en tu sistema y sigue los mismos pasos para clonar el repositorio y instalar las dependencias utilizando pip. Una vez hecho esto, puedes ejecutar la herramienta utilizando el comando:

```
python CiberMonkey.py
```

3. Interfaz Gráfica

Una vez ejecutada la herramienta, nos encontraremos con una interfaz similar a la siguiente:



Justo debajo se nos permite escribir el número de la herramienta que deseamos usar, las cuales explicaremos en el siguiente apartado.

4. Herramientas

Ahora haremos un pequeño recorrido por todas las herramientas que nos proporciona esta Tool Box para pentesting.

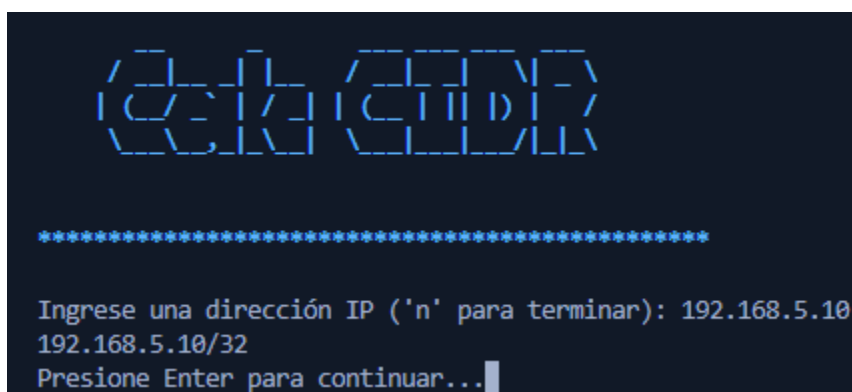
- **Opción 1 : Calculadora CIDR**

Una calculadora CIDR de IP es una herramienta que permite calcular subredes y direcciones IP dentro de una red utilizando la notación CIDR. Esta notación reemplaza al antiguo sistema de clases de direcciones IP y permite una asignación más eficiente de direcciones IP al permitir longitudes de prefijo variables. La calculadora toma una dirección IP base y calcula las direcciones IP disponibles dentro de esa subred.

Es útil para administrar redes y comprender cómo se dividen las direcciones IP en subredes más pequeñas.

Para esta explicación he introducido la dirección IP 192.168.5.10 y nos devuelve 192.168.5.10/32

El "/32" en la notación "192.168.5.10/32" representa la longitud del prefijo de la dirección IP en notación CIDR. En este caso, "/32" indica que la dirección IP se refiere a una única dirección específica en la red. En otras palabras, todas las 32 bits de la dirección IP están dedicadas a identificar este único dispositivo. En términos prácticos, esto significa que esta dirección IP no está dentro de una subred, sino que representa un host individual en la red, sin subdivisiones adicionales.



```
calcidr

*****

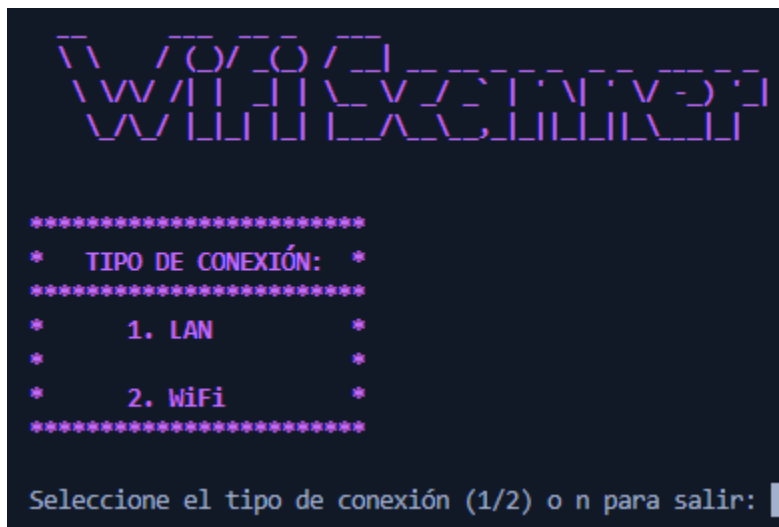
Ingrese una dirección IP ('n' para terminar): 192.168.5.10
192.168.5.10/32
Presione Enter para continuar...
```

Si pulsamos Enter, se nos permitirá insertar una nueva IP en la calculadora, si **escribimos n** y pulsamos Enter, se nos devolverá al menú principal

```
Ingrese una dirección IP ('n' para terminar): n
```

● Opción 2 : Wifi Scanner

Esta herramienta nos permite realizar un escaneo detallado de la red en la que estamos conectados. En primer lugar debemos elegir el modo en el que está nuestro equipo local conectado a la red: 1. Usando un cable LAN o 2. De forma inalámbrica WiFi



En mi caso estoy conectado Por cable LAN , por lo que elijo la opcion 1, acto seguido se nos pregunta el host de nuestra red, es decir la primera dirección de nuestro rango de red, en mi caso es: 192.168.3.1

```

Seleccione el tipo de conexión (1/2) o n para salir: 1
Ingrese la subred de su LAN (por ejemplo, 192.168.1.1/24): 192.168.3.1
Hosts en la red LAN:
IP: 192.168.3.1
█
```


Una vez finalizado el proceso de escaneo (esto puede durar varios minutos) se nos mostrarán todos los puertos abiertos de cada una de las IP dentro de nuestro rango, en mi caso los he ocultado por temas de privacidad

```
Seleccione el tipo de conexión (1/2) o n para salir: 1
Ingrese la subred de su LAN (por ejemplo, 192.168.1.1/24): 192.168.3.1
Hosts en la red LAN:
IP: 192.168.3.1
Puertos abiertos:
Puerto: 
Puerto: 
Puerto: 
Puerto: 
Puerto: 
-----
IP: 192.168.3.5
Puertos abiertos:
Puerto: 
Puerto: 
Puerto: 
Puerto: 
Puerto: 
Puerto: 
Puerto: 
Puerto: 
Puerto: 
Puerto: 
Puerto: 
Puerto: 
-----
```

En cambio si seleccionamos la opción 2, el escaneo de red nos mostrará el nombre de todas las redes a nuestro alcance, además de otra información relevante como el tipo de seguridad de cada red, el tipo de cifrado y los puertos abiertos.

Si **escribimos n** y pulsamos Enter, se nos devolverá al menú principal

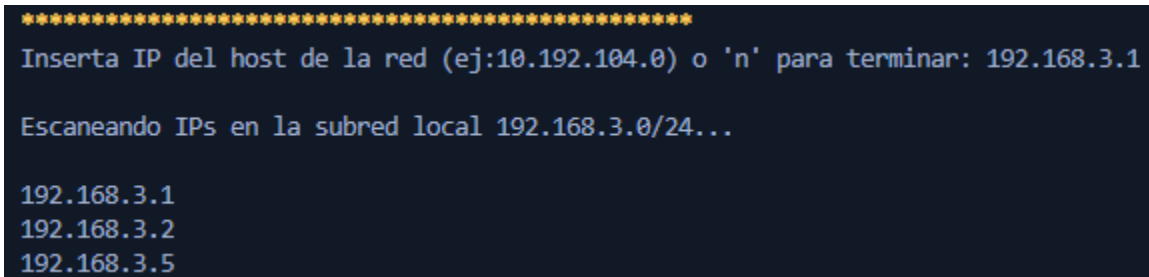
- **Opción 3 : Descubrir IP activa**

Esta herramienta nos permitirá saber las Ip activas dentro de una red a la que estemos conectados



Escribiendo la dirección de host de nuestra red, en mi caso 192.168.3.1 se nos hará un escaneo rápido de las IPs que están activas.

Si **escribimos n** y pulsamos Enter, se nos devolverá al menú principal

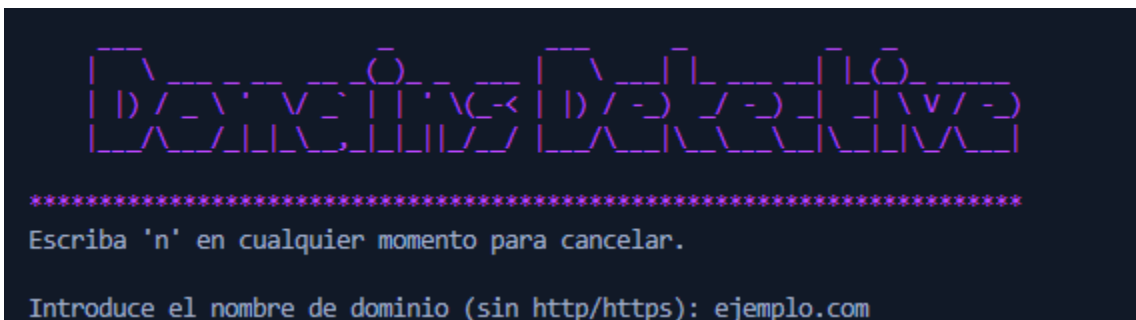


- **Opción 4 : Escucha de puertos**

- **Opción 5 : Listar subdominios**

En esta herramienta, haciendo uso de la herramienta Sublist3r, podremos listar todos los subdominios de una web además de poder añadirle una serie de restricciones o filtros.

En mi caso usaré <https://ejemplo.com>



A continuación se nos pedirán los valores para los filtros

```

Introduce el nombre de dominio (sin http/https): ejemplo.com
Ingresa el número de hilos (el valor predeterminado es 40):
Ingresa la ruta donde se guardará el archivo de resultados (el valor predeterminado es 'subdominios.txt'): C:\Users\MA\Downloads
¿Quieres especificar los puertos a escanear? (s/n): s
Ingresa los puertos a escanear (separados por comas): 443
¿Quieres ejecutar en modo silencioso? (s/n): s
¿Quieres ejecutar en modo detallado? (s/n): s
¿Quieres habilitar la fuerza bruta? (s/n): s
¿Quieres usar motores de búsqueda personalizados? (s/n): n

```

● Opción 6 : Ataque de DDoS

```

      _____
     /  _  _  \
    /  _  _  \
   /  _  _  \
  /  _  _  \
 /  _  _  \
/  _  _  \

*****
* Pulse Ctrl + C para parar el ataque (una vez activo)
* Escribe 'n' para volver al menu
-----
IP o URL objetivo (sin http o https):

```

En mi caso inserté la IP 192.168.3.15 en la que tengo levantado un servicio web con T-pot, acto seguido el programa empezará a mandar paquetes a esa dirección

```

      _____
     /  _  _  \
    /  _  _  \
   /  _  _  \
  /  _  _  \
 /  _  _  \
/  _  _  \

*****
IP Target: 192.168.3.15
Port: 443
Packets Sent: 1569

```

Si durante el ataque pulsamos Ctrl + C se detendrá de inmediato

```
*****  
[+] Attack Stopped  
* Pulse Ctrl + C para parar el ataque (una vez activo)  
  
* Escribe 'n' para volver al menu  
  
-----  
IP o URL objetivo (sin http o https):
```

- **Opción 7 : Crear contraseñas**

Esta funcionalidad nos permite crear contraseñas seguras teniendo en cuenta varios parámetros

```

*****
¿Desea generar una contraseña? (s/n): █

```

En mi caso haré la prueba con los siguientes valores:

- **Min** : Minúsculas
- **May** : Mayúsculas
- **Num** : Números
- **Espc** : Caracteres especiales

¿Desea generar una contraseña? (s/n): s

Ingrese la longitud de la contraseña: 15

Ingrese las opciones (min/may/num/espc): min/may/num/espc

La contraseña generada es: | =81erFc,iiC&g3

La contraseña generada es: | =81erFc,iiC&g3

¿Desea generar una contraseña? (s/n): ☐


Vamos a comprobar si esta contraseña generada es segura:

✓ ¡Buena contraseña!

- Tu contraseña es resistente al pirateo.
- Tu contraseña no aparece en ninguna base de datos de contraseñas filtradas.

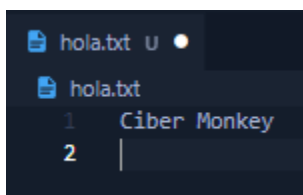
Tu contraseña puede ser descifrada con un ordenador común en...

3261 siglos

⏪  En ese tiempo puedes ir y volver a la Luna 13333 veces. ⏩

● Opción 8 : Encriptar o desencriptar archivos

En primer lugar creamos un archivo para las pruebas



Accedemos ahora si a la herramienta y pulsamos 1 para encriptar

```

_ _ _ O _ _ _ / _ _ _ _ _ _ _ _ _ O _ _ _ _ _
| | | / - > | C | H | Z | Z | / - > | \
| | | \ \ | \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \
*****
*          FILE          *
*****
*    1. Encriptar      *
*                      *
*    2. Desencriptar   *
*                      *
*****
¿Desea encriptar (1) o desencriptar (2)? 'n' para terminar: 

```



rellenamos los parámetros como la ruta del archivo objetivo y la contraseña

```
_ _ _ O _ / _ _ | _ O  
|_| || / -> |(C)| || /-Z T Z- I/-Z T \  
|| ||\ \_ \| \_\_/ \_\_/ \_\_/ \_\_/ ||
```

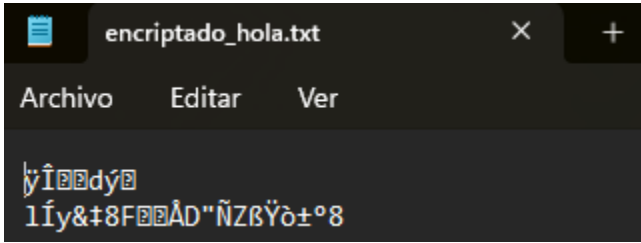
```
*****  
*          FILE          *  
*****  
*    1. Encriptar      *  
*                      *  
*    2. Desencriptar   *  
*****  
  
¿Desea encriptar (1) o desencriptar (2)? 'n' para terminar: 1  
Ingrese la ruta del archivo: M:\GitHub\Hacking_Tool\hola.txt  
Ingrese la clave de encriptación/desencriptación: micontrasena1
```

Ingrese la ruta donde se guardará el archivo encriptado/desencriptado: M:\GitHub\Hacking_Tool

Veremos que se ha creado un archivo nuevo en la ruta especificada anteriormente , este es el archivo encriptado

 hola.txt	26/03/2024 9:30	Documento de te...	1 KB
 encriptado_hola.txt	26/03/2024 9:35	Documento de te...	1 KB




Veremos que pasa si intentamos acceder a él de forma normal usando el bloc de notas



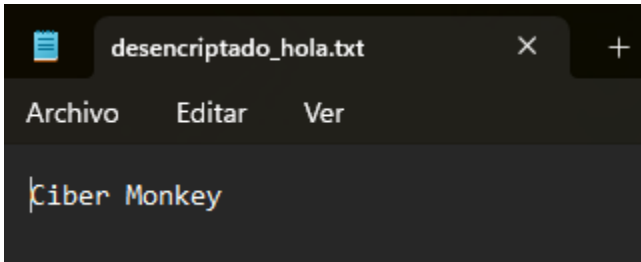
Para poder leer el texto contenido en este archivo volvemos a la herramienta pero pulsamos la opción 2 y debemos insertar la contraseña para descryptar, además de la ruta del archivo



se hará creado entonces un nuevo archivo, este corresponde al archivo completamente descriptado

 hola.txt	26/03/2024 9:30	Documento de te...	1 KB
 encriptado_hola.txt	26/03/2024 9:35	Documento de te...	1 KB
 desencriptado_hola.txt	26/03/2024 9:42	Documento de te...	1 KB

Si accedemos a él veremos que está completamente legible



- **Opción 9 : Generar datos**

Esta herramienta nos permitirá generar datos aleatorios sobre personas para llenar bases de datos y realizar pruebas u otros usos.

```

Data Generator

*****
¿Cuántas personas deseas generar? (Escribe 'n' para terminar): 2

```

Insertamos el número de personas que deseamos generar, en mi caso 2 y pulsamos enter.

```

Data Generator

*****
¿Cuántas personas deseas generar? (Escribe 'n' para terminar): 2
Datos de la persona #1
-----
DNI: 91846421K
Nombre: Elena Moreno
Email: emorenc@mail.com
Cuenta bancaria: Citibank - 4320216877
Fecha de nacimiento: 2014-02-23
Contraseña: JwBn5iKe
-----
Datos de la persona #2
-----
DNI: 072932600
Nombre: Ricardo Román
Email: rromán@example.com
Cuenta bancaria: Citibank - 5978233407
Fecha de nacimiento: 2015-08-10
Contraseña: #GI^Q*nB
-----

```

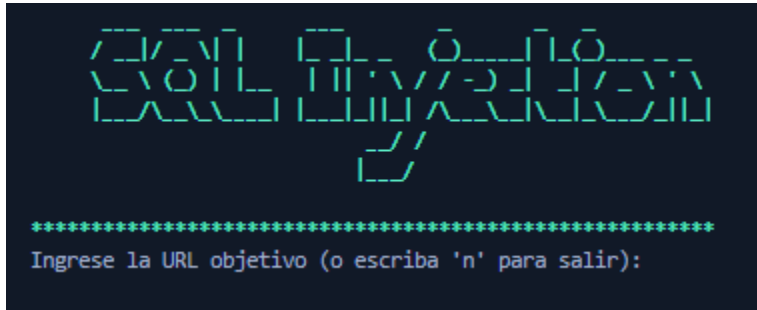
Esto nos devolverá datos interesantes como DNI, fecha de nacimiento, cuenta bancaria...

Si **escribimos n** y pulsamos Enter, se nos devolverá al menú principal

- **Los Datos son generados de forma aleatoria, cualquier coincidencia con la realidad es pura coincidencia**

- **Opción 10: Inyección SQL**

En esta ocasión esta herramienta se encarga de realizar inyecciones SQL sobre webs



Insertamos los datos requeridos como se muestra a continuación:



En mi caso realicé las pruebas usando una web alojada en una IP de mi rango de red y no encontré ninguna vulnerabilidad. En caso de haber encontrado alguna vulnerabilidad se nos mostrará que la transacción ha sido exitosa y con qué payload ha sido posible

```
print(Fore.BLACK + Back.GREEN + f"Inyección de SQL exitosa con payload: {payload}")
```

Ninguna web legítima fue vulnerada durante esta práctica.

- **Opción 11 : Auditar Bases de Datos SQL**

Esta opción está todavía en desarrollo, pero la idea principal de esta herramienta es poder auditar bases de datos SQL desde Python y poder encontrar tablas que almacenan datos de forma incorrecta o con errores de formato.

- **Opción 99 :Salir del programa**

Si escribimos 99 en el menú principal cerraremos la herramienta por completo.



```

  ____  ____  ____  ____  ____  ____  ____  ____  ____  ____
 |  _ \|  _ \|  _ \|  _ \|  _ \|  _ \|  _ \|  _ \|  _ \| | | | | | | | | | | | | | | | | | | | | | | | | | | |
 | |_| | |_| | |_| | |_| | |_| | |_| | |_| | |_| | |_| |
 |  __| |  __| |  __| |  __| |  __| |  __| |  __| |  __|
 |_____|_____|_____|_____|_____|_____|_____|_____|_____|
                                     |
----- By: MARDH -----|

=====
| 1. CALCULAR CIDR
| 2. WIFI SCANNER
| 3. DESCUBRIR IP ACTIVA
| 4. ESCUCHA DE PUERTOS
| 5. LISTAR SUBDOMINIOS
| 6. ATAQUE DDoS
| 7. CREAR CONTRASEÑA
| 8. EN / DESENCRIPTAR ARCHIVO
| 9. GENERAR DATOS
| 10. INYECCIÓN SQL
| 11. AUDITAR BD SQL (disabled)
|-----
| 99. SALIR DEL PROGRAMA
|=====

SELECCIONA UNA OPCIÓN: 99

  ____  ____  ____  ____  ____  ____  ____  ____  ____  ____
 |  _ \|  _ \|  _ \|  _ \|  _ \|  _ \|  _ \|  _ \|  _ \| | | | | | | | | | | | | | | | | | | | | | | | | | | |
 | |_| | |_| | |_| | |_| | |_| | |_| | |_| | |_| | |_| |
 |  __| |  __| |  __| |  __| |  __| |  __| |  __| |  __|
 |_____|_____|_____|_____|_____|_____|_____|_____|_____|

* LinkedIn: https://www.linkedin.com/in/mardh
* GitHub: https://github.com/Mayky23
```

5. Conclusión

Este proyecto ha sido una experiencia invaluable que me ha permitido profundizar en el entendimiento de numerosos ciberataques comunes, al mismo tiempo que he podido fortalecer mis habilidades como programador en Python.

Además, deseo expresar mi sincero agradecimiento a mi profesor Luis Robles del grado superior de Desarrollo de Aplicaciones Multiplataforma (DAM). Gracias a su dedicación y enseñanzas, he adquirido un conocimiento fundamental en la administración de equipos y redes que ha sido fundamental para el éxito de este proyecto y mi inicio profesional en la ciberseguridad.

