

Manual de uso del programa

HERRAMIENTAS DE CIBERSEGURIDAD

Programación para ciberseguridad | 17/05/2021

# **Partes del Proyecto**

El programa cuenta con múltiples archivos para su correcta ejecución

* Script Principal: PIA-PC.py
* Cuatro scripts externos con funciones que se pueden exportar al Script Principal, los cuales son: “Hash\_Script.py”, ”mailPia.py”, “Metadata\_Script.py”, “nmap\_Scripting.py”.
* Dos carpetas, una donde están las imágenes de prueba “ImagenesDePrueba” para ejecutar parte del script de mailPia.py y otra carpeta “PS\_Scripts” donde se encuentra un archivo ps1 “PowerShell” que se exportó correctamente a python.

**Script Principal**

Este archivo PIA-PC.py es la base de todo el proyecto.

* Librerías externas: Aquí es donde se hace el primer y más importante chequeo de paquetes: si el python del equipo no tiene instalado “google” (googlesearch), el programa lee el requirements.txt e instala TODAS las librerías faltantes.
* Importación de funciones: Después de las librerías, este script también importa (desde los demás scripts .py y .ps1), cada una de las funciones locales necesarias para el funcionamiento de las herramientas.
* Creación de la función \_\_main\_\_: No muy usada, pero esencial.
* Creación de funciones internas: Dentro de este script se encuentran las funciones necesarias para la ejecución de las herramientas Socket, Google API y Firewall.
* Definición de argumentos: Este es el único .py que debe ejecutar el usuario y es por eso que define y maneja los argumentos que se pueden ingresar. Para mayor información de un argumento, lea la documentación de las herramientas en las siguientes páginas.

**Hash**

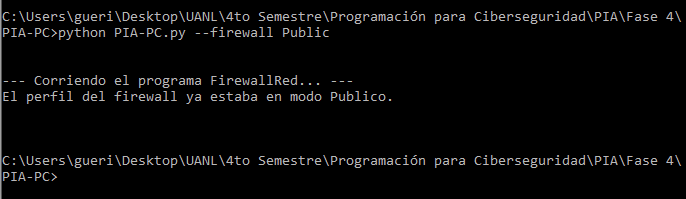
Para la ejecución hash primero hay que detectar cuál va ser el archivos que vamos a convertir en SHA-512 entonces ejecutamos el script con “PIA-PC.py -H (Ruta)” o bien “PIA-PC.py -hash (Ruta)”en ruta es necesario poner la ruta exacta en donde se encuentra el documento en donde se va crear el hash una vez pulsando enter se generará el hash automáticamente de dicho archivo.

**Programa Firewall**

Esta herramienta, por medio de un script de Powershell que está preparado para importarse al proyecto Python, identifica el perfil actual del firewall para luego mostrarlo o cambiarlo a petición del usuario. El argumento es --firewall o -F y sus únicas tres opciones son:

1. “-F Status”: Muestra el perfil actual del firewall.
2. “-F Public”: Establece el perfil como público. Si el firewall ya está en ese modo, la herramienta no cambia nada y solo imprime un mensaje.
3. “-F Private”: Establece el perfil como privado. Si el firewall ya está en ese modo, la herramienta no cambia nada y solo imprime un mensaje.

Ejemplo de ejecución:

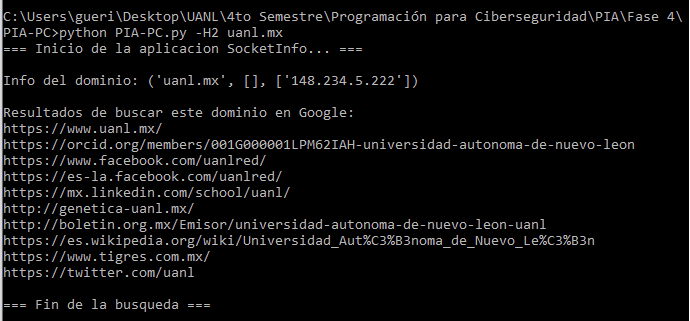


**Socket y Google API**

Estas dos herramientas trabajan juntas dentro de la función interna “SocketInfo()”. Primero se utiliza socket para obtener la información básica (sobre todo la IP), del dominio ingresado por el usuario. IMPORTANTE: El argumento solo acepta dominios, no links completos. Algunos ejemplos de dominios correctos son: “youtube.com”, “uanl.mx”, “twitch.tv”, etc.

Después, google recibe el mismo dominio (confirmado por socket) para luego “googlearlo”. El programa solo muestra los primeros diez resultados para no saturar la pantalla.

Argumento necesario: --hostname o -H2 Ej: “PIA-PC.py -H2 uanl.mx”

Ejemplo de ejecución:  


**Envío de correos**

Para realizar el envío de correos electrónicos es necesario saber cuál será la dirección de correo desde donde se enviara el mail, así como su contraseña para poder logearnos en el servidor correctamente, también será necesario conocer la dirección de correo a la que le llegara el mail y la ruta absoluta del archivo que se quiera adjuntar en este, toda esta información necesaria se enviará mediante argumentos.

Argumentos necesarios: “--remitente o -r1, --pswd o -p, --destinatario o -d, --ruta o -r2”

Ej: “PIA-PC.py -r1 [programa.segu@gmail.com](mailto:progra.segu@gmail.com) -p PiaPseguridad -d [programa.segu2@gmail.com](mailto:programa.segu2@gmail.com) -r2 C:\Users\Usuario\Desktop\adjunto.jpg”

**Metadatos**

Lo que hace este script es que, con la ruta de la carpeta en donde se encuentran las imágenes se pega en donde te dice “ruta de imágenes” después lo que hace este script es analizar cada una de ellas arrojando de esta manera los metadatos. Junto a esto se le añadió “GPSInfo” que te da la feo-localización de la imagen como lat y lon y distintos tipos de información. También se agregó la variable “archivo”para así poder almacenarla y que sea más fácil de leer, encontrarla y guardarla para el usuario.

El script no será capaz de encontrar metadatos si la foto no proviene directamente de un celular o cámara. Las imágenes descargadas de páginas o servicios web (facebook, twitter, telegram, etc.) generalmente no tienen metadatos.

**Programa Nmap**

Para ejecutar el script de nmap es necesario en la terminal de comandos colocarse en la dirección donde está almacenado el script de menú, al estar en dicha dirección lo siguiente será ingresar esto en la terminal: > PIA-PC.py -N -GWI [ingresar Gateway ip] en mi caso sería: > PIA-PC.py -N -GWI 192.168.1.1, si todo ha salido bien lo siguiente que pasara es que analizara la Gateway ip que ingresaste y almacenará y mostrara en terminal la lista de ip de los dispositivos conectados a tu red local, después se hará un escaneo individual ip por ip escaneando un rango de puertos definido entre 70-80 el cual se puede modificar modificando el código en sí en dos variables, being y end, el programa también irá imprimiendo la ip que escanea seguido de el estado de el rango de puertos a analizar, y así sucesivamente hasta terminar con los dispositivos en la lista, también imprimirá un mensaje de error si es que ocurrió un fallo al analizar un puerto, seguido de esto se analizará nuevamente hasta obtener un resultado y finalmente creará un archivo con el nombre ReporteNmap.csv donde almacenará los resultados obtenidos.

