¿ Alguien dijo Keylogger ?

FI - UNAM 2018

Presentan:

- Ochoa Ríos Luis Ernesto

/Leor8a

/Leor8a

- Palomeque González José Alonso

/palomeq87

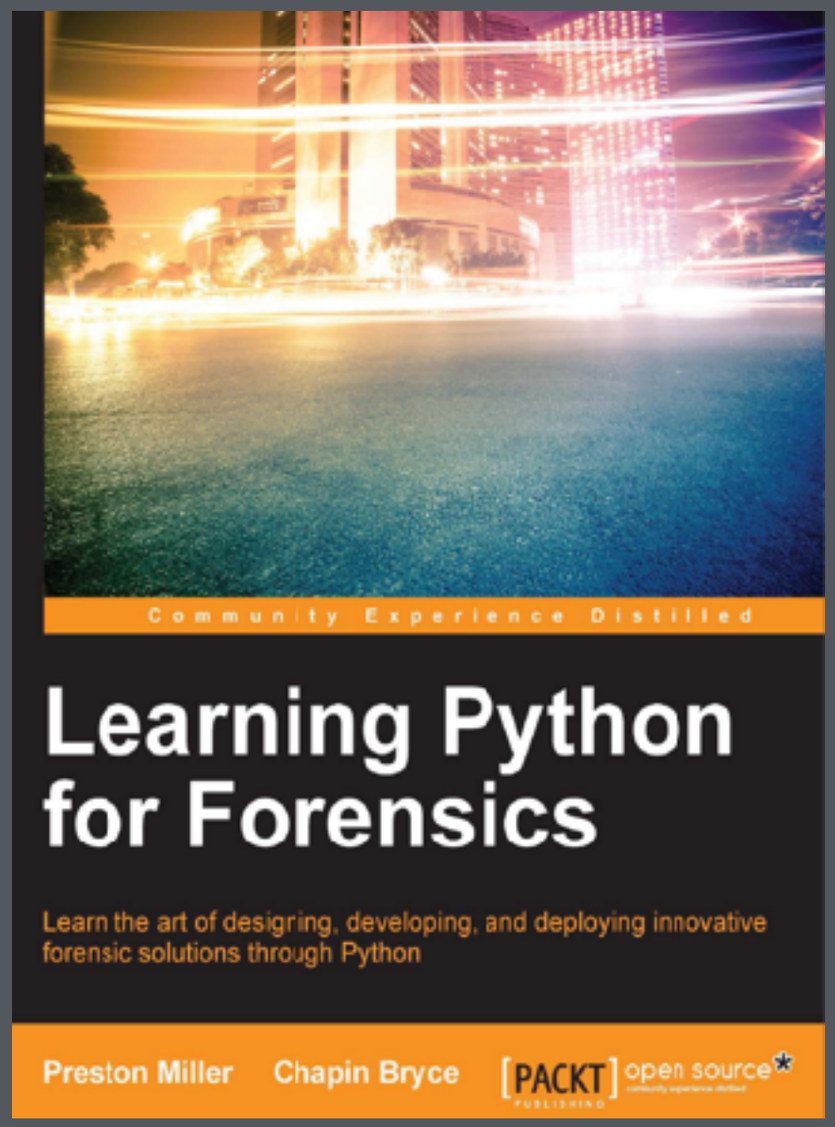
/karons0

ATENCIÓN

La siguiente información puede ser BIEN utilizada o MAL utilizada, nos deslindamos de toda MALA práctica.

Nosotros usaremos la información para fines EDUCATIVOS y PREVENTIVOS.

El libro de donde obtuvimos la información



Learning Python for Forensics

Capítulo 10

Muchos temas interesantes

Preston Miller, Chapin Bryce

(Todo es crédito de ellos...)

Derivado del inglés:

⌘ Key (Tecla)

⌘ Logger (Bitácora, Registrador)

De una manera más formal:

⌘ Son bien sabidas utilidades que pueden capturar pulsaciones de teclas y también sirven para obtener otro tipo de información del Runtime. Pueden ser tanto en Hardware como en Software.

Actualmente son herramientas que muchas empresas utilizan para medir el rendimiento de sus empleados, aunque éstos no lo sepan.

¿En serio?, así de fácil...🤯

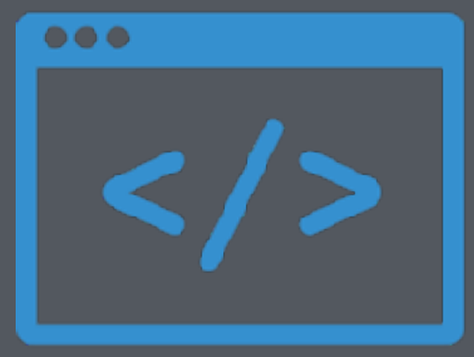
Pero si da un poco de miedo.

Keylogger

Man in the middle,
(Necesitamos un Intermediario)

HARDWARE

SOFTWARE



- Interactúan a Bajo Nivel.

- El código permite introducir el uso de Python con la API de Windows.

- Usaremos varias librerías de terceros que nos permiten comunicarnos con el Sistema Operativo y usar el lenguaje de Python.

Analizando el Keylogger

¿Porqué se combina un Monitor de Procesos con un Monitor Keylogger?

Usando la API de Windows

pyWin32: Windows API para las librerías de Python.

pyHooks: es una librería utilizada para interactuar con la Hooks API de Interfaces de Windows. Ésta nos permite monitorear gran variedad de eventos incluyendo mensajes de eventos, grabar macros y capturar eventos en el mouse y en el teclado.

WMI: Windows Management Instrumentation es un framework que nos permite administrar y manejar los Sistemas Windows. Obtendremos de ésta librería los nombres de los procesos, localizaciones, y ejecutar metadatos para nuestro monitor.

pythoncom: Windows define herramientas como los COM's (Component Object Models), son objetos basados en UML los cuales nos permiten compartir información de objetos(POO), sin importar el lenguaje con el cuál estemos programando. Pythoncom es una librería hecha con python para poder comunicarnos con estos objetos generalizados de Windows.

Analizando el Keylogger

El tiempo lo es todo...

Registraremos los procesos que se abren junto con el tiempo registrado. Podemos pedir todo tipo de información de los procesos, gracias a las librerías...

Como la fecha de creación, la ruta de ejecución del programa, el ID del proceso, entre otra (y mucha) información...

5 mini programas = un resultado final

1. Monitoreando los eventos del teclado

2. Obteniendo las capturas de Pantalla

3. Obteniendo información del Portapapeles

4. Monitoreando Procesos (de la API de Windows)

5. Correr programas de Python sin la necesidad de una terminal

6. Hay que juntarlo TODO...

Monitoreando los eventos del teclado

Obteniendo las capturas de Pantalla

Obteniendo información del Portapapeles

Monitoreando Procesos
(de la API de Windows)

Correr programas de Python sin la necesidad
de una terminal

No necesitamos de una ventanilla 🤫
(En lo oscurito, tras bambalinas ... se ve mas bonito)

Hay que juntarlo TODO...

Mucho que leer, poco tiempo que perder...

¿Dudas? ¿Sugerencias? ... (No dudes en preguntar)

- Ochoa Ríos Luis Ernesto

/Leor8a

/Leor8a

- Palomeque González José Alonso

/palomeq87

/karons0