Introducción

El objetivo principal es proveer los conocimientos sobre el funcionamiento de un malware, como se propaga, sus síntomas y consecuencias. Generalmente para el proceso de análisis de este tipo de amenazas, se distinguen dos tipos:

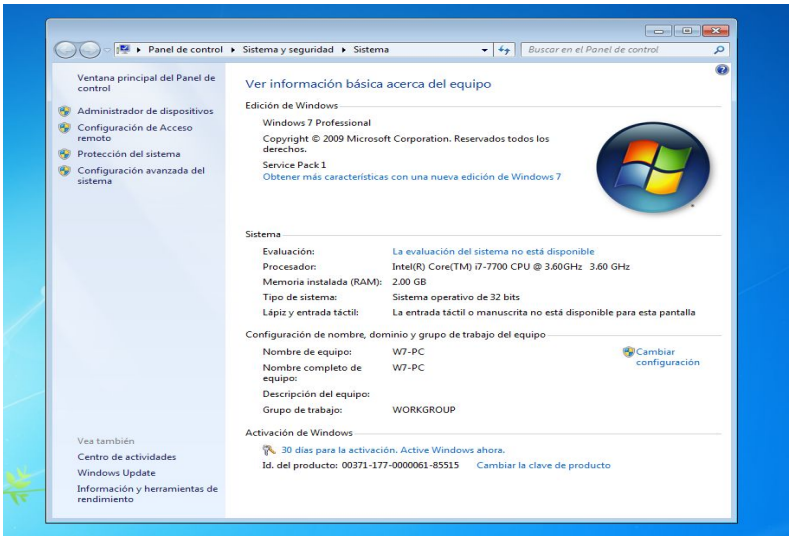
- **Análisis estático de malware:** También conocido como análisis de código, con el propósito de revisar el código binario sin ejecutarlo para comprender como se encuentra estructurado y su propósito. El proceso involucra distintas herramientas y técnicas que permitan determinar la parte maliciosa del programa o archivo. Así como obtener toda la información posible de la funcionalidad, recolectando indicadores técnicos o formas como nombres, hashes, tipo y tamaño de los archivos.

- **Análisis dinámico de malware:** Implica ejecutar el malware para conocer la manera en que interactúa con el sistema, sobre todo el impacto que genera después de la infección, también se le conoce como análisis de comportamiento. Revelando información como nombre de dominios, directorios, llaves de registro, direcciones ip, DLL, archivos de instalación.

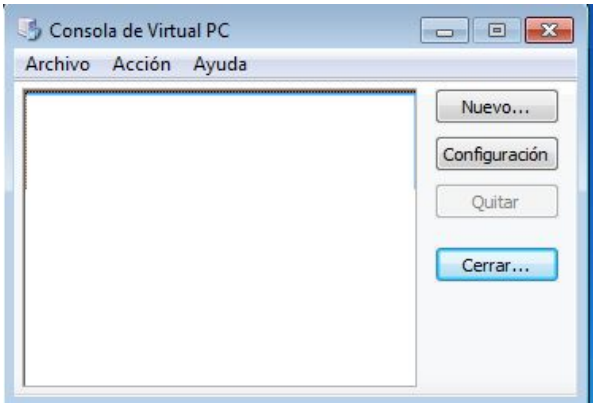
Desarrollo

1. Primeramente, se debe preparar el ambiente de trabajo por lo cual se utilizará la herramienta virtualbox con una máquina virtual con las siguientes características:

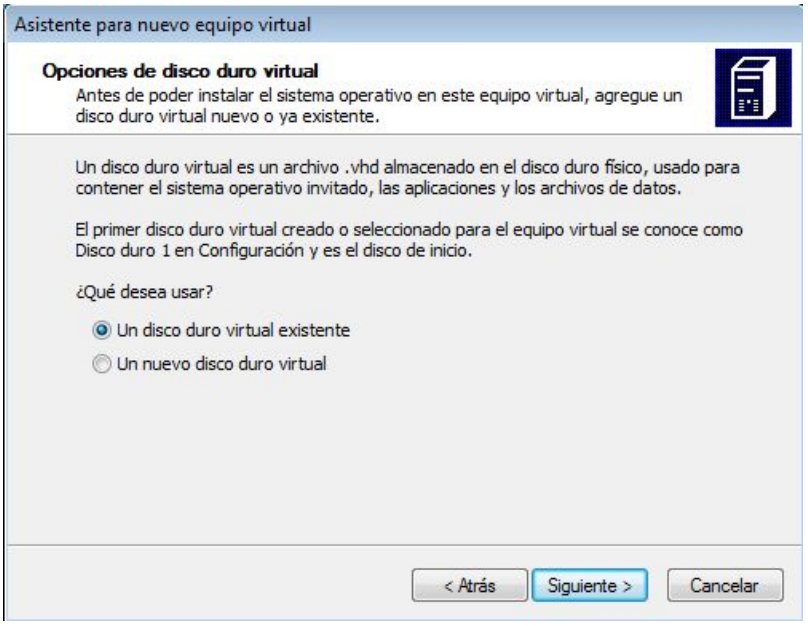
    - Sistema operativo Windows 7 de 32 bits.
    - 2Gb memoria RAM.
    - 30Gb disco duro.
    - Para aislar la red configuramos el adaptador de red como “Host only”
    - Deshabilitar carpetas compartidas.



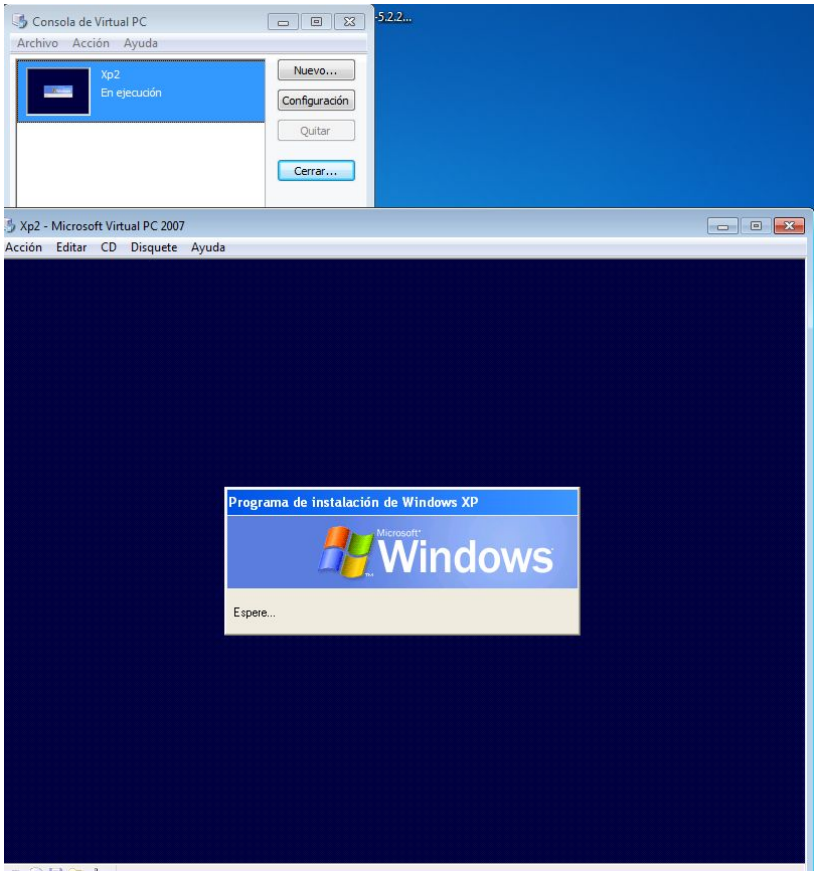
2. Instalación de Windows XP Mode en Windows 7 requiere que se cuenta con Microsoft virtual pc.



3. Se crea una nueva máquina virtual con el disco de Windows XP mode.



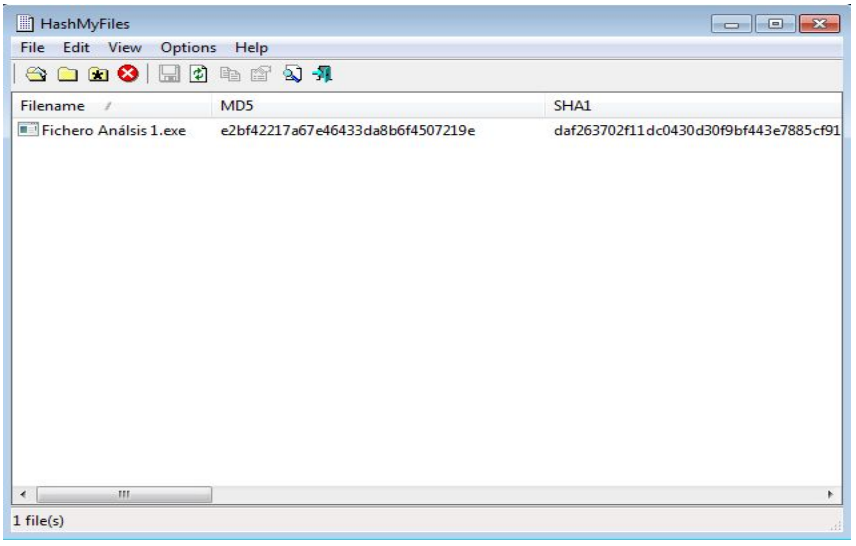
4. Una vez instalado se inicia la máquina virtual con las configuraciones requeridas y se procede al análisis.



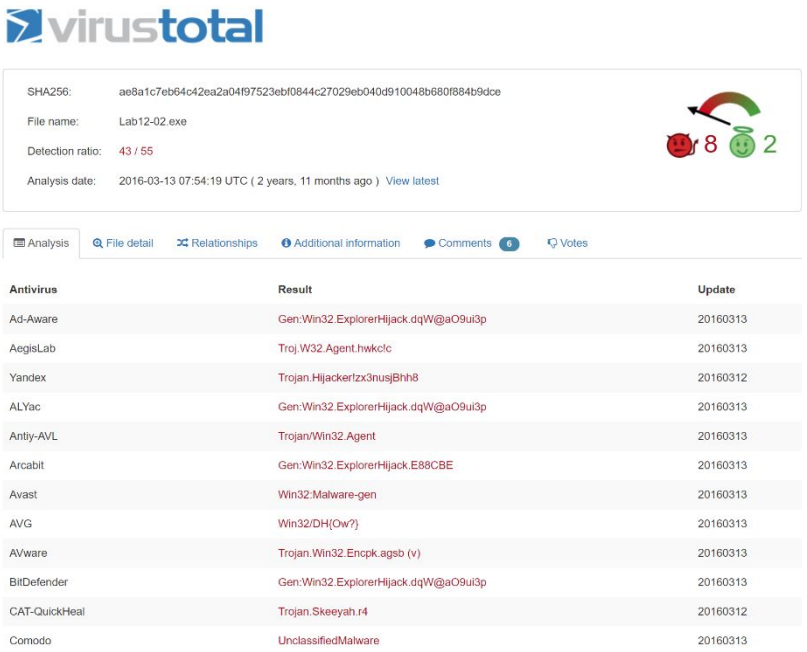
Análisis estático:

1. Aunque no es parte de este trabajo, se realizara un pequeño análisis estático para tener más información del ejecutable, por ejemplo, se verificara el valor hash del archivo con la herramienta “HashMyFiles”.

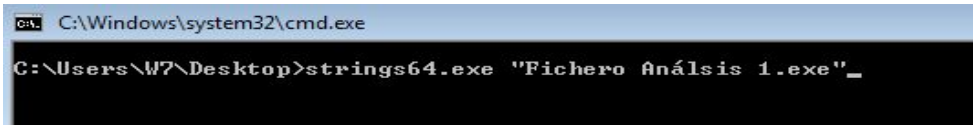
    - **SHA256:**ae8a1c7eb64c42ea2a04f97523ebf0844c27029eb040d910048b680f884b9dce

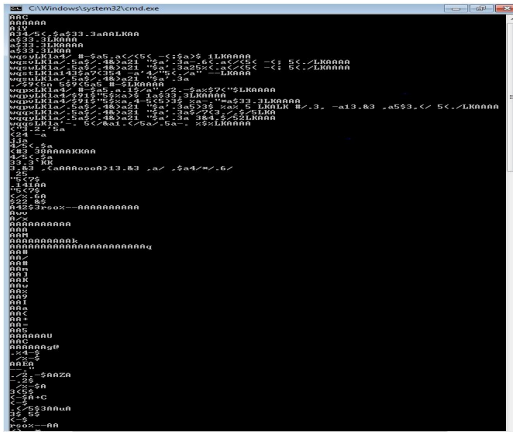


2. Verificando en la página virustotal las firmas hash, distintos antivirus lo catalogan como “Troyano”.

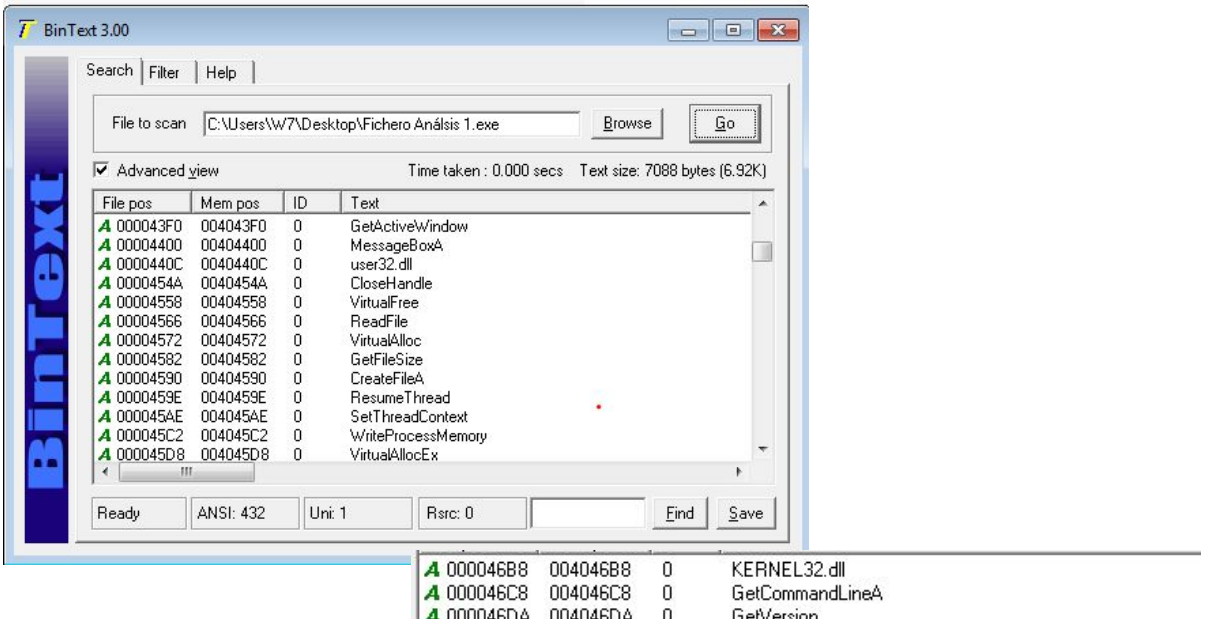


3. En este paso se tratarán de obtener las cadenas dentro del ejecutabl, que puede contener comandos para funciones específicas, para identificar posibles actos maliciosos del programa como: lectura de la memoria interna o cookies, etc.

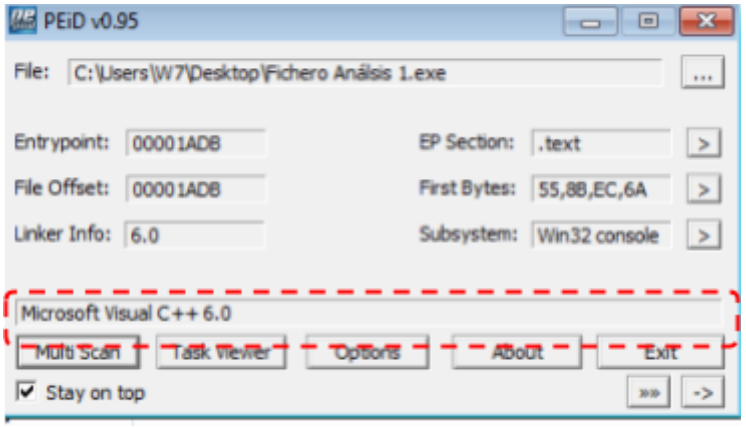




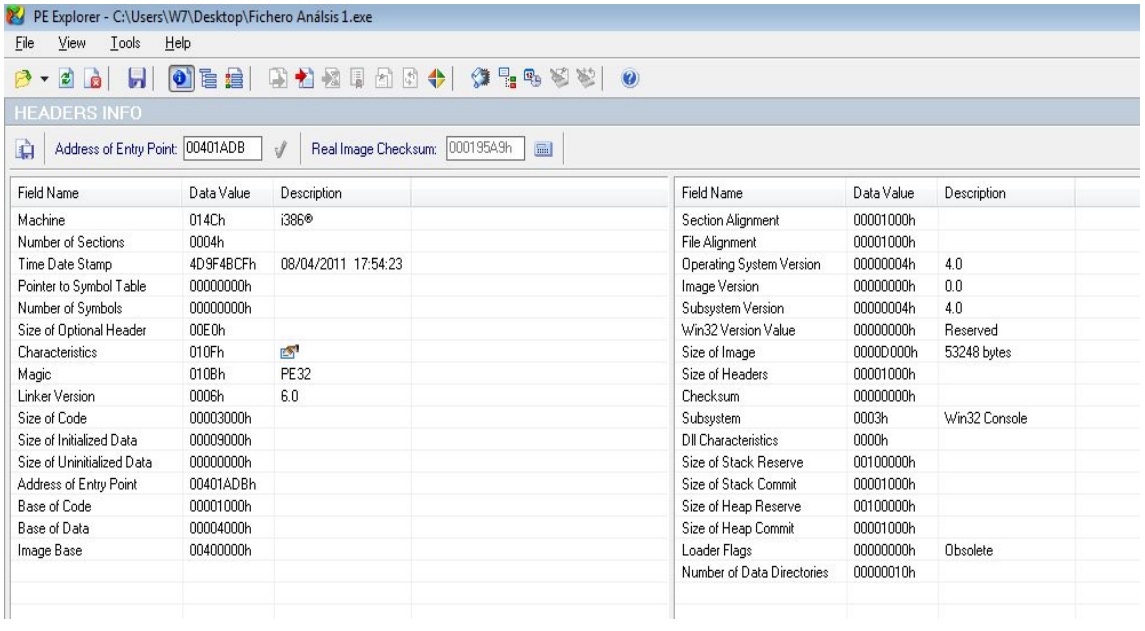
Para corroborar la información obtenida anteriormente y poder visualizarse más fácilmente se puede ejecutar la herramienta Bintext, el cual al finalizar el análisis aparecerán listadas todas las palabras contenida, encontrando cadenas bastante interesantes como algunos DDL que utiliza.



4. Lo siguiente es detectar si la amenaza utiliza alguna técnica de ofuscación, con la herramienta PEid. Además, muestra los primeros bytes para identificar el tipo de archivo y que herramienta se utilizó en su compilación, en este caso Visual C++ 6.0 y no se encuentra empaquetado.

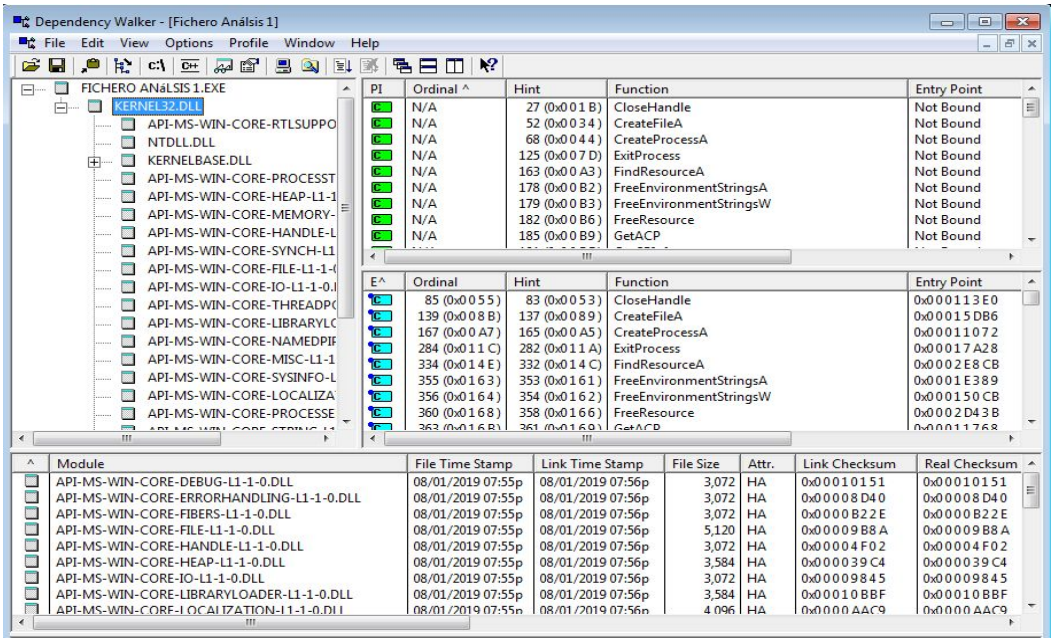


5. Otra herramienta que puede brindar información relevante sobre el formato Portable Executable (PE) o archivos ejecutables es PE Explorer, que contiene datos como fecha de creación o modificación, funciones utilizadas, compilaciones, DLL's, entre otras cosas.



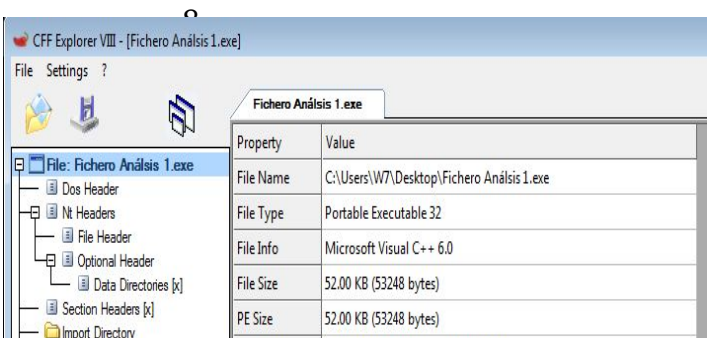
Nota: Se puede apreciar que se realizó o modifico con un equipo de 32 bits de fecha 08/04/2011 a las 17:54 horas

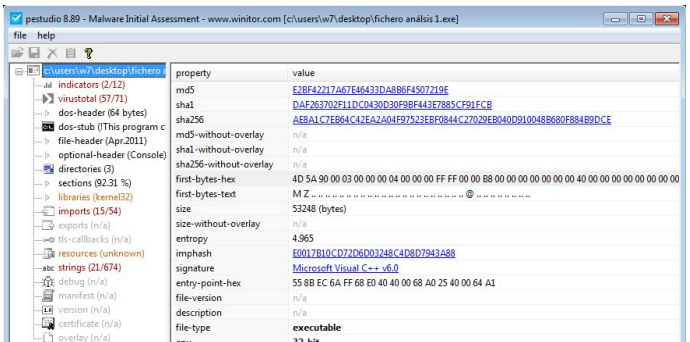
6. Con el propósito de conocer que dependencias utiliza y conocer un poco más su comportamiento al momento de ejecutarse, se puede emplear el programa "Dependency Walker".



Nota: Las dependencias utilizadas son **KERNEL32.DLL** el cual maneja la funcionalidad del Core, memoria, archivos y hardware. Otra biblioteca de enlace dinámico que utiliza es **NTDLL.DLL** que manipula las interfaces del kernel de Windows.

7. Todos estos valores también se pueden obtener fácilmente con las herramientas CFF Explorer y PEstudio, los cuales contienen varios programas utilizadas en esta práctica de análisis estático.

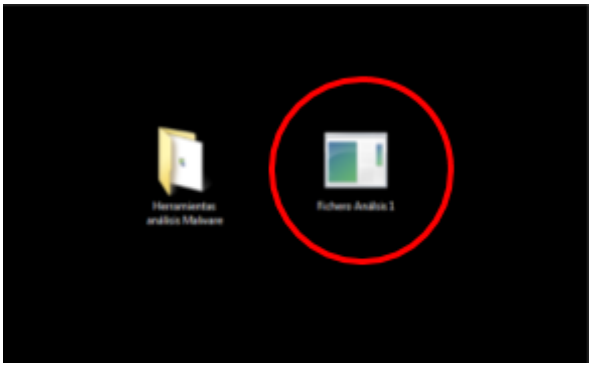




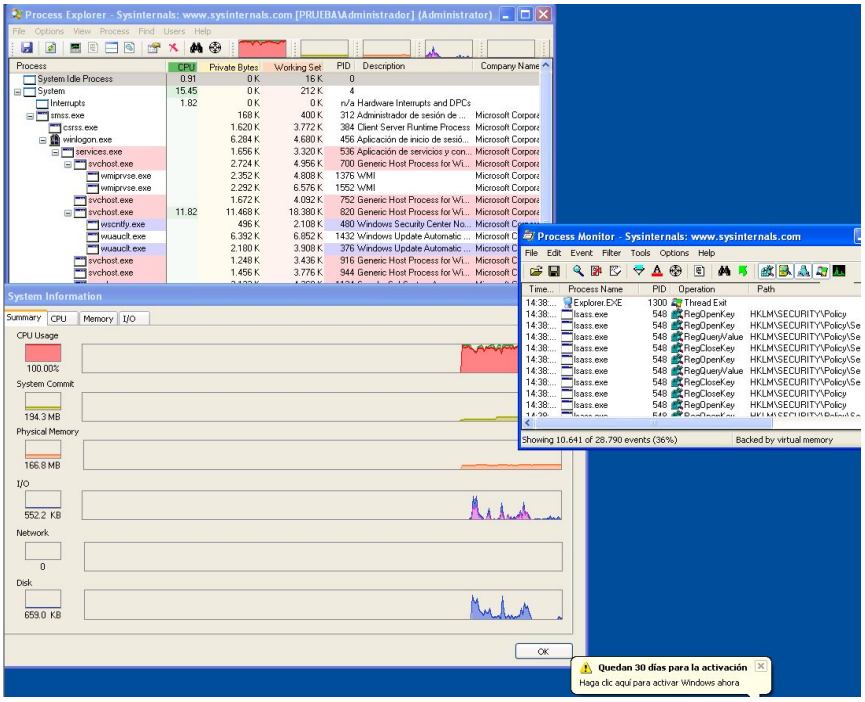
9. Con la información obtenida anteriormente podemos conocer un poco como funciona el malware sin la necesidad de ejecutarlo y tener una idea de los posibles elementos con lo que puede interactuar. Entre las conclusiones que se puede obtener es que cuenta con patrones que indican que intenta provocar un Buffer Over Flow.

Análisis dinámico:

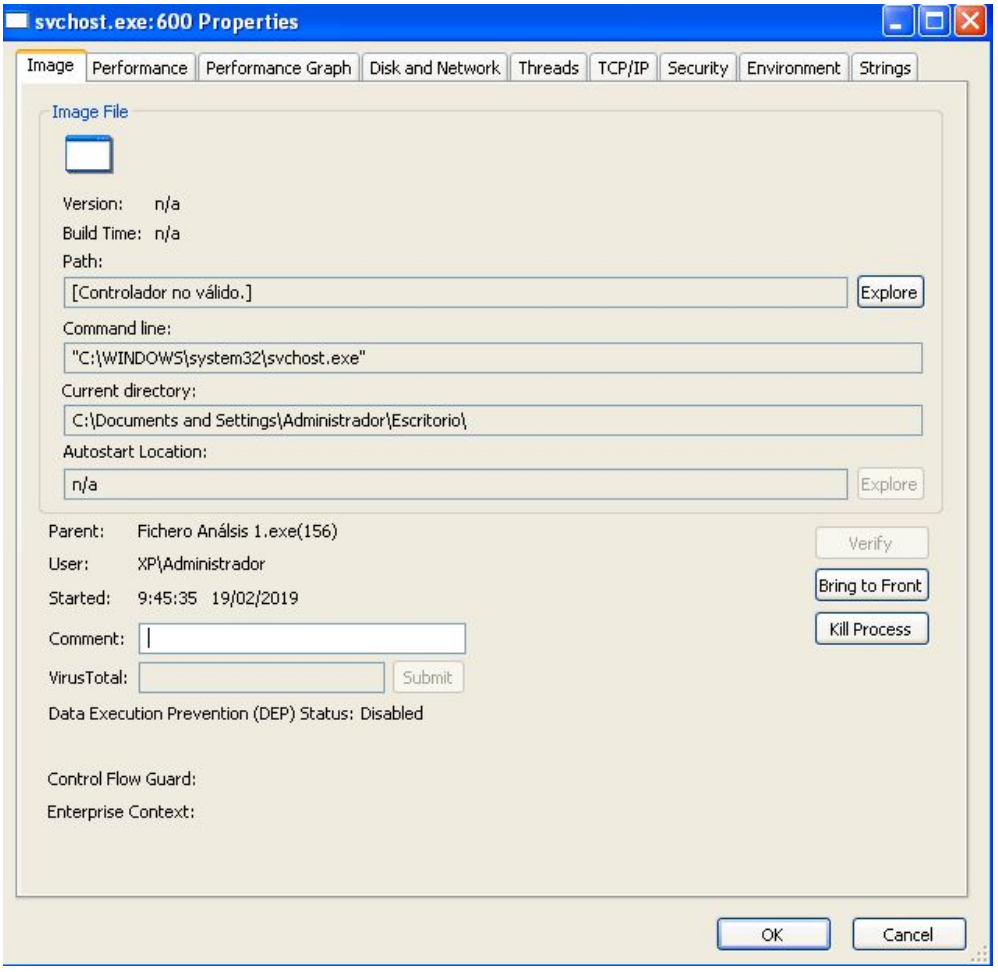
10. El siguiente paso es realizar un análisis dinámico, para eso debemos ejecutar el archivo en la máquina virtual.



11. Antes de ejecutar el archivo analizamos el comportamiento normal del equipo.



12. Se ejecuta el archivo “Fichero Análisis1.exe” para ver su comportamiento. El archivo genera un “svchost.exe” sin firmar con la opción de protección “Data Execution Prevention (DEP)” desactivada.



13. Analizando las conexiones activas los procesos antes mencionados no generan ningún puerto a las escucha o se establece una conexión.

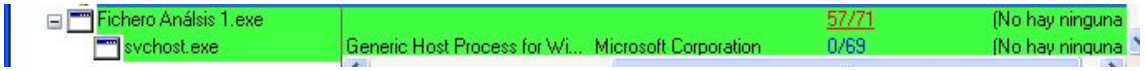
```
C:\Documents and Settings\Administrador>netstat -ano

Conexiones activas

  Proto  Dirección local        Dirección remota       Estado           PID
  TCP    0.0.0.0:135            0.0.0.0:0              LISTENING        776
  TCP    0.0.0.0:445            0.0.0.0:0              LISTENING        4
  TCP    0.0.0.0:3389           0.0.0.0:0              LISTENING        728
  TCP    10.0.2.12:139          0.0.0.0:0              LISTENING        4
  TCP    127.0.0.1:1025         0.0.0.0:0              LISTENING        1640
  UDP    0.0.0.0:445            *:*                                     4
  UDP    0.0.0.0:500            *:*                                     556
  UDP    0.0.0.0:4500           *:*                                     556
  UDP    10.0.2.12:123          *:*                                     844
  UDP    10.0.2.12:137          *:*                                     4
  UDP    10.0.2.12:138          *:*                                     4
  UDP    10.0.2.12:1900         *:*                                     1008
  UDP    127.0.0.1:123          *:*                                     844
  UDP    127.0.0.1:1026         *:*                                     844
  UDP    127.0.0.1:1900         *:*                                     1008
```

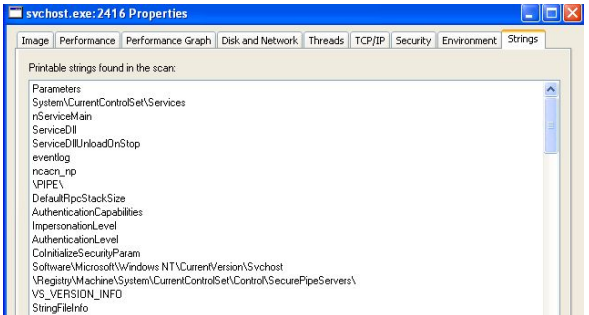
¿Qué observas al supervisar este malware con Process Explorer?

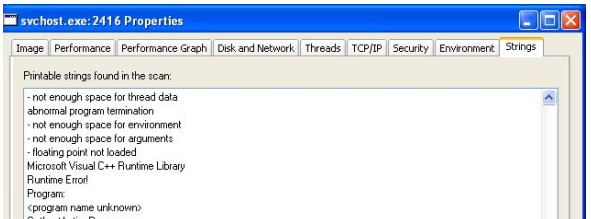
Se puede observar que al momento de ejecutarse el archivo Fichero Análisis 1.exe se transforma en svchost.exe, que no cuenta con la firma de Microsoft.



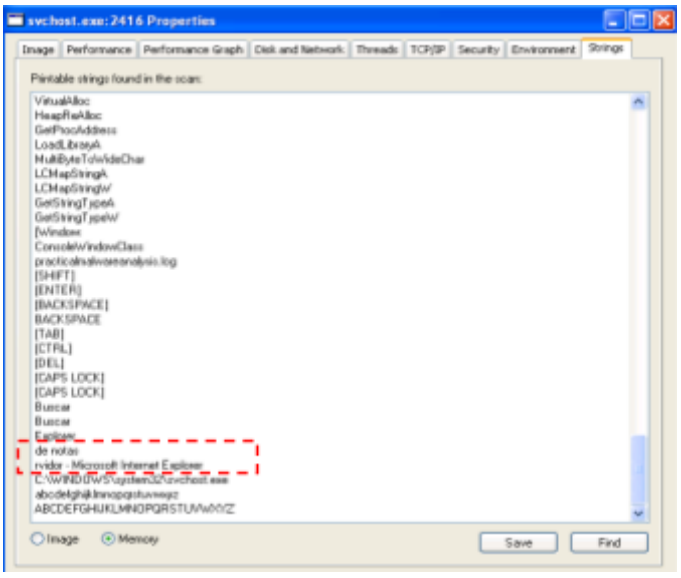
¿Puedes identificar modificaciones en la memoria?

Estos son los Strings que se guardan en la imagen y memoria, pero no concuerdan. Llama la atención que en memoria utiliza la función SetWindowsHookExA, la cual permite captar las teclas presionadas y movimiento del mouse.



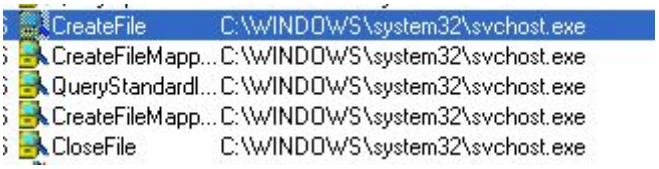


Como se puede apreciar en la siguiente imagen, se va guardando en la memoria del nuevo archivo svchost.exe todo lo que se realiza en el equipo. Como prueba, se abrió un bloc de notas e Internet Explorer y todas estas acciones están registradas y se genera un archivo llamado practicamalwareanalysis.log, en el cual se escriben todos los pasos realizados, con lo que se confirma que el malware tiene la función de keylogger.

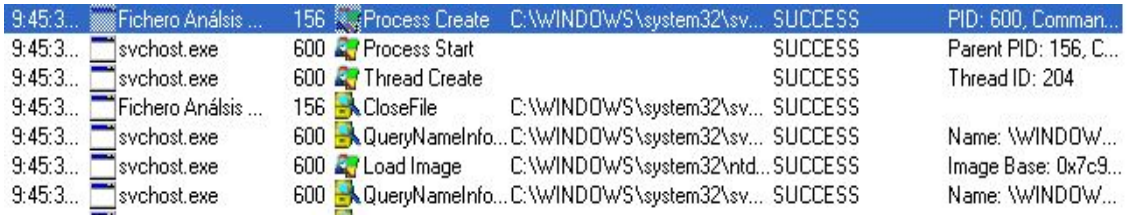


¿Qué archivos crea?

Reemplaza el archivo svchost.exe, pero sin la firma de Microsoft.



Además del proceso con PID 600 a nombre de svhost.exe, como padre el proceso con PID 156 Fichero Análisis 1.exe.



Este proceso crea el archivo “practicamalwareanalysis.log” en escritorio, donde se guardan las acciones realizadas por el usuario, como se aprecia en la imagen de abajo.