Panda Security

Introducción al análisis de malware.

panda

pandasecurity.com

Get-NetNeighbor

- Ximo Molina López
- t.me/expetec

- Yassin Said Esteller
- t.me/saiestyas

- Formamos parte del equipo de PandaLabs como técnicos de Adaptive Defense.

Panda Security

1. Introducción.
2. Laboratorio de análisis.
3. Introducción al análisis estático.
4. Introducción al análisis dinámico.
5. Ejercicios prácticos.

Introducción

¿Porqué aprender reversing?

El objetivo de la ingeniería inversa es obtener información acerca de un producto, para ello se determinan los diferentes elementos que lo componen, como estos interactúan y como fueron desarrollados.

En el ámbito del software la ingeniería inversa nos ayuda a comprender como funciona un programa o aplicación.

De este modo, podemos ver como un elemento afecta a nuestro sistema o incluso entender como llega a realizar ciertas tareas.

Introducción

¿Porqué aprender reversing?

La seguridad informática es el área de la informática y telemática que se enfoca en la protección de la infraestructura computacional y todo lo relacionado con ella.

Realizar tareas de ingeniería inversa nos brinda información muy útil como:

- Vectores de entrada. ¿Por qué canal ha llegado el malware?
- Indicadores de compromiso. ¿Qué hace sobre nuestro sistema?
- Tareas de mitigación o prevención. ¿Qué vulnerabilidades explota? ¿Podemos remediarlo?
- Impacto sobre la compañía. ¿Hubo fuga de datos?

Introducción

Tipos de amenazas

Podemos encontrar multitud de clasificaciones y maneras de definir al malware, esta es una de las más comunes:

- Minner: Obtiene recursos de la máquina para realizar minado de criptomonedas.
- Information stealer: Su objetivo es extraer información de un sistema para luego retransmitirlo al atacante.
- Backdoor: Código malicioso que se auto-instala con el fin de brindar acceso al atacante.
- Botnet: Similar al backdoor, solo que en este caso existe una red de máquinas esperando recibir ordenes desde un command and control.
- Downloader: Se encarga de descargar más contenido malicioso.
- Rootkit: Diseñado para ocultar su presencia y ejecutarse con privilegios de administrador.
- Ransomware, cifra ficheros y exige el pago de un rescate para eliminar la amenaza.

Introducción al análisis estático

Laboratorio de análisis – Primeros pasos

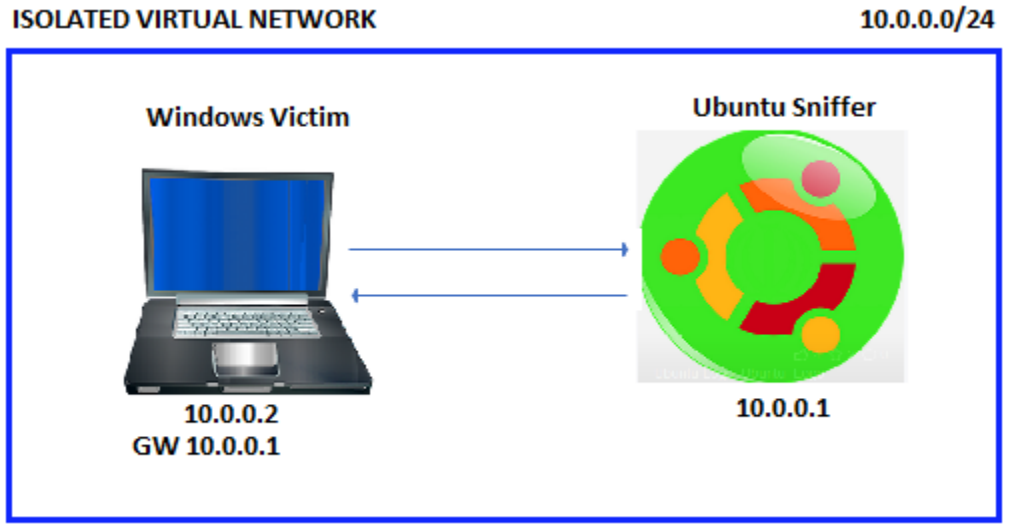
Para ello las herramientas que serán necesarias son:

- VirtualBox
- Una imagen de Windows, descargable de forma gratuita desde https://developer.microsoft.com/en-us/microsoft-edge/tools/vms/
- Internet Simulation Suite (InetSim)
- Una máquina complementaria, cualquier distribución de linux, para realizar tareas de sniffer.

Introducción al análisis estático

Laboratorio de análisis – Arquitectura de red.

En primer lugar deberemos crear dos máquinas virtuales una Windows y otra Linux. La arquitectura de red debe solo permitir la comunicación entre las dos máquinas pero no con el exterior.



Introducción al análisis estático

Laboratorio de análisis – Arquitectura de red.

Crear un laboratorio de malware es algo muy sencillo. Solo es necesario crear las dos máquinas virtuales y que tengan conexión en una red privada virtual (isolated virtual network).

Una vez instaladas y tengan comunicación a través de esa red privada debemos:

- Crear una instantánea de la imagen Windows y de la imagen Linux. En caso de infección permitirá volver a un estado anterior.
- Configurar el entorno de red.
- Instalar Inetsim en la máquina Linux.
- Instalar todas las herramientas para prodecer al análisis en Windows.

Introducción al análisis estático

Laboratorio de análisis – InetSim

INetSim es un software que permite simular los servicios típicos de internet en un entorno de laboratorio para realizar un análisis del comportamiento de red.

Entre los muchos protocolos que permite similar están:

- http/https.
- smtp/smtps.
- Pop3/pop3s.
- ftp/sftp.
- DNS.
- NTP.
- Etc.

Introducción al análisis estático

Introducción – Análisis estático

El análisis estático describe el proceso de analizar una muestra según la estructura del fichero sin llegar a ejecutarlo.

En esta fase la información a recopilar es:

- Identificación del archivo, hash MD5.
- Motores Antivirus, Virustotal como primera aproximación.
- Obtención de datos acerca del fichero: strings, headers, librerías, etc.

Introducción al análisis estático

Breve descripción de un PE

Cuando hablamos de un **PE** estamos hablando de un fichero **ejecutable** para OS **Windows**.

El resultado de compilar código se traduce en una estructura de datos, PE, que encapsula toda la información que necesita el loader de Windows para administrar el código del ejecutable.

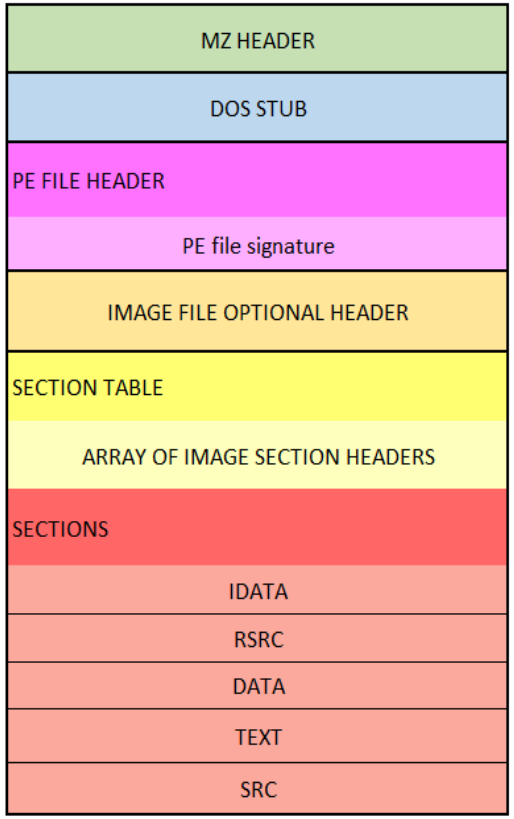
Esto significa que en el PE habrá información no solo relativa al código compilado sino también:

- Identificador de tipo de archivo. Magic number.
- Librerías que necesita el ejecutable.
- Tamaño de fichero en disco y en memoria virtual (en tiempo de ejecución).
- Arquitectura para la que fue compilado (x86 o x64).
- Etc.

Introducción al análisis estático

Estructura básica de un PE

La estructura de un PE representada de forma gráfica y de forma muy resumida puede entenderse como:



Introducción al análisis estático

Descripción de la estructura de un PE

La información almacenada en un PE es de mucha utilidad, veamos una pequeña descripción:

- **MZ HEADER.** Se trata de los primeros 64 bytes de todo PE. Contiene, entre otras cosas, el magic number.
    - El **Magic Number** se encuentra en los **dos primeros bytes** del PE y es el identificativo de que tipo de fichero es. Un **PE** empezará siempre con **4D 5A.**
- **Dos Stub** Usualmente suele contener la clásica cadena: *This program cannot be run in DOS mode.*

Introducción al análisis estático

Descripción de la estructura de un PE

- **PE File Header.** Contiene información relativa al resto de la estructura del fichero:
  - Dirección de memoria donde se encuentra el código.
  - Tamaño del archivo en disco y en memoria virtual.
  - Número de secciones.
  - Dirección de carga cuando el archivo sea mapeado en memoria.
  - Información relativa al alineamiento de las diferentes secciones.
  - Etc.

Introducción al análisis estático

Descripción de la estructura de un PE

- **Section Table:** Contiene información relativa a las secciones, para cada sección habrá una entrada en la section table.
- **Sections**: Contiene la información referida a recursos que necesita el ejecutable (imágnes, strings, etc), espacio de memoria a escribir en tiempo de ejecución, código, etc.

Aunque hay mucha información interesante la estructura del PE, veremos con un poco más de profundidad las **Sections** dado que están **directamente relacionadas con el código**.

Introducción al análisis estático

Descripción de la estructura de un PE - Sections

Las secciones típicamente a encontrar en un PE son:

- **.idata:** Almacena los datos relacionados con los imports que el programa necesita. Si una librería del sistema será cargada, aparecerá en la sección idata.
- **.rsrc:** Contiene los recursos necesarios que no son considerados como parte del ejecutable. Se encuentran imágenes, iconos, strings.
- **.data:** Contiene los datos del programa de entorno global. Son datos accesibles desde cualquier parte del programa.
- **.text:** Contiene las instrucciones que serán ejecutadas por la CPU. El resto de secciones contienen datos de apoyo. Es **la única** sección **que debe contener código.**
- **.reloc:** Esta sección entra en juego cuando ejecutamos el programa y hay que emplazar el PE en memoria. Indica cómo deben ser modificadas las direcciones de memoria del PE para mapear en memoria virtual.

Introducción al análisis estático

Presentación de las herramientas

Estás son algunas de las muchas herramientas para iniciar un análisis:

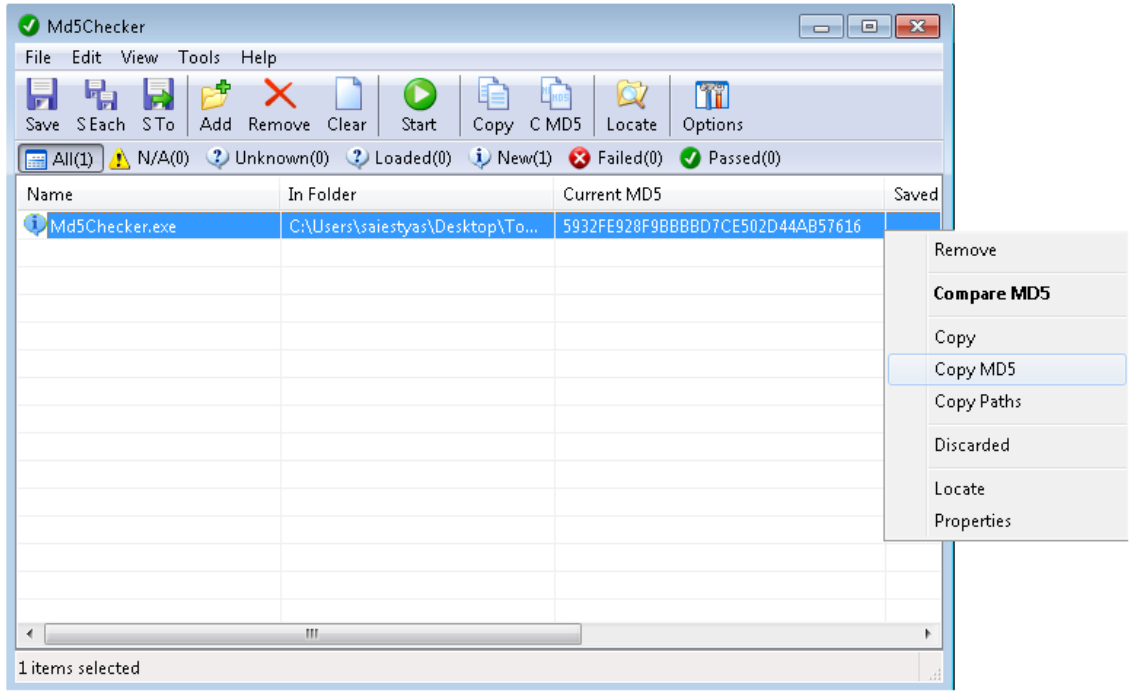
- **MD5Chequer**: Cálculo del hash MD5 del archivo.
- **Virustotal**: Página web donde se permite analizar ficheros con multitud de motores antivirus de forma gratuita.
- **HxD**: Editor hexadecimal.
- **Strings**: Muestra las cadenas contenidas en el PE. (Microsoft Sysinternals Suite).
- **Die – Detect it easy**: Herramienta que muestra si un archivo ha sido manipulado por un packer.
- **Resource Hacker:** Permite la extracción de recursos (.rsrc section). Se usa para extraer o manipular elementos del programa.
- **PeStudio:** Herramienta para investigar cualquier binario de Windows. Contiene mucha información relativa a las secciones y otros metadatos.

Introducción al análisis estático

Presentación de las herramientas - Md5Chequer

Todo archivo objeto de análisis debe ser identificado de forma unívoca. Por ello las muestras son nombradas con su MD5 tanto a nivel profesional como educativo.

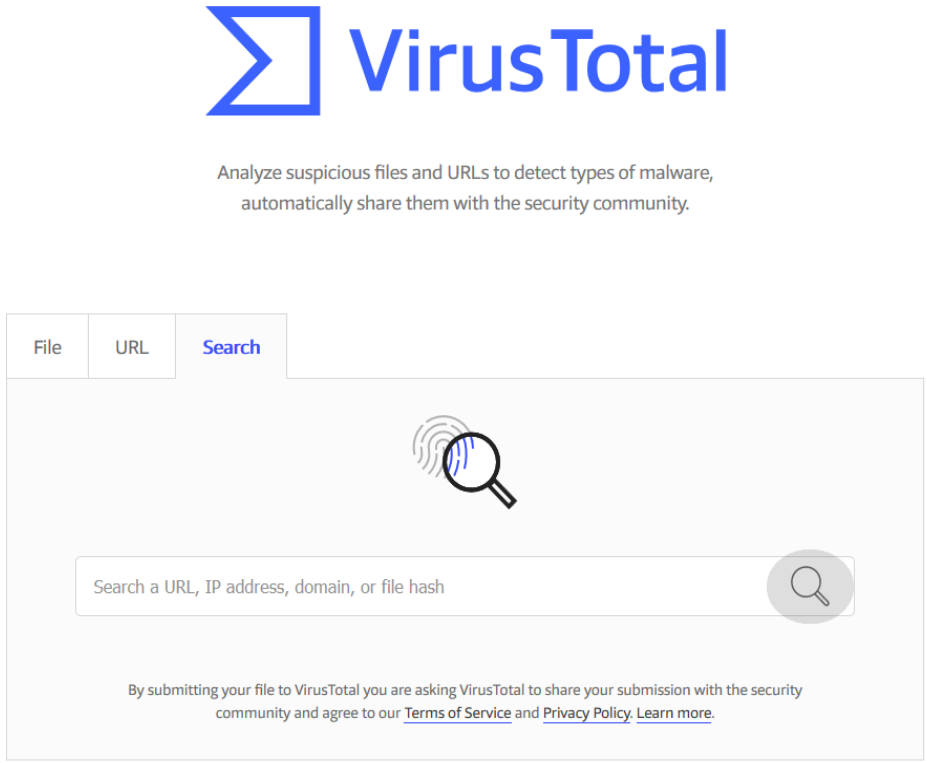
La herramienta MD5chequer es prácticamente auto-explicativa:



Introducción al análisis estático

Presentación de las herramientas – VirusTotal

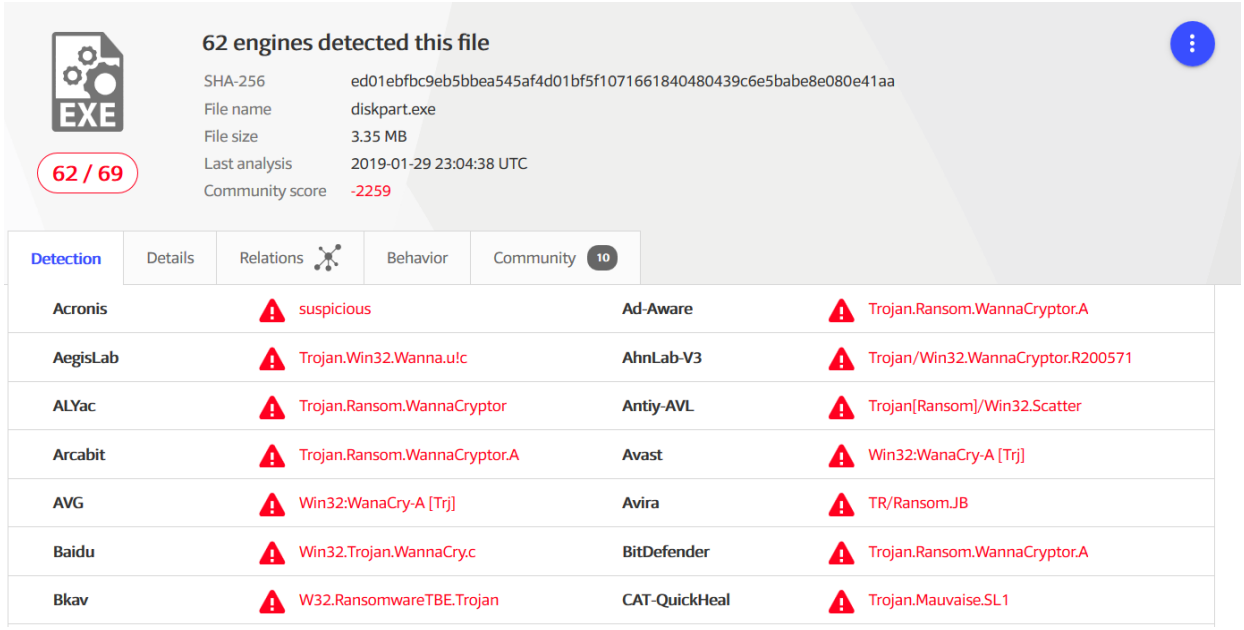
Virustotal es una de las webs más grandes de análisis online y dispone de más de 60 motores antivirus para realizar el análisis. Permite: subida de ficheros, análisis de urls o búsqueda de cadenas que identifiquen al archivo que queramos analizar.



Introducción al análisis estático

Presentación de las herramientas – Virustotal ejemplo de detección

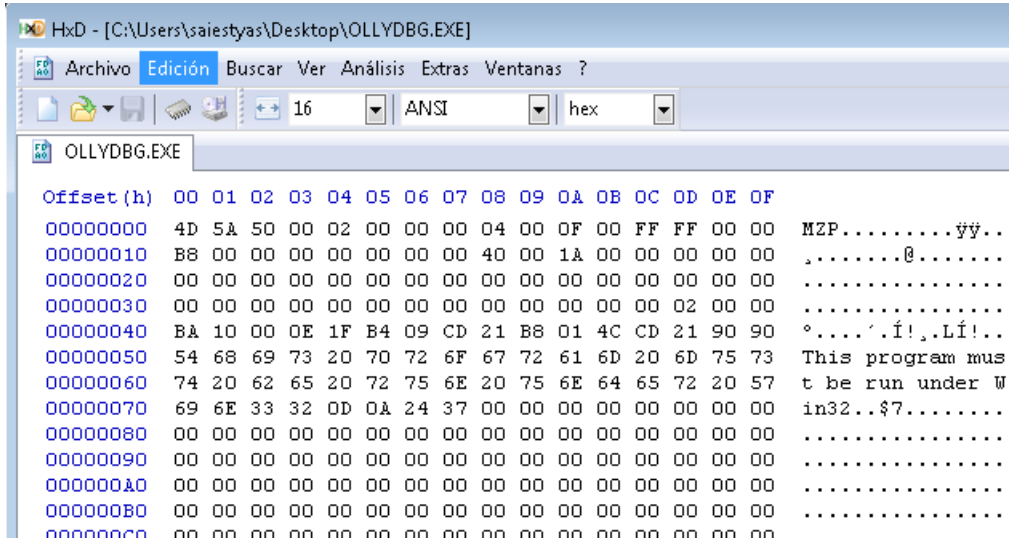
En este caso se ha introducido el MD5 de un malware relacionado con el famoso ataque WannaCry.



Introducción al análisis estático

Presentación de las herramientas – Editor hexadecimal

HxD es uno de tantos editores hexadecimales gratuitos. Dada su simplicidad de uso es objeto de uso:



Puede ser muy útil a la hora de buscar cadenas concretas a primer ojo o cuando queramos modificar bytes del archivo.

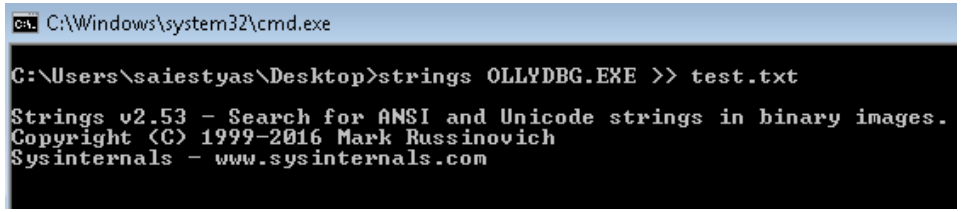
Introducción al análisis estático

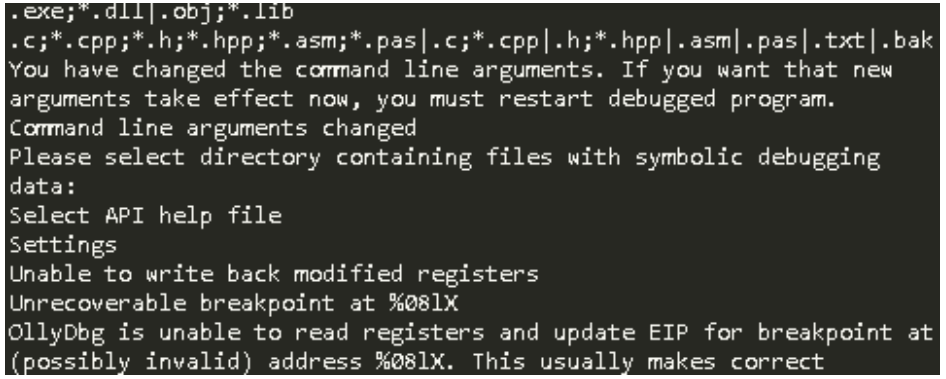
Presentación de las herramientas – Strings

El programa strings pertenece a las **Sysinternals suite** ofrecida por **Microsoft**.

En las Sysinternals Suite encontramos utilidades dirigidas a diagnóstico, traza de errores, monitorización, etc.

Tal y como su propio nombre indica Strings extrae las cadenas en formato ASCII o UNICODE de un PE, permitiéndonos almacenar su contenido en un fichero para su posterior análisis:





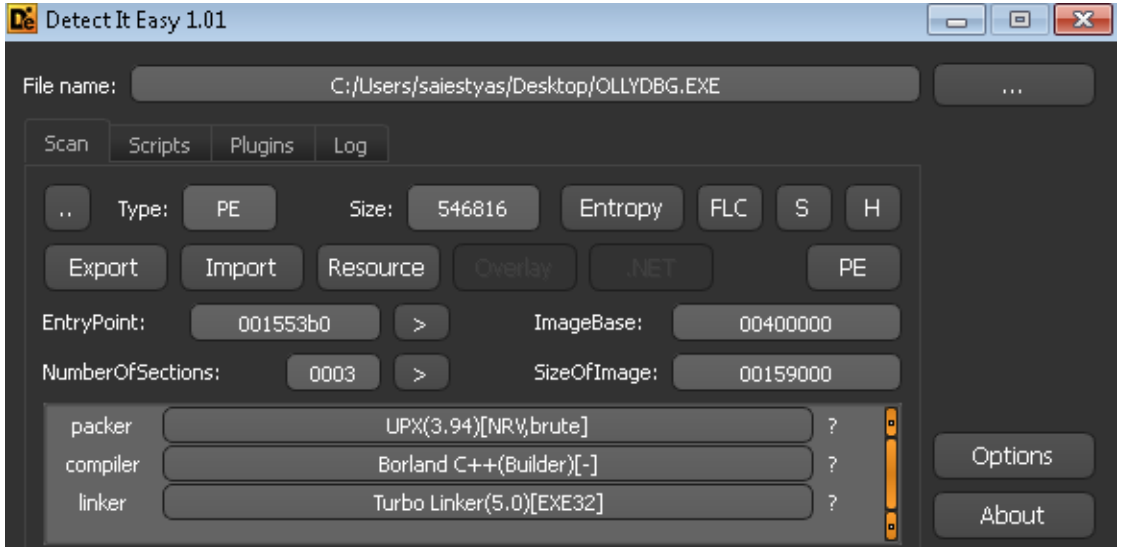
Introducción al análisis estático

Presentación de las herramientas – Detect it Easy (Die)

En muchas ocasiones los programas tanto maliciosos como no maliciosos son protegidos con capas de cifrado y también de ofuscación.

DiE es capaz de distinguir entre un gran abanico de packers junto con su versión.

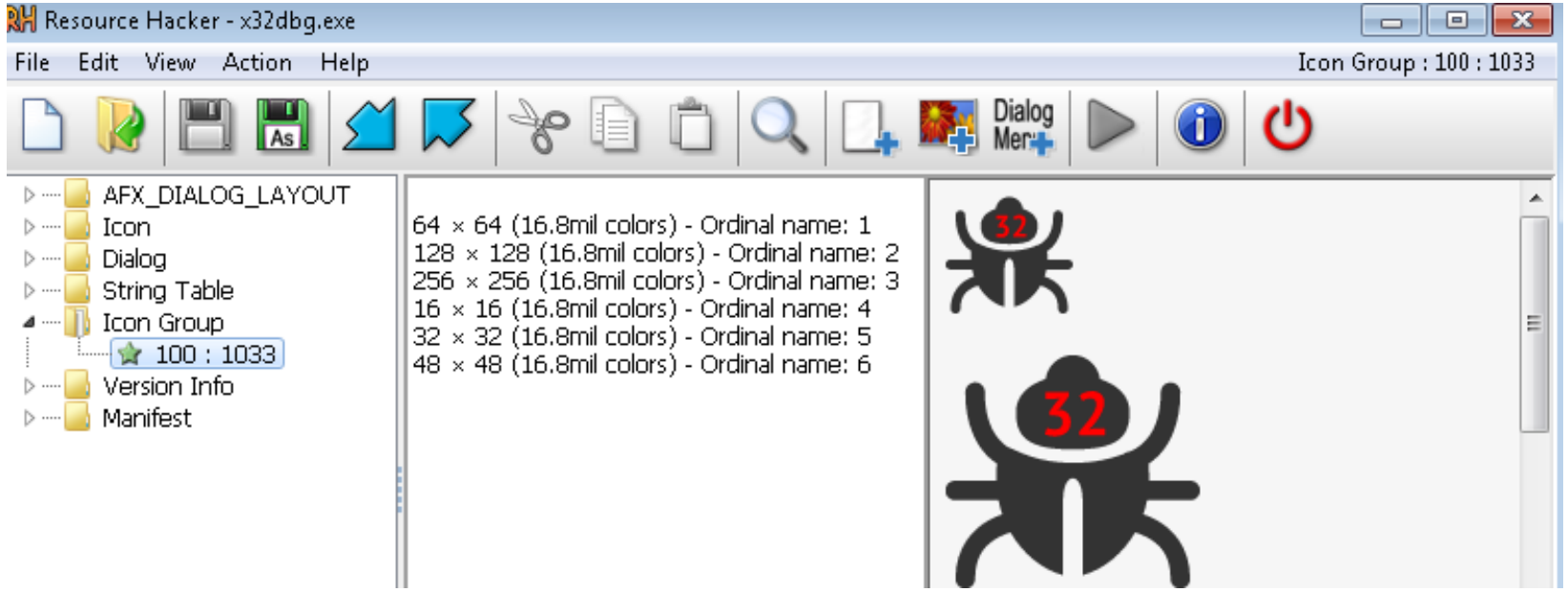
Los packers son esos programas que se encargan de introducir esa capa/s de cifrado u ofuscación.



Introducción al análisis estático

Presentación de las herramientas - Resource Hacker

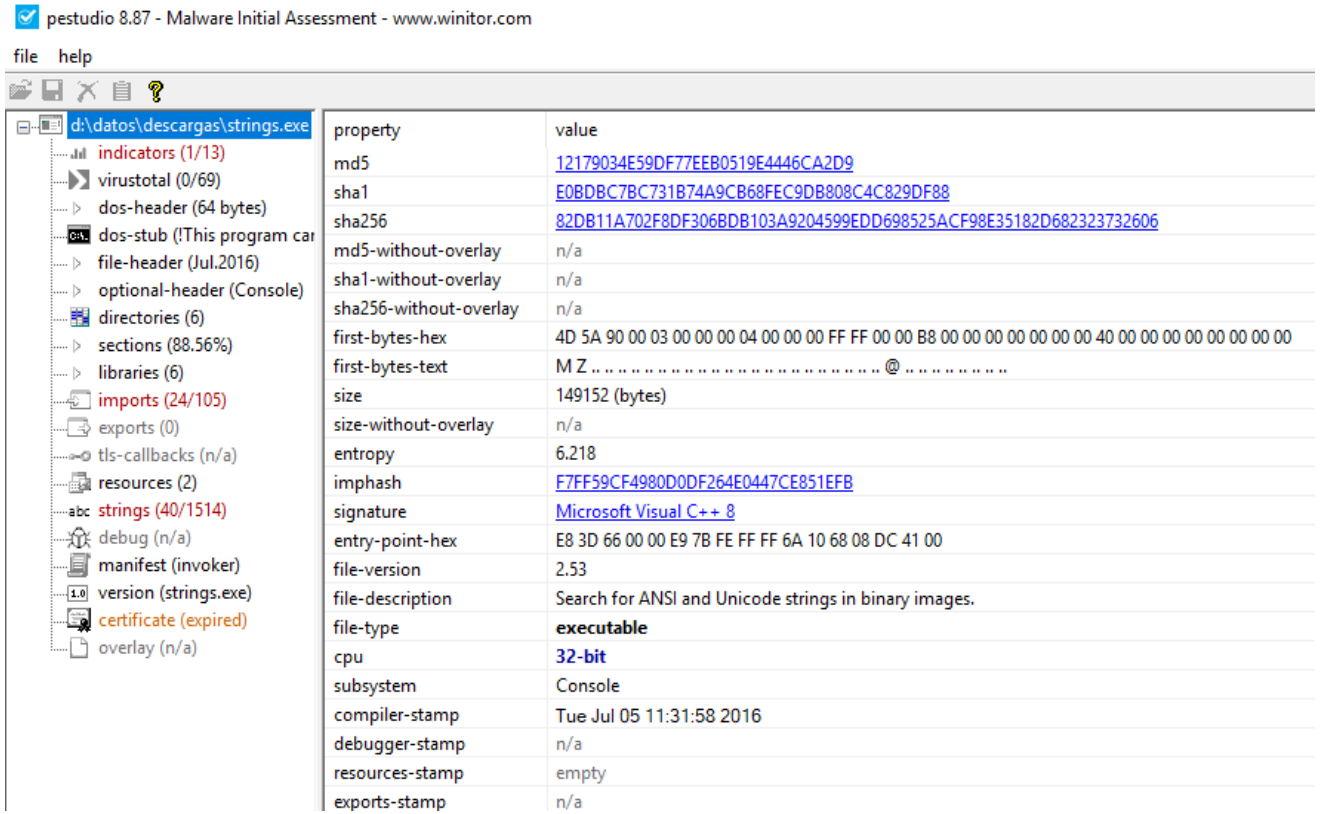
Resource Hacker es una herramienta útil con la que podemos: ver, extraer e incluso modificar recursos (.rsrc) de un PE.



Introducción al análisis estático

Presentación de las herramientas - PeStudio Visión general

PeStudio es una gran herramienta que aglutina funcionalidades de otras ofreciendo en un solo programa una gran suite de herramientas para el visionado de metadatos del PE.

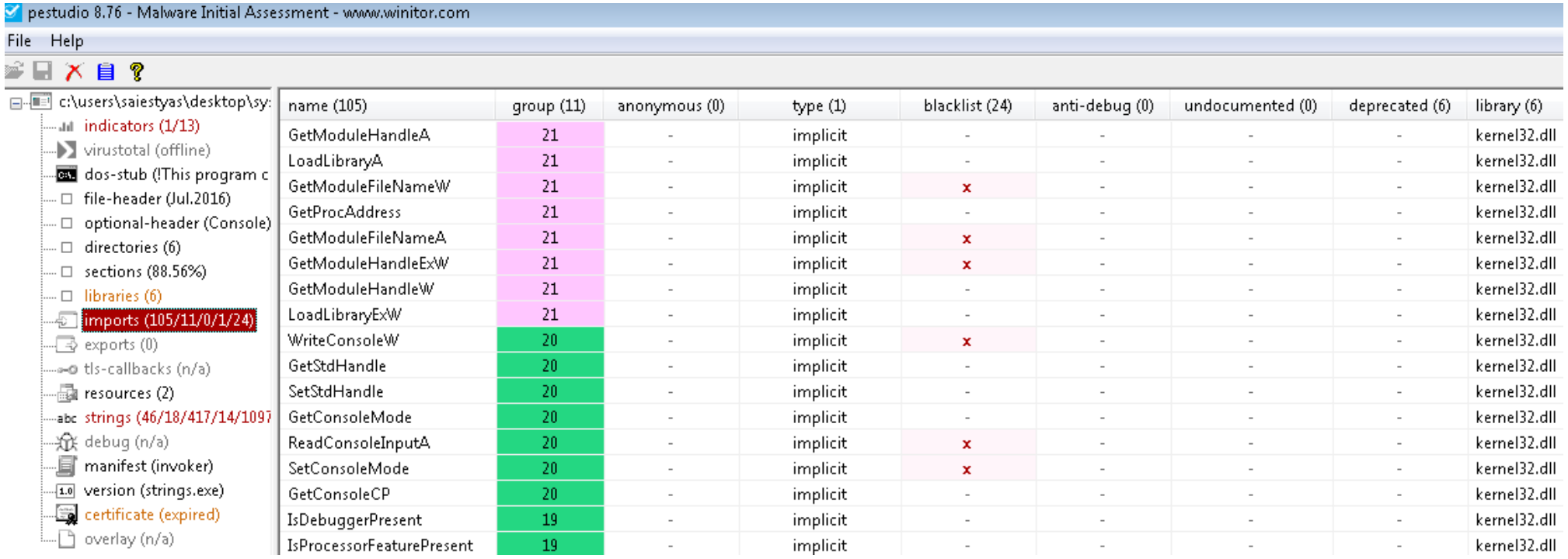


Introducción al análisis estático

Presentación de las herramientas - PeStudio – Imports y Exports

En el apartado **imports** podemos ver todas las **llamadas a funciones** de **librerías externas** realizadas por el programa. Esto puede ser un gran indicador puesto que en el caso que nos ocupa ahora

¿Qué implicaría una llamada a GetConnection?



name (105)	group (11)	anonymous (0)	type (1)	blacklist (24)	anti-debug (0)	undocumented (0)	deprecated (6)	library (6)
GetModuleHandleA	21	-	implicit	-	-	-	-	kernel32.dll
LoadLibraryA	21	-	implicit	-	-	-	-	kernel32.dll
GetModuleFileNameW	21	-	implicit	x	-	-	-	kernel32.dll
GetProcAddress	21	-	implicit	-	-	-	-	kernel32.dll
GetModuleFileNameA	21	-	implicit	x	-	-	-	kernel32.dll
GetModuleHandleExW	21	-	implicit	x	-	-	-	kernel32.dll
GetModuleHandleW	21	-	implicit	-	-	-	-	kernel32.dll
LoadLibraryExW	21	-	implicit	-	-	-	-	kernel32.dll
WriteConsoleW	20	-	implicit	x	-	-	-	kernel32.dll
GetStdHandle	20	-	implicit	-	-	-	-	kernel32.dll
SetStdHandle	20	-	implicit	-	-	-	-	kernel32.dll
GetConsoleMode	20	-	implicit	-	-	-	-	kernel32.dll
ReadConsoleInputA	20	-	implicit	x	-	-	-	kernel32.dll
SetConsoleMode	20	-	implicit	x	-	-	-	kernel32.dll
GetConsoleCP	20	-	implicit	-	-	-	-	kernel32.dll
IsDebuggerPresent	19	-	implicit	-	-	-	-	kernel32.dll
IsProcessorFeaturePresent	19	-	implicit	-	-	-	-	kernel32.dll

Introducción al análisis estático

Presentación de las herramientas - PeStudio – Imports reseñables

KERNELL32.dll	Acceso y manipulación de memoria, archivos, creación de procesos y hardware.
Advapi32.dll	Relacionado con operaciones de registro y administración de servicios.
User32.dll	Componentes de interfaz de usuario (GUI).
Ntdll.dll	Interface con el kernel de Windows.
WSock32.dll y Ws2_32.dll	Acceso a operaciones de red.
Wininet.dll	Alto nivel de red, tales como protocolos HTTP, FTP y NTP.

Introducción al análisis estático

Presentación de las herramientas - PeStudio – Funciones reseñables

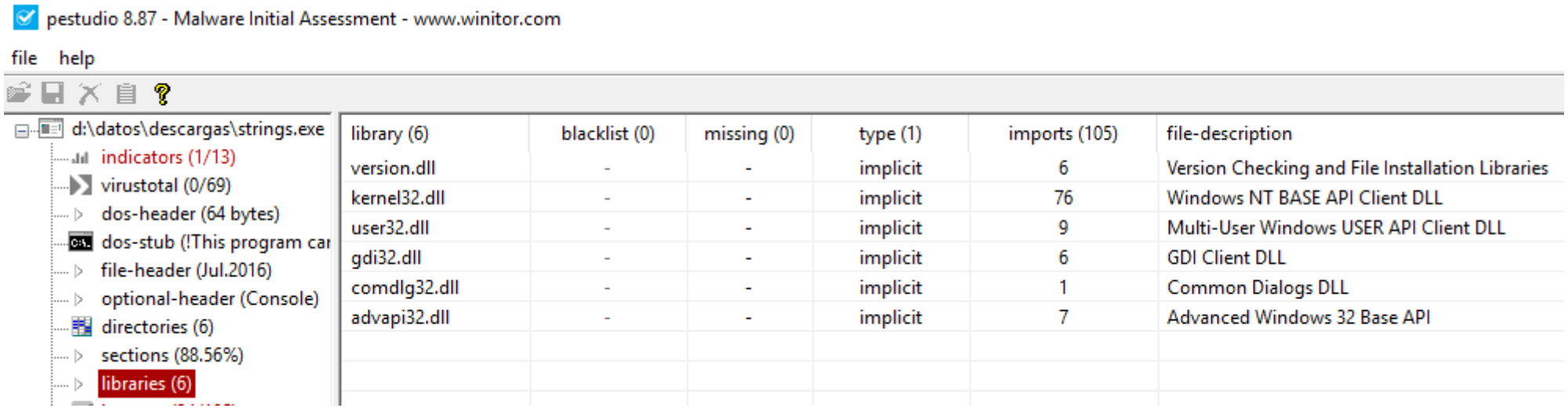
CreateProcess	Usadas para crear procesos / threads nuevos
CreateThread	
CreateToolhelp32Snapshot	Crea un snapshot de los procesos con su memoria, módulos y threads utilizados
Process32Next	Utilizada conjunto la anterior, itera entre los procesos del snapshot
EnumProcesses	Enumera los procesos actuales del sistema
CryptAcquireContext	Utilizada para inicializar el cifrado de Windows
FindResource	Busca un recurso en la sección .rsrc
CreateMutex	Crea un objeto exclusivo para asegurar que solo se ejecuta una instancia del programa
AdjustTokenPrivilegdes	Ajusta los privilegios del proceso
FindWindow	Busca el nombre de alguna en las ventanas abiertas
GetAsyncKeyState	Funciones para obtener el estado del teclado
GetKeyState	
GetTempPath	Obtienes el path del directorio %TEMP%
inet_addr	Convierte strings de IP en objetos útiles
InternetOpen	Inicia las funciones WinINet
InternetWriteFile/ReadFile	Escribe/lee información abierta por las funciones de WinINet

Introducción al análisis estático

Presentación de las herramientas - PeStudio – Libraries

Si con los imports y exports se observaban las llamadas a funciones de librerías, en libraries veremos solo las librerías que carga el PE.

De este modo si apareciera una librería que no cuadra con el contexto del programa podemos encontrar IOCS. Ie contener llamadas a la librería winsock.

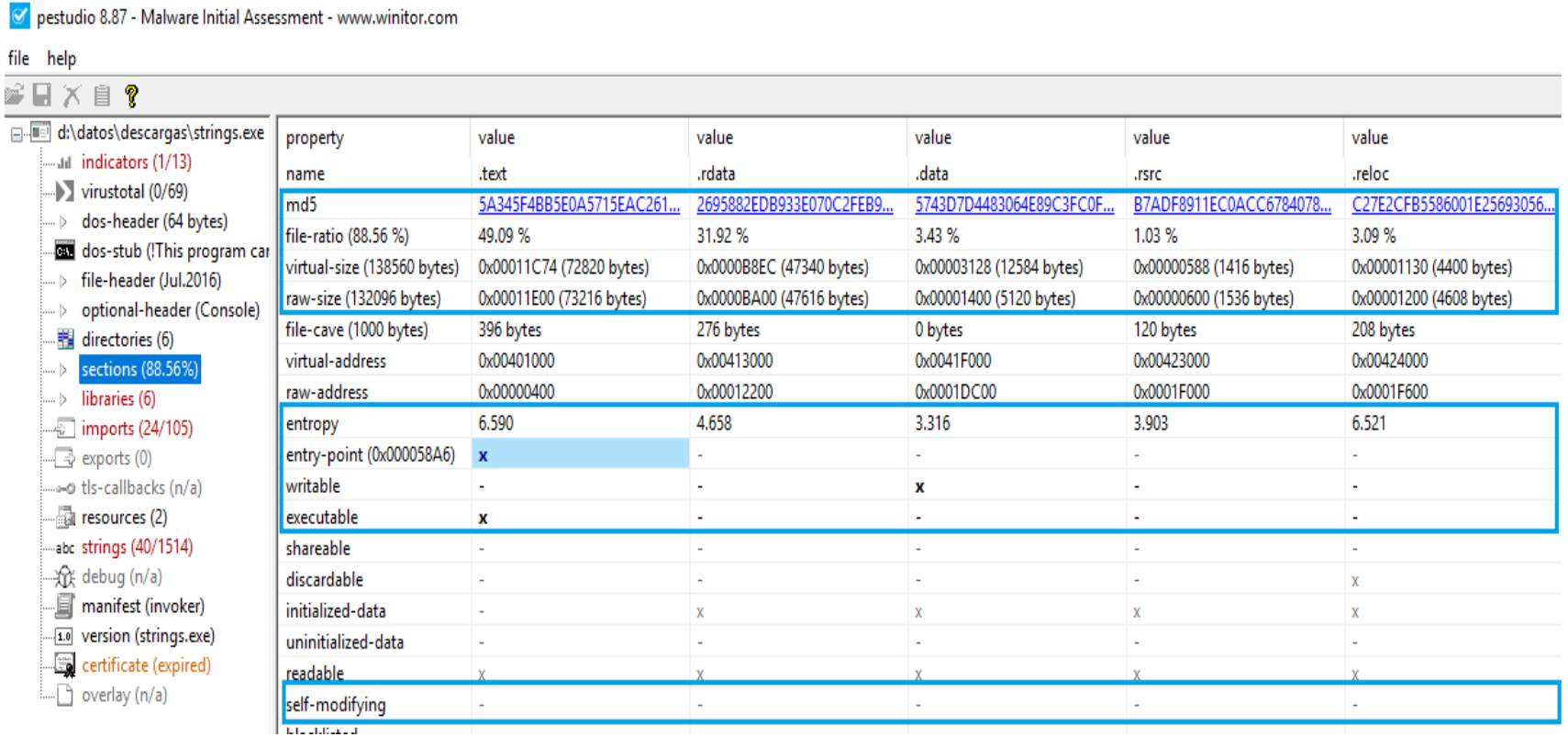


Introducción al análisis estático

Presentación de las herramientas - PeStudio – Sections

Quizás uno de los apartados más importantes a revisar. Podemos ver:

- Permisos de R/W/X
- Hash Md5
- Nivel de compresión
- Etc.



property	value	value	value	value	value
name	.text	.rdata	.data	.rsrc	.reloc
md5	5A345F4BB5E0A5715EAC261...	2695882EDB933E070C2FEB9...	5743D7D4483064E89C3FC0F...	B7ADF8911EC0ACC6784078...	C27E2CFB5586001E25693056...
file-ratio (88.56 %)	49.09 %	31.92 %	3.43 %	1.03 %	3.09 %
virtual-size (138560 bytes)	0x00011C74 (72820 bytes)	0x0000B8EC (47340 bytes)	0x00003128 (12584 bytes)	0x00000588 (1416 bytes)	0x00001130 (4400 bytes)
raw-size (132096 bytes)	0x00011E00 (73216 bytes)	0x0000BA00 (47616 bytes)	0x00001400 (5120 bytes)	0x00000600 (1536 bytes)	0x00001200 (4608 bytes)
file-cave (1000 bytes)	396 bytes	276 bytes	0 bytes	120 bytes	208 bytes
virtual-address	0x00401000	0x00413000	0x0041F000	0x00423000	0x00424000
raw-address	0x00000400	0x00012200	0x0001DC00	0x0001F000	0x0001F600
entropy	6.590	4.658	3.316	3.903	6.521
entry-point (0x000058A6)	x	-	-	-	-
writable	-	-	x	-	-
executable	x	-	-	-	-
shareable	-	-	-	-	-
discardable	-	-	-	-	x
initialized-data	-	x	x	x	x
uninitialized-data	-	-	-	-	-
readable	x	x	x	x	x
self-modifying	-	-	-	-	-

Introducción al análisis estático

Presentación de las herramientas - PeStudio – Otros datos relevantes

PeStudio es una herramienta muy versátil y la información que contiene es muy extensa algunos de los apartados que han quedado atrás pero también relevantes son:

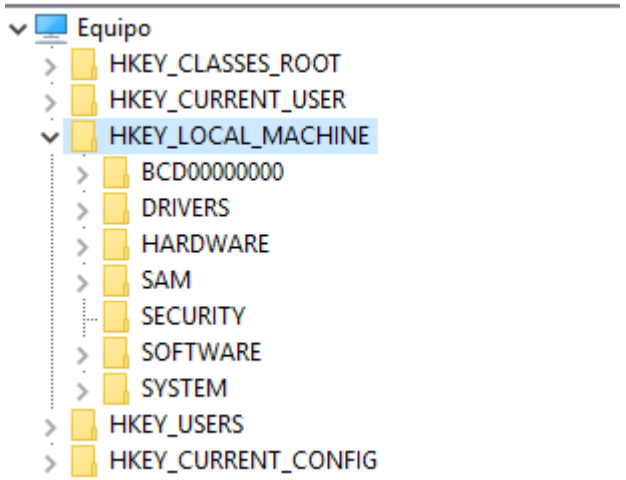
- Strings
- Version
- Overlay
- Debug
- Resources

¿Y todas las herramientas anteriores? PeStudio aglutina a todas ellas, ¿Significa eso que podemos prescindir de ellas y solo usar PeStudio?

Introducción al análisis dinámico

Pequeño resumen del funcionamiento de Windows – Windows Registry

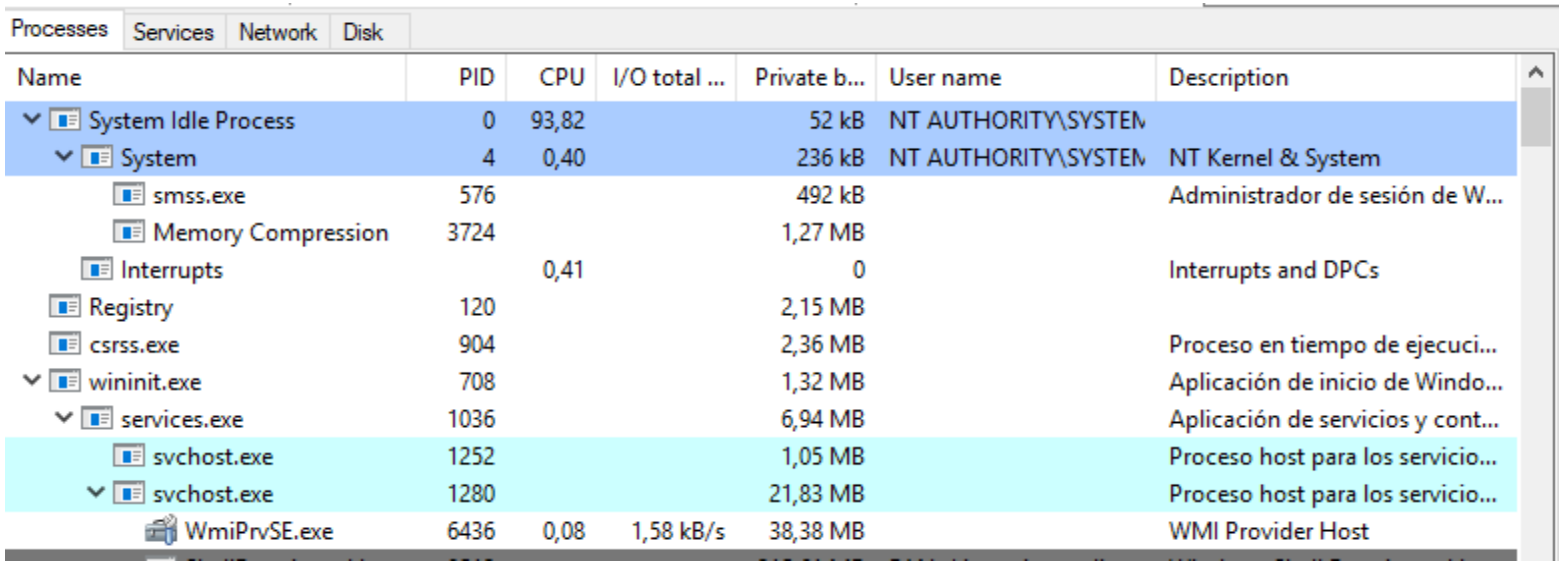
- Se trata de una base de datos utilizada para almacenar información de configuración y opciones de componentes de Windows y también aplicaciones de terceros.
- El malware utiliza el registro comúnmente para obtener persistencia en el sistema, obtener información del ordenador, establecer datos de configuración, etc.
- Buen lugar para obtener indicadores de compromiso.



Introducción al análisis dinámico

Pequeño resumen del funcionamiento de Windows – Procesos

- Todo el sistema operativo basa su funcionamiento en la ejecución de los procesos, que corresponden a las aplicaciones y servicios que se ejecuta en cada momento.
- Los procesos contienen en memoria el mapeado de cada una de las secciones del PE, el sistema operativo se encarga de preparar la memoria para su correcta ejecución.
- Existe malware que puede inyectarse en la memoria de otros procesos o en procesos nuevos creados,



Name	PID	CPU	I/O total ...	Private b...	User name	Description
System Idle Process	0	93,82		52 kB	NT AUTHORITY\SYSTEM	
System	4	0,40		236 kB	NT AUTHORITY\SYSTEM	NT Kernel & System
smss.exe	576			492 kB		Administrador de sesión de W...
Memory Compression	3724			1,27 MB		
Interrupts		0,41		0		Interrupts and DPCs
Registry	120			2,15 MB		
csrss.exe	904			2,36 MB		Proceso en tiempo de ejecuci...
wininit.exe	708			1,32 MB		Aplicación de inicio de Windo...
services.exe	1036			6,94 MB		Aplicación de servicios y cont...
svchost.exe	1252			1,05 MB		Proceso host para los servicio...
svchost.exe	1280			21,83 MB		Proceso host para los servicio...
WmiPrvSE.exe	6436	0,08	1,58 kB/s	38,38 MB		WMI Provider Host

Introducción al análisis dinámico

Pequeño resumen del funcionamiento de Windows – Procesos comunes

svchost.exe	nombre genérico de proceso anfitrión para servicios
lsass.exe	responsable de cumplir la política de seguridad en el sistema
services.exe	inicia, detiene e interactúa con procesos de servicios de Windows
winlogon.exe	carga el perfil de usuario, bloqueo del sistema, verificación de las credenciales de usuario
csrss.exe	encargado de la consola de windows y del proceso de apagado de la GUI
Smss.exe	ejecutado durante el proceso de arranque, tiene como hijos winlogon.exe y csrss.exe (entre otros)
ntvdm.exe	permite ejecutar MS-DOS aplicaciones en ordenadores actuales

Introducción al análisis dinámico

Introducción – Análisis dinámico

El análisis dinámico describe el proceso de analizar una muestra según su comportamiento durante la ejecución.

La información importante a recopilar es:

- Uso de red, tanto conexiones internas como externas.
- Creación o cualquier tipo de manipulación de ficheros.
- Carga o uso de elementos no visibles durante el análisis estático.
- Interacción con el sistema de ficheros.
- Cambios en los procesos de ejecución.
- Etc.

Introducción al análisis dinámico

Presentación de las herramientas - Clasificación

Existen diferentes tipos de herramientas para el análisis dinámico pero en su gran mayoría se pueden dividir en 3 grupos según su funcionamiento:

- Herramienta de análisis basadas en Hooks.
- Herramientas basadas en la diferencia o variación.
- Herramientas basadas en las notificaciones.

Introducción al análisis dinámico

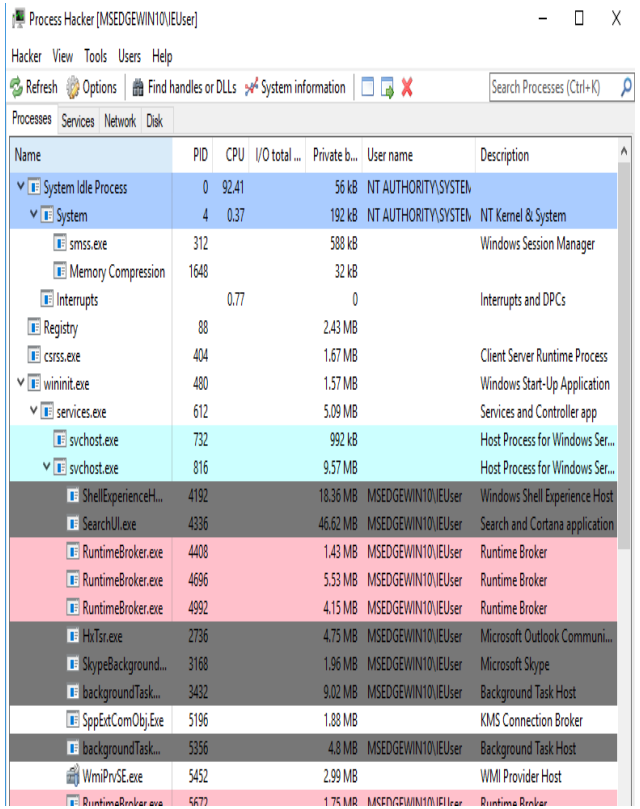
Presentación de las herramientas - Herramienta de análisis basadas en hooks

Estas herramientas interceptan las llamadas a las funciones o los elementos del sistema para monitorizar en tiempo real, tanto a nivel de usuario como de sistema.

Se pueden observar:

- Conexiones de red.
- Árbol de procesos.
- Interacción con el sistema de archivos.
- Servicios activos.

Ej, Process Hacker, SysAnalisis.



Introducción al análisis dinámico

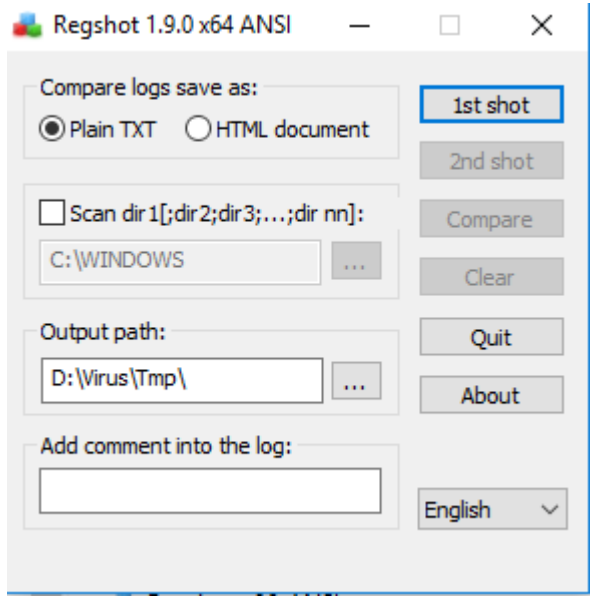
Presentación de las herramientas - Herramientas basadas en la diferencia

Se encargan de realizar la captura del estado del sistema antes y después de la ejecución del archivo para comprobar las diferencias entre el estado anterior y posterior.

Se pueden observar:

- Cambios en las claves de registro.

Ej, Regshot.



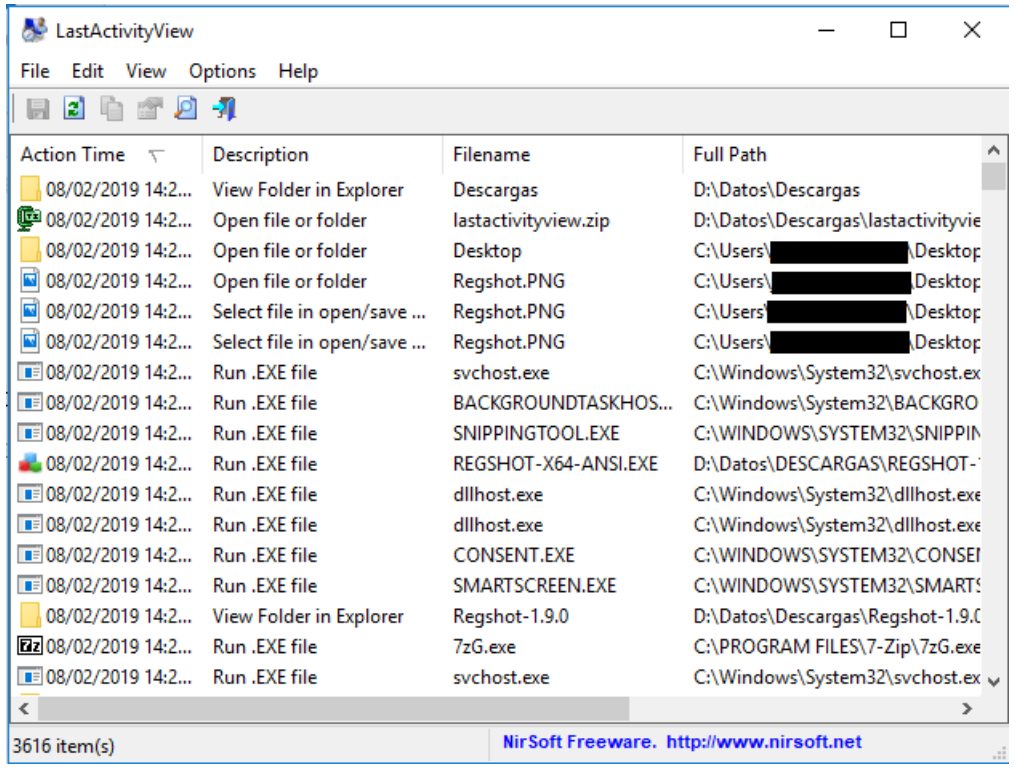
Introducción al análisis dinámico

Presentación de las herramientas - Herramientas basadas en las notificaciones

Estas herramientas registran las rutinas de notificación del sistema cuando suceden una serie de eventos. Se utilizan comúnmente para realizar análisis forense

La diferencia con las herramientas de hooks radica en que estas registran los eventos propios del sistema.

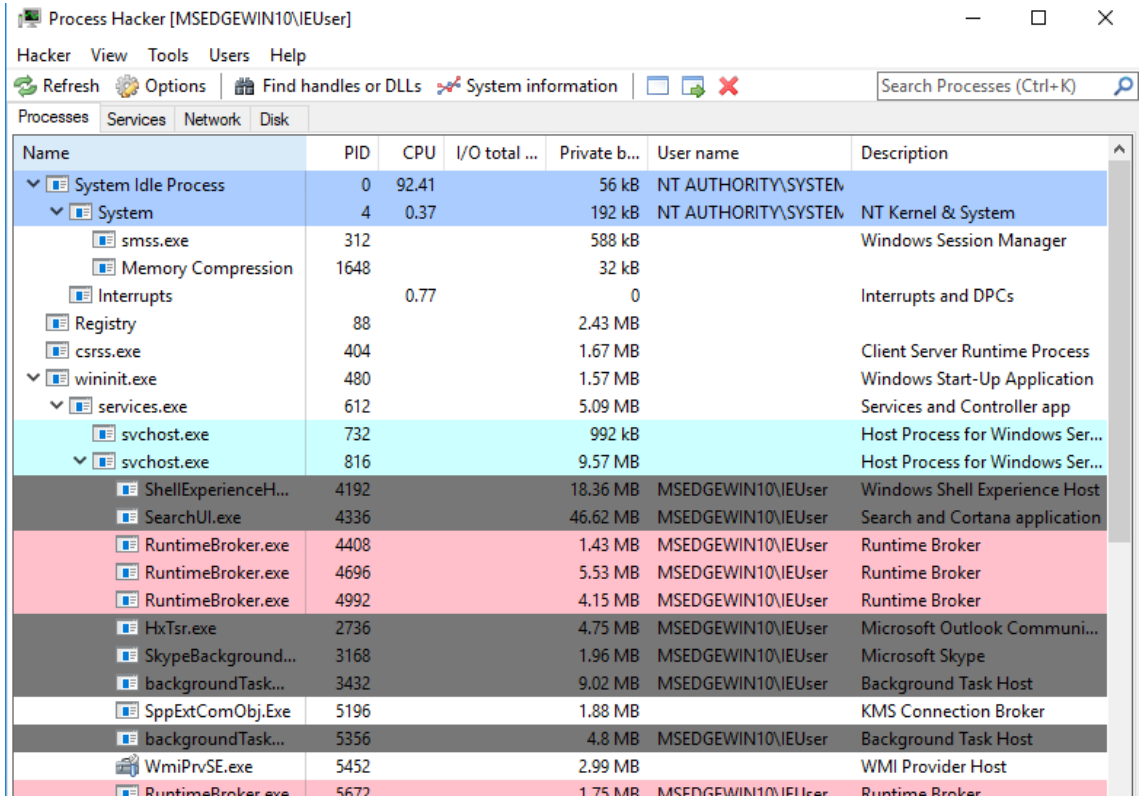
Ej, LastActivityView.



Introducción al análisis dinámico

Presentación de las herramientas - Process Hacker

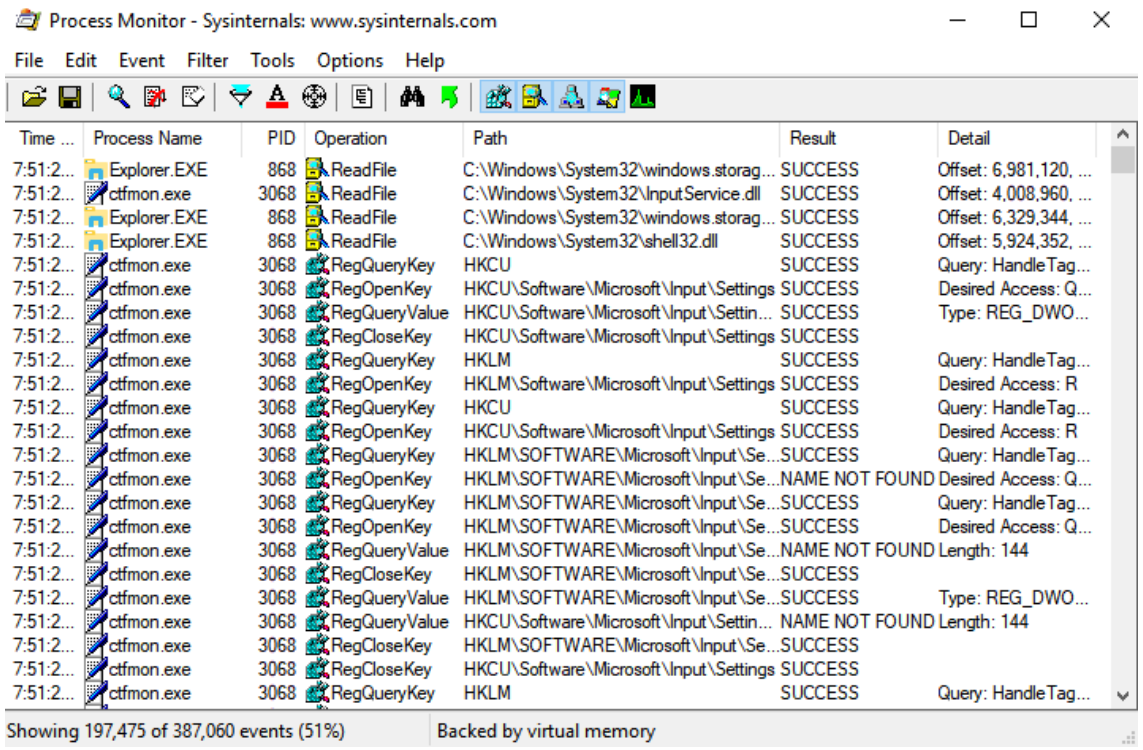
Gracias a esta herramienta se puede hacer un primer modelaje de comportamiento dada la información que ofrece. Además es capaz de mostrar información relevante como los permisos, el contenido de la memoria, los threads abiertos, etc.



Introducción al análisis dinámico

Presentación de las herramientas – Process Monitor

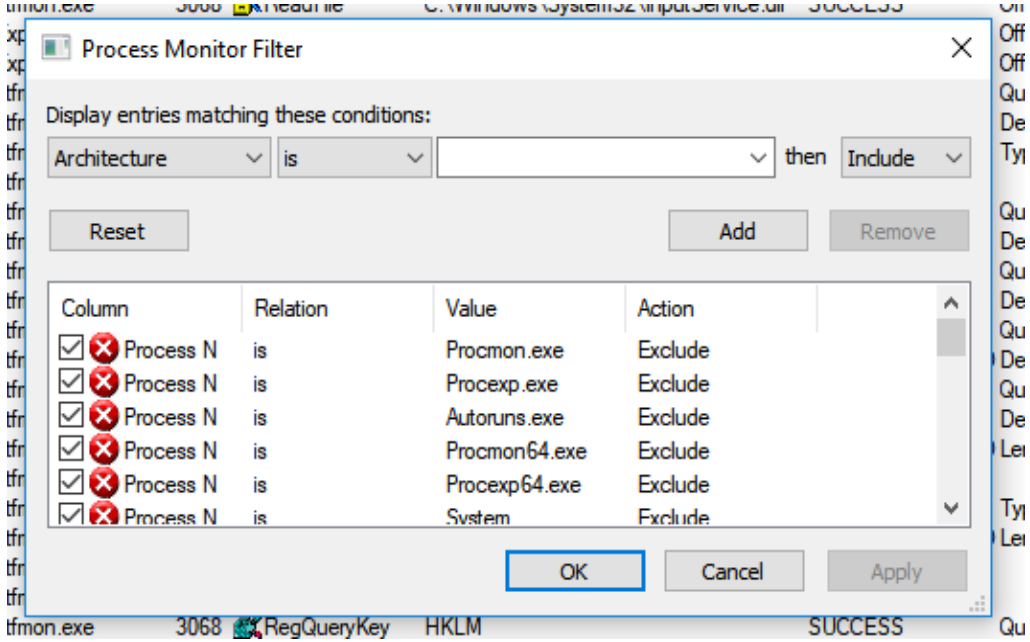
Esta herramienta esta incluida en la Sysinternals Suite de Microsoft, se trata de una herramienta que captura todos los eventos del registro, sistema de ficheros, red y actividad de threads y procesos.



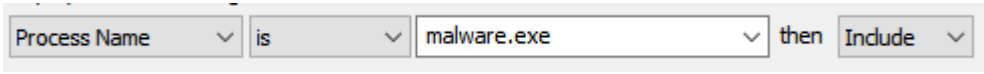
Introducción al análisis dinámico

Presentación de las herramientas – Process Monitor

Una de las mejores funcionalidades es la poner filtros en la captura de los eventos, de esta forma puedes filtrar para obtener los resultados asociados a la actividad que deseas.



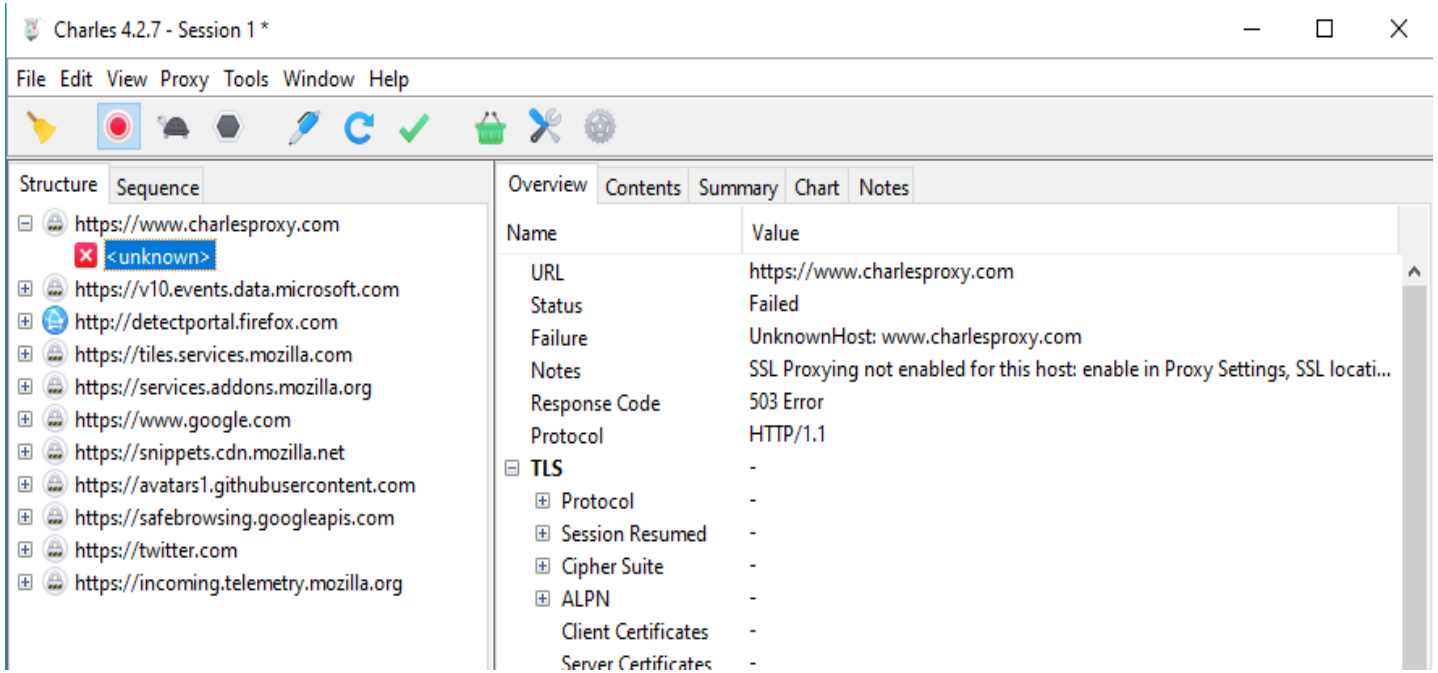
Filtro que muestra los eventos relativos al ejecutable malware.exe



Introducción al análisis dinámico

Presentación de las herramientas - Charles Proxy – Proxy de red

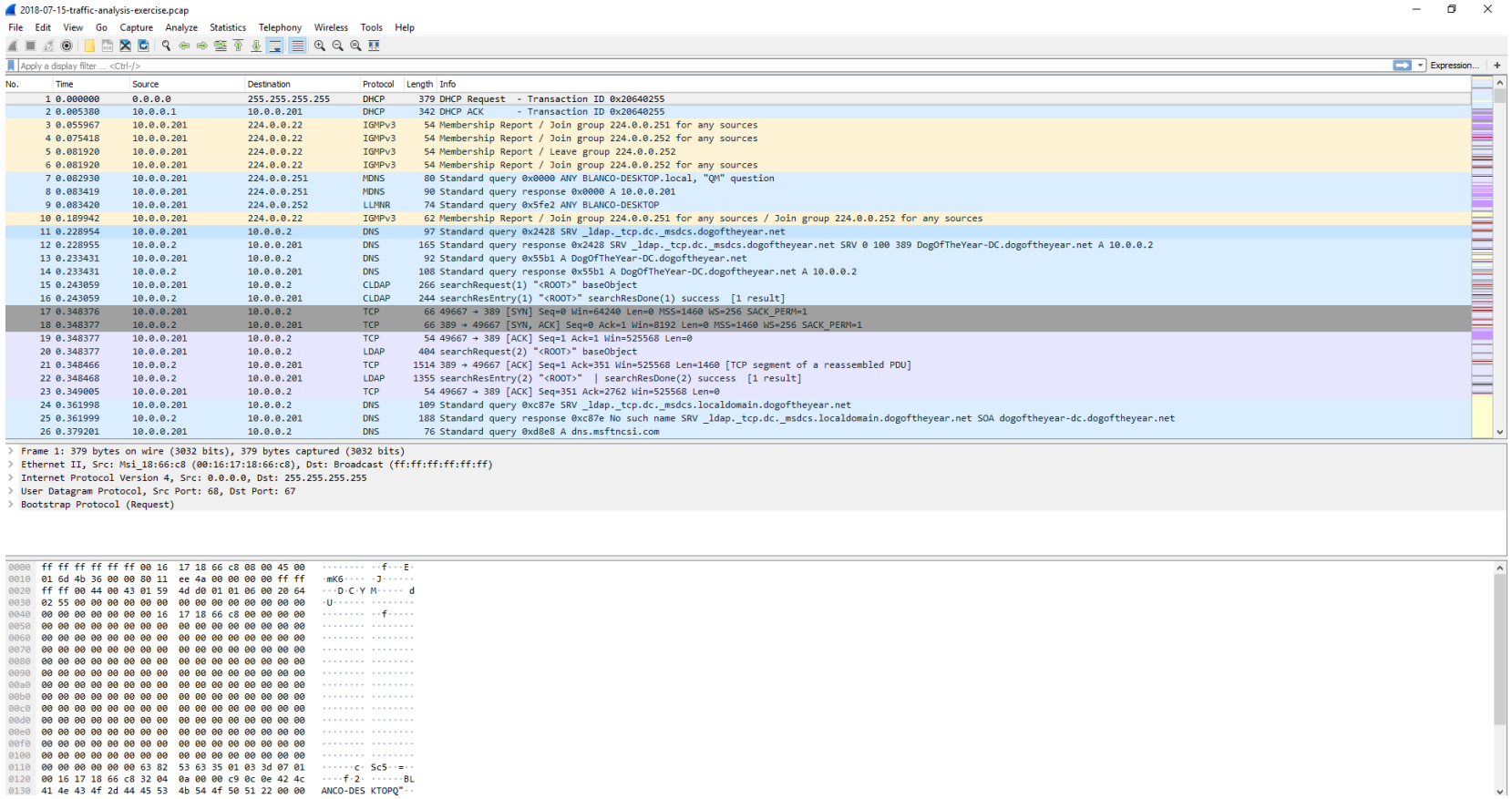
Esta herramienta es un HTTP proxy, con ella puedes obtener las conexiónes HTTP que puede tener la sample.



Introducción al análisis dinámico

Presentación de las herramientas - Wireshark

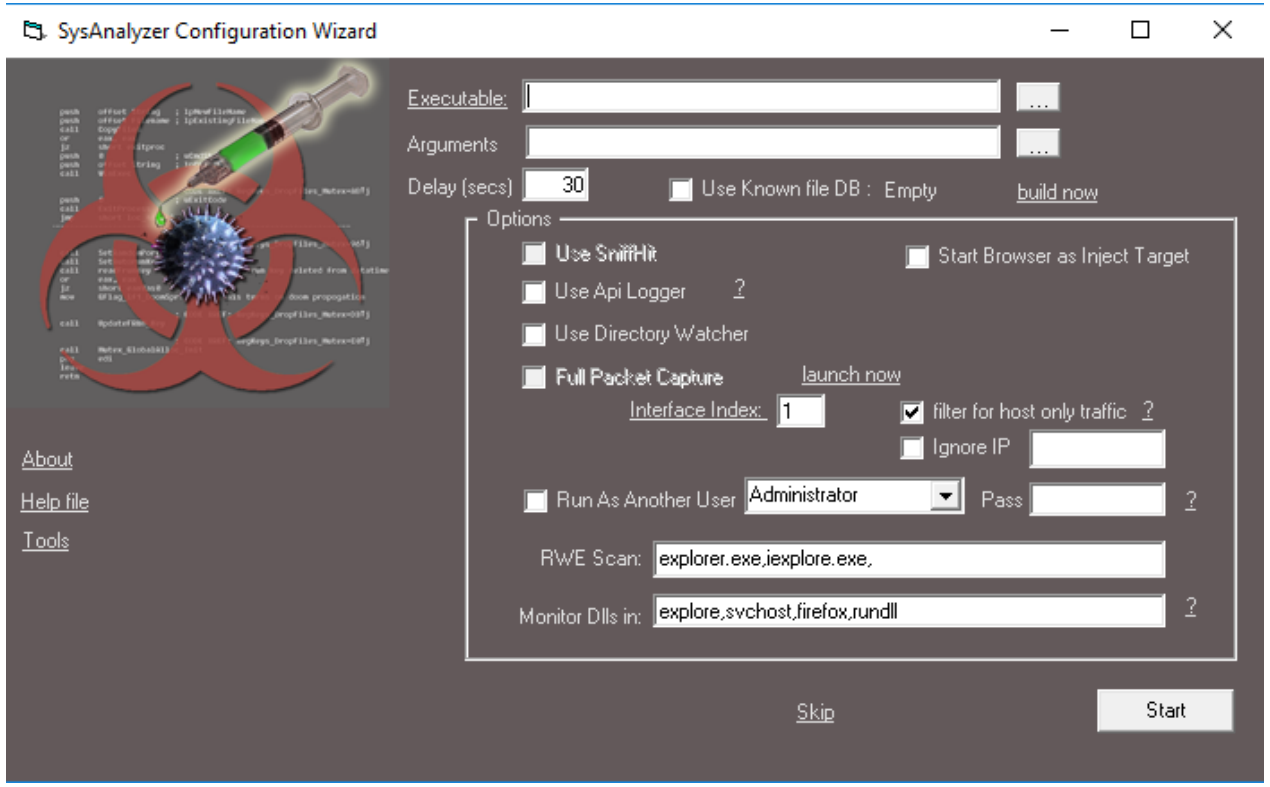
Analizador de protocolos de red, permite ver todo el tráfico que pasa a través de una red.



Introducción al análisis dinámico

Presentación de las herramientas - SysAnalizer

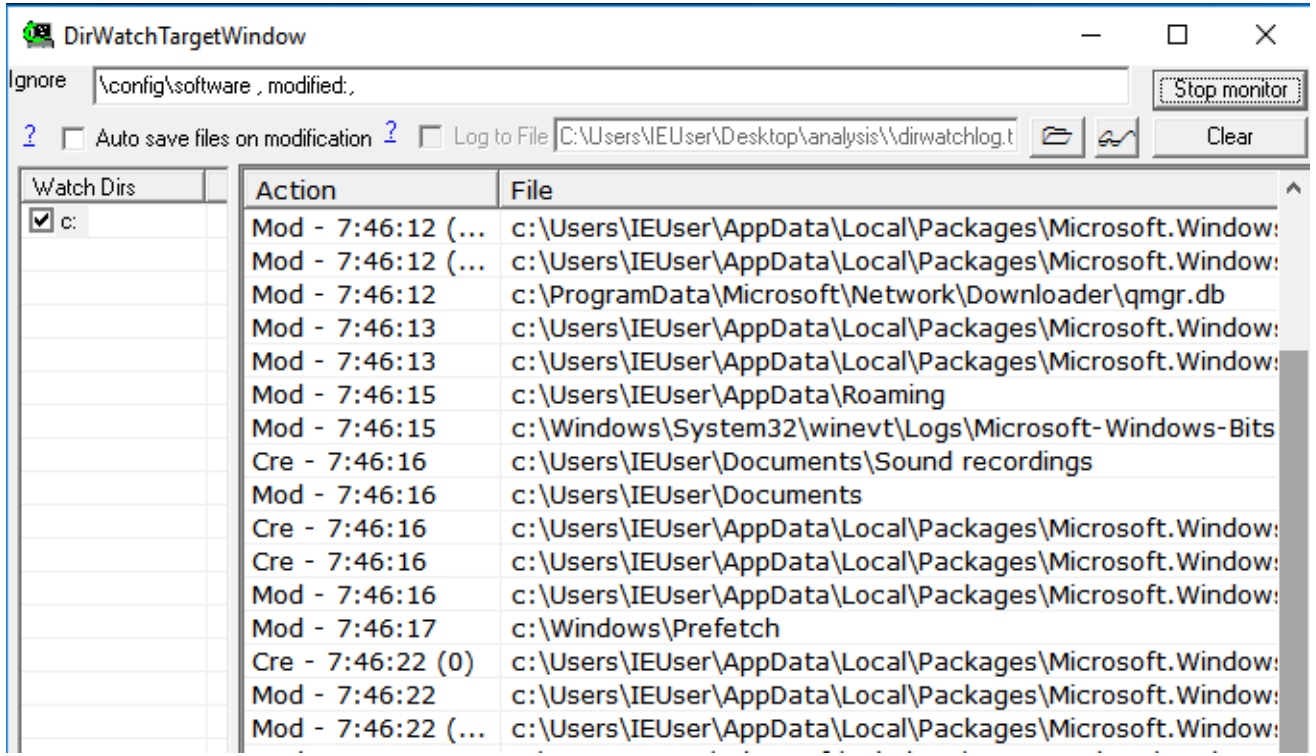
Esta herramienta permite la monitorización en tiempo real de la ejecución de una muestra, una vez finalizada la monitorización se obtiene un informe detallado con la recopilación de la información que ha obtenido en la ejecución.



Introducción al análisis dinámico

Presentación de las herramientas DirWatch

Esta utilidad pertenece a la Suite de SysAnalizer permite la monitorización de una carpeta y captura los eventos de escritura/lectura en la carpeta.

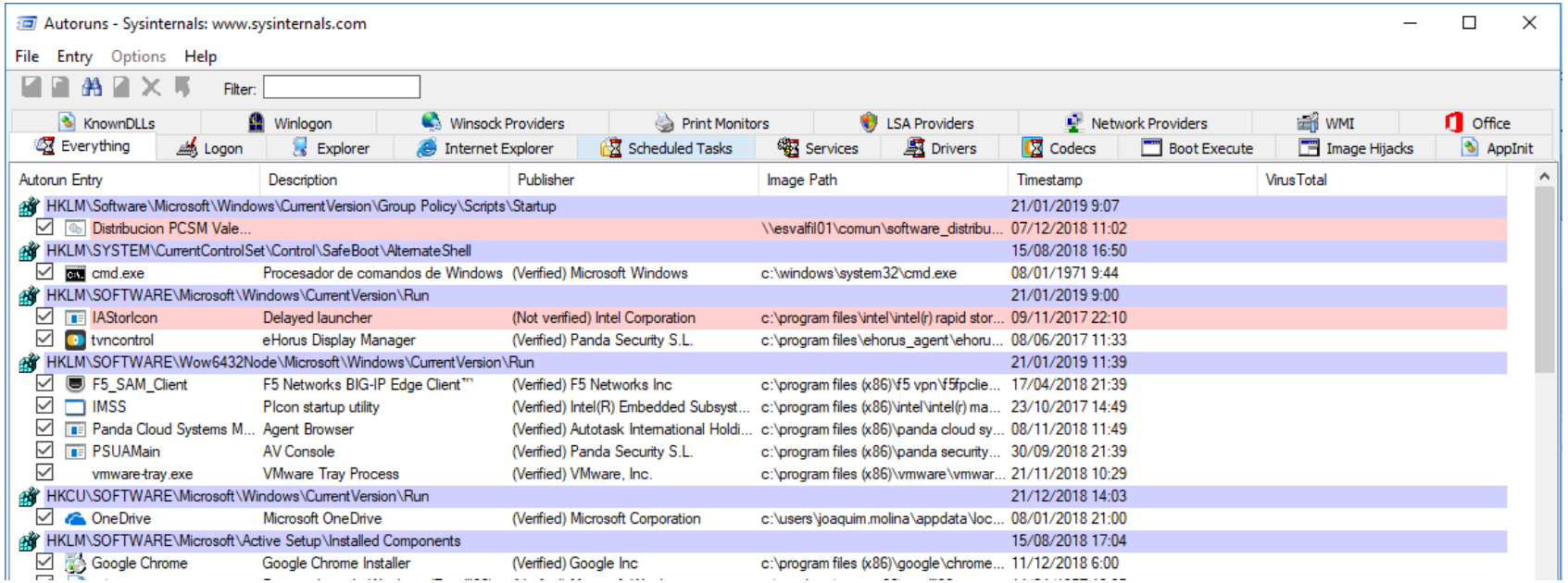


Introducción al análisis dinámico

Prentación de las herramientas Sysinternals Suite

Es un paquete de utilidades para realizar tareas de administración en los ordenadores Windows, contiene diferentes programas que te permiten analizar procesos, el disco, red, etc.

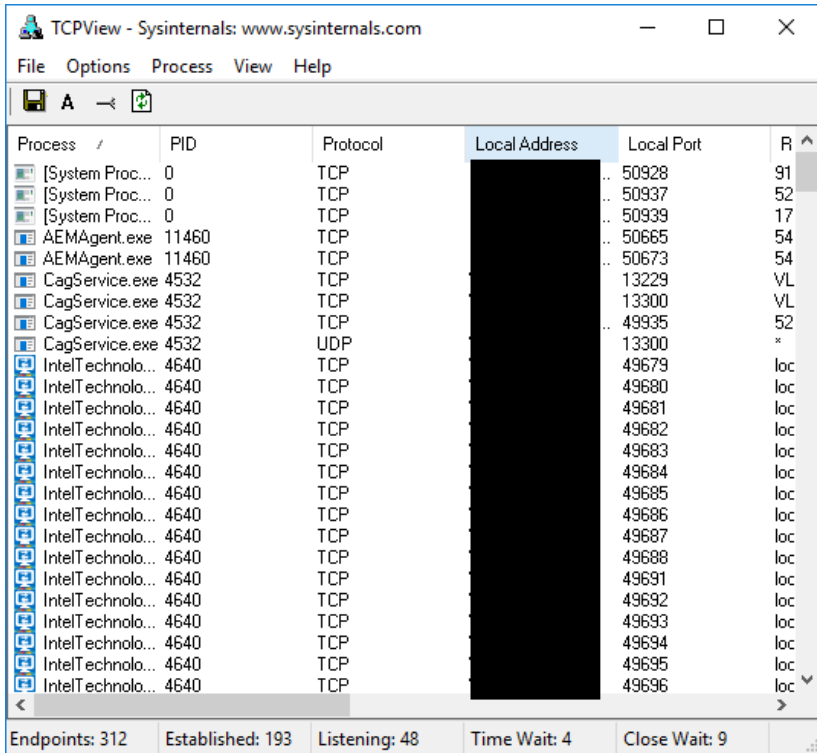
Una de las utilidades más utilizadas es Autoruns, que enumera la lista de todos los programas y procesos que se encuentran en el inicio del sistema.



Introducción al análisis dinámico

Prentación de las herramientas Sysinternals Suite

Con la utilidad TcpView puedes ver las conexiones TCP/UDP que se encuentran activas en el ordenador en ese momento, útil para determinar conexiones persistentes con servidores externos.



¡Manos a la obra!

Objetivo: conseguir la mayor información posible sobre la amenaza.

El departamento de soporte ha detectado una amenaza de malware en diversos equipos corporativos dentro de la red de un cliente, a continuación nos envían al laboratorio las siguientes muestras.

Vuestro trabajo es descubrir que acciones realiza y obtener la mayor cantidad de información posible.

Bibliografía

Documentación

1. **Practical Malware Analysis: A Hands-On Guide to Dissecting Malicious Software.** Michael Sikorski, Andrew Honig
2. **Learning Malware Analysis. Monnapa K.A.**
3. **MSDN, Documentación de la API de Windows** https://docs.microsoft.com/en-us/windows/desktop/index


Reinventing
Cybersecurity.
panda
pandasecurity.com