

Laboratorio 4

Security Data Science
Maria Jose Castro #181202
Maria Ines Vasquez #18250

Análisis estático

1. **Utilice la herramienta pefile para examinar el PE header y obtenga las DLL y las APIs que los ejecutables llaman. ¿Qué diferencias observa entre los ejemplos? ¿Existe algún indicio sospechoso en la cantidad de DLLs y las APIs llamadas?**

Previo a desempquetado:

En este caso el archivo no tiene una gran cantidad de llamadas al API, lo que desde un inicio se ve sospechoso.

Dentro de las llamadas DLL podemos ver que hace una llamada al **Kernel32**, que esta tiene la función de manipular memoria, archivos, crear procesos y hardware. También las llamadas al **User32** que se encarga de los componentes de interfaz de usuario, botones y acciones del usuario. Por último la llamada a **WS2_32** ya que dentro de este se hacen las llamadas a las redes, por lo que nos indica que tendrán conexiones a internet.

El archivo sample_qwrty_dk2 tiene secciones que se encuentran empaquetadas o con extensión UPX, versus sample_vg655_25th.exe tiene extensiones como txt.

```

security@security-VirtualBox:~/Downloads/LAB4/Users/jyas
ata Science/Laboratorios/LAB04/MALWR2$ python3 sa.py
/MALWR
b'UPX0\x00\x00\x00\x00' 0x1000 0x5000 0
b'UPX1\x00\x00\x00\x00' 0x6000 0x1000 4096
b'.rsrc\x00\x00\x00' 0x7000 0x1000 512
Llamadas DLL:
b'KERNEL32.DLL'
Llamadas a funciones:
    b'LoadLibraryA'
    b'ExitProcess'
    b'GetProcAddress'
    b'VirtualProtect'
Llamadas DLL:
b'MSVCRT.dll'
Llamadas a funciones:
    b'atol'
Llamadas DLL:
b'SHELL32.dll'
Llamadas a funciones:
    b'SHChangeNotify'
Llamadas DLL:
b'USER32.dll'
Llamadas a funciones:
    b'LoadStringA'
Llamadas DLL:
b'WS2_32.dll'
Llamadas a funciones:
    b'closesocket'
TimeStamp : Thu May 14 17:12:40 2009 UTC

```

2. **Obtenga la información de las secciones del PE Header. ¿Qué significa que algunas secciones tengan como parte de su nombre “upx”? Realice el procedimiento de desempaquetado para obtener las llamadas completas de las APIs.**

Posteriormente se desempaqueta el archivo y posteriormente se hace el mismo análisis del archivo utilizando Pe Header. Podemos ver que ahora las llamadas que tenemos son mucho más, especialmente en la llamada al Kernel32.

Las secciones como parte del nombre UPX son secciones que están comprimidas, en este caso utilizando ingeniería inversa, puesto que al no mostrarlas las el archivo no parezca sospechoso.

```

security@security-VirtualBox:~/Downloads/Users/jyass/Documents/UVG/202
cience/Laboratorios/LAB04/MALWR2$ python3 sa.py
/MALWR
b'.text\x00\x00\x00' 0x1000 0xea6 4096
b'.rdata\x00\x00' 0x2000 0x67e 2048
b'.data\x00\x00\x00' 0x3000 0x628 512
b'.rsrc\x00\x00\x00' 0x4000 0x80 512
Llamadas DLL:
b'KERNEL32.dll'
Llamadas a funciones:
b'CloseHandle'
b'WaitForSingleObject'
b'CreateEventA'
b'ExitThread'
b'Sleep'
b'GetComputerNameA'
b'CreatePipe'
b'DisconnectNamedPipe'
b'TerminateProcess'
b'WaitForMultipleObjects'
b'TerminateThread'
b'CreateThread'
b'CreateProcessA'
b'DuplicateHandle'
b'GetCurrentProcess'
b'ReadFile'
b'PeekNamedPipe'
b'SetEvent'
b'WriteFile'
b'SetProcessPriorityBoost'
b'SetThreadPriority'
b'GetCurrentThread'
b'SetPriorityClass'
b'lstrcatA'
b'lstrcpyA'
b'GetEnvironmentVariableA'
b'GetShortPathNameA'
b'GetModuleFileNameA'
b'GetStartupInfoA'
b'GetModuleHandleA'

```

```

Llamadas DLL:
b'MSVCRT.dll'
Llamadas a funciones:
b'_controlfp'
b'_beginthread'
b'_strntcmp'
b'_sprintf'
b'_atoi'
b'_strchr'
b'_free'
b'_malloc'
b'_exit'
b'_XcptFilter'
b'_exit'
b'_acmdln'
b'_getmainargs'
b'_inittterm'
b'_setusermatherr'
b'_adjust_fdiv'
b'_p_commode'
b'_p_fmode'
b'_set_app_type'
b'_except_handler3'
b'_itoa'
Llamadas DLL:
b'SHELL32.dll'
Llamadas a funciones:
b'ShellExecuteExA'
b'SHChangeNotify'
Llamadas DLL:
b'USER32.dll'
Llamadas a funciones:
b'LoadStringA'
Llamadas DLL:
b'WS2_32.dll'
Llamadas a funciones:
b'htons'
b'connect'
b'socket'
b'WSAStartup'
b'send'
b'inet_addr'
b'recv'
b'closesocket'
TimeDateStamp : Thu May 14 17:12:40 2009 UTC

```

3. Según el paper “Towards Understanding Malware Behaviour by the Extraction of API Calls”, ¿en qué categoría sospechosas pueden clasificarse estos ejemplos en base a algunas de las llamadas a las APIs que realizan? Muestre una tabla con las APIs sospechosas y la categoría de malware que el paper propone.

Con respecto a las Apis sospechosas encontradas en el archivo y su clasificación según el paper:

Comportamiento	Categoría de Malware	Llamadas al API
1	Búsqueda de archivos para infectar	
2	Copiar/Borrar archivos	CloseHandle,
3	Obtener la información del archivo	GetShortPathNameA, GetModuleFileNameA
4	Mover archivos	
5	Archivos de lectura/escritura	ReadFile, WriteFile,

6	Cambio de atributos del archivo	
---	---------------------------------	--

Estas son las recomendadas o encontradas en el paper:

TABLE 1
MAIN MALICIOUS BEHAVIOUR GROUPS OF API CALL FEATURES

Behaviour	Malware Category	API Function Calls
Behaviour 1	Search Files to Infect	FindClose, FindFirstFile, FindFirstFileEx, FindFirstFileName, TransactedW, FindFirstFileNameW, FindFirstFileTransacted, FindFirstStream, TransactedW, FindFirstStreamW, FindNextFile, FindNextFileNameW, FindNextStreamW, SearchPath.
Behaviour 2	Copy/Delete Files	CloseHandle, CopyFile, CopyFileEx, CopyFileTransacted, CreateFile, CreateFileTransacted, CreateHardLink, CreateHardLink, Transacted, CreateSymbolicLink, CreateSymbolic, LinkTransacted, DeleteFile, DeleteFileTransacted.
Behaviour 3	Get File Information	GetBinaryType, GetCompressed, FileSize, GetCompressedFile, SizeTransacted, GetFileAttributes, GetFileAttributesEx, GetFileAttributes, Transacted, GetFileBandwidth, Reservation, GetFileInformation, ByHandle, GetFileInformation, ByHandleEx, GetFileSize, GetFileSizeEx, GetFileType, GetFinalPathName, ByHandle, GetFullPathName, GetFullPathName, Transacted, GetLongPathName, GetLongPathName, Transacted, GetShortPathName, GetTempFileName, GetTempPath.
Behaviour 4	Move Files	MoveFile, MoveFileEx, MoveFileTransacted, MoveFileWithProgress.
Behaviour 5	Read/Write Files	OpenFile, OpenFileById, ReOpenFile, ReplaceFile, WriteFile, CreateFile, CloseHandle.
Behaviour 6	Change File Attributes	SetFileApisToANSI, SetFileApisToOEM, SetFileAttributes, SetFileAttributesTransacted, SetFileBandwidthReservation, SetFileInformationByHandle, SetFileShortName, SetFileValidData

4. Para el archivo “sample_vg655_25th.exe” obtenga el HASH en base al algoritmo SHA256.

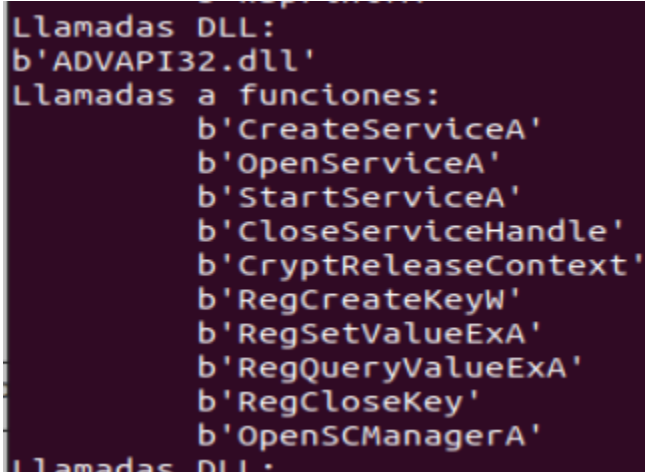
En este caso para cada sección tenemos un distinto hash para el SHA256. En este caso estas son todas para cada una de las secciones del archivo.

```
security@security-VirtualBox:~/Downloads/Users/jyass/Documents/UVG/2022/
/MALWR
b'.text\x00\x00\x00'
HASH 920e964050a1a5dd60dd00083fd541a2
Sha256 55cda830ff2543783350fb781ed2b2f77e72aa123134d2513acfb944487773054
b'.rdata\x00\x00'
HASH 2c42611802d585e6eed68595876d1a15
Sha256 a2acc94d242d28b6dd0a0859ec59ecc7f6b98d4ea09346b819d486b8827d2d79
b'.data\x00\x00\x00'
HASH 83506e37bd8b50cacabd480f8eb3849b
Sha256 110357de37bd422f6c68b66035e4652b99767819353f4c398953249a930fa823
b'.rsrc\x00\x00\x00'
HASH f99ce7dc94308f0a149a19e022e4c316
Sha256 418c45aa8ad5b74ea7a820a4cf19b2fbc688502752d600a7800d3cbe1d058e44
Llamadas DLL:
b'KERNEL32.dll'
Llamadas a funciones:
```

5. Para el archivo “sample_vg655_25th.exe”, ¿cuál es el propósito de la DLL ADVAPI32.dll?

En este caso el **ADVAPI32** es una biblioteca que utiliza muchas APIs, y este es altamente necesario para el funcionamiento del sistema. Este archivo tiene código de máquina y se encarga de algunas funciones un poco más avanzadas como registro y administración de servicios.

Posiblemente puede que inyecta código en estas librerías y también puede que limite cuales las acciones permitidas dentro del sistema, acceden a registros de memoria, cambiando archivos, etc.



Llamadas DLL:
b'ADVAPI32.dll'
Llamadas a funciones:
b'CreateServiceA'
b'OpenServiceA'
b'StartServiceA'
b'CloseServiceHandle'
b'CryptReleaseContext'
b'RegCreateKeyW'
b'RegSetValueExA'
b'RegQueryValueExA'
b'RegCloseKey'
b'OpenSCManagerA'
Llamadas DLL:

6. Para el archivo "sample_vg655_25th.exe", ¿cuál es el propósito de la API CryptReleaseContext?

La función libera el identificador de un proveedor de servicios criptográficos (CSP) y un contenedor de claves. En cada llamada a esta función, el recuento de referencias en el CSP se reduce en uno. Cuando el recuento de referencias llega a cero, el contexto se libera por completo y ya no puede ser utilizado por ninguna función de la aplicación.

Una aplicación llama a esta función después de finalizar el uso del CSP. Después de llamar a esta función, el identificador CSP liberado ya no es válido. Esta función no destruye contenedores de claves ni pares de claves.

7. Con la información recopilada hasta el momento, indique para el archivo "sample_vg655_25th.exe" si es sospechoso o no, y cuál podría ser su propósito.

El archivo se ve altamente sospechoso puesto que tiene accesos a leer, escribir y accesos a memoria. Posiblemente este archivo se encargue de encriptar archivos o de mover su ubicación a donde no estén.

Ya que cuenta con la función **CryptReleaseContext** puede bloquear el acceso a los archivos. También podemos ver que tiene llamadas a internet y acceso a APIs de redes.

Análisis dinámico

8. Utilice la plataforma de análisis dinámico <https://www.hybrid-analysis.com> y cargue el archivo “sample_vg655_25th.exe”. ¿Se corresponde el HASH de la plataforma con el generado? ¿Cuál es el nombre del malware encontrado? ¿En qué consiste este malware?

En este caso el HASH encontrado por el sitio fue un SHA 256:

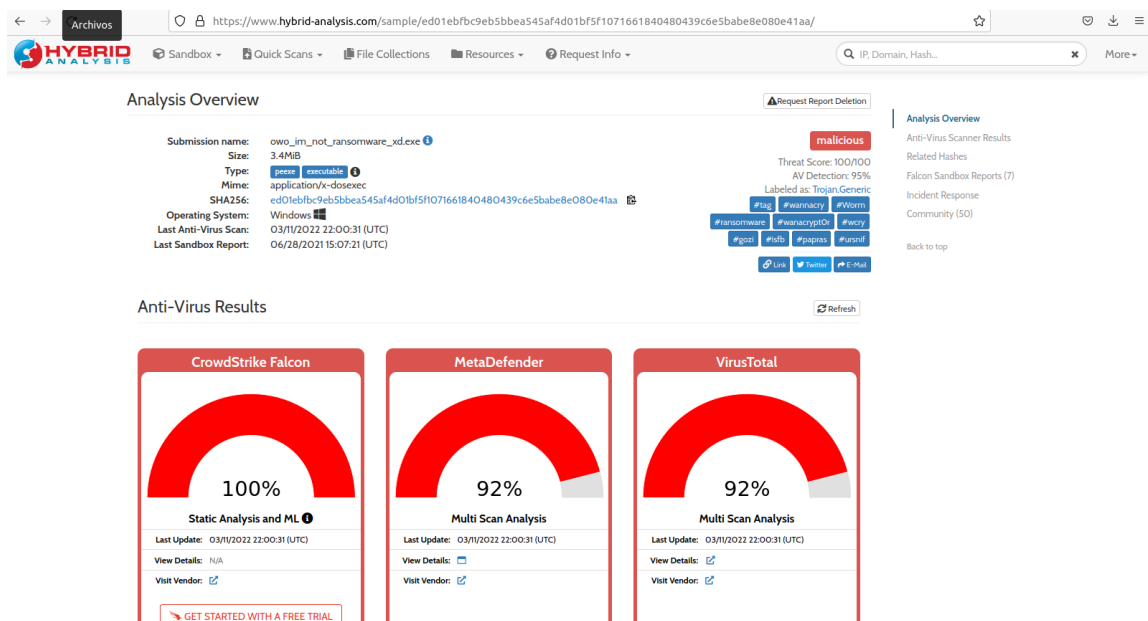
ed01ebfbc9eb5bbea545af4d01bf5f1071661840480439c6e5babe8e080e41aa

Según el análisis hecho estos NO corresponden, tampoco con los vistos anteriormente en ninguna de las secciones.

```
TimeDateStamp : Sat Nov 20 09:05:05 2010 UTC
security@security-VirtualBox:~/Downloads/Users/jyass/Documents/UVG/2022/Security Data Science/Laboratorios/LAB04/MALWR2$ python3 sa.py
/MALWR
SHA256 6caeca67b7c6a82989f4e7cefb5312a13e59151ae84f3ba6964c70e799729bac
```

El nombre encontrado es Wanna Cry

Un Ransomware es un tipo de malware que impide a los usuarios acceder a su sistema o a sus archivos personales y que exige el pago de un rescate para poder acceder de nuevo a ellos. Los creadores de ransomware piden que el pago se efectúe mediante criptomonedas o tarjetas de crédito. En muchos países existen regulaciones donde se cataloga como ilegal hacer el pago para recuperar archivos.



9. Muestre las capturas de pantalla sobre los mensajes que este malware presenta a usuario. ¿Se corresponden las sospechas con el análisis realizado en el punto 7?

Estas imágenes corresponden a lo que vemos en el punto 7 puesto que está llamada **CryptReleaseContext** bloquea los archivos de la computadora y

efectivamente los mensajes que muestra el ransomware corresponden a que nuestros archivos fueron secuestrados y el acceso a ellos está bloqueado.

