Análisis de Malware

Prácticas

1. Objetivos

Conocer las principales herramientas usadas por los analistas de malware para desensamblar y depurar.

2. Descripción

Durante esta práctica, realizada de manera paralela al desarrollo teórico del tema, realizaremos una pequeña introducción a algunas de las herramientas disponibles para el análisis de malware, deteniéndonos en dos en particular, el desensamblador IDA Pro y el depurador OllyDbg.

3. Realización

3.1 Análisis estático básico

Software requerido: Dependency Walker, PEiD, Resource Hacker, upx
Material adicional: Conexión a Internet, Ejercic1.exe, Ejercic1.dll, Ejercic2.exe.

Ejercicio 1: Usando únicamente las herramientas descritas y los conocimientos adquiridos para responder a las siguientes preguntas sobre los ficheros Ejercic1.exe y Ejercic1.dll:

1. Carga los archivos en http://www.VirusTotal.com/ y revisa los informes. ¿Coinciden con alguna firma antivirus existente?
2. ¿Cuándo se compilaron estos archivos?
3. ¿Hay alguna indicación de que cualquiera de estos archivos esté empaquetado u ofuscado? En caso afirmativo, ¿cuáles son estos indicadores?
4. ¿Alguna importación indica qué hace este malware? En caso afirmativo, ¿cuál y por qué?
5. ¿Existen otros archivos o indicadores basados en el host que puede buscar en los sistemas infectados?
6. ¿Qué indicadores basados en el uso de la red se podrían utilizar para encontrar este malware en máquinas infectadas?
7. ¿Cuál es el propósito de estos archivos?

Ejercicio 2: Analiza el fichero Ejercic2.exe:

1. Carga el archivo Ejercic2.exe en http://www.VirusTotal.com/. ¿Coincide con alguna firma

antivirus existente?
2. ¿Hay alguna indicación de que este archivo está empaquetado u ofuscado? En caso afirmativo, ¿cuáles son estos indicadores? Si el archivo está empaquetado, descomprímelo si es posible.
3. ¿Alguna importación insinúa la funcionalidad de este programa? Si es así, ¿qué importaciones son y qué te dicen?
4. ¿Qué indicadores basados en el host o en la red podrían utilizarse para identificar este malware en máquinas infectadas?

3.2 Análisis dinámico básico

Software requerido: SysInternals suite, fakenet, Wireshark, regshot, VMWare Player
Material adicional: Conexión a Internet, Ejercic3.exe, Ejercic4.dll, Ejercic5.exe, Ejercic6.exe

Ejercicio 3: Analizar el malware que se encuentra en el archivo Ejercic3.exe utilizando herramientas vistas hasta ahora.

1. ¿Cuáles son las importaciones y las cadenas de este malware?
2. ¿Cuáles son los indicadores basados en el host del malware?
3. ¿Existen firmas basadas en el uso de la red útiles para este malware? Si es así, ¿Cuáles son?

Ejercicio 4: Analiza el malware que se encuentra en el archivo Ejercic4.dll utilizando herramientas vistas hasta ahora.

1. ¿Cómo se puede instalar este malware?
2. ¿Cómo lograría que este malware se ejecutara después de la instalación?
3. ¿Cómo puede encontrar el proceso bajo el cual este malware se está ejecutando?
4. ¿Qué filtros se pueden configurar en procmon para recopilar información?
5. ¿Cuáles son los indicadores basados en el host del malware?
6. ¿Hay alguna firma útil basada en red para este malware?

Ejercicio 5: Ejecuta el malware que se encuentra en el archivo Ejercic5.exe mientras lo monitorizas utilizando herramientas básicas de análisis dinámico en un entorno seguro.

1. ¿Qué observas al supervisar este malware con Process Explorer?
2. ¿Puedes identificar cualquier modificación de la memoria en vivo?
3. ¿Cuáles son los indicadores basados en el host del malware?
4. ¿Cuál es el propósito de este programa?

Ejercicio 6: Analiza el malware que se encuentra en el archivo Ejercic6.exe utilizando herramientas básicas de análisis dinámico.

1. ¿Qué sucede cuando ejecuta este archivo?

2. ¿Qué causa el bloqueo en el análisis dinámico?
3. ¿Hay otras formas de ejecutar este programa?

3.3 Análisis estático avanzado: Ensamblador x86

Software requerido: objdump -D -Mintel,i386 -b binary -m i386 ejercic7.bin
Material adicional: Ejercic7.bin, Catálogo de instrucciones de Intel

Ejercicio 7: Utilizando objdump, describe el propósito de la siguiente pieza de código:

EB1F5B31C088430B884318895B198D4B0C894B1D894321B00B8D4B198D5321CD80E8D
CFFFFFF2F2F2F2F62696E2F636174232F2F6574632F70617373776423414A495448414A49
54484B50

3.4 Análisis estático avanzado: IDA Pro

Software requerido: IDA Pro
Material adicional: Ejercic8.exe, Ejercic9.dll, Ejercic10.exe, Ejercic11.exe, Ejercic11.dll

Ejercicio 8: Utilizando el IDA Pro sobre el fichero Ejercic8.exe

1. ¿En qué rutina se realiza la verificación del número de serie?
2. ¿Cómo podría hacer para que el software diera como válido cualquier número de serie?
3. Genera un ejecutable modificado que admita cualquier número de serie

Ejercicio 9: Analiza el malware que se encuentra en el archivo Ejercic9.dll utilizando sólo IDA Pro.

1. ¿Cuál es la dirección de DllMain?
2. Utiliza la ventana Imports para buscar gethostbyname. ¿Dónde está la importación?
3. ¿Cuántas funciones llaman gethostbyname?
4. Centrándote en la llamada a gethostbyname ubicada en 0x10001757, ¿Qué solicitud de DNS se realizará?
5. ¿Cuántas variables locales IDA Pro ha reconocido para la subrutina en 0x10001656?
6. ¿Cuántos parámetros ha reconocido IDA Pro para la subrutina en 0x10001656?
7. Utilice la ventana Strings para localizar la cadena \ cmd.exe / c en el desensamblado. ¿Dónde está localizado?
8. ¿Qué sucede en el área de código que hace referencia a \ cmd.exe / c?
9. En la misma área, en 0x100101C8, parece que dword_1008E5C4 es una variable global que ayuda a decidir qué camino tomar. ¿Cómo se setea dword_1008E5C4? (Pista: utiliza las

referencias cruzadas de dword_1008E5C4.)
10. En la subrutina a 0x1000FF58 hay una serie de memcmp para comparar cadenas. ¿Qué sucede si la comparación de cadenas con robotwork tiene éxito (memcmp devuelve 0)?
11. ¿Qué hace el export PSLIST?
12. Utiliza el modo de gráfico para representar gráficamente las referencias cruzadas de sub_10004E79. ¿Qué funciones de la API se pueden llamar desde esta función? Basándote en las funciones de la API, ¿cómo llamarías a esta función?
13. ¿Cuántas funciones de la API de Windows DllMain llama directamente? ¿Cuántas a una profundidad de 2?
14. En 0x10001358, hay una llamada a Sleep (una función de API que toma un parámetro que contiene el número de milisegundos a dormir). Mirando hacia atrás a través del código, ¿cuánto tiempo dormirá el programa si este código se ejecuta?
15. En 0x10001701 hay una llamada a socket. ¿Cuáles son los tres parámetros?
16. ¿Utilizando la página de MSDN de la función socket y la funcionalidad de renombrado de constantes en IDA Pro, puede hacer que los parámetros sean más legibles? ¿Cuáles son los parámetros después de aplicar los cambios?
17. Busque el uso de la instrucción in (opcode 0xED). Esta instrucción se utiliza con una cadena mágica VMXh para detectar si el malware está corriendo en VMware. ¿Está en uso en este malware? Utilizando las referencias cruzadas a la función que ejecuta la instrucción in, ¿hay más pruebas de detección de VMware?

Ejercicio 10: Analiza el malware que se encuentra en el archivo Ejercic10.exe

1. ¿Cómo asegura este programa que continúe funcionando (tiene persistencia) cuando se reinicia el equipo?
2. ¿Para qué utiliza un mutex?
3. ¿Cuál sería una buena firma basada en host para detectar este programa?
4. ¿Cuál sería buena firma basada en la red para detectar este malware?
5. ¿Cuál es el propósito de este programa?
6. ¿Cuándo finalizará este programa?

Ejercicio 11 (OPCIONAL): Para este ejercicio, obtuvimos el archivo ejecutable malicioso, Ejercic11.exe y la DLL Ejercic11.dll, antes de ser ejecutados por la víctima. Esto es importante porque el malware puede cambiar al ejecutarse. Ambos archivos se encontraron en el mismo directorio en la máquina víctima. Si ejecutas el programa, debes asegurarte de que ambos archivos están en el mismo directorio en la máquina de análisis. (Encontrarás una IP que comienza por 127, en la versión real de este malware, esta dirección se conecta a una máquina remota, pero la hemos configurado para que se conecte a localhost para protegerte).

ADVERTENCIA: Este ejercicio puede causar daños considerables en tu PC y puede ser difícil de quitar una vez instalado. No ejecutes este archivo sin una máquina virtual con una instantánea tomada antes de la ejecución.

3.5 Análisis dinámico avanzado: OllyDbg

Software requerido: IDA Pro, OllyDbg y todo el utilizado hasta ahora
Material adicional: Ejercic12.exe, Ejercic13.exe, Ejercic14.exe, DLL1.dll, DLL2.dll, DLL3.dll, Ejercic8.ex

Ejercicio 12: Analiza el malware que se encuentra en el archivo Ejercic12.exe utilizando OllyDbg y IDA Pro para responder a las siguientes preguntas. (Este malware es el mismo del ejercicio 6)

1. ¿Cómo se puede instalar este malware?
2. ¿Cuáles son las opciones de línea de comandos para este programa? ¿Qué contraseña necesita?
3. ¿Cómo puedes utilizar OllyDbg para corregir permanentemente este malware, para que no requiera la contraseña?
4. ¿Cuáles son indicadores basados en el host de este malware?
5. ¿Cuáles son las diferentes instrucciones que puede recibir a través de la red?
6. ¿Hay alguna firma útil basada en red para este malware?

Ejercicio 13: Analiza el malware que se encuentra en el archivo Ejercic13.exe utilizando OllyDbg para responder a las siguientes preguntas.

1. ¿Qué cadenas ves estáticamente en el binario?
2. ¿Qué sucede cuando se ejecuta este binario?
3. ¿Cómo puede hacer que se ejecute la payload maliciosa?
4. ¿Qué sucede en 0x00401133?
5. ¿Qué argumentos se pasan a la subrutina 0x00401089?
6. ¿Qué nombre de dominio utiliza este malware?
7. ¿Qué rutina de codificación se está utilizando para ofuscar el nombre de dominio?
8. ¿Para qué sirve la llamada CreateProcessA en 0x0040106E?

Ejercicio 14: Analiza el malware que se encuentra en el archivo Ejercic14.exe utilizando OllyDbg y IDA Pro.
Este malware carga tres DLL incluidas (DLL1.dll, DLL2.dll y DLL3.dll) que se han creado para solicitar la misma ubicación de carga de memoria. Por lo tanto, al ver estos DLL en OllyDbg frente a IDA Pro, el código puede aparecer en diferentes ubicaciones de memoria. El propósito de este ejercicio es hacer que te sientas cómodo al encontrar la ubicación correcta del código dentro de IDA Pro cuando estés buscando código en OllyDbg.

1. ¿Qué archivos DLL son importados por Ejercic14.exe?
2. ¿Cuál es la dirección base solicitada por DLL1.dll, DLL2.dll y DLL3.dll?
3. Cuando utilizas OllyDbg para depurar Ejercic14.exe, ¿cuál es la dirección asignada para: DLL1.dll, DLL2.dll y DLL3.dll?
4. Cuando Ejercic14.exe llama a una función de importación de DLL1.dll, ¿qué hace esta función de importación?
5. Cuando Ejercic14.exe llama WriteFile, ¿cuál es el nombre de archivo que escribe?
6. Cuando Ejercic14.exe crea un trabajo utilizando NetScheduleJobAdd, ¿de dónde obtiene los datos del segundo parámetro?
7. Mientras ejecutas o depuras el programa, verás que imprime tres fragmentos de datos misteriosos. ¿Qué son?
8. ¿Cómo se puede cargar DLL2.dll en IDA Pro para que coincida con la dirección de carga utilizada por OllyDbg?

Ejercicio 15 (OPCIONAL): Desarrolla un keygen para Ejercic8.exe