Reverse-Engineering Cheat Sheet

Por Lenny Zeltser § Aprende a darle la vuelta al Malware
http://www.zeltser.com/reverse-malware

Traducido por Jose Selvi
http://www.pentester.es

Aproximación General

1. Preparate un laboratorio controlado y aislado donde examinar los especímenes de malware.
2. Haz un análisis del comportamiento para examinar las interacciones del especimen con el entorno.
3. Haz un análisis del código estático para entender el funcionamiento interno del especimen.
4. Haz un análisis del código dinámico para entender los aspectos más difíciles del código.
5. Si es necesario, desempaqueta el especimen.
6. Repite los pasos 2, 3, y 4 (el orden puede variar) hasta que cumplas con los objetivos del análisis.
7. Documenta lo que has encontrado y limpia el laboratorio para futuros análisis.

Análisis del comportamiento

Estate listo para volver a un estado correcto mediante dd, VMware snapshots, CoreRestore, Ghost, SteadyState, etc.

Monitorización de interacciones locales (Process Monitor, Process Explorer) y de la red (Wireshark, tcpdump).

Detecta los principales cambios locales (RegShot, Autoruns).

Redirige el tráfico de red (hosts file, DNS, Honeyd).

Activa los servicios (IRC, HTTP, SMTP, etc.) así como los necesites para provocar nuevo comportamiento en el especimen.

IDA Pro para análisis estático

Buscar texto	Alt+T
Mostrar ventana de cadenas	Mays+F12
Mostrar operando como hexadecimal	Q
Insertar comentario	:
Seguir salto o llamada	Enter
Volver a la anterior vista	Esc
Ir a la siguiente vista	Ctrl+Enter
Mostrar ventana de nombres	Mays+F4
Mostrar diagrama de flujo de las funciones	F12
Mostrar diagrama de llamadas a funciones	Ctrl+F12
Ir al *“entry point”* del programa	Ctrl+E
Ir a una dirección específica	G
Renombrar variable o función	N
Mostrar lista de nombres	Ctrl+L
Mostrar lista de segmentos	Ctrl+S
Mostrar referencias cruzadas de una función	Seleccionar nombre de la función » Ctrl+X
Mostrar pila de la función actual	Ctrl+K

OllyDbg para análisis dinámico

Saltar dentro de instrucción (*“Step Into”*)	F7
Saltar sobre la instrucción (*“Step Over”*)	F8
Ejecutar hasta el siguiente *breakpoint*	F9
Ejecutar hasta el siguiente *return* de función	Ctrl+F9
Mostrar anterior/siguiente instrucción ejecutada	- / +
Volver a la vista anterior	*
Mostrar mapa de memoria	Alt+M
Seguir expresiones en vista	Ctrl+G
Insertar comentario	;
Seguir salto o llamada en vista	Enter
Mostrar listado de nombres	Ctrl+N
Nueva búsqueda binaria	Ctrl+G
Siguiente resultado de búsqueda binaria	Ctrl+L
Mostrar listado de breakpoints	Alt+B
Añadir instrucción en lugar de la seleccionada	Seleccionar instrucción » Espacio
Editar datos en memoria u opcode de instrucciones	Seleccionar datos o instrucción » Ctrl+E
Mostrar cadena SEH	View » SEH chain
Mostrar parches	Ctrl+P

Evadiendo defensas de Malware

Para intentar desempaquetar rapidamente, infecta el sistema y vuelca la memoria con LordPE o OllyDump.

Para un desempaquetamiento más quirúrjico, localiza el *Entry Point* original (OEP) después de que se ejecute el desempaquetador.

Si no se puede desempaquetar limpiamente, examina el especimen empaquetado mediante análisis dinámico mientras está en ejecución.

Cuando desempaquetas en OllyDbg, intenta usar SFX (bytewise) y la opción "*Find OEP by Section Hop*" de OllyDump.

Oculta a OllyDbg mediante HideOD y OllyAdvanced.

Un JMP o CALL a EAX puede mostrarnos el OEP, posiblemente precedido de POPA o POPAD.

Busca saltos rebuscados mediante SEH, RET, CALL, etc.

Si el empaquetador usa SEH, anticipate al OEP siguiendo las áreas de la pila usadas para almacenar los manejadores del empaquetador.

Decodifica los datos protegidos examinando los resultados de decodificar una función mediante análisis dinámico de código.

Corrige los problemas con la cabecera PE con XPELister, LordPE, ImpREC, PEiD, etc.

Para acercarte al OEP, intenta con un *breakpoint* en las llamadas del desempaquetador a *LoadLibraryA* o *GetProcAddress*.

Registros x86 y usos comunes

EAX	Sumas, multiplicaciones, resultado de las funciones
ECX	Contador
EBP	Base para referenciar a los argumentos de la función (EBP+value) y a las variables locales (EBP-value)
ESP	Apunta a la cima de la pila; cambia mediante PUSH, POP, y otros
EIP	Apunta a la siguiente instrucción
EFLAGS	Contiene flags que almacenan resultados de computación (por ejemplo, Zero flag y Carry flag)

Los conceptos de análisis de Malware tras este resumen son cubiertos en el curso *SEC610: Reverse-Engineering Malware* del SANS Institute, impartido por Lenny Zeltser.