Practical Reverse Engineering Exercises - Write Ups

Chapter 1 – Exercise 3 – part1 (questions 1 to 3) (20th of July 2014)

TASK

"1. Repeat the walk-through by yourself. Draw the stack layout, including parameters and local variables.

2. In the example walk-through, we did a nearly one-to-one translation of the assembly code to C. As an exercise, re-decompile this whole function so that it looks more natural. What can you say about the developer's skill level/experience? Explain your reasons. Can you do a better job?

3. In some of the assembly listings, the function name has a @ prefix followed by a number. Explain when and why this decoration exists."

Excerpt from: *"Practical Reverse Engineering: x86, x64, ARM, Windows Kernel, Reversing Tools, and Obfuscation",*

Bruce Dang, Alexandre Gazet, Elias Bachaalany, Sebastien Josse, ISBN: 978-1-118-78731-1

MY ANSWERS

EXERCISE 3.1 – PRIMER STACK LAYOUT

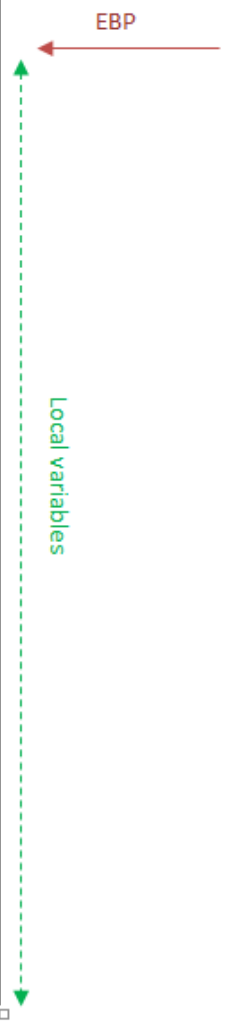
Here is the representation of the stack for Sample J's DllMain stack layout

Author: ePsiLoN (info at epsilon-labs dot com)

Practical Reverse Engineering Exercises - Write Ups

Chapter 1 – Exercise 3 – part1 (questions 1 to 3) (20th of July 2014)

EBP offset	Variable name	Notes
+ 10h	lpvReserved	DLL arguments
+ 0Ch	fdwReason	
+ 8	hinstDLL	
+ 4	Return address	
- 4		stores IDT register
- 6		
- 8	ebp-8 (var_8)	
- 0Ch		ebp-130h (pe) PROCESSENTRY32 structure
...		
- 108h		
- 10Ch	pe.szExeFile	
- 110h	pe.dwFlags	
- 114h	pe.pcPriClassBase	
- 118h	pe.th32ParentProcessID	
- 11Ch	pe.cntThreads	
- 120h	pe.th32ModuleID	
- 124h	pe.th32DefaultHeapID	
- 128h	pe.th32ProcessID	
- 12Ch	pe.cntUsage	
- 130h	pe.dwSize	



EXERCISE 3.2 – PRIMER C TRANSLATION

TODO

This exercise is still in my TODO list.

EXERCISE 3.3 – DECORATION EXPLANATION

The calling convention for those functions is __stdcall. According to http://msdn.microsoft.com/en-us/library/zxk0tw93.aspx calling convention decoration the function names are prefixed with underscore (_), the function name is followed by the at sign (@) followed by the number of bytes (in decimal) in the argument list. Usually _stdcall is used when calling Win32 API functions.

Author: ePsiLoN (info at epsilon-labs dot com)