

目录

回顾.....	1
Winasm_0.exe 分析.....	2
常用程序的特征.....	2
分析 BC++ 程序特征	2
分析 Delphi 程序特征	4
分析 VS 程序.....	5
分析 vc6.0 和易语言	5

回顾

1. IDA 的快捷键有哪些？

空格	切换选项卡
R	转换为 ASCII 码字符
N	重命名
G	打开跳转窗口
INS	打开创建结构体窗口(结构体窗口)
Q	将局部变量转为本身数值
Shift+F4	打开名称窗口
M	转换常量数据为宏定义
回车	选中地址按回车进入到地址内部
D	定义数据类型
Alt+Q	转换结构体类型(局部堆栈窗口)
F5	将反汇编反编译为 C 代码
Alt+M	添加标签
A	转换为 Ascii 码字符串
Tab	将反汇编反编译为 C 代码
ESC	返回上一步
:	添加注释
Alt+G	切换指令集(ARM 指令与 thumb 指令转换)
C	将数据转为代码
Ctrl+P	打开函数窗口
Alt+T	搜索文本
Ctrl+X	交叉引用
F7	单步步入(调试器)
F8	单步步过(调试器)
?	打开计算器
Ctrl+M	打开标签窗口
Ctrl+Alt+B	断点列表
F1	查看帮助文档
Ctrl+F7	运行到函数返回地址

Shift+F9	打开结构体窗口
U	将当前选中的数据会代码转为未定义
Ctrl+K	打开局部堆栈窗口
Shift+F3	打开函数窗口
Shift+F12	字符串窗口
Alt+K	打开设置堆栈窗口
P	将代码转为函数

Winasm_0.exe 分析

```

0334A      pExceptionInfo = dword ptr 8
0334A      enter    0, 0          ; push ebp
0334A      mov     ebp, esp      ; mov ebp, esp
0334E      push    esi          ; 保存寄存器环境
0334F      mov     esi, [ebp+pExceptionInfo]
03352      lodsd             ; eax = [esi] = pExceptionRecord
03352      ; esi += 4
03353      mov     eax, [eax]      ; eax = [eax] = [pExceptionRecord] = ExceptionCode
03353      ; = 0xC0000005
03355      and     eax, 0DEADFFh ; eax = 0xC0000005 & 0xDEADFF = 5
0335A      shl     eax, 5         ; eax << 5 = 0xA0
0335D      mov     ebx, eax      ; ebx = eax = 0xA0
0335F      lea     eax, [eax+402FAEh] ; eax = 0xA0 + 0x402FAE = 0x40304E
03365      mov     esi, [esi]    ; esi = [esi] = pContextRecord
03367      mov     [esi+CONTEXT._Edi], eax ; 修改 Context.edi = 40304E
0336D      add     eax, 2DCh
03372      mov     [ebx+esi], eax ; 修改 Context.ESI = 40332A
03375      mov     eax, 400FDFh
0337A      add     eax, ebx
0337C      mov     [esi+CONTEXT._Eip], eax ; 修改 Context.eip = 40107F
03382      xor     eax, eax
03384      dec     eax            ; 返回值是-1 继续回到修改后EIP处执行
03385      pop     esi          ; 恢复寄存器环境
03386      leave                    ; mov esp,ebp
03386      ; pop ebp

```

常用程序的特征

1. BC++
2. Delphi
3. 易语言
4. VB
5. VC++

分析 BC++程序特征

1. OEP 特征

- 2. 区段名称
- 3. 链接器版本

1. Oep 特征

```

.text:004012FC
.text:004012FC
.text:004012FC
.text:004012FC EB 10
.text:004012FC
.text:004012FE 66 62+aFbCHook
.text:00401300 00
.text:00401300 E9
.text:0040130A 98 40+
.text:0040130E
.text:0040130E
.text:0040130E
loc_40130E:
.text:0040130E A1 8B+
.text:00401313 C1 E0+
.text:00401316 A3 8F+
.text:0040131B 52

```

```

public start
proc near
jmp short loc_40130E

db 'fb:C++HOOK'
db 00h
db 0E9h
dd offset __CPPdebugHook

; -----
; CODE XREF: start
mov eax, dwTlsIndex
shl eax, 2
mov dword_46408F, eax
push edx

```

地址	HEX 数据	反汇编	注释
004012FC	EB 10	JMP SHORT BC6.0040130E	
004012FE	66	DB 66	CHAR 'f'
004012FF	62	DB 62	CHAR 'b'
00401300	3A	DB 3A	CHAR ':'
00401301	43	DB 43	CHAR 'c'
00401302	2B	DB 2B	CHAR '+'
00401303	2B	DB 2B	CHAR '+'
00401304	48	DB 48	CHAR 'H'
00401305	4F	DB 4F	CHAR 'O'
00401306	4F	DB 4F	CHAR 'O'
00401307	4B	DB 4B	CHAR 'K'
00401308	90	NOP	
00401309	E9	DB E9	
0040130A	98 404600	DD OFFSET BC6.__CPPdebugHook	
0040130E	A1 8B404600	MOV EAX, DWORD PTR DS:[0x46408B]	

二进制特征：EB 10 66 62 3A 43 2B 2B 48 4F 4F 4B 90

第一个 API 调用，GetModuleHandleA、

API 调用 IAT 时，采用模式是 FF25

地址	HEX 数据	反汇编	注释
00405B9C	FF25 E4814500	JMP DWORD PTR DS:[<&kernel32.GetModuleH	kernel3
00405BA2	8BC0	MOV EAX,EAX	
00405BA4	FF25 E0814500	JMP DWORD PTR DS:[<&kernel32.LocalAlloc	kernel3
00405BAA	8BC0	MOV EAX,EAX	
00405BAC	FF25 DC814500	JMP DWORD PTR DS:[<&kernel32.TlsGetValu	kernel3

- 2. 区段名称
- 分的比较细

节查看器

名称	V. 偏移	V. 大小	R. 偏移	R. 大小	标志
.text	00001000	00063000	00000600	00062800	60000020
.data	00064000	00009000	00062E00	00004800	C0000040
.tls	0006D000	00001000	00067600	00000200	C0000040
.rdata	0006E000	00001000	00067800	00000200	50000040
.idata	0006F000	00003000	00067A00	00002200	40000040
.edata	00072000	00001000	00069C00	00000200	40000040
.rsrc	00073000	00006000	00069E00	00005200	40000040
.reloc	00079000	00007000	0006F000	00006E00	50000040

关闭(C)

3. 链接器版本

5.0

分析 Delphi 程序特征

1. OEP 特征

地址	偏移	汇编	注释
00454A14	\$ 55	PUSH EBP	
00454A15	. 8BEC	MOV EBP,ESP	
00454A17	. 83C4 F0	ADD ESP,-0x10	
00454A1A	. B8 34484500	MOV EAX,Delphi7.00454834	UNICODE ";"
00454A1F	. E8 3C12FBFF	CALL Delphi7.00405C60	
00454A24	. A1 58604500	MOV EAX,DWORD PTR DS:[0x456058]	
00454A29	. 8B00	MOV EAX,DWORD PTR DS:[EAX]	
00454A2B	. E8 ACE5FFFF	CALL Delphi7.00452FDC	
00454A30	. 8B00 3C614500	MOV ECX,DWORD PTR DS:[0x45613C]	Delphi7.00457BFC
00454A36	. A1 58604500	MOV EAX,DWORD PTR DS:[0x456058]	
00454A3B	. 8B00	MOV EAX,DWORD PTR DS:[EAX]	
00454A3D	. 8B15 7C454500	MOV EDX,DWORD PTR DS:[0x45457C]	Delphi7.004545C8
00454A43	. E8 ACE5FFFF	CALL Delphi7.00452FF4	
00454A48	. A1 58604500	MOV EAX,DWORD PTR DS:[0x456058]	
00454A4D	. 8B00	MOV EAX,DWORD PTR DS:[EAX]	
00454A4F	. E8 20E6FFFF	CALL Delphi7.00453074	
00454A54	. E8 03F3FAFF	CALL Delphi7.00403D5C	
00454A59	. 8D40 00	LEA EAX,DWORD PTR DS:[EAX]	

5 个 CALL，第一个 CALL 内有 API 调用，GetModuleHandleA

2. 区段名称

名称	V. 偏移	V. 大小	R. 偏移	R. 大小	标志
CODE	00001000	00053A5C	00000400	00053C00	60000020
DATA	00055000	000011CC	00054000	00001200	C0000040
BSS	00057000	00000C05	00055200	00000000	C0000000
.idata	00058000	0000211A	00055200	00002200	C0000040
.tls	0005B000	00000010	00057400	00000000	C0000000
.rdata	0005C000	00000018	00057400	00000200	50000040
.reloc	0005D000	00006160	00057600	00006200	50000040
.rsrc	00064000	00006A00	0005D800	00006A00	50000040

关闭(C)

3. 链接器版本

2.25

分析 VS 程序

以链接器版本为例，分析 VC 程序

VC6.0	6.0
VC2003	7.0, 7.1
VC2005	8.0
VS2008	9.0
VS2010	10.0
VS2012	11.0
VS2013	12.0
VS2015	14.0
VS2017	14.1

分析 vc6.0 和易语言

OEP

地址	HEX	反汇编	注释
0044848D		PUSH EBP	
0044848E	.	MOV EBP,ESP	
00448490	.	PUSH -0x1	
00448492	.	PUSH 易语言.0046E818	
00448497	.	PUSH 易语言.0044CCBC	SE
0044849C	.	MOV EAX,DWORD PTR FS:[0]	
004484A2	.	PUSH EAX	
004484A3	.	MOV DWORD PTR FS:[0],ESP	
004484AA	.	SUB ESP,0x58	
004484AD	.	PUSH EBX	
004484AE	.	PUSH ESI	
004484AF	.	PUSH EDI	
004484B0	.	MOV [LOCAL.6],ESP	
004484B3	.	CALL DWORD PTR DS:[&KERNEL32.GetVersion	ker
004484B9	.	XOR EDX,EDX	
004484BB	.	MOV DL,AH	
004484BD	.	MOV DWORD PTR DS:[0x4040C0],EDX	

申请局部空间是：sub esp,0x58

第一个 API 调用是 GetVersion