

Back To User를 이용한 lena 4번 문제풀이

Back To User Mode란?

특정 이벤트를 이벤트가 일어나기 전으로
설정해두고 Call 명령이 일어난 다음 바로
다음 위치를 잡을 수 있는 기능이다.

참고사항

필자의 PC 환경은 win10 64bit이다.
필자의 PC에서 Back To User 기능을
사용해보려 했지만, 되지 않아 구글링을 해본 결과
64bit에서는 해당 기능이 되지 않는다고 한다.
그래서 Vitual Box에 32bit 운영체제를 올렸다.
추가적으로 Ollydbg는 1.10버전을 사용했다.

문제 풀이1

The screenshot shows the OllyDbg interface with the following components:

- Disassembly Window:** Shows assembly code for the `ntdll.KiFastSystemCallRet` function. The code includes instructions like `LEA ESP, DWORD PTR SS:[ESP]`, `INT 2E`, `RETN`, `PUSH EBP`, `MOV EBP, ESP`, `LEA ESP, DWORD PTR SS:[ESP+8]`, `CALL ntdll.RtlCaptureContext`, and `ADD DWORD PTR SS:[ESP+4], 4`.
- File Selection Dialog:** An "Open 32-bit executable" dialog box is open, showing a file list with "PixtopianBook" selected. The file name field contains "PixtopianBook" and the file type is set to "Executable file (*.exe)".
- Registers Window:** Shows the state of the CPU registers. The `EAX` register is highlighted with the value `000000C0`. Other registers like `ECX`, `EDX`, `EBX`, `ESP`, `EBP`, `ESI`, and `EDI` are also visible.
- Hex Dump Window:** Located at the bottom, it displays the memory dump corresponding to the assembly code, showing hex values and their ASCII representations.

Ollydbg를 이용해 PixtopianBook.exe를 Open한다.

문제 풀이2

OllyDbg - PixtopianBook.exe - [CPU - main thread, module Pixtopia]

File View Debug Plugins Options Window Help

0044036E \$ 55 PUSH EBP
0044036F . 8BEC MOV EBP,ESP
00440371 . 6A FF PUSH -1
00440373 . 68 00440000 PUSH Pixtopia.00440000
00440378 . 68 00440000 PUSH Pixtopia.00440000
0044037D . 64:A1 00000000 MOV EAX,DWORD PTR FS:[0]
00440383 . 50 PUSH EAX
00440384 . 64:8925 00000000 MOV DWORD PTR FS:[0],ESP
0044038B . 83EC 58 SUB ESP,58
0044038E . 53 PUSH EBX
0044038F . 56 PUSH ESI
00440390 . 57 PUSH EDI
00440391 . 8965 E8 MOV DWORD PTR SS:[EBP-18],ESP
00440394 . FF15 8C524700 CALL DWORD PTR DS:[C:\WINDOWS\system32\kernel32.GetVersion]
0044039A . 3302 XOR EDX,EDX
0044039C . 8004 MOV DL,AH
0044039E . 8915 AC9E4900 MOV DWORD PTR DS:[499EAC],EDX
004403A4 . 8BC8 MOV ECX,EAX
004403A6 . 81E1 FF000000 AND ECX,0FF
004403AC . 890D A89E4900 MOV DWORD PTR DS:[499EA8],ECX
004403B2 . C1E1 08 SHL ECX,8
004403B5 . 03CA ADD ECX,EDX
004403B7 . 890D A49E4900 MOV DWORD PTR DS:[499EA4],ECX
004403BD . C1E8 10 SHR EAX,10
004403C0 . A3 A09E4900 MOV DWORD PTR DS:[499EA0],EAX
004403C5 . 6A 01 PUSH 1
004403C7 . E8 7A330000 CALL Pixtopia.00440374
004403CC . 59 POP ECX
004403CD . 85C0 TEST EAX,EAX
004403CF . 75 08 JNZ SHORT Pixtopia.004403D9
004403D1 . 6A 1C PUSH 1C
004403D3 . E8 C3000000 CALL Pixtopia.0044039B

SE handler installation

kernel32.GetVersion

Registers (FPU)

EAX 0012C6F0 UNICODE "C:\Windows\system32\msintf.dll"
ECX 00000004
EDX 00000020
EBX 00000000
ESP 0012BFD0
EBP 0012BF24
ESI 7FFDF000
EDI 0012BF88

EIP 778A70F4 ntdll.KiFastSystemCallRet

C 0 ES 0023 32bit 0(FFFFFFFF)
P 1 CS 001B 32bit 0(FFFFFFFF)
A 0 SS 0023 32bit 0(FFFFFFFF)
Z 1 DS 0023 32bit 0(FFFFFFFF)
S 0 FS 003B 32bit 7FFDF000(FFF)
T 0 GS 0000 NULL
D 0
0 0 LastErr ERROR_NO_TOKEN (000003F0)

EFL 00000246 (NO,NB,E,BE,NS,PE,GE,LE)

ST0 empty 0.0
ST1 empty 0.0
ST2 empty 0.0
ST3 empty 0.0
ST4 empty 0.0
ST5 empty 0.0
ST6 empty 0.0
ST7 empty 0.0

FST 4020 Cond 1 0 0 0 Err 0 0 1 0 0 0 0 (EQ)
FCW 027F Prec NEAR,53 Mask 1 1 1 1 1 1

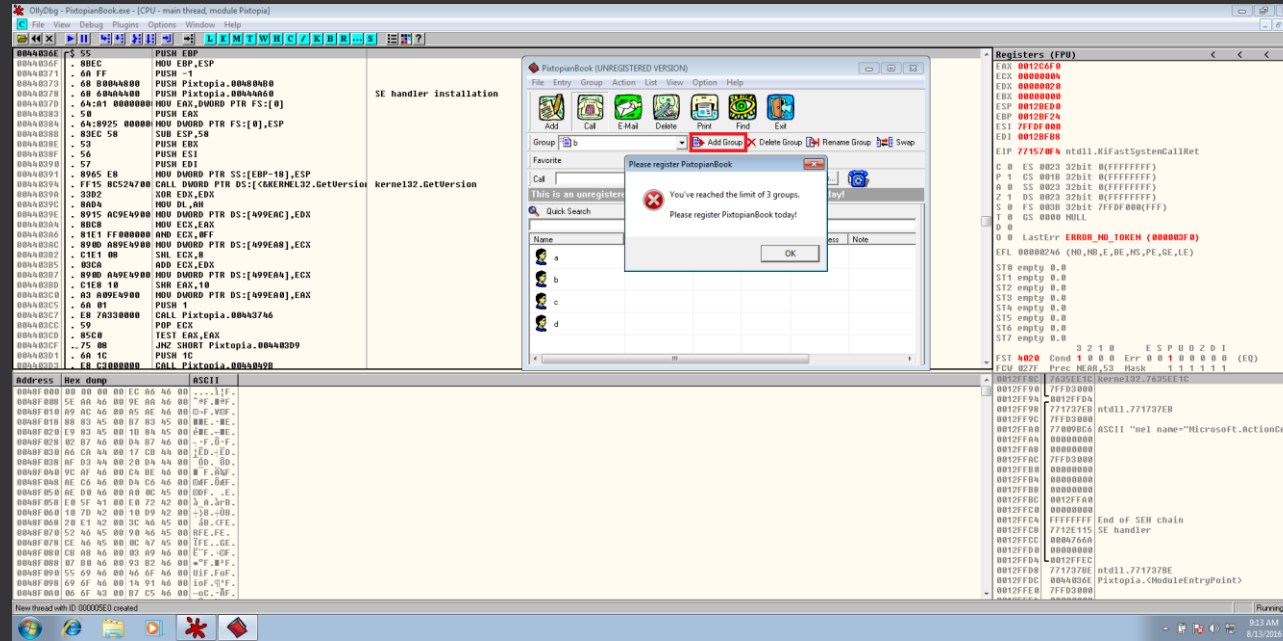
0012FF8C 779FEE1C kernel32.779FEE1C
0012FF90 77FD9000
0012FF94 0012FFD4
0012FF98 778C37EB ntdll.778C37EB
0012FF9C 77FD9000
0012FFA0 77915A02 ntdll.77915A02
0012FFA4 00000000
0012FFA8 00000000
0012FFAC 77FD9000
0012FFB0 00000000
0012FFB4 00000000
0012FFB8 00000000
0012FFBC 0012FFA0
0012FFC0 00000000
0012FFC4 FFFFFFFF End of SEH chain
0012FFC8 7787E115 SE handler
0012FFCC 0000B70E
0012FFD0 00000000
0012FFD4 0012FFEC
0012FFD8 778C37BE ntdll.778C37BE
0012FFDC 0044036E Pixtopia.<ModuleEntryPoint>
0012FFE0 77FD9000

Module C:\Windows\system32\WINNSI.DLL

Running

프로그램을 열고 F9(Run) 또는 **빨간색** 박스를
클릭 하면 프로그램이 실행되는 것을 볼 수 있다.

문제 풀이3



빨간색 박스를 클릭하면 그룹을 만들 수 있는데

기존에는 3개이상 만들려고 하면 화면과 같이

에러창이 출력된다.

이제 패치를 통해 그룹을 더 생성해보도록 할 것이다.

문제 풀이 4

The screenshot displays the OllyDbg interface with the following components:

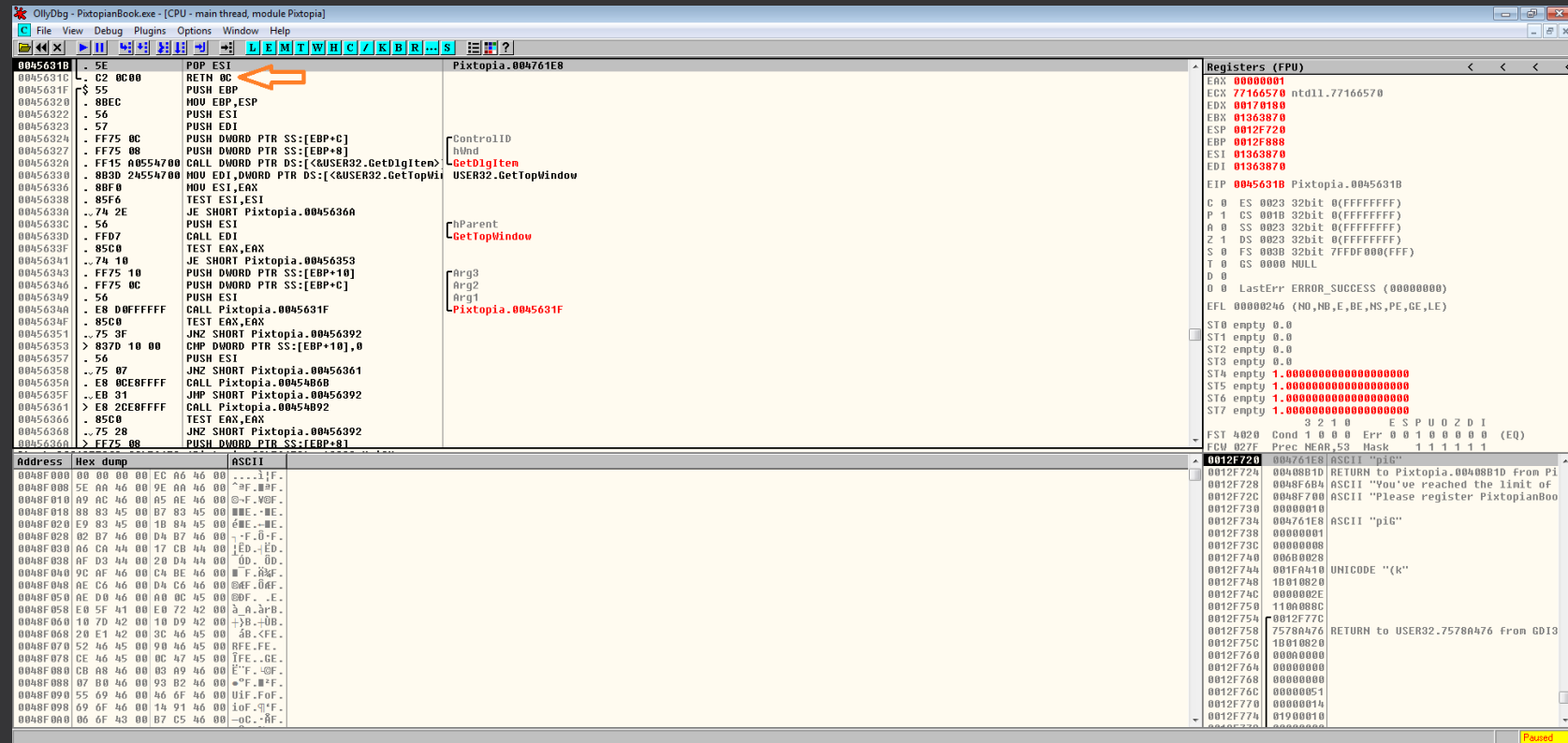
- Assembly Window:** Shows assembly code for the 'ntdll' module. The instruction at address 80A424 is highlighted: `ADD DWORD PTR SS:[ESP+4], 4`.
- Registers Window:** Shows the CPU registers. The EIP register is at 771570F4, and the ESP register is at 0012F3E4.
- Dialog Box:** A 'Please register PixtopianBook' dialog box is open in the center, displaying an error message: 'You've reached the limit of 3 groups. Please register PixtopianBook today!'. The 'OK' button is visible.
- Debugger Interface:** The bottom right corner of the debugger window features a 'Back to user' button, which is highlighted by an orange arrow.

에러메시지 창이 뜬 상태에서 Ollydbg에서 Pasue(F12)를 하고

Execute till user code (Alt + F9)를 누르면 우측 하단 처럼

Back To User mode로 진입하는걸 확인 할 수 있다.

문제 풀이5



이전 페이지에서 봤던 에러메시지 창에 있는 OK 버튼을 누르면 해당 코드창으로 넘어가게 된다.

그럼 F8을 눌러서 화살표가 가리키고 있는 RETN으로 진입해보자.

문제 풀이6

The screenshot shows the OllyDbg interface with the following details:

- Assembly Window:** Displays assembly code from address 00408A00 to 00408B3F. Instruction 00408B1D is highlighted in red. The instruction is `JL SHORT Pixtopia.00408B34`. The comment for this instruction is `ASCII "Please register PixtopianBook"`.
- Registers (FPU) Window:** Shows the current state of registers. EIP is 00408B1D, pointing to `Pixtopia.00408B1D`.
- ASCII Dump Window:** Shows the ASCII dump of the memory at the current instruction pointer. The message `ASCII "Please register PixtopianBook"` is visible.

RETN을 타고 넘어오면 00408B1D로 오게되는데

스크롤을 조금 올려보면 아까 봤던 메시지가 적혀있는 부분을 보게된다.

자세히 보면 빨간 박스 부분의 JL문을 지나면서 해당 에러창이

출력되는 것을 알 수 있다. 그렇다면 해당 부분을 JMP(무조건 점프)로

바꿔주게 되면 어떻게 될까?

문제 풀이7

OllyDbg - PixtopianBook.exe - [CPU - main thread, module Pixtopia]

```
00408AE8 . 53          PUSH EBX
00408AEC . 8BD9       MOV EBX,ECX
00408AEE . 56         PUSH ESI
00408AEF . 6A 00     PUSH 0
00408AF1 . 8B83 A4000000 MOV ECX,DWORD PTR DS:[EBX+A4]
00408AF7 . 6A 00     PUSH 0
00408AF9 . 68 46010000 PUSH 146
00408AFE . 50         PUSH EAX
00408AFF . FF15 D0564700 CALL DUWORD PTR DS:[<&USER32.SendMessageA@00401000]
0040B005 . 83F8 03   CMP EAX,3
0040B008 . EB 2A     JMP SHORT Pixtopia.00408B34
0040B00A . 6A 10     PUSH 10
0040B00C . 68 00F74000 PUSH Pixtopia.0048F700
0040B011 . 68 D4F64800 PUSH Pixtopia.0048F604
0040B014 . 8BCB     MOV ECX,EBX
0040B018 . E8 D0D70400 CALL Pixtopia.004562ED
0040B01D . 5E       POP ESI
0040B01E . 5B       POP EBX
0040B01F . 8B8C24 340100 MOV ECX,DWORD PTR SS:[ESP+134]
0040B026 . 64:8900 000000 MOV DWORD PTR FS:[0],ECX
0040B02D . 81C4 40010000 ADD ESP,140
0040B033 . C3       RETN
0040B034 . 6A 00     PUSH 0
0040B036 . 8D4C24 24 LEA ECX,DWORD PTR SS:[ESP+24]
0040B03A . E8 21E20100 CALL Pixtopia.00426D00
0040B03F . 6A 00     PUSH 0
0040B041 . 8D4C24 24 LEA ECX,DWORD PTR SS:[ESP+24]
0040B045 . C78424 480100 MOV DWORD PTR SS:[ESP+148],0
0040B050 . E8 8B502000 CALL Pixtopia.0042DBE0
0040B055 . 6A 00     PUSH 0
0040B057 . 68 A8F64800 PUSH Pixtopia.0048F608
0040B05C . 68 DCF64800 PUSH Pixtopia.0048F60C
```

Registers (FPU)

```
ERX 00000001
ECX 77166570 ntdll.77166570
EDX 00170180
EBX 01363870
ESP 0012F734
EBP 0012F888
ESI 004761E8 ASCII "pic"
EDI 01363870
EIP 00408B1D Pixtopia.00408B1D
C 0 ES 0023 32bit 0(FFFFFFFF)
P 1 CS 001B 32bit 0(FFFFFFFF)
A 0 SS 0023 32bit 0(FFFFFFFF)
Z 1 DS 0023 32bit 0(FFFFFFFF)
S 0 FS 0038 32bit 7FDF00(FFF)
T 0 GS 0000 NULL
D 0
D 0
D 0 LastErrr ERROR_SUCCESS (00000000)
EFL 00000246 (NO,NB,E,OE,NS,PE,GE,LE)
ST0 empty 0.0
ST1 empty 0.0
ST2 empty 0.0
ST3 empty 0.0
ST4 empty 1.000000000000000000000000
ST5 empty 1.000000000000000000000000
ST6 empty 1.000000000000000000000000
ST7 empty 1.000000000000000000000000
FST 4020 Cond 1 0 0 0 Err 0 0 1 0 0 0 0 0 (EQ)
FCW 027F Prec NEAR,53 Mask 1 1 1 1 1 1
```

0012F734 004761E8 ASCII "pic"

```
0012F738 00000001
0012F73C 00000000
0012F740 00600028
0012F744 001FA440 UNICODE "{k}"
0012F748 10010020
0012F74C 0000002E
0012F750 110A088C
0012F754 0012F77C
0012F758 7578A476 RETURN to USER32.7578A476 from GD13
0012F75C 10010020
0012F760 000A0000
0012F764 00000000
0012F768 00000000
0012F76C 00000051
0012F770 00000014
0012F774 01900010
0012F778 00000000
0012F77C 75755D4E RETURN to USER32.75755D4E from USER
0012F780 75777961 RETURN to USER32.75777961 from USER
0012F784 000301B4
0012F788 0012F7BC
```

JMP문으로 바꾸어주게 되면서 표기된 00408B34부분으로 점프(Jump)하게 되어

해당 메시지가 출력이 되지 않을 것이다.

확인해보도록 하자.

결과 확인

The screenshot shows the OllyDbg interface with the following components:

- Assembly Window:** Displays assembly code for the main thread. Key instructions include:
 - 00408AEB: PUSH EBX
 - 00408AEC: MOV EBX, ECX
 - 00408AEE: PUSH ESI
 - 00408AEF: PUSH 0
 - 00408AF1: MOV EAX, DWORD PTR DS:[EBX+4]
 - 00408AF7: PUSH 0
 - 00408AF9: PUSH 146
 - 00408AFE: PUSH EAX
 - 00408AFF: CALL DWORD PTR DS:[<USER32.SendMessageA
 - 00408B05: CMP EAX, 3
 - 00408B08: JMP SHORT Pixtopia.00408B34
 - 00408B0A: PUSH 10
 - 00408B0C: PUSH Pixtopia.0048F700
 - 00408B11: PUSH Pixtopia.0048F6B4
 - 00408B16: MOV ECX, EBX
 - 00408B18: CALL Pixtopia.004562ED
 - 00408B1E: POP ESI
 - 00408B20: POP EBX
 - 00408B26: MOV DWORD PTR FS:[0], ECX
 - 00408B2D: ADD ESP, 140
 - 00408B33: RETN 0
 - 00408B34: PUSH 0
 - 00408B36: LEA ECX, DWORD PTR SS:[ESP+24]
 - 00408B3A: CALL Pixtopia.00426D06
 - 00408B3F: PUSH 0
 - 00408B41: LEA ECX, DWORD PTR SS:[ESP+24]
 - 00408B45: MOV DWORD PTR SS:[ESP+148], 0
 - 00408B50: CALL Pixtopia.0042D8E0
 - 00408B55: PUSH 0
 - 00408B57: PUSH Pixtopia.0048F6A8
 - 00408B5C: PUSH Pixtopia.0048F69C
- Registers (FPU):** Shows the current state of registers, including EAX (00000001), ECX (77166570), EDX (00170180), and ESP (0012F734).
- Hex Dump:** Shows memory addresses from 0048F000 to 0048F0A0 with their corresponding hex and ASCII values.
- PixtopianBook (UNREGISTERED VERSION) Window:** A secondary application window is open, displaying a 'New Group' dialog box. The dialog has a 'Group name' field and 'OK' and 'Cancel' buttons. A red arrow points to the 'Add Group' button in the main window's toolbar.

처음과 같이 그룹 추가 버튼을 누르면 아까와는 다르게
그룹추가가 가능 하게 된 것을 확인 할 수 있다.

GOOD~~~~

감사합니다.^^