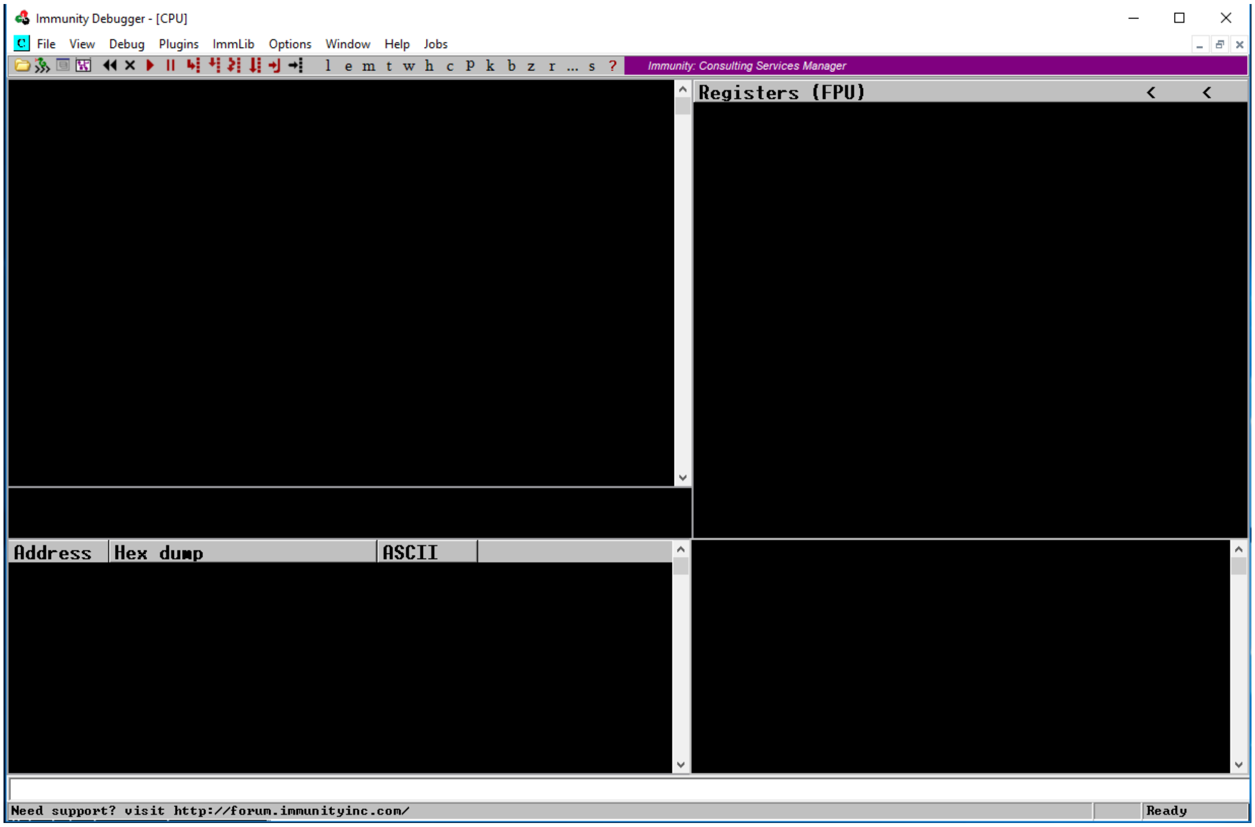
Working with Immunity Debugger

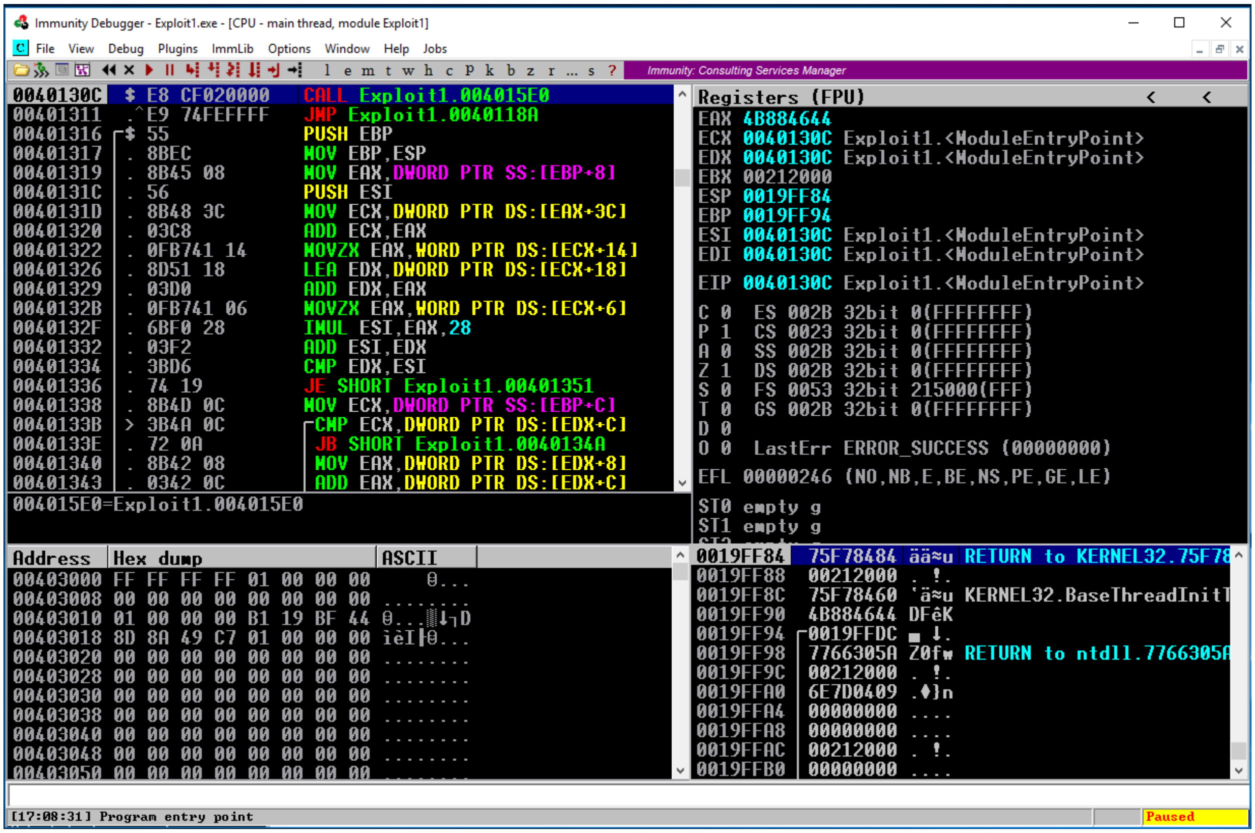
In this Lab we want to have a basic understanding of Immunity Debugger which we will be using for the rest of the course. From the Virtual Machine given, open Immunity Debugger to reach something similar to the following screen:


Immunity Debugger - [CPU]
C File View Debug Plugins ImmLib Options Window Help Jobs
l e m t w h c P k b z r ... s ?
Immunity: Consulting Services Manager
Registers (FPU)
Address
Hex dump
ASCII
Ready

Need support? visit http://forum.immunityinc.com/

Load any application, for example the putty.exe (or anything else). It does not matter for this lab; we only want to have a basic understanding.

You should end up with something similar to the following:


Immunity Debugger - Exploit1.exe - [CPU - main thread, module Exploit1]
File View Debug Plugins ImmLib Options Window Help Jobs
l e m t w h c P k b z r ... s ?
Immunity: Consulting Services Manager
Paused

```
0040130C  $ E8 CF020000    CALL Exploit1.004015E0
00401311  .^E9 74FEFFFF    JMP Exploit1.0040118A
00401316 ┌$ 55             PUSH EBP
00401317 │. 8BEC           MOV EBP,ESP
00401319 │. 8B45 08        MOV EAX,DWORD PTR SS:[EBP+8]
0040131C │. 56             PUSH ESI
0040131D │. 8B48 3C        MOV ECX,DWORD PTR DS:[EAX+3C]
00401320 │. 03C8           ADD ECX,EAX
00401322 │. 0FB741 14      MOVZX EAX,WORD PTR DS:[ECX+14]
00401326 │. 8D51 18        LEA EDX,DWORD PTR DS:[ECX+18]
00401329 │. 03D0           ADD EDX,EAX
0040132B │. 0FB741 06      MOVZX EAX,WORD PTR DS:[ECX+6]
0040132F │. 6BF0 28        IMUL ESI,EAX,28
00401332 │. 03F2           ADD ESI,EDX
00401334 │. 3BD6           CMP EDX,ESI
00401336 │. 74 19          JE SHORT Exploit1.00401351
00401338 │. 8B4D 0C        MOV ECX,DWORD PTR SS:[EBP+C]
0040133B │> 3B4A 0C        ┌CMP ECX,DWORD PTR DS:[EDX+C]
0040133E │. 72 0A          │JB SHORT Exploit1.0040134A
00401340 │. 8B42 08        │MOV EAX,DWORD PTR DS:[EDX+8]
00401343 │. 0342 0C        │ADD EAX,DWORD PTR DS:[EDX+C]
004015E0=Exploit1.004015E0
```

```
Registers (FPU)
EAX 4B884644
ECX 0040130C Exploit1.<ModuleEntryPoint>
EDX 0040130C Exploit1.<ModuleEntryPoint>
EBX 00212000
ESP 0019FF84
EBP 0019FF94
ESI 0040130C Exploit1.<ModuleEntryPoint>
EDI 0040130C Exploit1.<ModuleEntryPoint>

EIP 0040130C Exploit1.<ModuleEntryPoint>

C 0  ES 002B 32bit 0(FFFFFFFF)
P 1  CS 0023 32bit 0(FFFFFFFF)
A 0  SS 002B 32bit 0(FFFFFFFF)
Z 1  DS 002B 32bit 0(FFFFFFFF)
S 0  FS 0053 32bit 215000(FFF)
T 0  GS 002B 32bit 0(FFFFFFFF)
D 0
O 0  LastErr ERROR_SUCCESS (00000000)

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

ST0 empty g
ST1 empty g
```

```
Address  Hex dump                 ASCII
00403000 FF FF FF FF 01 00 00 00      θ...
00403008 00 00 00 00 00 00 00 00  ........
00403010 01 00 00 00 B1 19 BF 44  θ...▒↓┐D
00403018 8D 8A 49 C7 01 00 00 00  ìèI╟θ...
00403020 00 00 00 00 00 00 00 00  ........
00403028 00 00 00 00 00 00 00 00  ........
00403030 00 00 00 00 00 00 00 00  ........
00403038 00 00 00 00 00 00 00 00  ........
00403040 00 00 00 00 00 00 00 00  ........
00403048 00 00 00 00 00 00 00 00  ........
00403050 00 00 00 00 00 00 00 00  ........
```

```
0019FF84   75F78484 ää≈u RETURN to KERNEL32.75F78
0019FF88   00212000 . !.
0019FF8C   75F78460 `ä≈u KERNEL32.BaseThreadInitT
0019FF90   4B884644 DFêK
0019FF94  ┌0019FFDC ▄ ↓.
0019FF98  │7766305A Z0fw RETURN to ntdll.7766305A
0019FF9C  │00212000 . !.
0019FFA0  │6E7D0409 .♦}n
0019FFA4  │00000000 ....
0019FFA8  │00000000 ....
0019FFAC  │00212000 . !.
0019FFB0  │00000000 ....
```

```
[17:08:31] Program entry point
```

Task #1: Basic navigation - Panes:

- What is the Top left pane for?

- What is the Bottom left pane for?

- What is the Top right pane for?

- What is the Bottom right pane for?

General:

- What does paused mean?
- Which instruction should be executed after resuming the program?
- What is the first value in the stack?
- What does it point to in memory?

Windows:

- Where is the loaded modules window and what is it for?
- Where is the log window and what is it for?
- Where is the CPU window and what is it for?
- Where is the breakpoints window and what is it for?

Task #2: Finding program sections

Find the .text section of your application and tell me, what is the access available to this section? Why?

Task #3: Other navigations

Please navigate and try to understand the different debugging features available, how to add and remove a breakpoint, how to search for commands, add comments, and following commands. You will be learning more during the course, but these are most of the basics that we need for now.

Task #4: Open vs Attach

What is the difference between “File → Open” and “File → Attach”?

Task #5: Reflection

Please reflect on what you learned in this lab.