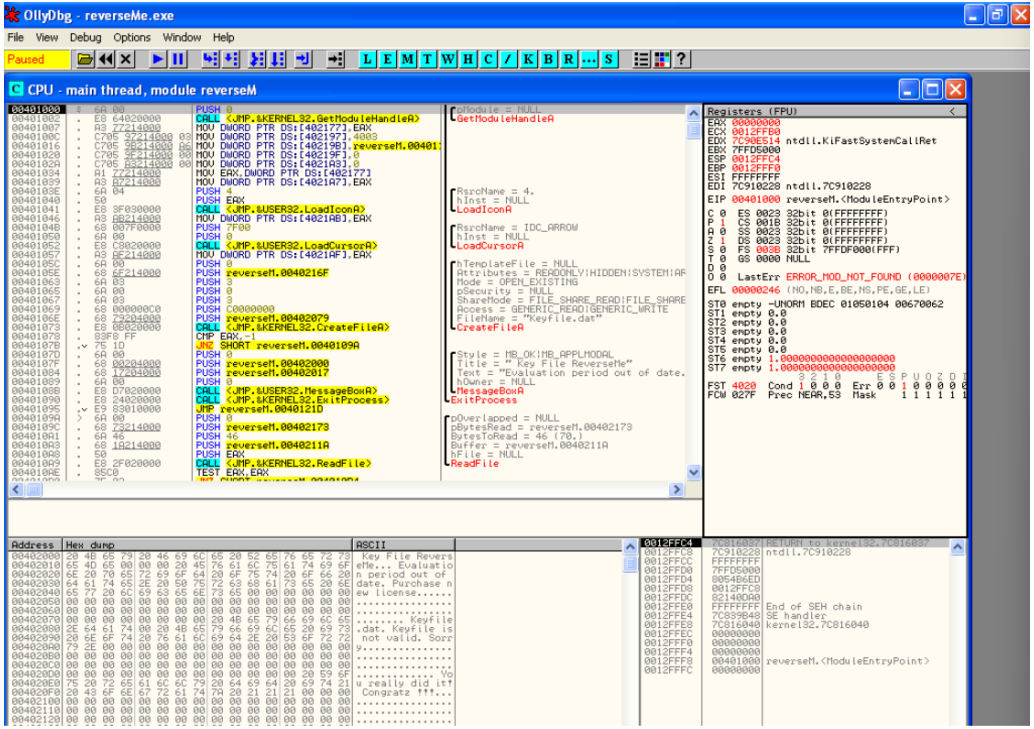
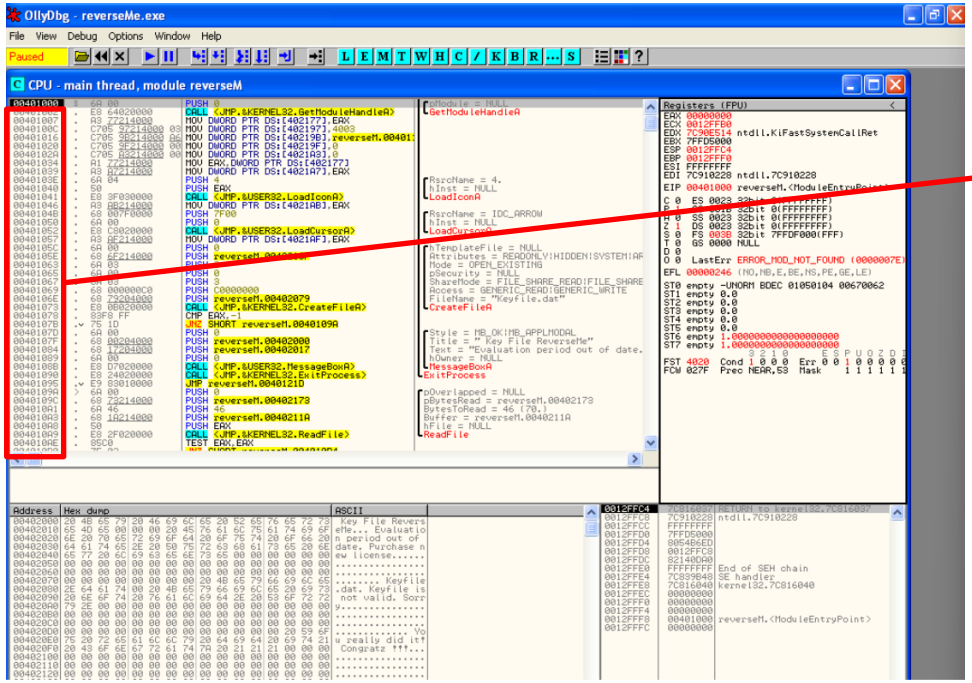
Lena Tutorial

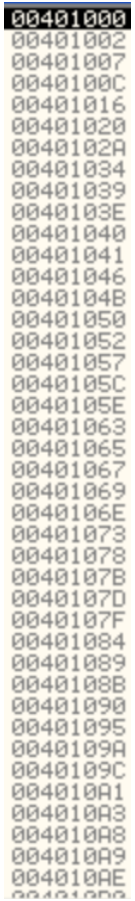
정보보호학과 신재형

Ollydbg

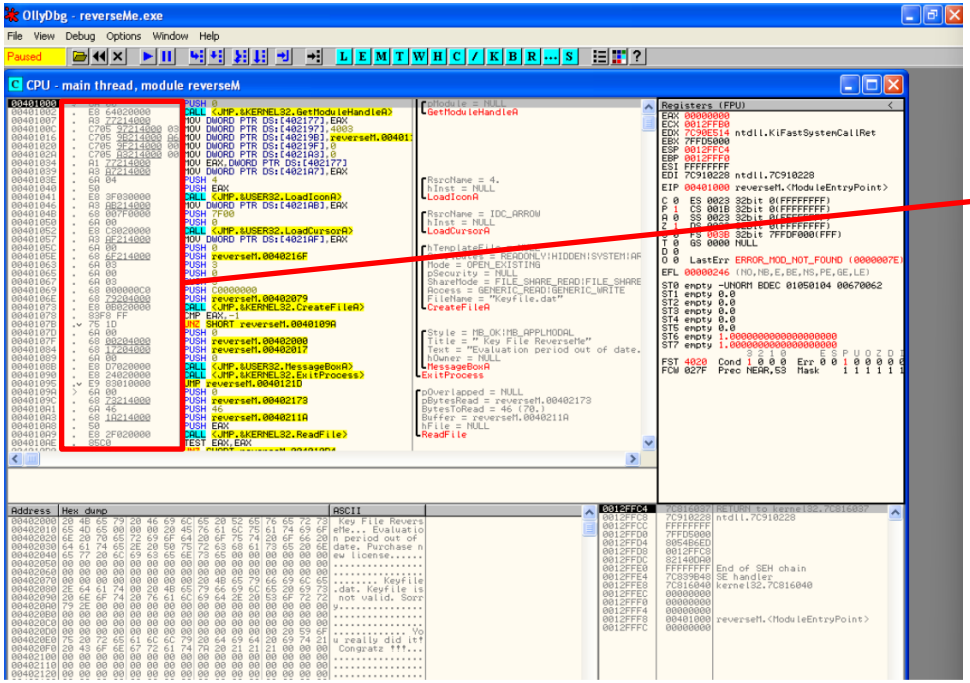

OllyDbg - reverseMe.exe
File View Debug Options Window Help
Paused
CPU - main thread, module reverseM
Registers (FPU)
Address Hex dump ASCII

Virtual Memory



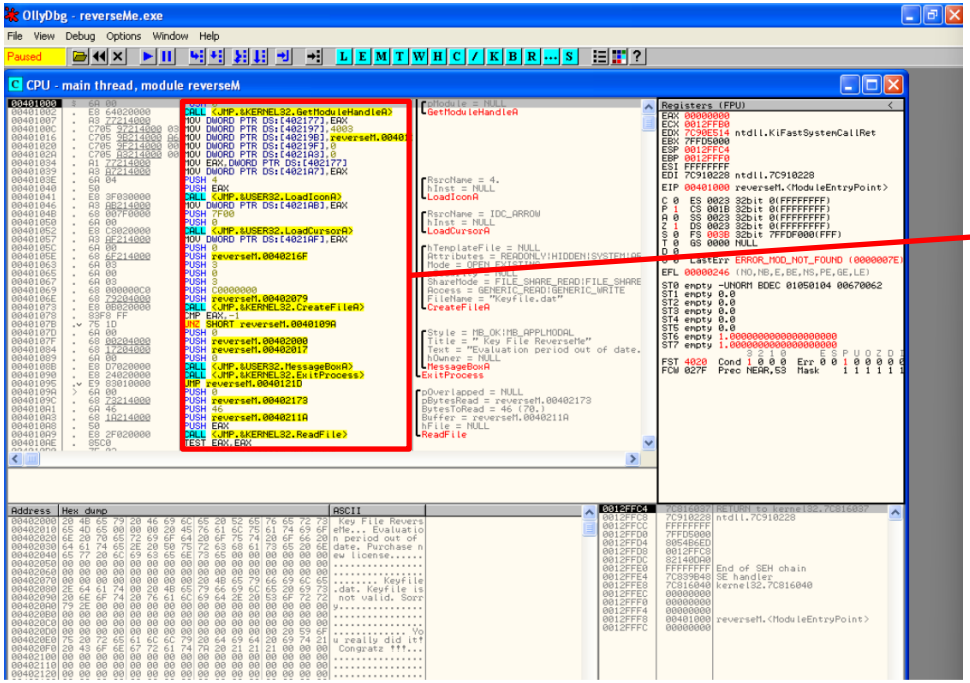


Machine Code



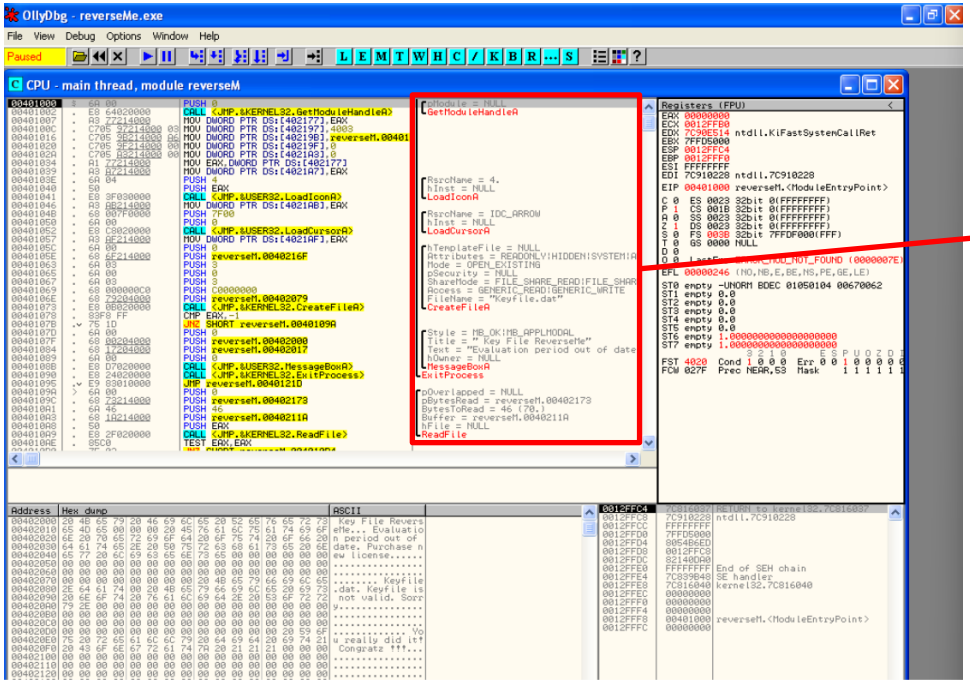

```
$  6A 00
.  E8 64020000
.  A3 77214000
.  C705 97214000 03400000
.  C705 9B214000 A6114000
.  C705 9F214000 00000000
.  C705 A3214000 00000000
.  A1 77214000
.  A3 A7214000
.  6A 04
.  50
.  E8 3F030000
.  A3 AB214000
.  68 007F0000
.  6A 00
.  E8 C8020000
.  A3 AF214000
.  6A 00
.  68 6F214000
.  6A 03
.  6A 00
.  6A 03
.  68 000000C0
.  68 79204000
.  E8 0B020000
.  83F8 FF
.v 75 1D
.  6A 00
.  68 00204000
.  68 17204000
.  6A 00
.  E8 D7020000
.  E8 24020000
.v E9 83010000
>  6A 00
.  68 73214000
.  6A 46
.  68 1A214000
.  50
.  E8 2F020000
.  85C0
```

Asembler



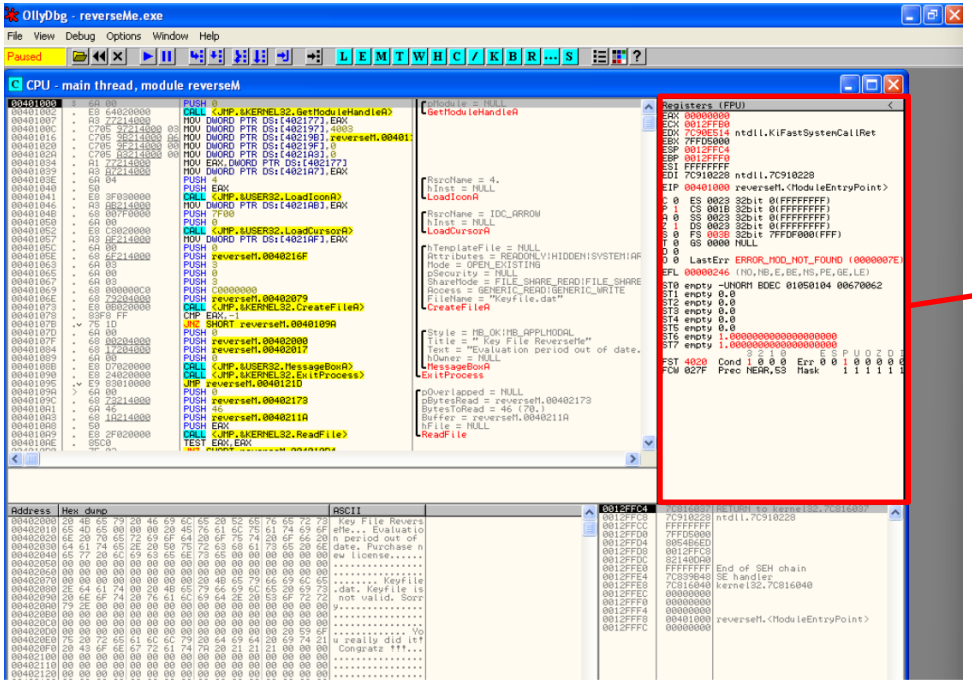

```
PUSH 0
CALL <JMP.&KERNEL32.GetModuleHandleA>
MOV DWORD PTR DS:[402177],EAX
MOV DWORD PTR DS:[402197],4003
MOV DWORD PTR DS:[40219B],reverseM.0040
MOV DWORD PTR DS:[40219F],0
MOV DWORD PTR DS:[4021A3],0
MOV EAX,DWORD PTR DS:[402177]
MOV DWORD PTR DS:[4021A7],EAX
PUSH 4
PUSH EAX
CALL <JMP.&USER32.LoadIconA>
MOV DWORD PTR DS:[4021AB],EAX
PUSH 7F00
PUSH 0
CALL <JMP.&USER32.LoadCursorA>
MOV DWORD PTR DS:[4021AF],EAX
PUSH 0
PUSH reverseM.0040216F
PUSH 3
PUSH 0
PUSH 3
PUSH C0000000
PUSH reverseM.00402079
CALL <JMP.&KERNEL32.CreateFileA>
CMP EAX,-1
JNZ SHORT reverseM.0040109A
PUSH 0
PUSH reverseM.00402000
PUSH reverseM.00402017
PUSH 0
CALL <JMP.&USER32.MessageBoxA>
CALL <JMP.&KERNEL32.ExitProcess>
JMP reverseM.0040121D
PUSH 0
PUSH reverseM.00402173
PUSH 46
PUSH reverseM.0040211A
PUSH EAX
CALL <JMP.&KERNEL32.ReadFile>
TEST EAX,EAX
```

Annoate



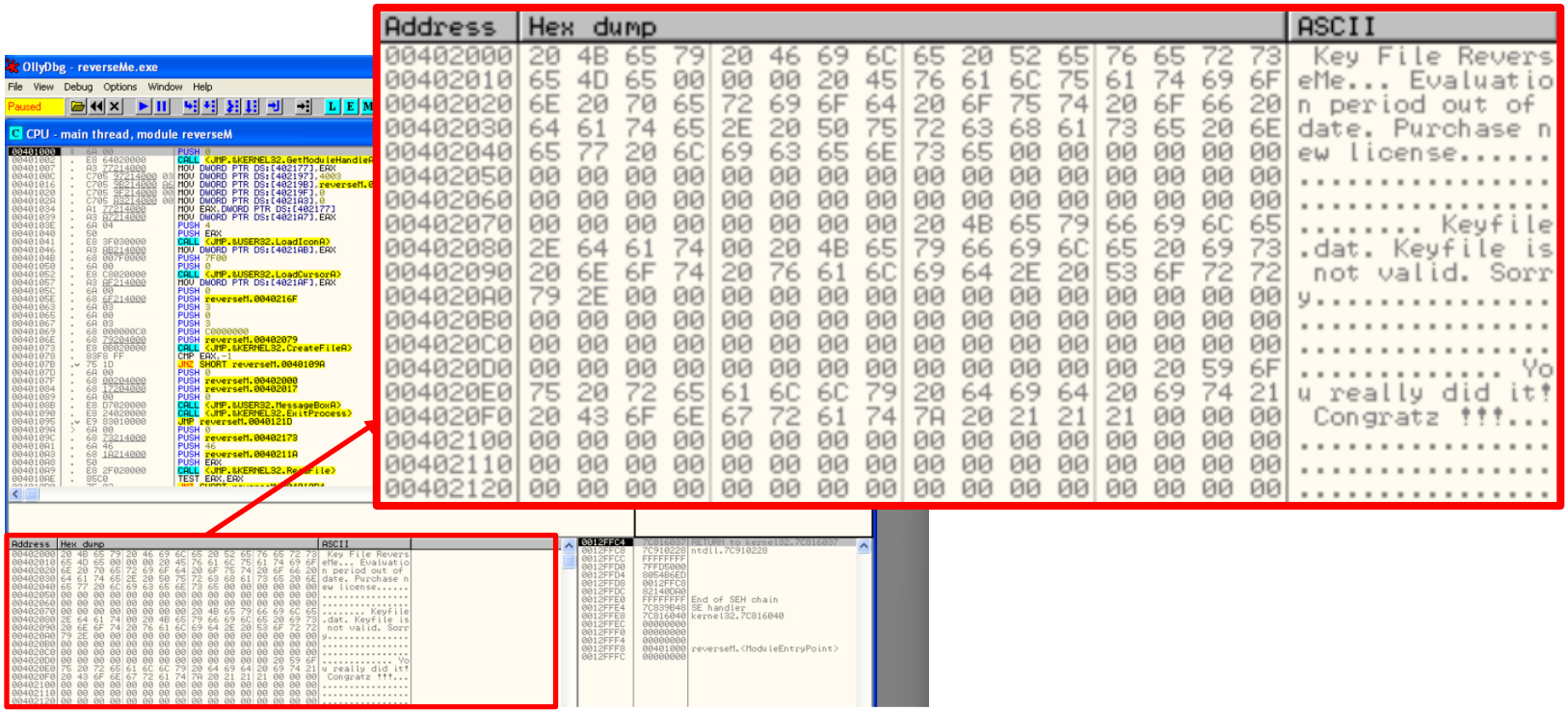

```
pModule = NULL
GetModuleHandleA

RsrcName = 4.
hInst = NULL
LoadIconA

RsrcName = IDC_ARROW
hInst = NULL
LoadCursorA

hTemplateFile = NULL
Attributes = READONLY|HIDDEN|SYSTEM|ARCHIVE|TEMPORARY|402048
Mode = OPEN_EXISTING
pSecurity = NULL
ShareMode = FILE_SHARE_READ|FILE_SHARE_WRITE
Access = GENERIC_READ|GENERIC_WRITE
FileName = "Keyfile.dat"
CreateFileA

Style = MB_OK|MB_APPLMODAL
Title = " Key File ReverseMe"
Text = "Evaluation period out of date. Purchase new license"
hOwner = NULL
MessageBoxA
ExitProcess

pOverlapped = NULL
pBytesRead = reverseM.00402173
BytesToRead = 46 (70.)
Buffer = reverseM.0040211A
hFile = NULL
ReadFile
```

Registers

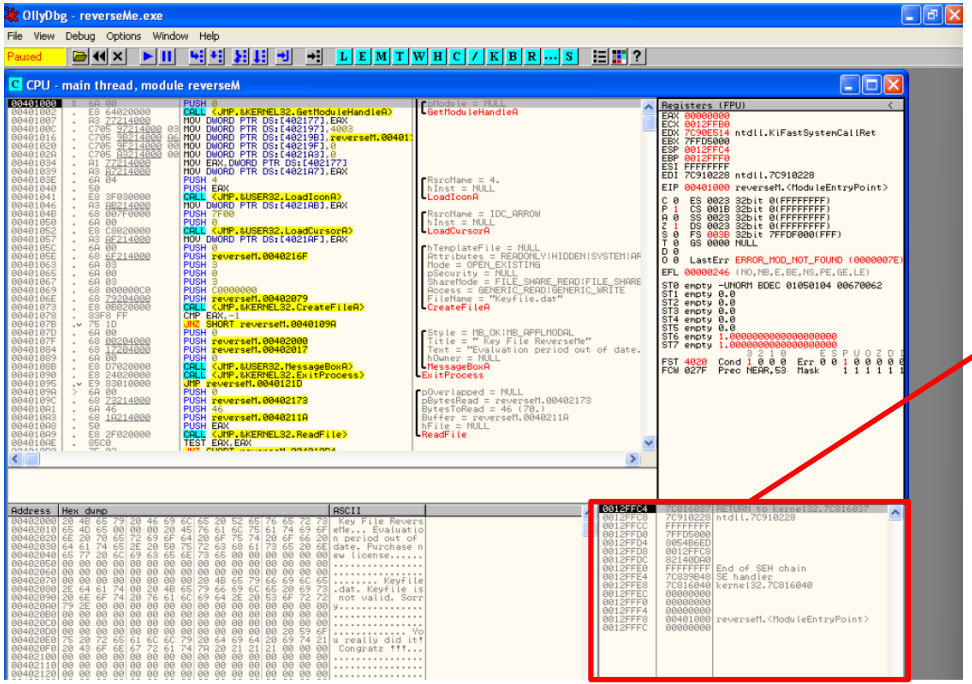


Registers (FPU)
EAX 00000000
ECX 0012FFB0
EDX 7C90E514 ntdll.KiFastSystemCallRet
EBX 7FFD5000
ESP 0012FFC4
EBP 0012FFF0
ESI FFFFFFFF
EDI 7C910228 ntdll.7C910228
EIP 00401000 reverseM.<ModuleEntryPoint>
C 0 ES 0023 32bit 0(FFFFFFFF)
P 1 CS 001B 32bit 0(FFFFFFFF)
A 0 SS 0023 32bit 0(FFFFFFFF)
Z 1 DS 0023 32bit 0(FFFFFFFF)
S 0 FS 003B 32bit 7FFDF000(FFF)
T 0 GS 0000 NULL
D 0
O 0 LastErr ERROR_MOD_NOT_FOUND (0000007E)
EFL 00000246 (NO,NB,E,BE,NS,PE,GE,LE)
ST0 empty -UNORM BDEC 01050104 00670062
ST1 empty 0.0
ST2 empty 0.0
ST3 empty 0.0
ST4 empty 0.0
ST5 empty 0.0
ST6 empty 1.0000000000000000000
ST7 empty 1.0000000000000000000
FST 4020 Cond 1 0 0 0 Err 0 0 1 0 0 0 0 0
FCW 027F Prec NEAR,53 Mask 1 1 1 1 1 1

Hex Dump


Address | Hex dump | ASCII
00402000 20 4B 65 79 20 46 69 6C 65 20 52 65 76 65 72 73 Key File Revers
00402010 65 4D 65 00 00 00 20 45 76 61 6C 75 61 74 69 6F eMe... Evaluatio
00402020 6E 20 70 65 72 69 6F 64 20 6F 75 74 20 6F 66 20 n period out of
00402030 64 61 74 65 2E 20 50 75 72 63 68 61 73 65 20 6E date. Purchase n
00402040 65 77 20 6C 69 63 65 6E 73 65 00 00 00 00 00 00 ew license.....
00402050 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ....
00402060 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ....
00402070 00 00 00 00 00 00 00 00 20 4B 65 79 66 69 6C 65 .... Keyfile
00402080 2E 64 61 74 00 20 4B 65 79 66 69 6C 65 20 69 73 .dat. Keyfile is
00402090 20 6E 6F 74 20 76 61 6C 69 64 2E 20 53 6F 72 72 not valid. Sorr
004020A0 79 2E 00 00 00 00 00 00 00 00 00 00 00 00 00 00 y...............
004020B0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ....
004020C0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ....
004020D0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 20 59 6F ....Yo
004020E0 75 20 72 65 61 6C 6C 79 20 64 69 64 20 69 74 21 u really did it!
004020F0 20 43 6F 6E 67 72 61 74 7A 20 21 21 21 00 00 00 Congratz !!!...
00402100 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ....
00402110 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ....
00402120 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ....

Stack



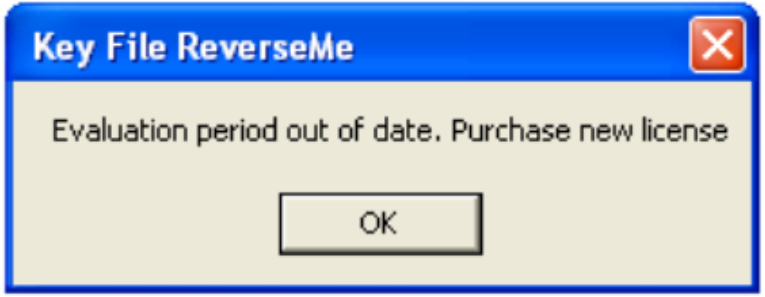
ESP 0012FFC4
EBP 0012FFF0

0012FFC4	7C816037	RETURN to kernel32.7C816037
0012FFC8	7C910228	ntdll.7C910228
0012FFCC	FFFFFFFF	
0012FFD0	7FFD5000	
0012FFD4	8054B6ED	
0012FFD8	0012FFC8	
0012FFDC	82140DA0	
0012FFE0	FFFFFFFF	End of SEH chain
0012FFE4	7C839B48	SE handler
0012FFE8	7C816040	kernel32.7C816040
0012FFEC	00000000	
0012FFF0	00000000	
0012FFF4	00000000	
0012FFF8	00401000	reverseM.<ModuleEntryPoint>
0012FFFC	00000000	

Basics of Assembler
Assembler : the basics
Wij

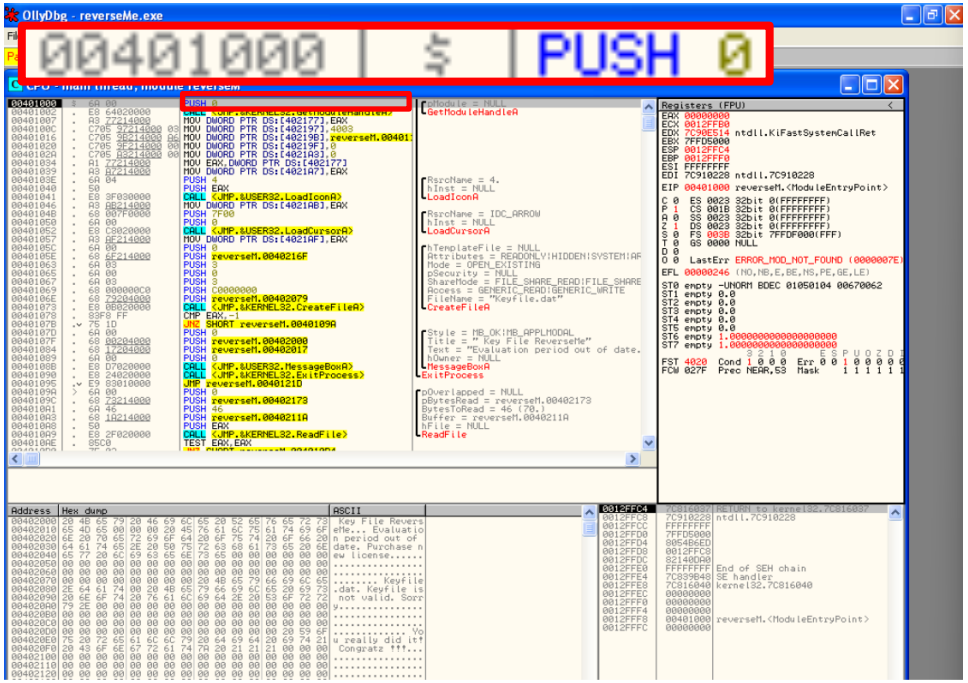
LENA
151
reverseMe

Shortcut Keys
HTML Document
11 KB



PUSH 0

Stack




```
0012FFC4   7C816037   RETURN to kernel32.7C816037
0012FFC8   7C910228   ntdll.7C910228
0012FFCC   FFFFFFFF
0012FFD0   7FFD5000
0012FFD4   8054B6ED
0012FFD8   0012FFC8
0012FFDC   82140DA0
0012FFE0   FFFFFFFF   End of SEH chain
0012FFE4   7C839B48   SE handler
0012FFE8   7C816040   kernel32.7C816040
0012FFEC   00000000
0012FFF0   00000000
0012FFF4   00000000
0012FFF8   00401000   reverseM.<ModuleEntryPoint>
0012FFFC   00000000
```

```
0012FFC0   00000000   pModule = NULL
0012FFC4   7C816037   RETURN to kernel32.7C81603
0012FFC8   7C910228   ntdll.7C910228
0012FFCC   FFFFFFFF
0012FFD0   7FFD5000
0012FFD4   8054B6ED
0012FFD8   0012FFC8
0012FFDC   82140DA0
0012FFE0   FFFFFFFF   End of SEH chain
0012FFE4   7C839B48   SE handler
0012FFE8   7C816040   kernel32.7C816040
0012FFEC   00000000
0012FFF0   00000000
0012FFF4   00000000
0012FFF8   00401000   reverseM.<ModuleEntryPoint
```

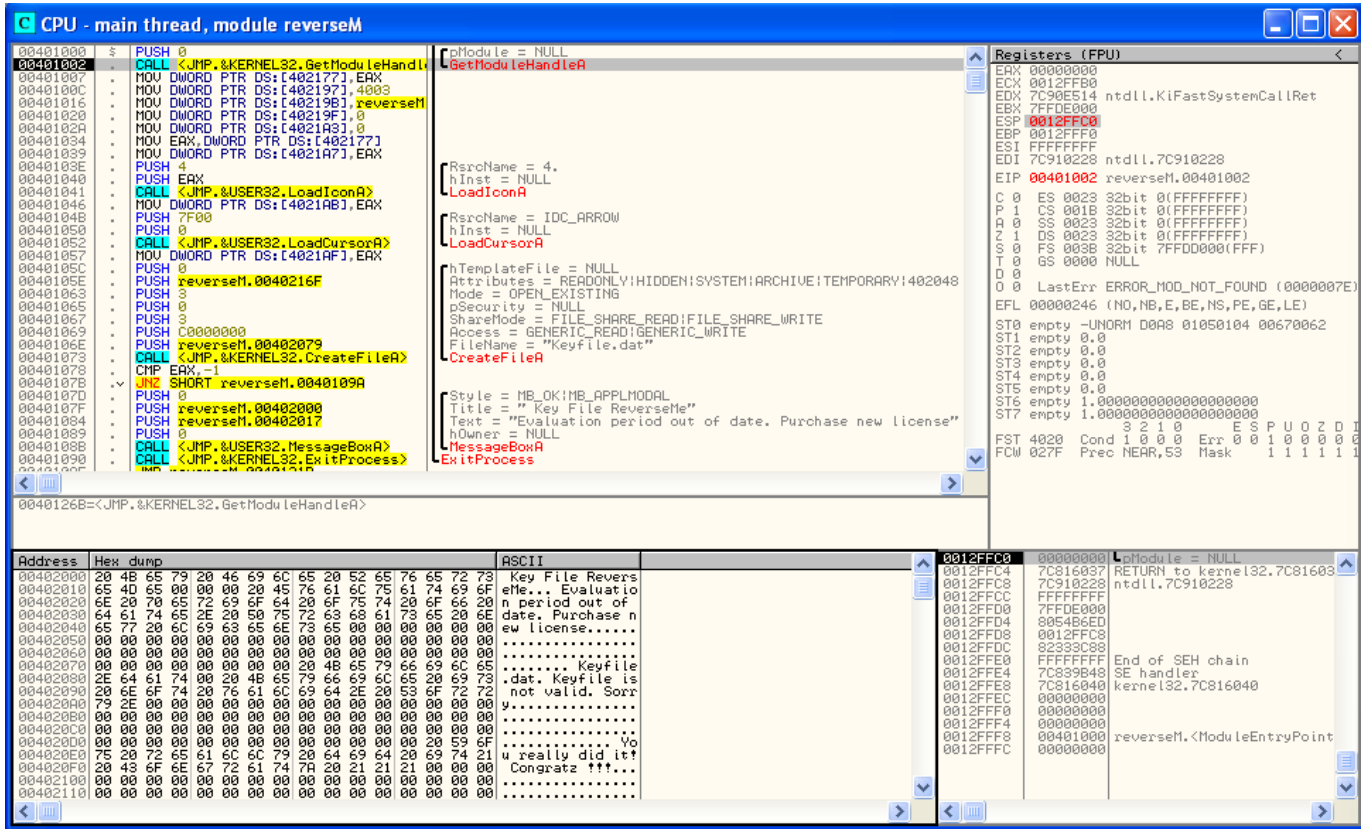
PUSH 0

Registers

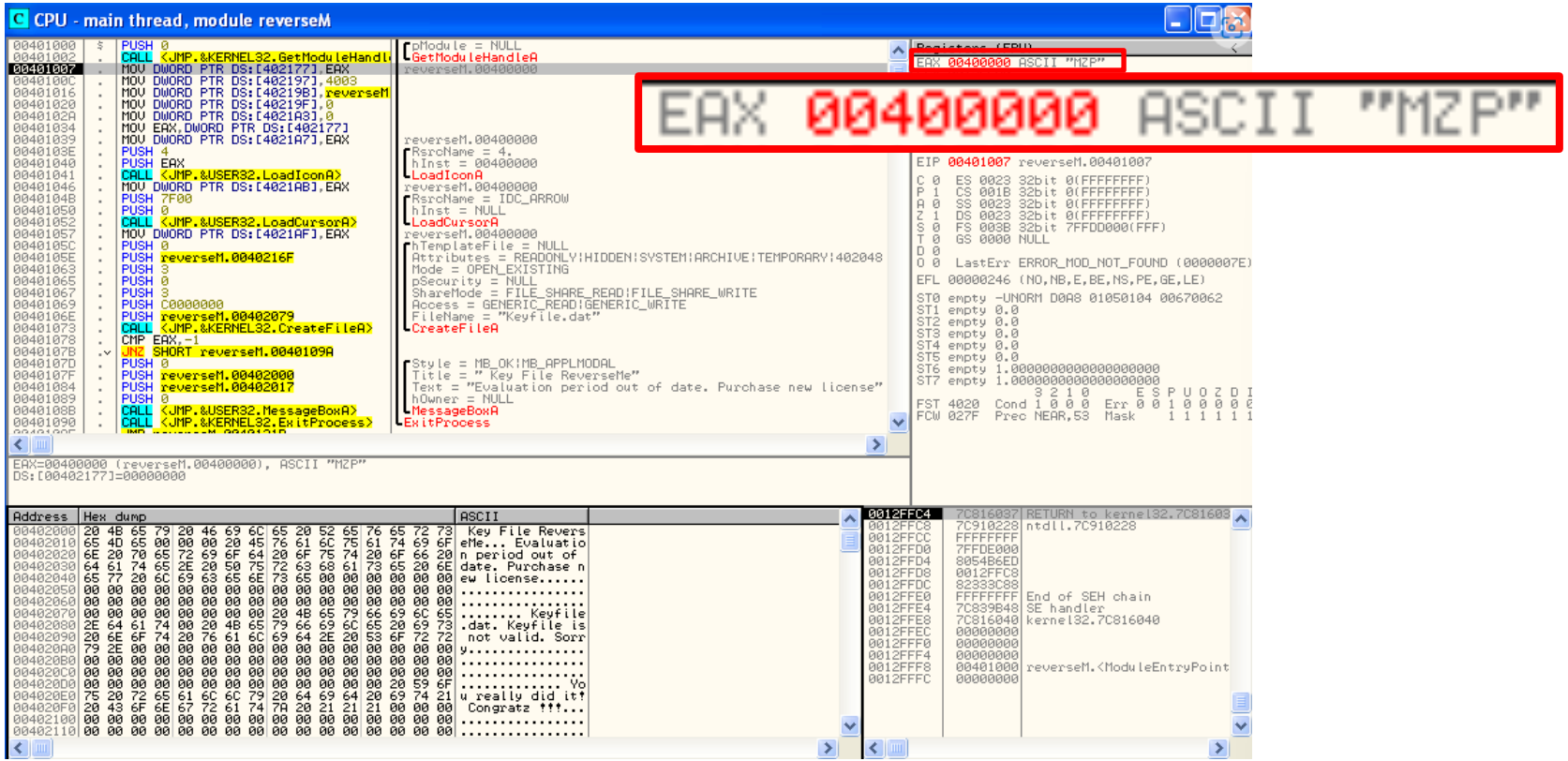
```
ESP 0012FFC4
EBP 0012FFF0
ESI FFFFFFFF
EDI 7C910228 ntdll.7C910228
EIP 00401000 reverseM.<ModuleEntryPoint>
```

```
ESP 0012FFC0
EBP 0012FFF0
ESI FFFFFFFF
EDI 7C910228 ntdll.7C910228
EIP 00401002 reverseM.00401002
```

CALL <JMP.&KERNEL32.GetModuleHandleA>

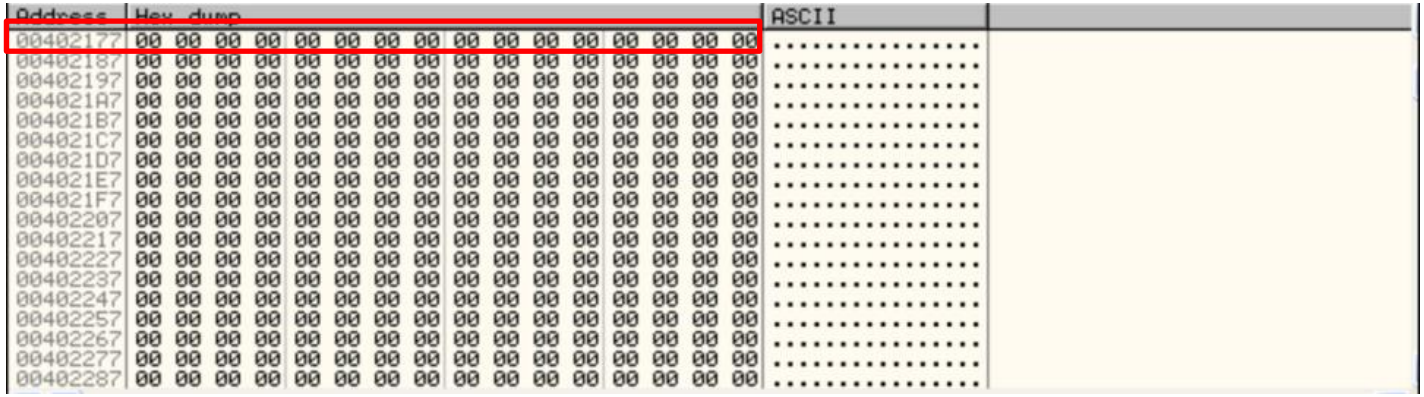


MOV DWORD PTR DS: [402177], EAX



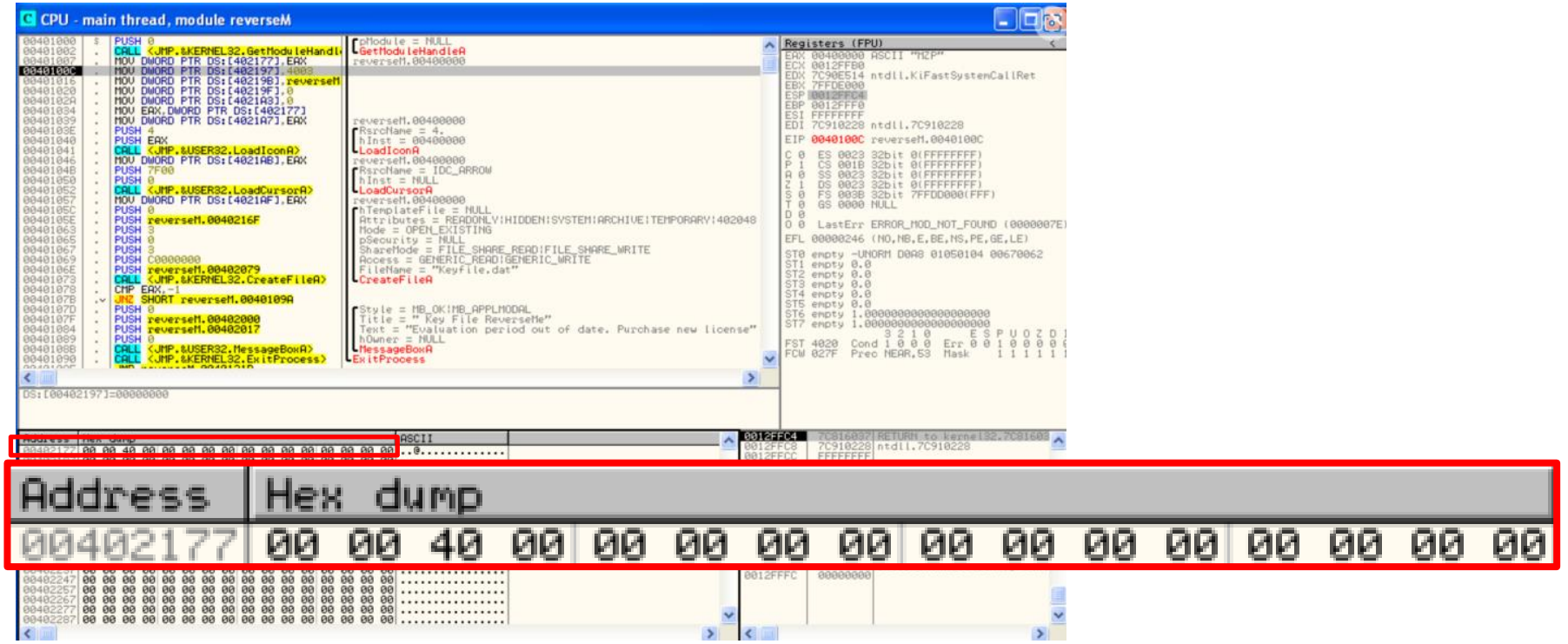
Dump

00402177

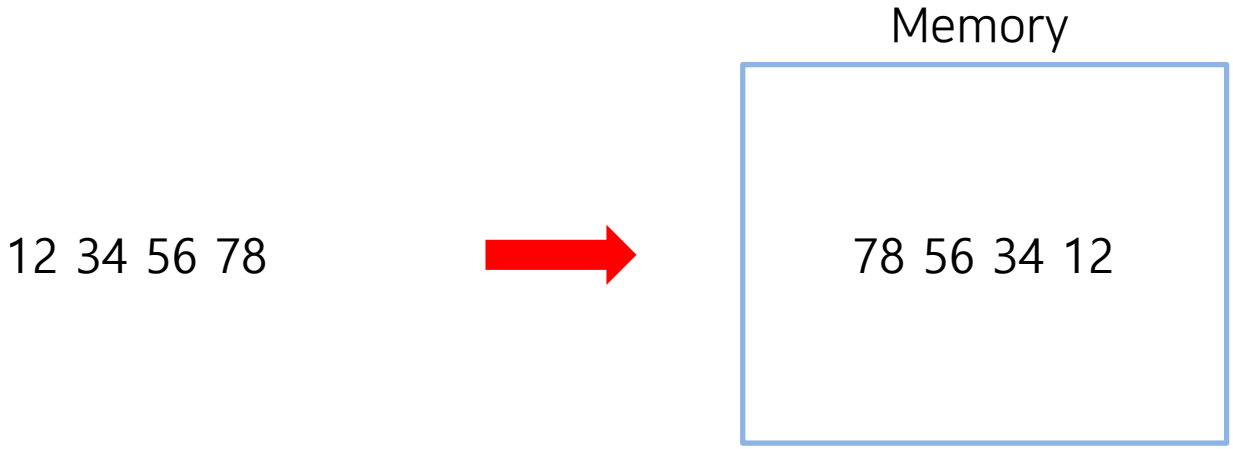


Address	Hex dump	ASCII
00402177	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	....
00402187	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	....
00402197	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	....
004021A7	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	....
004021B7	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	....
004021C7	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	....
004021D7	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	....
004021E7	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	....
004021F7	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	....
00402207	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	....
00402217	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	....
00402227	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	....
00402237	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	....
00402247	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	....
00402257	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	....
00402267	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	....
00402277	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	....
00402287	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	....

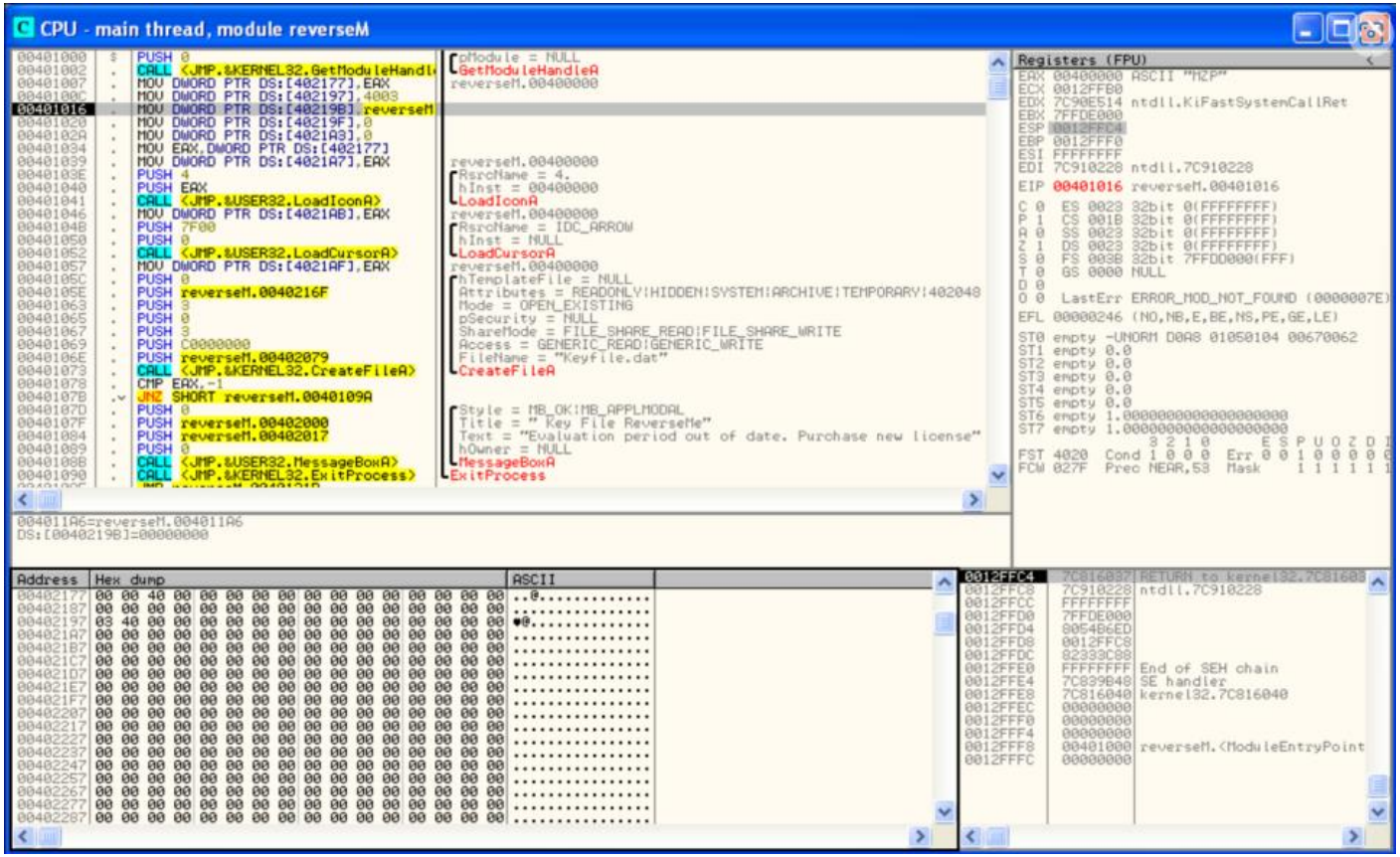
MOV DWORD PTR DS: [402197], 4003



리틀 엔디안 방식

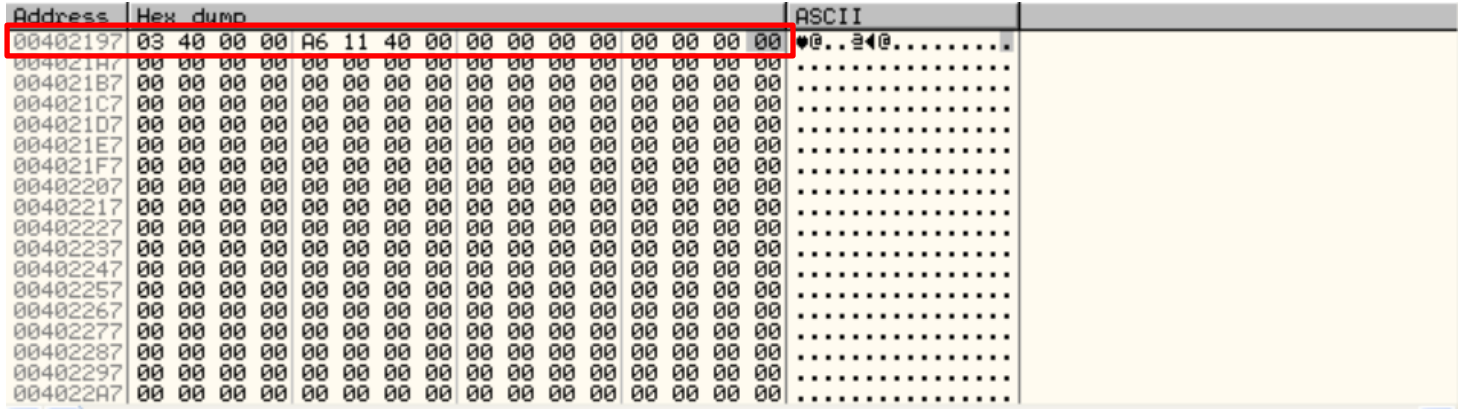


MOV DWORD PTR DS: [40219B], reverseM



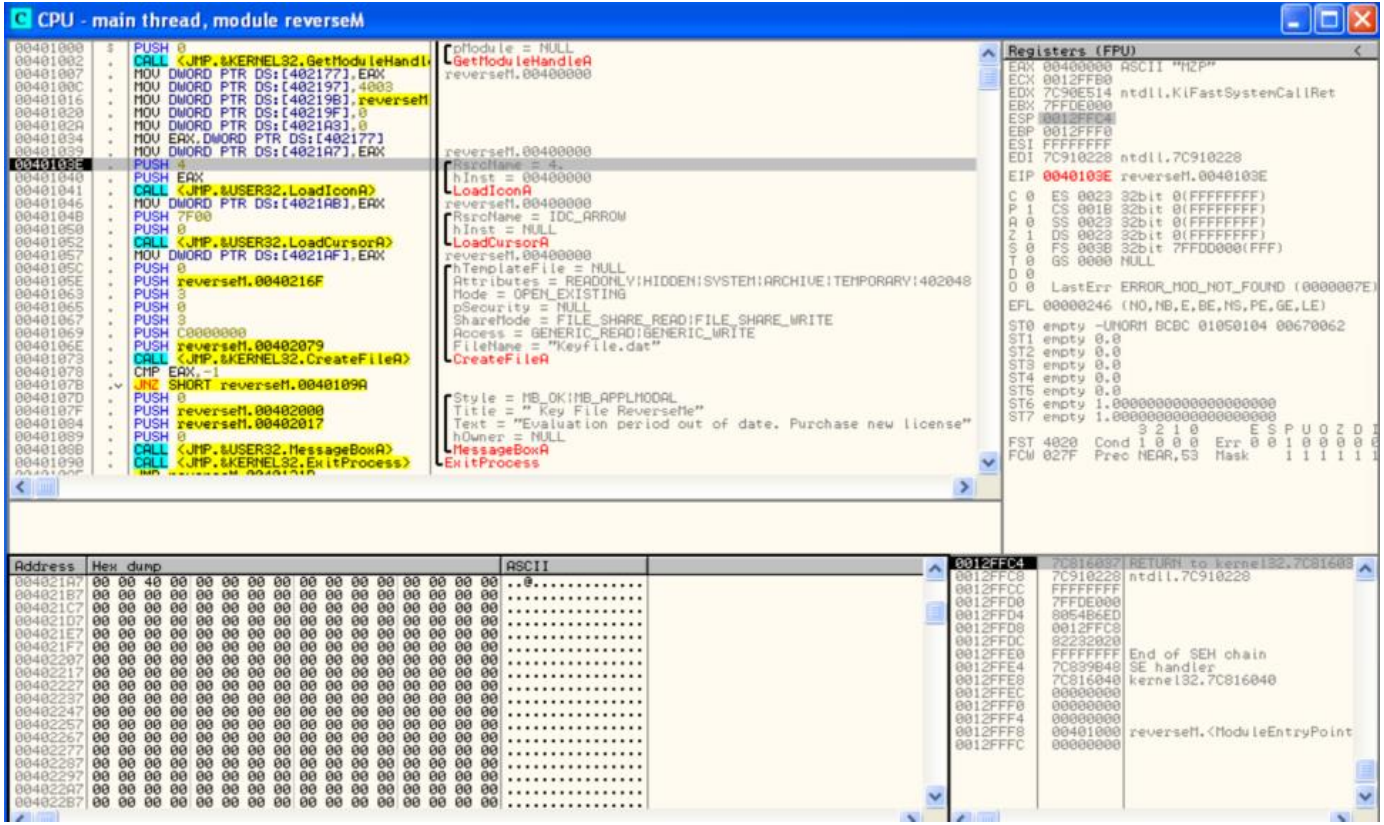
Dump

00402197



Address	Hex dump	ASCII
00402197	03 40 00 00 A6 11 40 00 00 00 00 00 00 00 00 00	♥@..¦◄@.....
004021A7	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
004021B7	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
004021C7	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
004021D7	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
004021E7	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
004021F7	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
00402207	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
00402217	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
00402227	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
00402237	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
00402247	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
00402257	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
00402267	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
00402277	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
00402287	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
00402297	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....
004022A7	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	.....

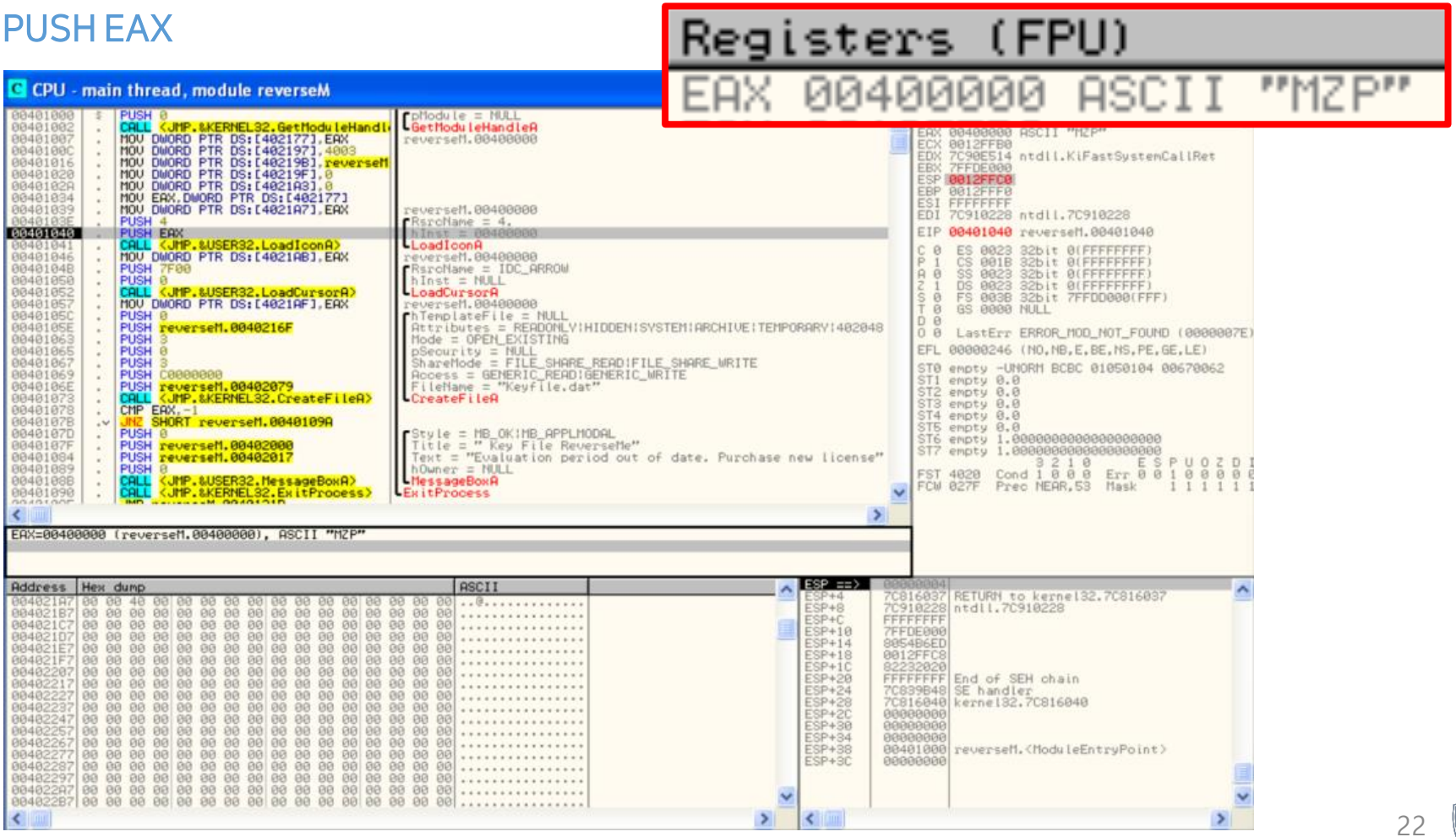
PUSH 4


CPU - main thread, module reverseM
Registers (FPU)
Address
Hex dump
ASCII

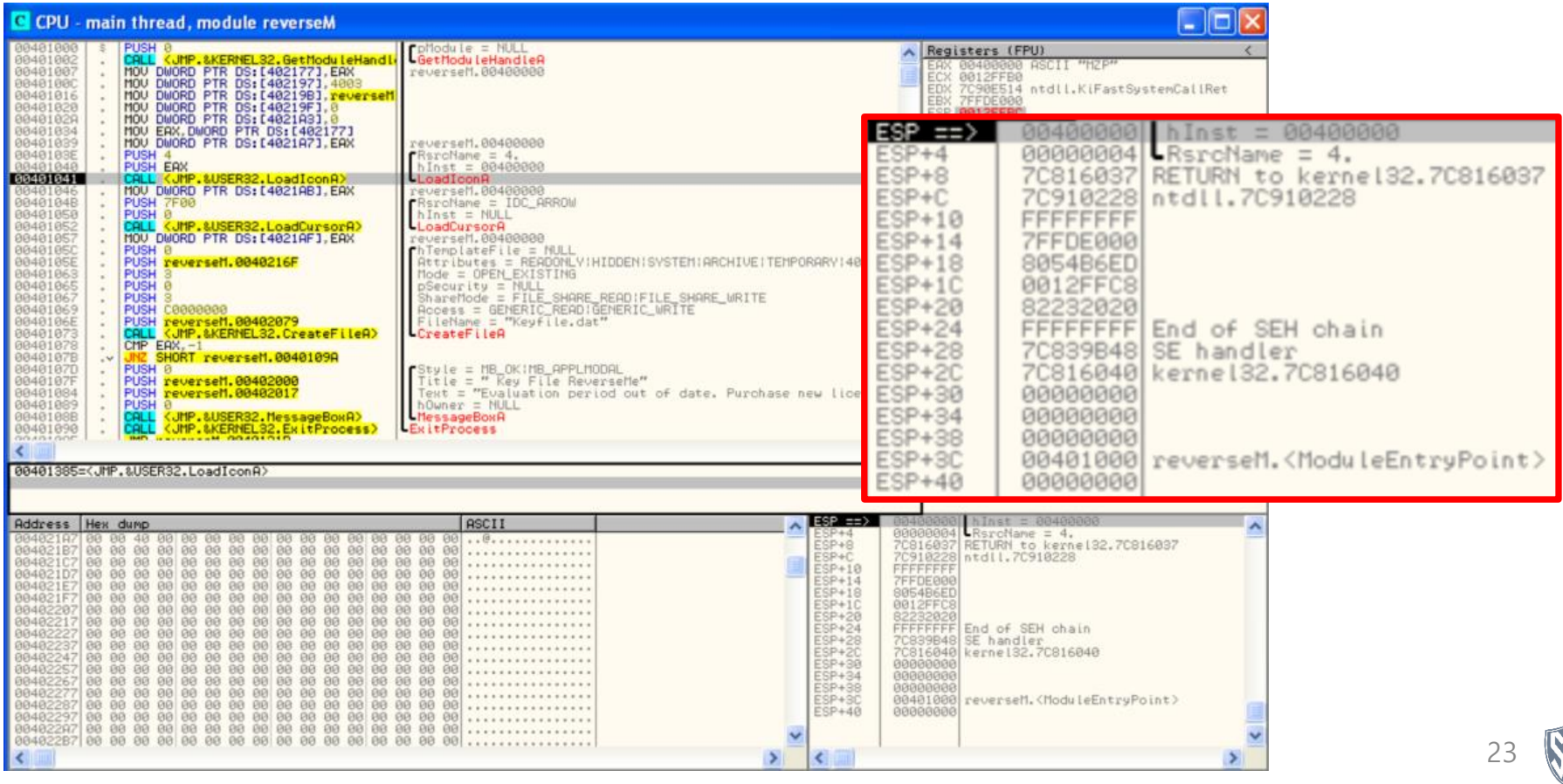
Stack


ESP ==> 00000004
ESP+4 7C816037 RETURN to kernel32.7C816037
ESP+8 7C910228 ntdll.7C910228
ESP+C FFFFFFFF
ESP+10 7FFDE000
ESP+14 8054B6ED
ESP+18 0012FFC8
ESP+1C 82232020
ESP+20 FFFFFFFF End of SEH chain
ESP+24 7C839B48 SE handler
ESP+28 7C816040 kernel32.7C816040
ESP+2C 00000000
ESP+30 00000000
ESP+34 00000000
ESP+38 00401000 reverseM.<ModuleEntryPoint>
ESP+3C 00000000

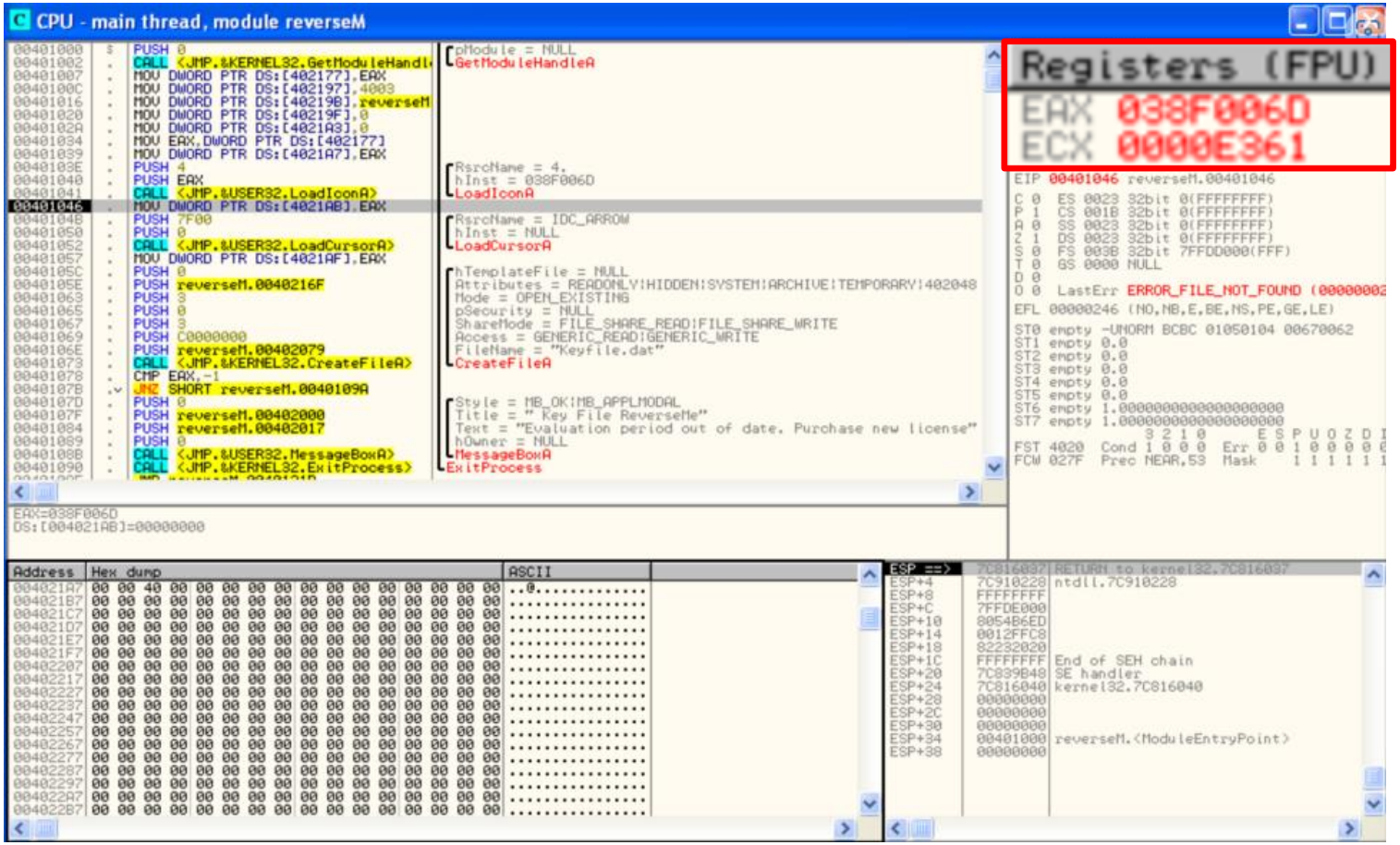
PUSH EAX


Registers (FPU)
EAX 00400000 ASCII "MZP"
CPU - main thread, module reverseM
00401040 PUSH EAX
EAX=00400000 (reverseM.00400000), ASCII "MZP"
Address Hex dump ASCII
ESP ==>

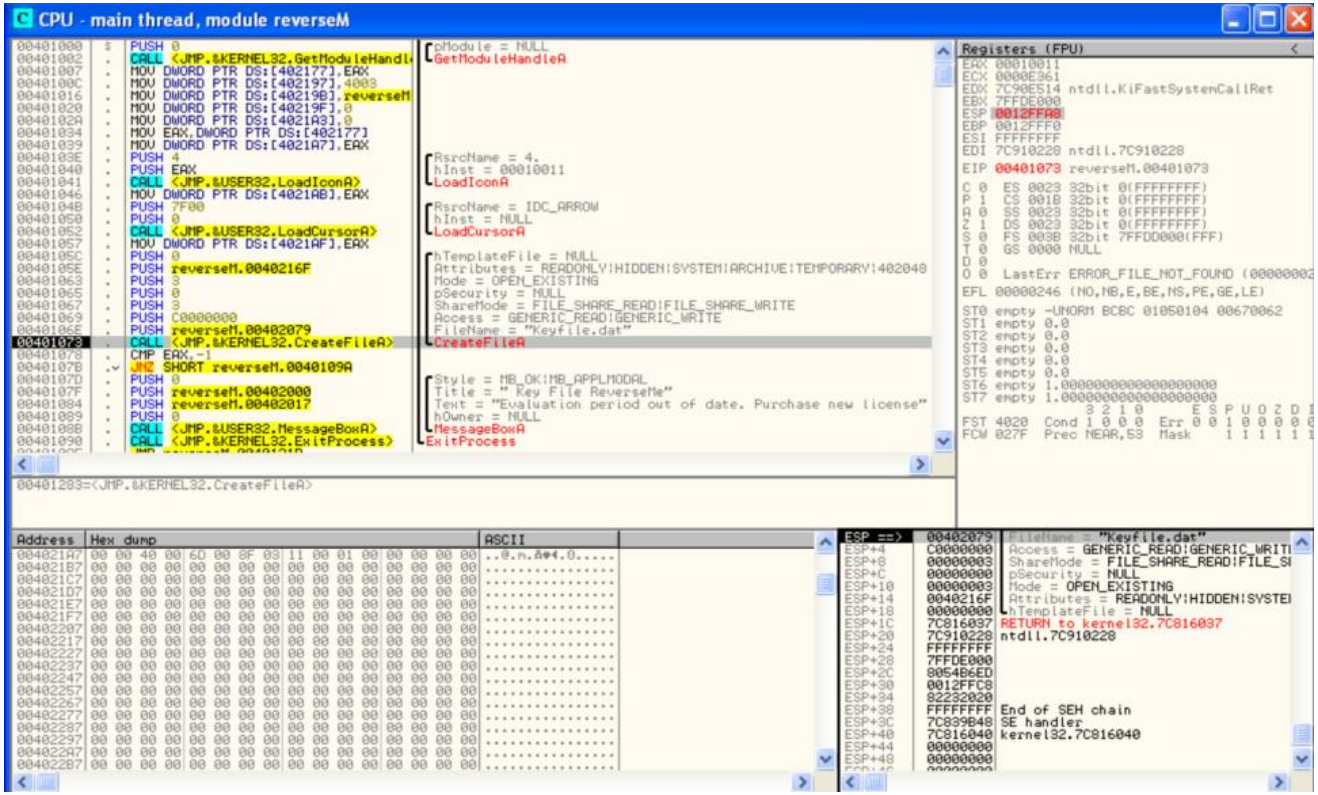
CALL <JMP.&USER32.LoadIconA>



MOV DWORD PTR DS: [4021AB], EAX



CALL <JMP.&KERNEL32.CreateFileA>



CALL <JMP.&KERNEL32.CreateFileA>

CreateFileA msdn

검색 : CreateFileA msdn

C++ Copy

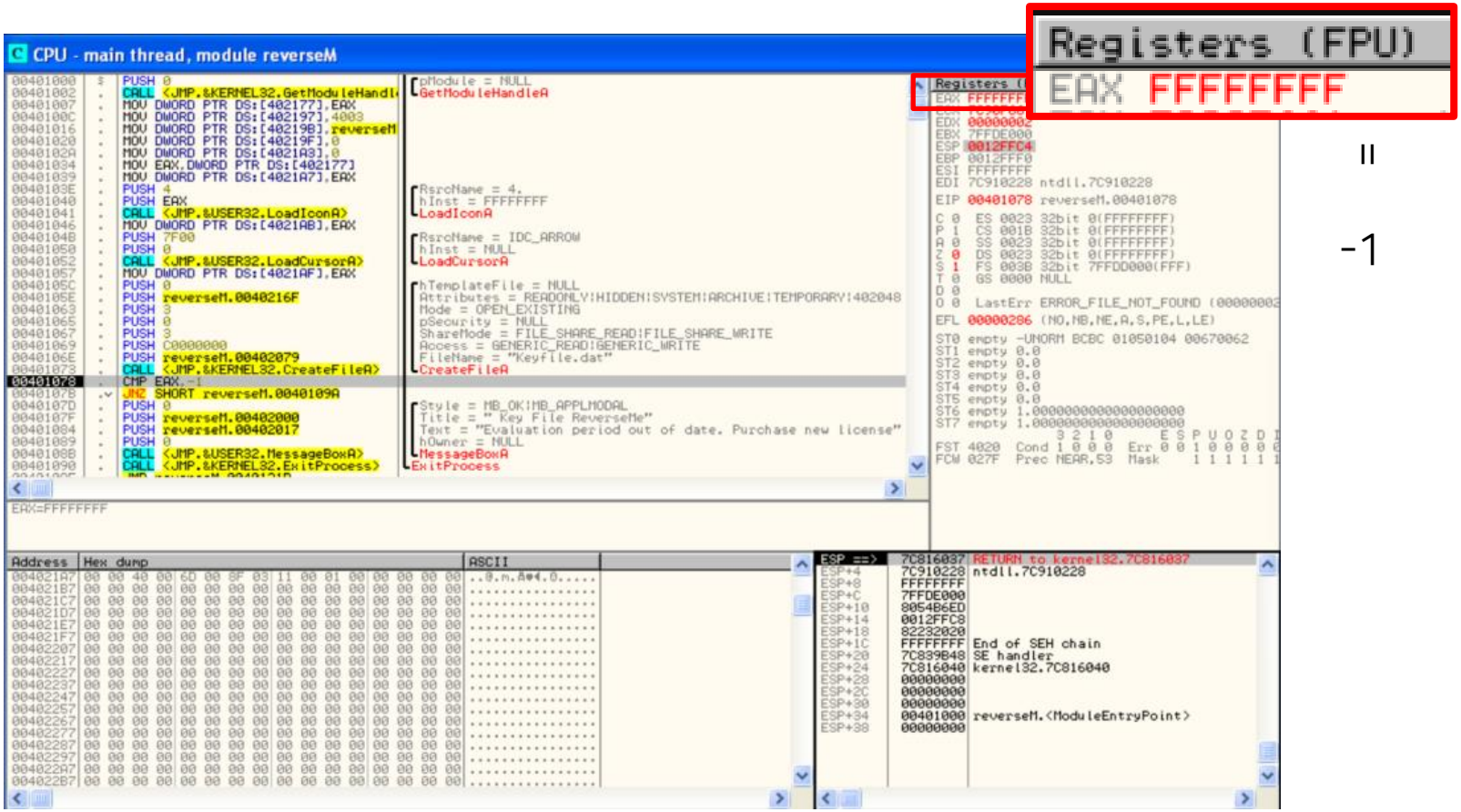
```
HANDLE CreateFileA(
  [in]           LPCSTR                lpFileName,
  [in]           DWORD                 dwDesiredAccess,
  [in]           DWORD                 dwShareMode,
  [in, optional] LPSECURITY_ATTRIBUTES lpSecurityAttributes,
  [in]           DWORD                 dwCreationDisposition,
  [in]           DWORD                 dwFlagsAndAttributes,
  [in, optional] HANDLE                hTemplateFile
);
```

CALL <JMP.&KERNEL32.CreateFileA>

```
0040105C   .   PUSH 0
0040105E   .   PUSH reverseM.0040216F
00401063   .   PUSH 3
00401065   .   PUSH 0
00401067   .   PUSH 3
00401069   .   PUSH C0000000
0040106E   .   PUSH reverseM.00402079
00401073   .   CALL <JMP.&KERNEL32.CreateFileA>
```

```
lpFileName,
dwDesiredAccess,
dwShareMode,
lpSecurityAttributes,
dwCreationDisposition,
dwFlagsAndAttributes,
hTemplateFile
```

CMP EAX, -1



CMP EAX, -1

CreateFileA msdn

Return value

If the function succeeds, the return value is an open handle to the specified file, device, named pipe, or mail slot.

If the function fails, the return value is **INVALID_HANDLE_VALUE**. To get extended error information, call GetLastError.

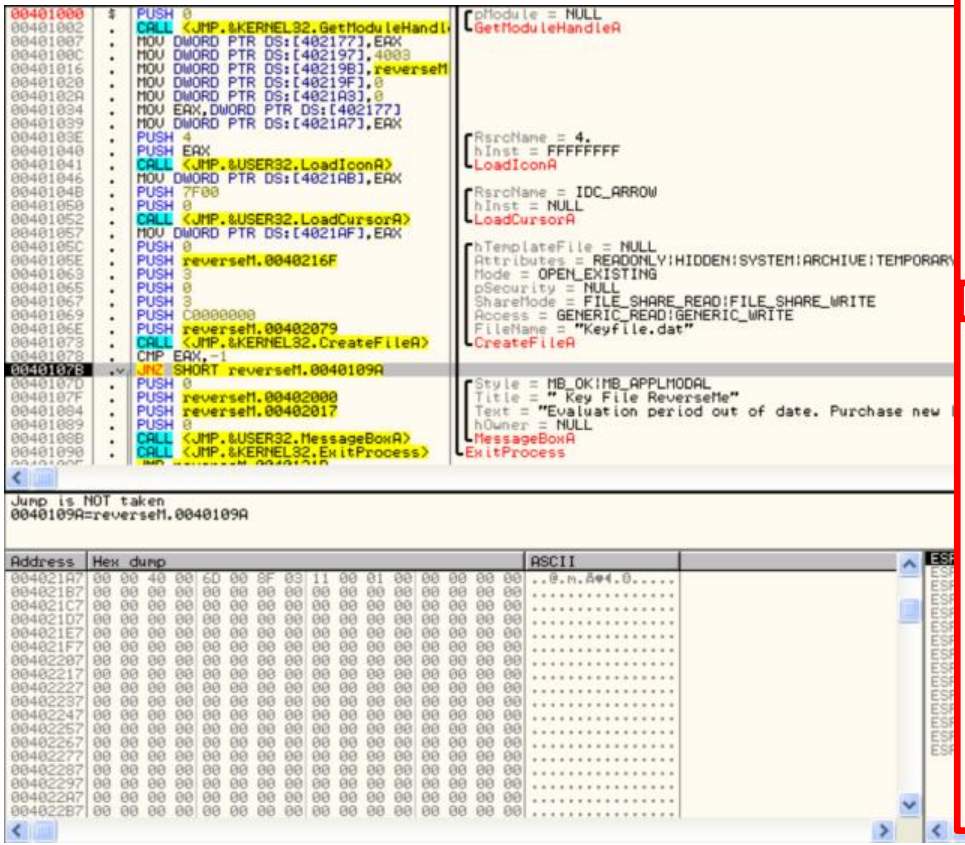
EAX = FFFFFFFF(-1)

CMP EAX, -1

```
PUSH 0
PUSH reverseM.0040216F
PUSH 3
PUSH 0
PUSH 3
PUSH C0000000
PUSH reverseM.00402079
CALL <JMP.&KERNEL32.CreateFileA>
```

```
hTemplateFile = NULL
Attributes = READONLY|HIDDEN|SYSTEM|ARCHIVE|TEMPORARY|402048
Mode = OPEN_EXISTING
pSecurity = NULL
ShareMode = FILE_SHARE_READ|FILE_SHARE_WRITE
Access = GENERIC_READ|GENERIC_WRITE
FileName = "Keyfile.dat"
CreateFileA
```

JNZ SHORT reverseM.0040109A

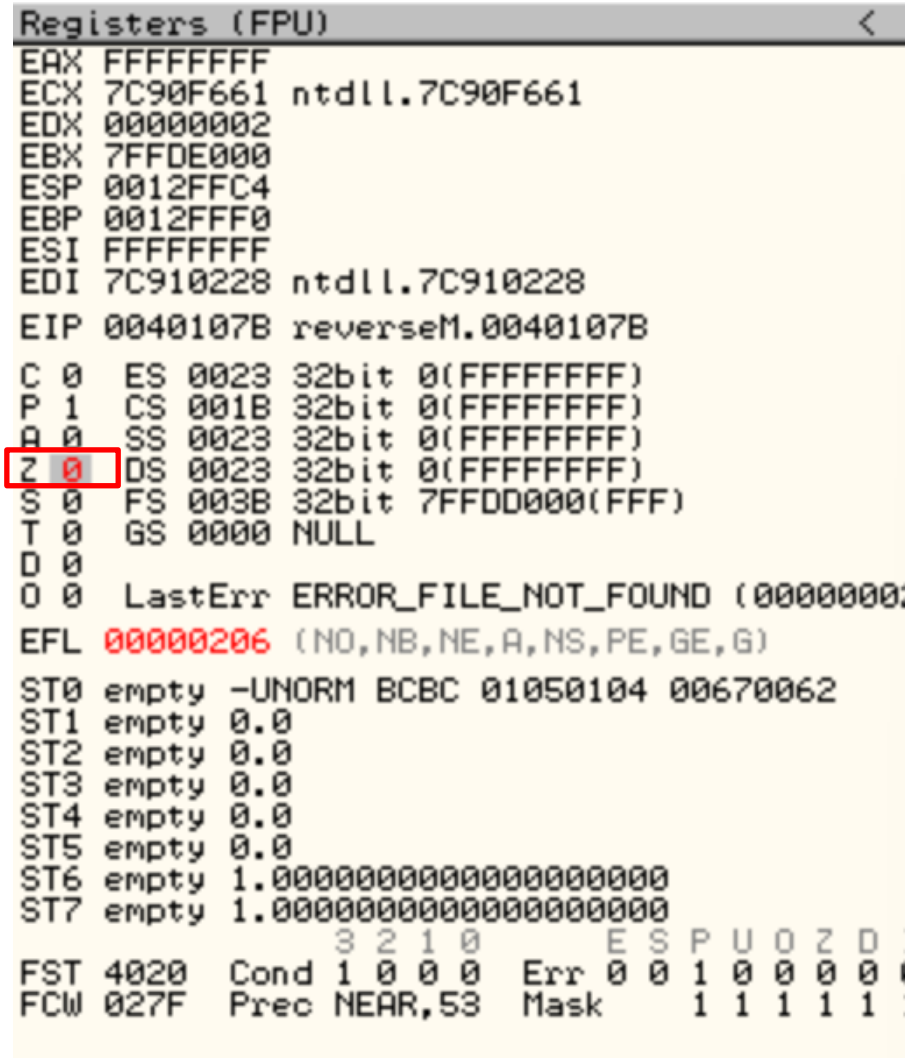


MessageBoxA

0040107D

```
0040107D  .  PUSH 0                                  Style = MB_OK|MB_APPLMODAL
0040107F  .  PUSH reverseM.00402000                  Title = " Key File ReverseMe"
00401084  .  PUSH reverseM.00402017                  Text = "Evaluation period out of date. Purchase new license"
00401089  .  PUSH 0                                  hOwner = NULL
0040108B  .  CALL <JMP.&USER32.MessageBoxA>          MessageBoxA
00401090  .  CALL <JMP.&KERNEL32.ExitProcess>        ExitProcess
00401095  .v JMP reverseM.0040121D
```

Registers


Registers (FPU)
EAX FFFFFFFF
ECX 7C90F661 ntdll.7C90F661
EDX 00000002
EBX 7FFDE000
ESP 0012FFC4
EBP 0012FFF0
ESI FFFFFFFF
EDI 7C910228 ntdll.7C910228
EIP 0040107B reverseM.0040107B
C 0 ES 0023 32bit 0(FFFFFFFF)
P 1 CS 001B 32bit 0(FFFFFFFF)
A 0 SS 0023 32bit 0(FFFFFFFF)
Z 0 DS 0023 32bit 0(FFFFFFFF)
S 0 FS 003B 32bit 7FFDD000(FFF)
T 0 GS 0000 NULL
D 0
O 0 LastErr ERROR_FILE_NOT_FOUND (0000000
EFL 00000206 (NO,NB,NE,A,NS,PE,GE,G)
ST0 empty -UNORM BCBC 01050104 00670062
ST1 empty 0.0
ST2 empty 0.0
ST3 empty 0.0
ST4 empty 0.0
ST5 empty 0.0
ST6 empty 1.0000000000000000000
ST7 empty 1.0000000000000000000
3 2 1 0 E S P U O Z D
FST 4020 Cond 1 0 0 0 Err 0 0 1 0 0 0 0
FCW 027F Prec NEAR,53 Mask 1 1 1 1 1

CALL <JMP.&KERNEL32.ReadFile>

```
PUSH 0                                  pOverlapped = NULL
PUSH reverseM.00402173                  pBytesRead = reverseM.00402173
PUSH 46                                 BytesToRead = 46 (70.)
PUSH reverseM.0040211A                  Buffer = reverseM.0040211A
PUSH EAX                                hFile = FFFFFFFF
CALL <JMP.&KERNEL32.ReadFile>           ReadFile
```

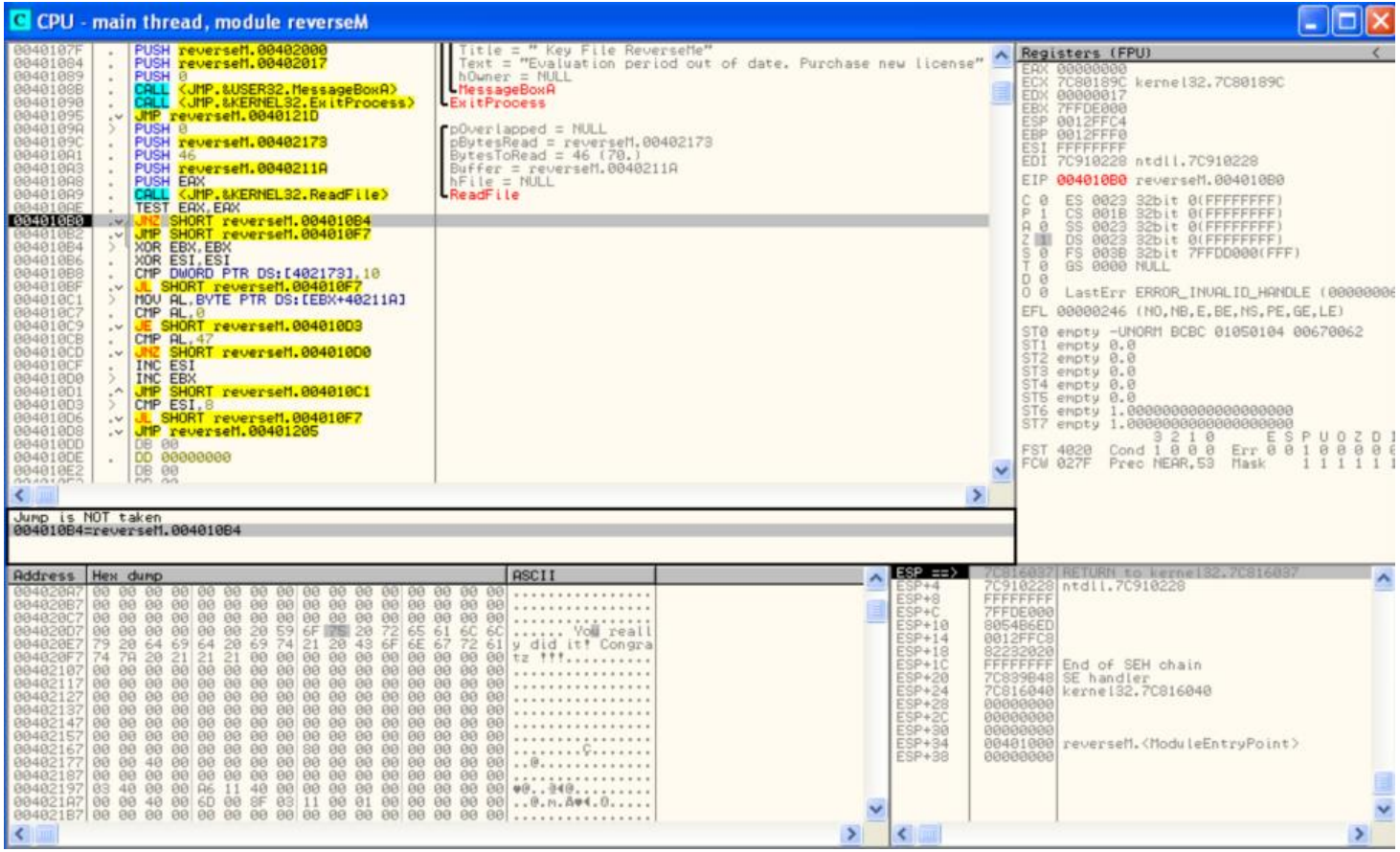
CALL <JMP.&KERNEL32.ReadFile>

ReadFile msdn

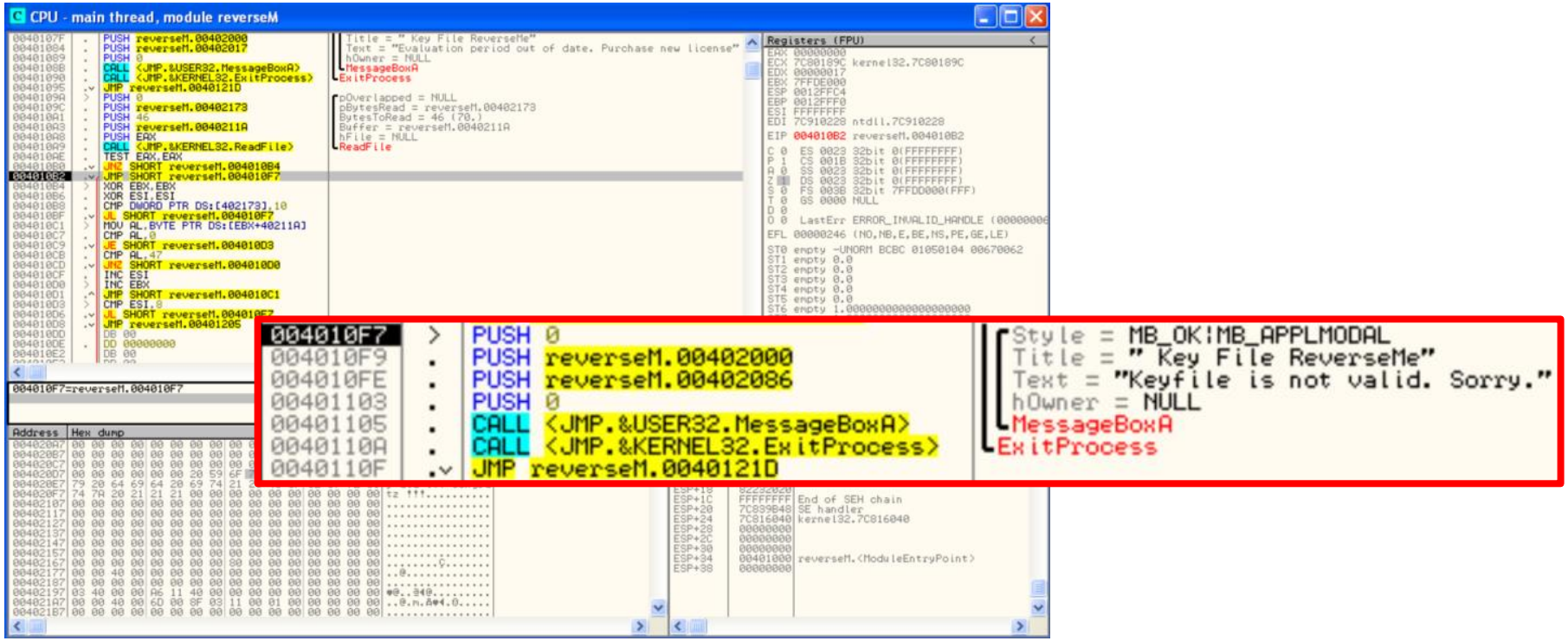
C++ Copy

```
BOOL ReadFile(
  [in]                HANDLE       hFile,
  [out]               LPVOID       lpBuffer,
  [in]                DWORD        nNumberOfBytesToRead,
  [out, optional]     LPDWORD      lpNumberOfBytesRead,
  [in, out, optional] LPOVERLAPPED lpOverlapped
);
```

JNZ SHORT reverseM.004010B4

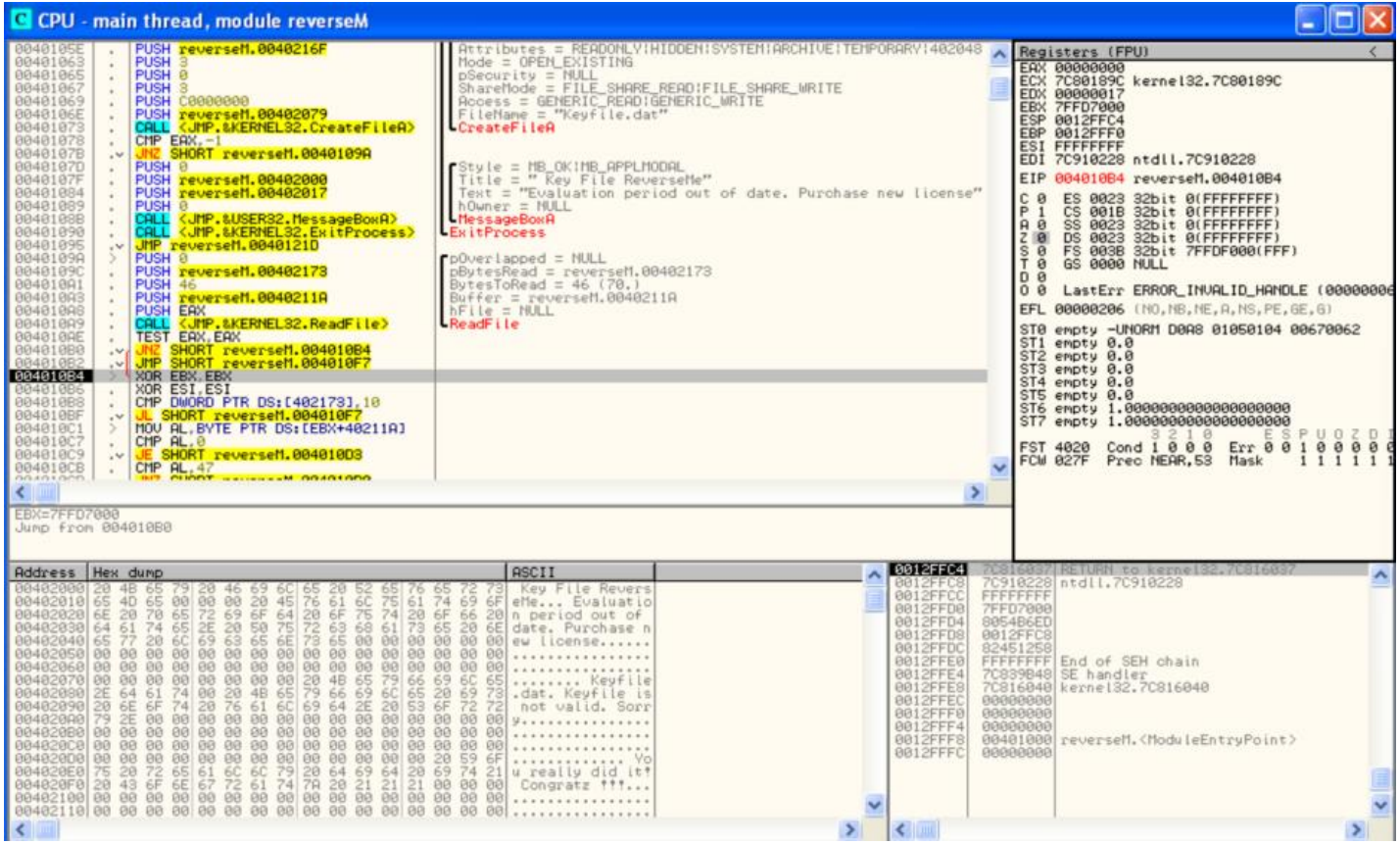


JMP SHORT reverseM.004010F7



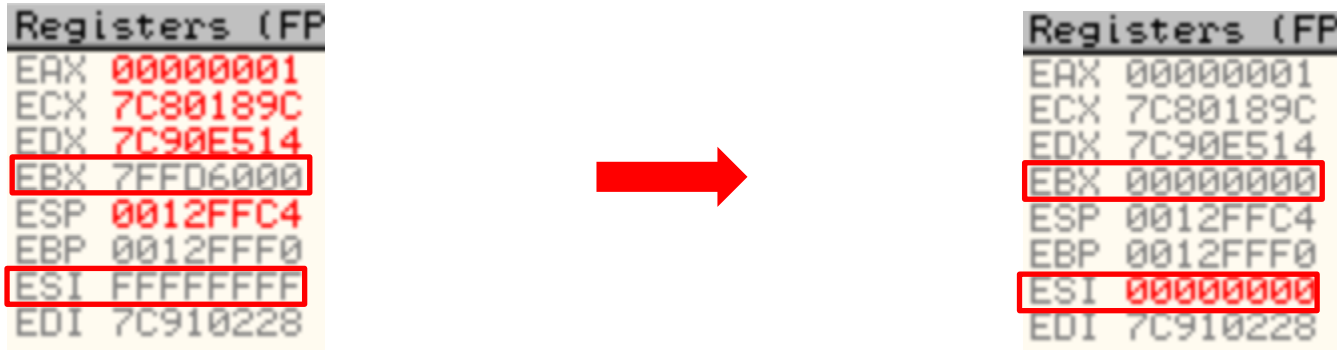
XOR EBX, EBX

XOR ESI, ESI



XOR EBX, EBX

XOR ESI, ESI



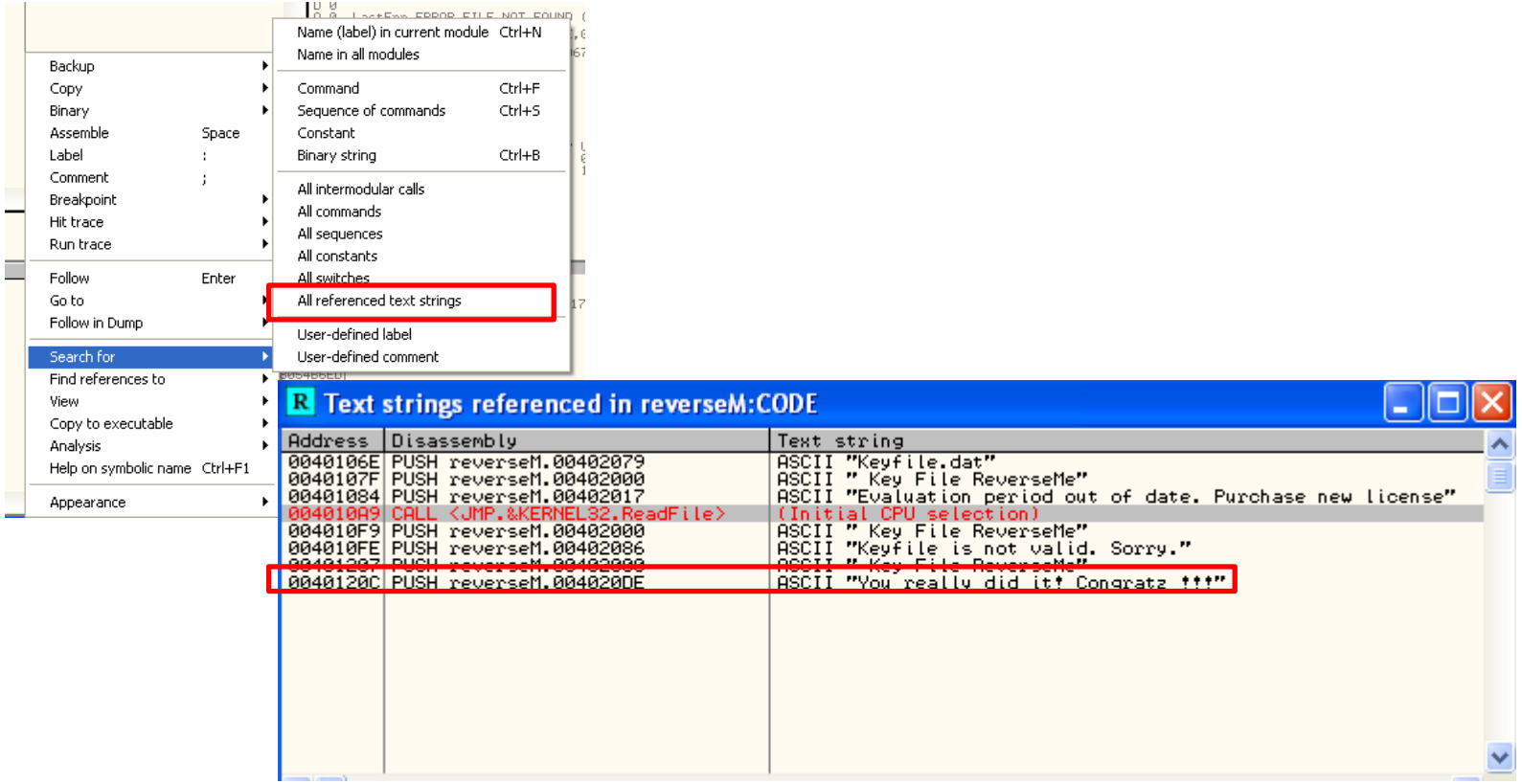
MessageBoxA

```
004010F7  >  PUSH 0                                    Style = MB_OK|MB_APPLMODAL
004010F9  .  PUSH reverseM.00402000                    Title = " Key File ReverseMe"
004010FE  .  PUSH reverseM.00402086                    Text = "Keyfile is not valid. Sorry."
00401103  .  PUSH 0                                    hOwner = NULL
00401105  .  CALL <JMP.&USER32.MessageBoxA>            MessageBoxA
0040110A  .  CALL <JMP.&KERNEL32.ExitProcess>          ExitProcess
0040110F  .v JMP reverseM.0040121D
```

```
004010B8  .  CMP DWORD PTR DS:[402173],10
004010BF  .v JL SHORT reverseM.004010F7
004010C1  >  MOV AL,BYTE PTR DS:[EBX+40211A]
004010C7  .  CMP AL,0
004010C9  .v JE SHORT reverseM.004010D3
004010CB  .  CMP AL,47
004010CD  .v JNZ SHORT reverseM.004010D0
004010CF  .  INC ESI
004010D0  >  INC EBX
004010D1  .^ JMP SHORT reverseM.004010C1
004010D3  >  CMP ESI,8
004010D6  .v JL SHORT reverseM.004010F7
004010D8  .v JMP reverseM.00401205
```

해결!!

Search for – All referenced text strings

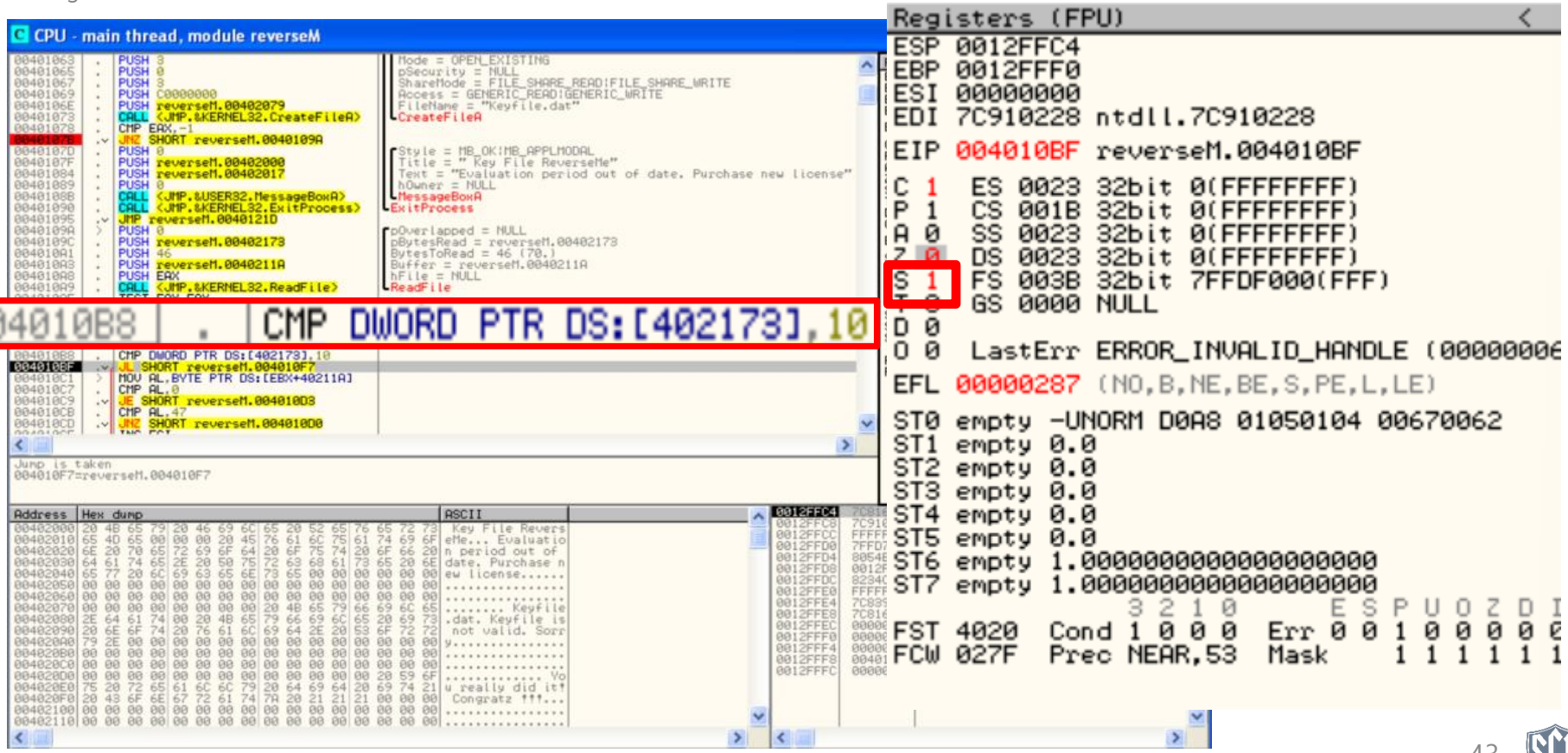


PUSH reversM.004020DE

```
00401205  >  PUSH 0                                  Style = MB_OK|MB_APPLMODAL
00401207  .  PUSH reverseM.00402000                  Title = " Key File ReverseMe"
0040120C  .  PUSH reverseM.004020DE                  Text = "You really did it! Congratz !!!"
00401211  .  PUSH 0                                  hOwner = NULL
00401213  .  CALL <JMP.&USER32.MessageBoxA>          MessageBoxA
00401218  .  CALL <JMP.&KERNEL32.ExitProcess>        ExitProcess
0040121D  >  RETN
```

JL SHORT reverseM.004010F7

registers



MessageBoxA

```
00401205  >  PUSH 0                                  Style = MB_OK|MB_APPLMODAL
00401207  .  PUSH reverseM.00402000                  Title = " Key File ReverseMe"
0040120C  .  PUSH reverseM.004020DE                  Text = "You really did it! Congratz !!!"
00401211  .  PUSH 0                                  hOwner = NULL
00401213  .  CALL <JMP.&USER32.MessageBoxA>          MessageBoxA
00401218  .  CALL <JMP.&KERNEL32.ExitProcess>        ExitProcess
0040121D  >  RETN
0040121E     DB 00
0040121F     DB 00
00401220     DB 00
00401221     DB 00
00401222     DB 00
00401223     DB 00
00401224     DB 00
00401225     DB 00
00401226     DB 00
00401227     DB 00
00401228     DB 00
00401229     DB 00
0040122A     DB 00
0040122B     DB 00
0040122C     DB 00
0040122D     DB 00
```

Q&A