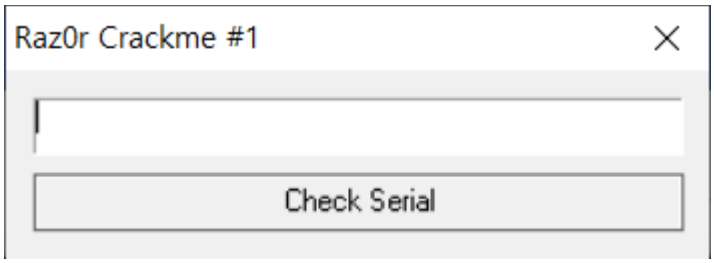
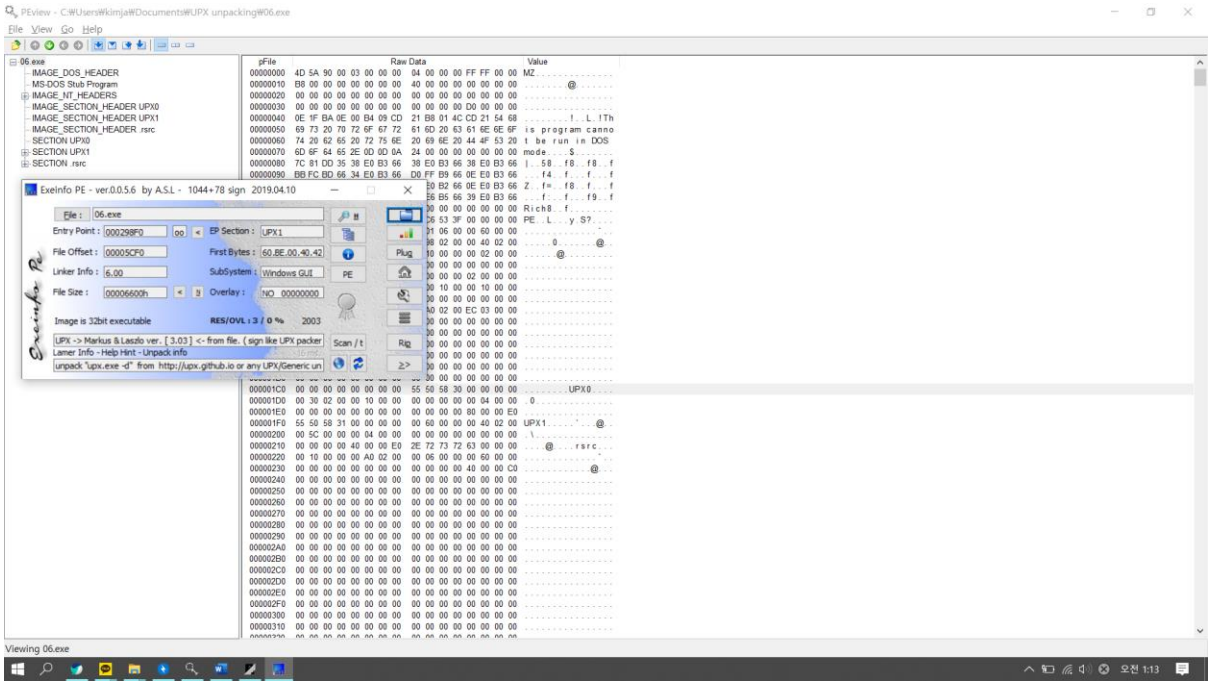
05.exe - Unpack을 한 후 Serial을 찾으시오. (정답인증은 OEP + Serial)



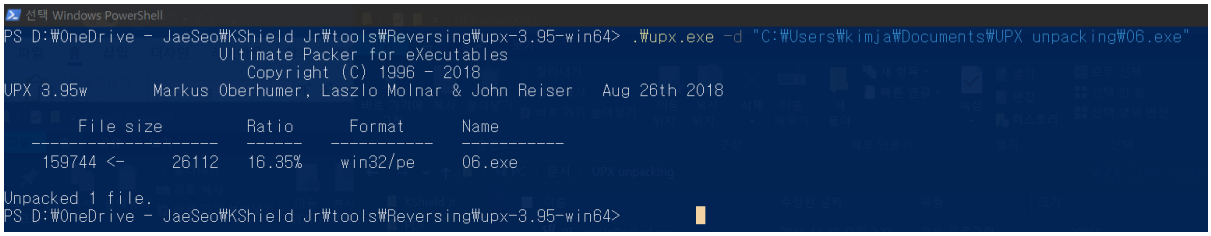
프로그램을 실행을 하면 이런 식으로 Input Box가 있고 입력을 하면 Serial를 체크를 하는 프로그램이다.

일단 디버깅을 하기 전에 PE 분석을 해본다.

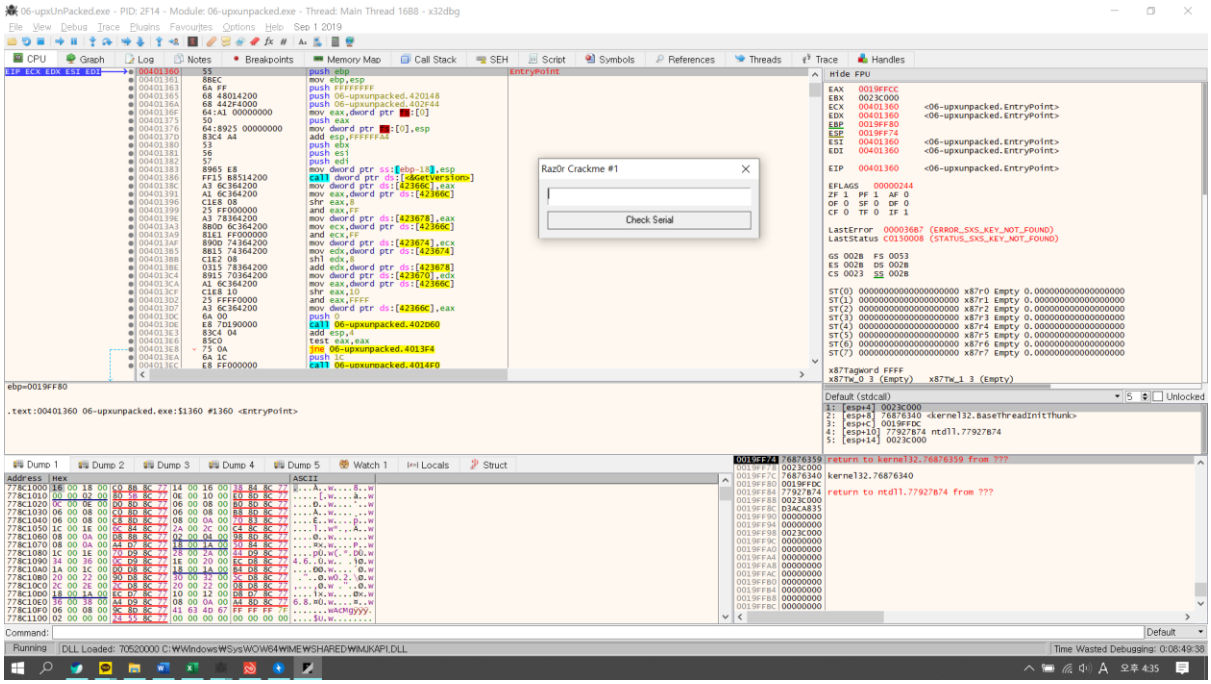


PE 분석을 통해 UPX를 통해 Packing이 되어있다는 것을 확인했다.

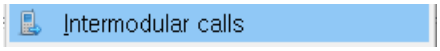
UPS unpacking을 해본다.



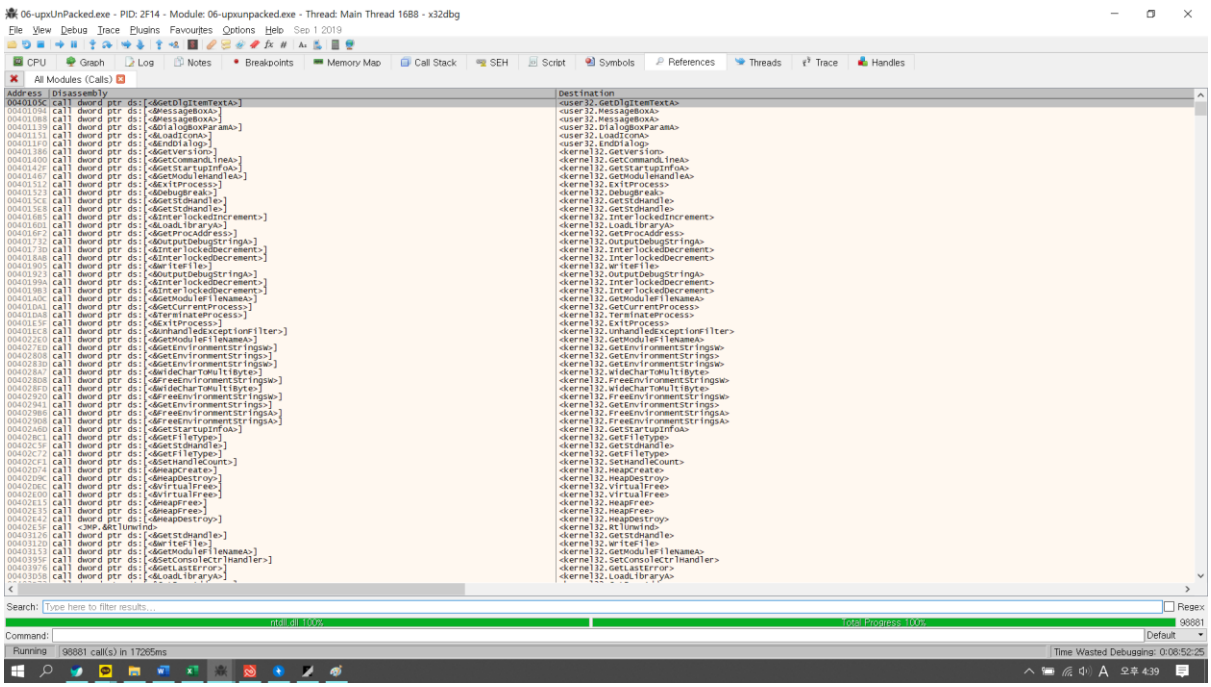
이제 unpacking된 파일을 가지고 x32dbg로 분석을 해본다.



일단 프로그램을 동작 시켜 GUI가 구성이 시작되는 main 함수부분을 찾아본다.



인터럽트가 발생하는 것을 검색해본다.

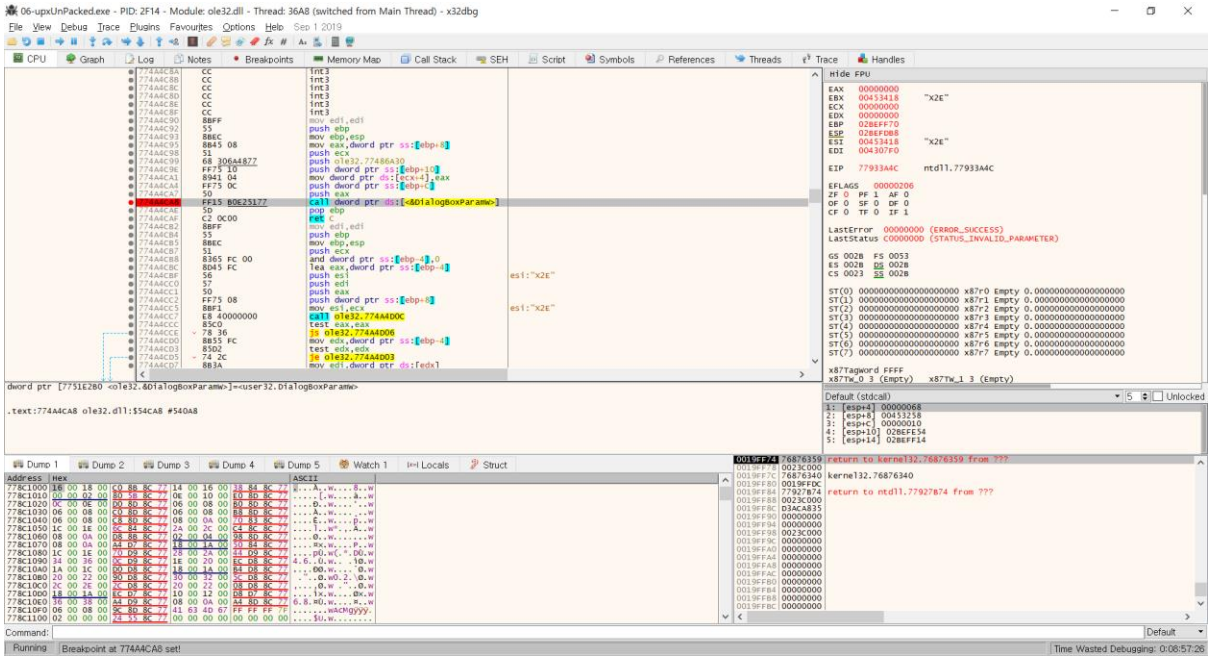


이때 main에서 동작될 것으로 예상되는 GUI를 만드는 함수에 관련된 키워드를 검색해본다.

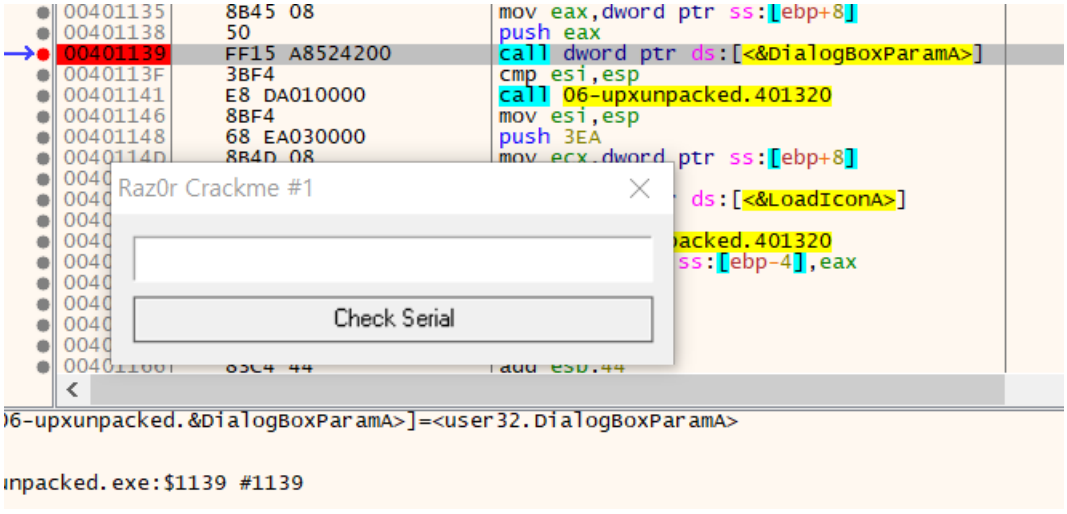
키워드(gui, box, dialog 등)

Address	Disassembly	Destination
00401094	call dword ptr ds:[<&MessageBoxA>]	<user32.MessageBoxA>
004010B8	call dword ptr ds:[<&MessageBoxA>]	<user32.MessageBoxA>
00401139	call dword ptr ds:[<&DialogBoxParamA>]	<user32.DialogBoxParamA>
6406EE9D	call dword ptr ds:[<&GetRgnBox>]	<gdi32.GetRgnBox>
6406EEB3	call dword ptr ds:[<&GetWindowRgnBox>]	<user32.GetWindowRgnBox>
6406EEED	call dword ptr ds:[<&GetWindowRgnBox>]	<user32.GetWindowRgnBox>
642D1D53	call dword ptr ds:[<&MessageBoxW>]	<user32.MessageBoxW>
642DFEA9	call dword ptr ds:[<&MessageBoxW>]	<user32.MessageBoxW>
717B7940	call dword ptr ds:[<&GetWindowRgnBox>]	<user32.GetWindowRgnBox>
717D3DB1	call dword ptr ds:[<&GetClipBox>]	<gdi32.GetClipBox>
717D3F64	call dword ptr ds:[<&GetClipBox>]	<gdi32.GetClipBox>
717DA2E4	call dword ptr ds:[<&GetRgnBox>]	<gdi32.GetRgnBox>
75B553C1	call dword ptr ds:[<&NtGdiGetRgnBox>]	<win32u.NtGdiGetRgnBox>
75B559DC	call dword ptr ds:[<&GetClipBoxImpl>]	<gdi32full.GetClipBoxImpl>
76503B71	call dword ptr ds:[<&GetRgnBox>]	<gdi32.GetRgnBox>
76524C87	call dword ptr ds:[<&NtUserGetComboBoxInfo>]	<win32u.NtUserGetComboBoxInfo>
76535A18	call dword ptr ds:[<&NtUserGetListBoxInfo>]	<win32u.NtUserGetListBoxInfo>
7654671E	call dword ptr ds:[<&GetClipBox>]	<gdi32.GetClipBox>
76743D15	call dword ptr ds:[<&ZwCreateLowBoxToken>]	<ntdll.ZwCreateLowBoxToken>
7724F1DE	call dword ptr ds:[<&MessageBoxW>]	<user32.MessageBoxW>
7725286F	call dword ptr ds:[<&MessageBoxW>]	<user32.MessageBoxW>
77255E82	call dword ptr ds:[<&DialogBoxParamA>]	<user32.DialogBoxParamA>
77255F67	call dword ptr ds:[<&DialogBoxParamW>]	<user32.DialogBoxParamW>
772560EA	call dword ptr ds:[<&MessageBoxW>]	<user32.MessageBoxW>
7731B63A	call dword ptr ds:[<&GetClipBox>]	<gdi32.GetClipBox>
7731C6F6	call dword ptr ds:[<&NtGdiSetUMPDSandboxState>]	<win32u.NtGdiSetUMPDSandboxState>
77323D39	call dword ptr ds:[<&GetRgnBox>]	<gdi32.GetRgnBox>
77363A77	call dword ptr ds:[<&GetClipBox>]	<gdi32.GetClipBox>
77499077	call dword ptr ds:[<&MessageBoxW>]	<user32.MessageBoxW>
774A4CA8	call dword ptr ds:[<&DialogBoxParamW>]	<user32.DialogBoxParamW>

이때 이러한 결과값이 나오게 되는데 처음 실행된 것으로 의심이 되는 DialogBoxParamw를 들어가 확인해본다.

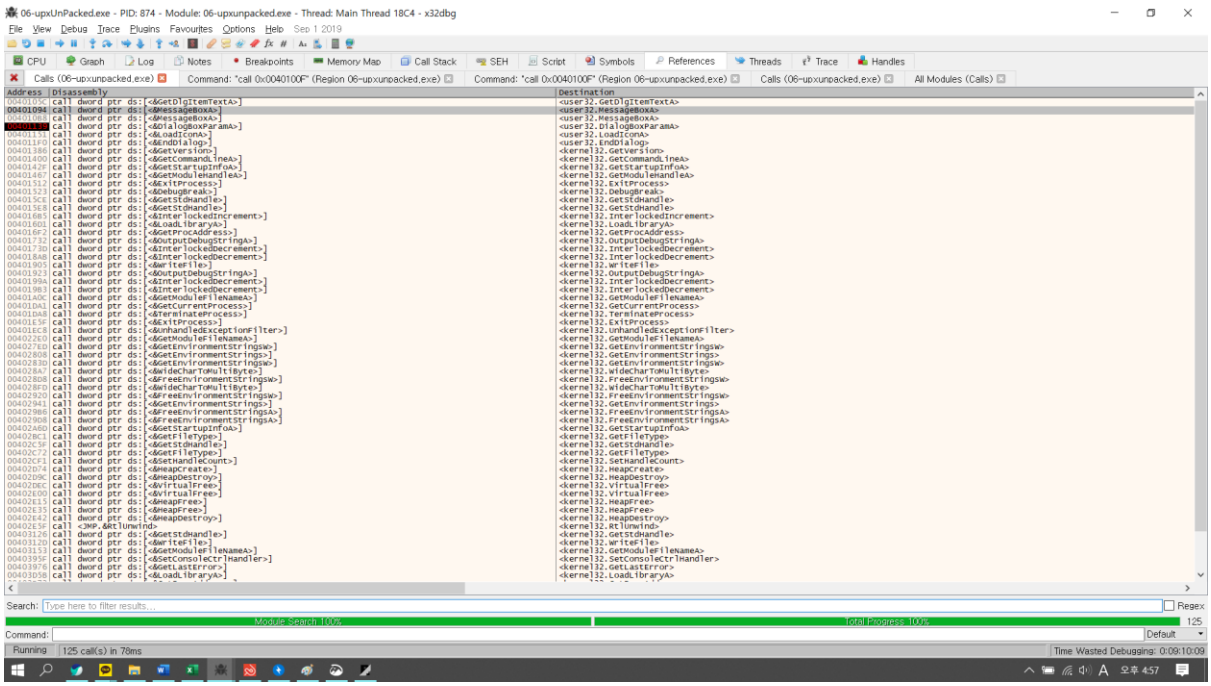


그리고 Break Point를 걸어 실제로 맞는지 확인을 해본다.



Break Point를 지나자 Gui창이 뜨는 것을 통해 main 함수 영역을 찾을 것을 볼 수 있다.

이제 다시 문제로 돌아와 Serial key 값을 찾아야 하는데 Check 버튼을 눌렀을 때 발생하는 인터럽트를 검색하여 찾아본다.



그 결과 위와 같이 다양한 호출을 볼 수 있는데 이때 안내 메시지 창으로 의심이 되는 MessageBox 내부로 들어가 본다.

메시지와 관련된 문자열을 발견 할 수 있다 이때 jne를 통해 정확한 serial key 인지 아닌지 판별된 ZF 플래그로 메시지를 출력하는 것으로 예상이 된다.

```
00401064   E8 B7020000   call 06-upxunpacked.401320
00401069   68 D4354200   push 06-upxunpacked.4235D4          4235D4:"check"
0040106E   68 302A4200   push 06-upxunpacked.422A30          422A30:"AD46DFS547"
00401073   E8 18020000   call 06-upxunpacked.401290
00401078   83C4 08       add esp,8
0040107B   85C0          test eax,eax
0040107D   75 24         jne 06-upxunpacked.4010A3
0040107F   8BF4          mov esi,esp
00401081   6A 40         push 40
00401083   68 48004200   push 06-upxunpacked.420048          420048:"Good Job!"
00401088   68 38004200   push 06-upxunpacked.420038          420038:"You got it ;)"
0040108D   8B0D 38364200 mov ecx,dword ptr ds:[423638]
00401093   51            push ecx
00401094   FF15 B4524200 call dword ptr ds:[<&MessageBoxA>]
0040109A   3BF4          cmp esi,esp
0040109C   E8 7F020000   call 06-upxunpacked.401320
004010A1   EB 22         jmp 06-upxunpacked.4010C5
004010A3   8BF4          mov esi,esp
004010A5   6A 10         push 10
004010A7   68 30004200   push 06-upxunpacked.420030          420030:"ERROR"
004010AC   68 1C004200   push 06-upxunpacked.42001C          42001C:"Wrong serial!!!"
004010B1   8B15 38364200 mov edx,dword ptr ds:[423638]
004010B7   52            push edx
004010B8   FF15 B4524200 call dword ptr ds:[<&MessageBoxA>]
```

Jne 하는 부분에 Break Point를 걸고 버튼을 눌러 확인을 해본다.

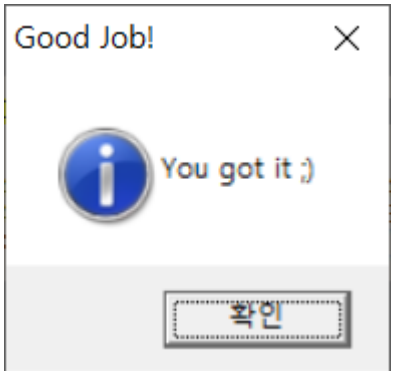
ZF 플래그가 0으로 에러 메시지를 출력하는 곳으로 넘어가게 되는 것을 볼 수 있다.

한번 ZF를 1으로 수정하여 작동시켜 본다.

```
004010B1   85C0          test eax,eax
0040107D   75 24         jne 06-upxunpacked.4010A3
0040107F   8BF4          mov esi,esp
00401081   6A 40         push 40
00401083   68 48004200   push 06-upxunpacked.420048          420048:"Good Job!"
00401088   68 38004200   push 06-upxunpacked.420038          420038:"You got it ;)"
0040108D   8B0D 38364200 mov ecx,dword ptr ds:[423638]       ecx:"asdas"
00401093   51            push ecx                           ecx:"asdas"

EIP   0040107D

EFLAGS   000003C4
ZF 1  PF 1  AF 0
OF 0  SF 1  DF 0
CF 0  TF 1  IF 1
```

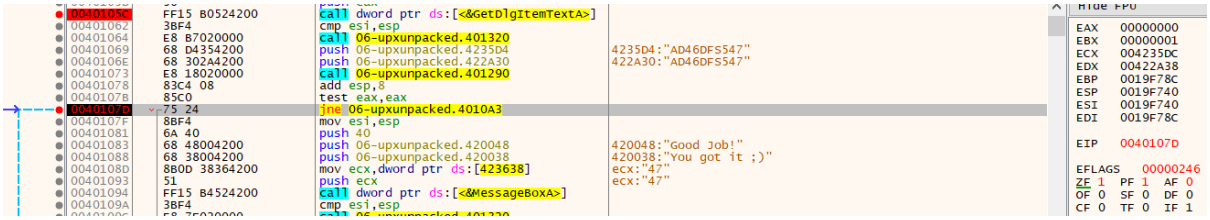


그 결과 성공했다는 메시지가 나오게 된다.

이때 위에서 작동하는 곳을 분석해서 시리얼 키와 어떤 식으로 비교를 하는지 분석해본다.

```
0040104C   68 D4354200   push 06-upxunpacked.4235D4          4235D4:"asdas"
00401051   68 E8030000   push 3E8
00401056   A1 38364200   mov eax,dword ptr ds:[423638]
0040105B   50            push eax
0040105C   FF15 B0524200 call dword ptr ds:[<&GetDlgItemTextA>]
00401062   3BF4          cmp esi,esp
00401064   E8 B7020000   call 06-upxunpacked.401320
00401069   68 D4354200   push 06-upxunpacked.4235D4          4235D4:"asdas"
0040106E   68 302A4200   push 06-upxunpacked.422A30          422A30:"AD46DFS547"
00401073   E8 18020000   call 06-upxunpacked.401290
00401078   83C4 08       add esp,8
0040107B   85C0          test eax,eax
0040107D   75 24         jne 06-upxunpacked.4010A3
```

일단 맨처음에 입력한 값을 GetDigitemTextA를 통해 입력 값을 가져와 push를 하고 serial key 값으로 의심이 되는 문자를 push 한 후 비교하는 기능을 하는 것으로 의심이 되는 함수를 호출 한 다음 리턴값을 eax에 가져와 and 연산을 하는 것을 볼 수 있다.이때 시리얼 키로 의심이 되는 키값을 넣고 다시 한번 해본다.



예상했던 것처럼 ZF가 1이고 EAX는 0으로 성공 메시지를 출력하는 부분으로 진행되는 것을 확인할 수 있다.

이제 Unpacking된 파일의 엔트리 포인트 값 OEP 코드 00401360+시리얼키를 추가 해서 정답을 구한다.

정답: 00401360AD46DFS547