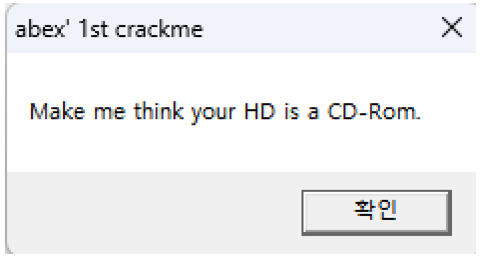
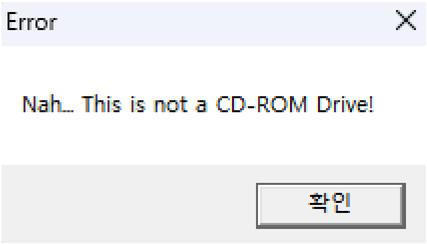
abex' crackme #1



abex' 1st crackme

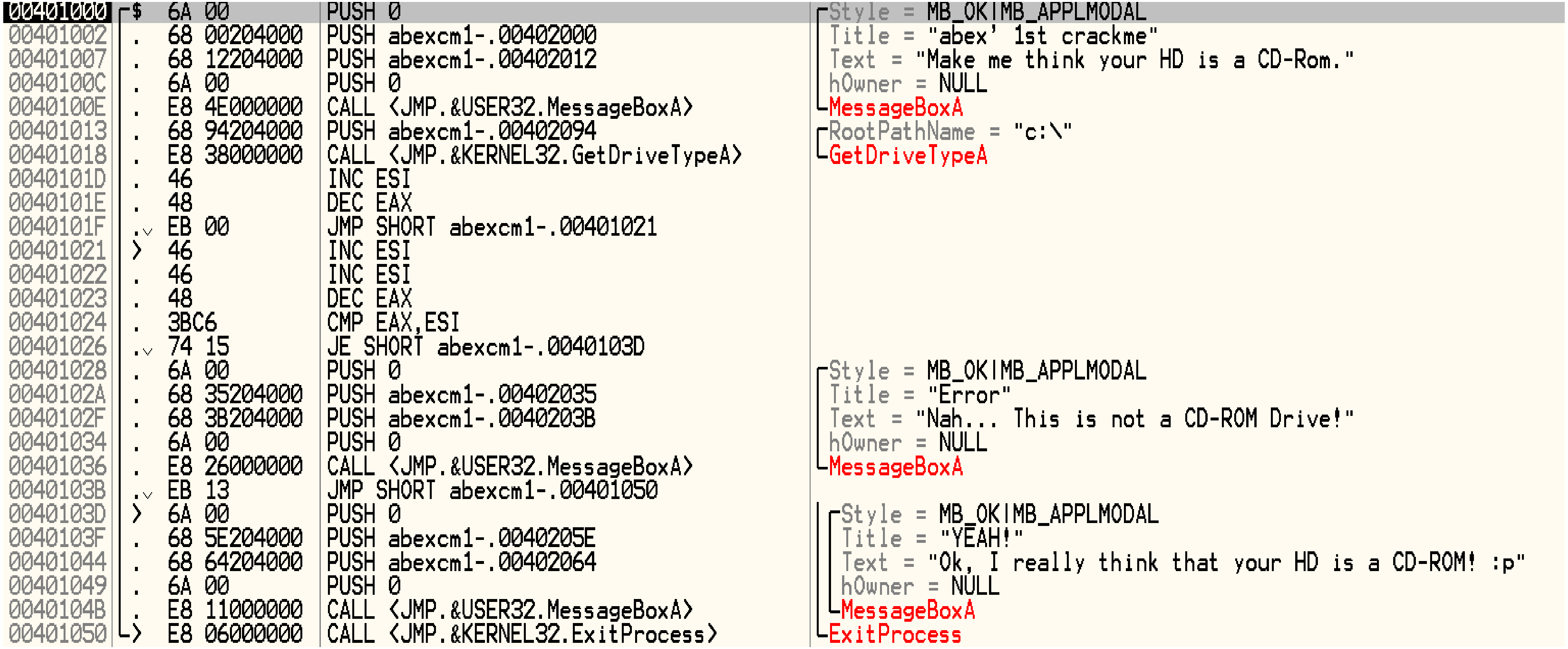
Make me think your HD is a CD-Rom.



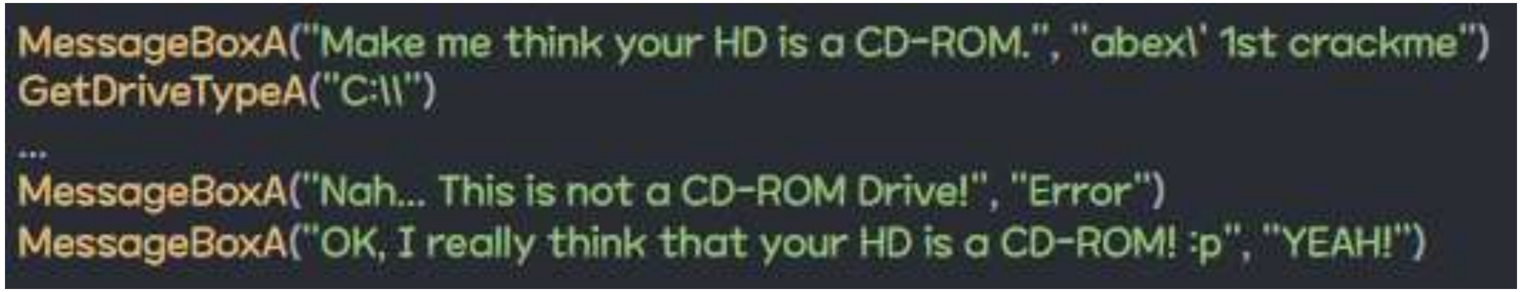
Error

Nah... This is not a CD-ROM Drive!

crackme 프로그램은 말 그대로 크랙 연습 목적으로 작성되어 공개된 프로그램
디버거를 실행하기 전에 파일을 실행



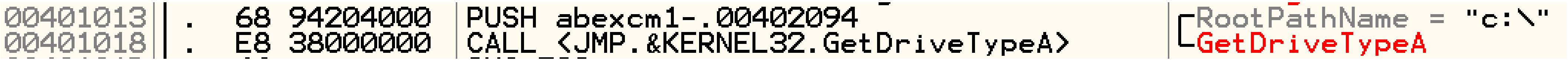
```
00401000 ┌$  6A 00          PUSH 0                                   ┌Style = MB_OK|MB_APPLMODAL
00401002 │.  68 00204000    PUSH abexcm1-.00402000                   │Title = "abex' 1st crackme"
00401007 │.  68 12204000    PUSH abexcm1-.00402012                   │Text = "Make me think your HD is a CD-Rom."
0040100C │.  6A 00          PUSH 0                                   │hOwner = NULL
0040100E │.  E8 4E000000    CALL <JMP.&USER32.MessageBoxA>           └MessageBoxA
00401013 │.  68 94204000    PUSH abexcm1-.00402094                   ┌RootPathName = "c:\"
00401018 │.  E8 38000000    CALL <JMP.&KERNEL32.GetDriveTypeA>       └GetDriveTypeA
0040101D │.  46             INC ESI
0040101E │.  48             DEC EAX
0040101F │.v EB 00          JMP SHORT abexcm1-.00401021
00401021 │>  46             INC ESI
00401022 │.  46             INC ESI
00401023 │.  48             DEC EAX
00401024 │.  3BC6           CMP EAX,ESI
00401026 │.v 74 15          JE SHORT abexcm1-.0040103D
00401028 │.  6A 00          PUSH 0                                   ┌Style = MB_OK|MB_APPLMODAL
0040102A │.  68 35204000    PUSH abexcm1-.00402035                   │Title = "Error"
0040102F │.  68 3B204000    PUSH abexcm1-.0040203B                   │Text = "Nah... This is not a CD-ROM Drive!"
00401034 │.  6A 00          PUSH 0                                   │hOwner = NULL
00401036 │.  E8 26000000    CALL <JMP.&USER32.MessageBoxA>           └MessageBoxA
0040103B │.v EB 13          JMP SHORT abexcm1-.00401050
0040103D │>  6A 00          PUSH 0                                   │┌Style = MB_OK|MB_APPLMODAL
0040103F │.  68 5E204000    PUSH abexcm1-.0040205E                   ││Title = "YEAH!"
00401044 │.  68 64204000    PUSH abexcm1-.00402064                   ││Text = "Ok, I really think that your HD is a CD-ROM! :p"
00401049 │.  6A 00          PUSH 0                                   ││hOwner = NULL
0040104B │.  E8 11000000    CALL <JMP.&USER32.MessageBoxA>           │└MessageBoxA
00401050 └>  E8 06000000    CALL <JMP.&KERNEL32.ExitProcess>         └ExitProcess
```



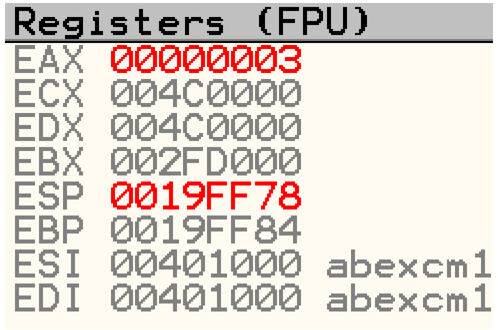
```
MessageBoxA("Make me think your HD is a CD-ROM.", "abex\' 1st crackme")
GetDriveTypeA("C:\\")
...
MessageBoxA("Nah... This is not a CD-ROM Drive!", "Error")
MessageBoxA("OK, I really think that your HD is a CD-ROM! :p", "YEAH!")
```

GetDriveType() API로 C 드라이브의 타입을 얻어오는데(대부분 HDD 타입이 리턴) 이걸 조작하여 CD-ROM 타입으로 인식하도록 만들어
"OK, I really think that your HD is a CD-ROM! :p“
메시지 박스가 출력되도록 하는 것

GetDriveTypeA() 호출

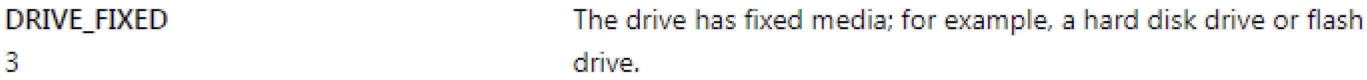


```
00401013  .  68 94204000  PUSH abexcm1-.00402094                  ┌RootPathName = "c:\"
00401018  .  E8 38000000  CALL <JMP.&KERNEL32.GetDriveTypeA>      └GetDriveTypeA
```

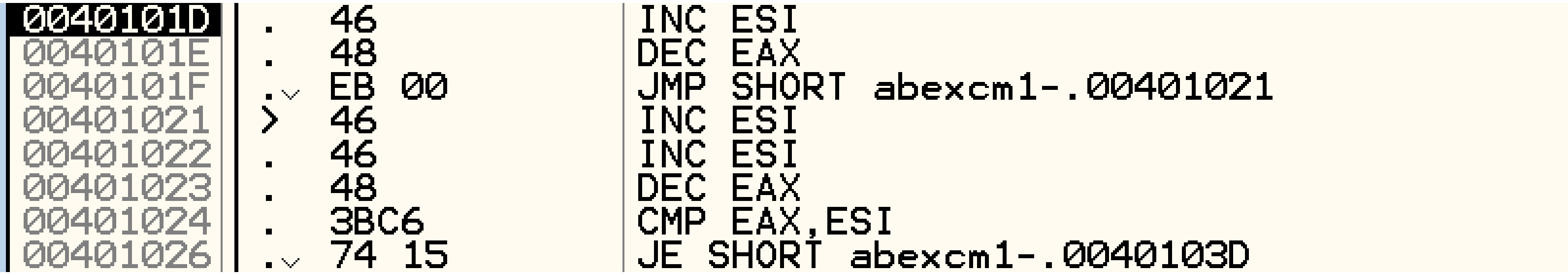


```
Registers (FPU)
EAX 00000003
ECX 004C0000
EDX 004C0000
EBX 002FD000
ESP 0019FF78
EBP 0019FF84
ESI 00401000 abexcm1
EDI 00401000 abexcm1
```

리턴 값(EAX)은 3(DRIVE_FIXED)



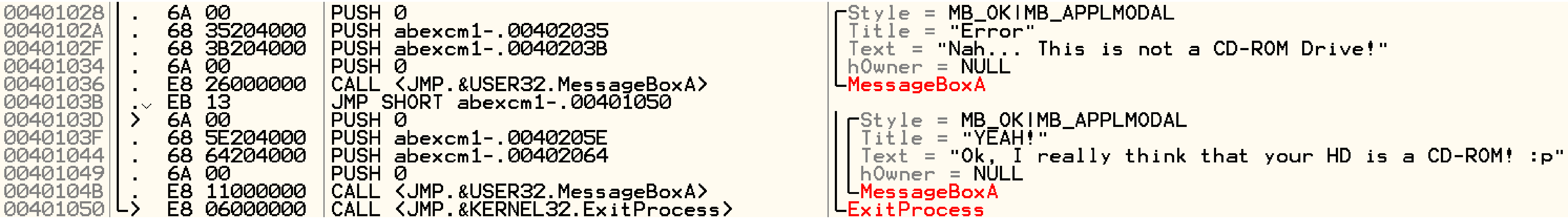
DRIVE_FIXED 3	The drive has fixed media; for example, a hard disk drive or flash drive.



```
0040101D  .    46      INC ESI
0040101E  .    48      DEC EAX
0040101F  .⌄   EB 00   JMP SHORT abexcm1-.00401021
00401021  >    46      INC ESI
00401022  .    46      INC ESI
00401023  .    48      DEC EAX
00401024  .    3BC6    CMP EAX,ESI
00401026  .⌄   74 15   JE SHORT abexcm1-.0040103D
```

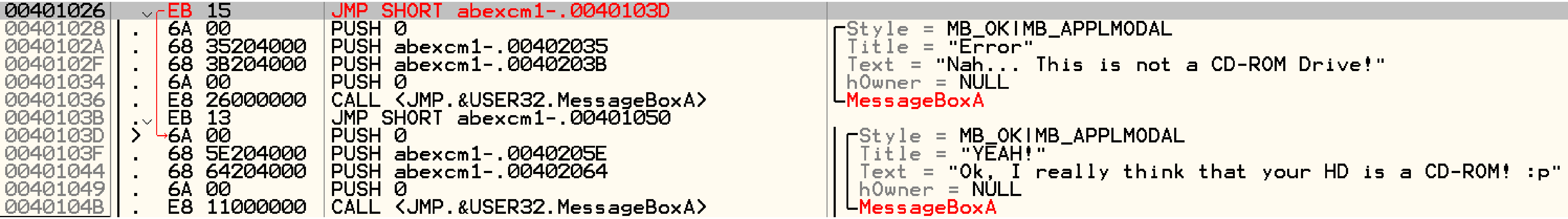
- 40101F은 의미 없는 JMP 명령 (garbage code)
  Garbage code는 디버깅을 방해하고 리버서를 혼란시키기 위해 고의적으로 추가된 것

- 401024에서EAX(1)와 ESI(2) 비교
  401026 - JE(Jump if Equal) 조건 분기 명령, 두 값이 같으면 40103D로 점프하고,
  다르면 밑(401028)으로 진행. 40103D 주소는 제작자가 원하는 메시지 박스 출력 코드



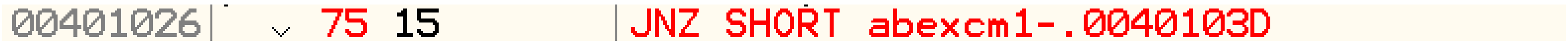
```
00401028  |.  6A 00          PUSH 0                                  ┌Style = MB_OK|MB_APPLMODAL
0040102A  |.  68 35204000    PUSH abexcm1-.00402035                  │Title = "Error"
0040102F  |.  68 3B204000    PUSH abexcm1-.0040203B                  │Text = "Nah... This is not a CD-ROM Drive!"
00401034  |.  6A 00          PUSH 0                                  │hOwner = NULL
00401036  |.  E8 26000000    CALL <JMP.&USER32.MessageBoxA>          └MessageBoxA
0040103B  |.v EB 13          JMP SHORT abexcm1-.00401050
0040103D  |>  6A 00          PUSH 0                                  │┌Style = MB_OK|MB_APPLMODAL
0040103F  |.  68 5E204000    PUSH abexcm1-.0040205E                  ││Title = "YEAH!"
00401044  |.  68 64204000    PUSH abexcm1-.00402064                  ││Text = "Ok, I really think that your HD is a CD-ROM! :p"
00401049  |.  6A 00          PUSH 0                                  ││hOwner = NULL
0040104B  |.  E8 11000000    CALL <JMP.&USER32.MessageBoxA>          │└MessageBoxA
00401050  └>  E8 06000000    CALL <JMP.&KERNEL32.ExitProcess>        └ExitProcess
```

패치



```
00401026   ┌EB 15         JMP SHORT abexcm1-.0040103D
00401028 │.│6A 00         PUSH 0                                ┌Style = MB_OK|MB_APPLMODAL
0040102A │.│68 35204000   PUSH abexcm1-.00402035                │Title = "Error"
0040102F │.│68 3B204000   PUSH abexcm1-.0040203B                │Text = "Nah... This is not a CD-ROM Drive!"
00401034 │.│6A 00         PUSH 0                                │hOwner = NULL
00401036 │.│E8 26000000   CALL <JMP.&USER32.MessageBoxA>        └MessageBoxA
0040103B │.│EB 13         JMP SHORT abexcm1-.00401050
0040103D │>└6A 00         PUSH 0                               │┌Style = MB_OK|MB_APPLMODAL
0040103F │.  68 5E204000  PUSH abexcm1-.0040205E               ││Title = "YEAH!"
00401044 │.  68 64204000  PUSH abexcm1-.00402064               ││Text = "Ok, I really think that your HD is a CD-ROM! :p"
00401049 │.  6A 00        PUSH 0                               ││hOwner = NULL
0040104B │.  E8 11000000  CALL <JMP.&USER32.MessageBoxA>       │└MessageBoxA
```

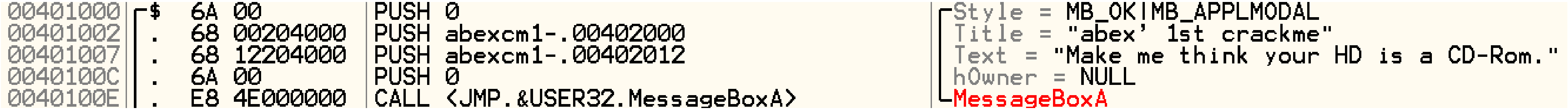
조건 분기(JE) 명령어를 점프 명령어로 바꾼다



```
00401026      75 15        JNZ SHORT abexcm1-.0040103D
```

조건 분기 명령어 JNE(jump if it’s not equal) 사용

스택에 파라미터를 전달하는 방법

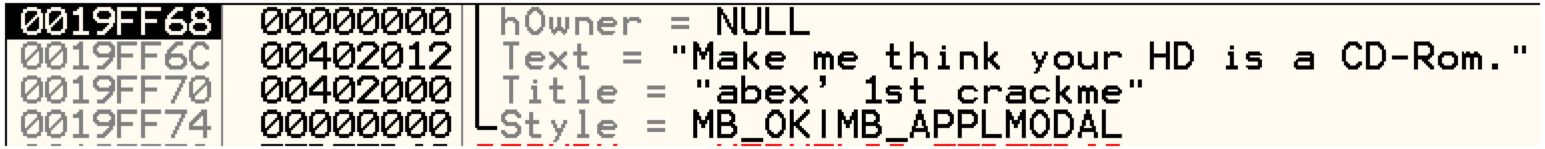


```
00401000 ┌$  6A 00        PUSH 0                                  ┌Style = MB_OK|MB_APPLMODAL
00401002 │.  68 00204000  PUSH abexcm1-.00402000                  │Title = "abex' 1st crackme"
00401007 │.  68 12204000  PUSH abexcm1-.00402012                  │Text = "Make me think your HD is a CD-Rom."
0040100C │.  6A 00        PUSH 0                                  │hOwner = NULL
0040100E │.  E8 4E000000  CALL <JMP.&USER32.MessageBoxA>          └MessageBoxA
```

401000~40100E 주소 사이의 명령어를 보면 MessageBoxA() 함수를 호출하기 전 4번의 PUSH 명령어를 사용하여 필요한 파라미터를 역순으로 입력

위 어셈블리 코드를 C 언어로 번역하면

MessageBox(NULL, "Make me think your HD is a CD-Rom.", "abex' 1st crackme", MB_OK|MB_APPLMODAL);



```
0019FF68   00000000  │hOwner = NULL
0019FF6C   00402012  │Text = "Make me think your HD is a CD-Rom."
0019FF70   00402000  │Title = "abex' 1st crackme"
0019FF74   00000000  └Style = MB_OK|MB_APPLMODAL
```

c언어 소스코드에서 함수에 넘기는 파라미터의 순서가 어셈블리 언어에서는 역순으로 넘어감
스택은 FILO(First In Last Out) 구조이기에 역순으로 넣어두면 받는 쪽에서 올바른 순서로 꺼냄

Thank you