Basic RCE L03

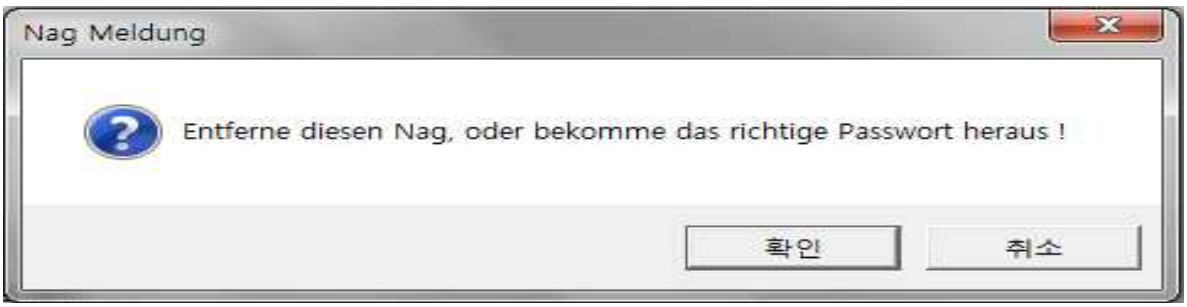
Korea :

비주얼베이직에서 스트링 비교함수 이름은?

English :

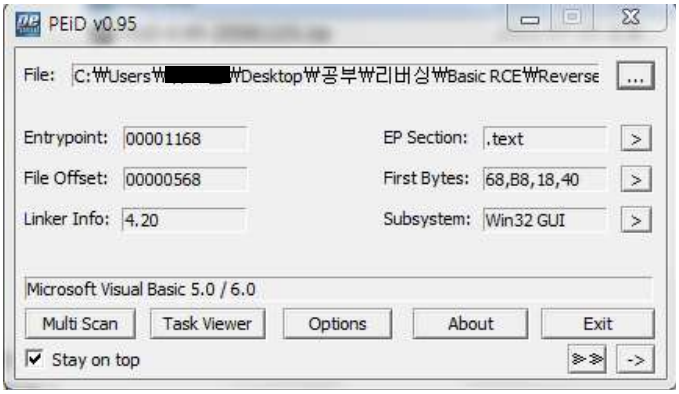
What is the name of the Visual Basic function that compares two strings?

역시 일단 프로그램을 실행시켜보자.

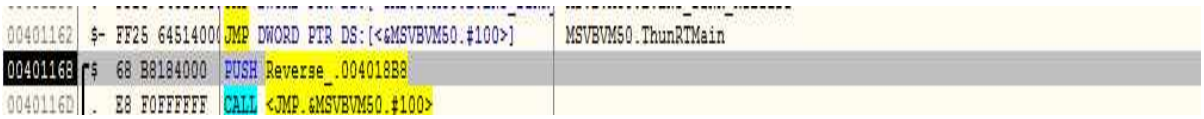


무슨 말인지 정확히는 모르겠다.
일단 취소를 누르면 프로그램이 끝나고, 확인을 누르면

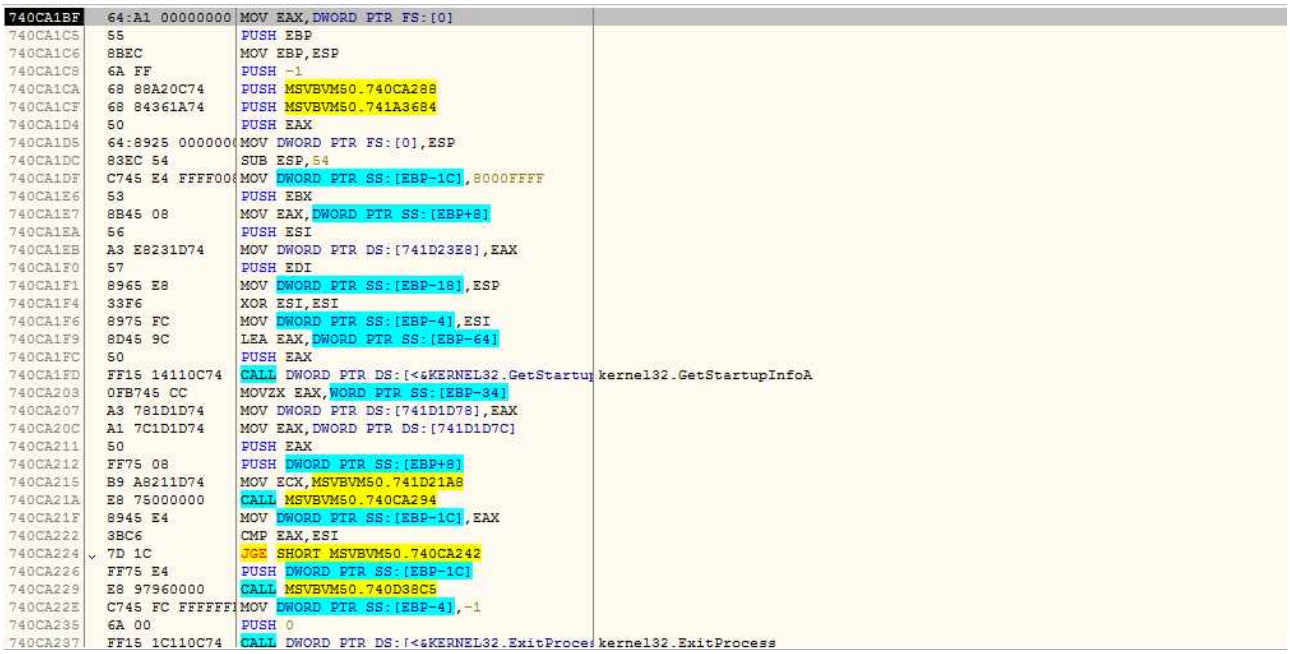
이런 창이 하나 더 뜬다. 시리얼을 입력해서 맞나 안맞나 확인하는 프로그램인 것 같다.
PEID로 프로그램을 열어보자.



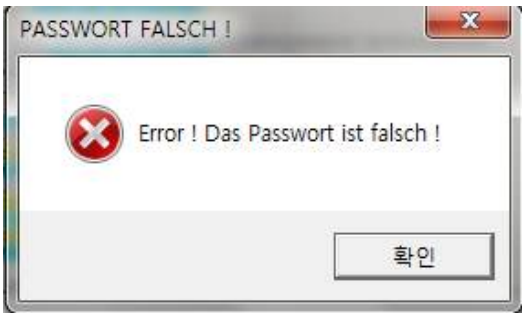
PEID 라는 프로그램으로 열면은 이 프로그램이 패킹되었는지,무슨 언어로 됐는지 등을 알 수 있다.
이 프로그램은 Visual Basic 으로 만들어 졌다. 그래서 문제가 Visual Basic 의 스트링 비교 함수를 구하는 건가 보다.
이번엔 올리디버거로 열어 보자.



처음 코드가 이것이다. F8로 한줄씩 실행시켜보면 CALL <JMP.&MSVBVM50.#100> 에서 프로그램이 실행 된다. 그럼 이번에는 Ctrl+F2를 눌러서 다시 처음으로 되돌아 간후
CALL <JMP.&MSVBVM50.#100>에서 F7을 눌러 내부로 들어가 보자. 그러면 00401162에 있는
JMP DWORD PTR DS:[<&MSVBVM50.#100>] 으로 오게 될 것이다. F8로 넘어가자. 그러면
이번엔 뭔가 제대로된 코드가 나온다.



메시지가 뜨고 제대로 된 실행은 740CA21A주소의 CALL MSVBVM50.740CA294 에서 될 것이다.
하지만 우리는 스트링 비교함수를 알면 된다. 프로그램을 실행한 후 시리얼을 입력하는 곳에 아무거나 입력하고 Registrieren 버튼을 누르자. 그러면 틀렸다는 뜻인 것 같은 메시지가 하나 뜰 것이다.



다시 Ctrl+F2를 눌러서 처음 화면으로 간 후 오른쪽 키를 눌러서 [Search for] - [All referenced text strings]를 누르면은 프로그램 내의 저장된 문자열들을 볼 수 있다. 그럼 창이 하나 뜨는데 맨

밑으로 내려보면

```
0040217C DD Reverse_.00401C84                ASCII "Form"
004021A4 DD Reverse_.00401CFC                ASCII "mnuprog"
004021CC DD Reverse_.00401D14                ASCII "Command1"
004021F4 DD Reverse_.00401D20                ASCII "Command2"
0040221C DD Reverse_.00401D2C                ASCII "mnuabout"
00402244 DD Reverse_.00401D48                ASCII "Text1"
0040226C DD Reverse_.00401D60                ASCII "Label2"
00402294 DD Reverse_.00401D68                ASCII "mnuexit"
004022BC DD Reverse_.00401D70                ASCII "Label3"
004022E4 DD Reverse_.00401D78                ASCII "Label1"
004028BD PUSH Reverse_.00401DDC              UNICODE "2G83G35Hs2"
004028F5 MOV DWORD PTR SS:[EBP-84],Reverse_.0040 UNICODE "Danke, das Passwort ist richtig !"
00402A2A PUSH Reverse_.00401DDC              UNICODE "2G83G35Hs2"
00402A69 MOV DWORD PTR SS:[EBP-84],Reverse_.0040 UNICODE "Error ! Das Passwort ist falsch !"
00402AA9 MOV DWORD PTR SS:[EBP-84],Reverse_.0040 UNICODE "PASSWORT FALSCH !"
00402C85 MOV DWORD PTR SS:[EBP-7C],Reverse_.0040 UNICODE "Entferne diesen Nag, oder bekomme das richtige Password heraus !"
00402CBE MOV DWORD PTR SS:[EBP-7C],Reverse_.0040 UNICODE "Nag Meldung"
00402E28 MOV DWORD PTR SS:[EBP-5C],Reverse_.0040 UNICODE "VB5-CrackMe 1.0 by Blaster99 [DCD]"
00402F9A PUSH Reverse_.00401FEC              UNICODE "Visible"
00403060 PUSH Reverse_.00401FEC              UNICODE "Visible"
```

이런게 있을 것이다. 실패했을때의 메시지인 "Error! Das Passwort ist falsch!"에 커서를 놓고 더블 클릭을 하면 그 코드에 갈 수 있다. 해보자.

```
00402A69  .  C785 7CFFFFF1 MOV DWORD PTR SS:[EBP-84],Reverse_.0040 UNICODE "Error ! Das Passwort ist falsch !"
00402A73  .  C785 74FFFFF1 MOV DWORD PTR SS:[EBP-8C],8
00402A7D  .  E8 AAE6FFFF   CALL <JMP.&MSVBVM50.__vbaVarCopy>
00402A82  .  8D95 74FFFFF1 LEA EDX,DWORD PTR SS:[EBP-8C]
00402A88  .  8D4D DC       LEA ECX,DWORD PTR SS:[EBP-24]
00402A8B  .  C785 7CFFFFF1 MOV DWORD PTR SS:[EBP-84],10
00402A95  .  899D 74FFFFF1 MOV DWORD PTR SS:[EBP-8C],EBX
00402A9B  .  E8 86E6FFFF   CALL <JMP.&MSVBVM50.__vbaVarMove>
00402AA0  .  8D95 74FFFFF1 LEA EDX,DWORD PTR SS:[EBP-8C]
00402AA6  .  8D4D CC       LEA ECX,DWORD PTR SS:[EBP-34]
00402AA9  .  C785 7CFFFFF1 MOV DWORD PTR SS:[EBP-84],Reverse_.0040 UNICODE "PASSWORT FALSCH !"
```

이렇게 코드가 나올 것이다. 여기서 조금 더 위로 올려보면

```
004028E3  .v 0F84 F3000000 JE Reverse_.004029DC
004028E9  .  6A 08         PUSH 8
004028EB  .  8D95 74FFFFF1 LEA EDX,DWORD PTR SS:[EBP-8C]
004028F1  .  5E            POP ESI
004028F2  .  8D4D AC       LEA ECX,DWORD PTR SS:[EBP-54]
004028F5  .  C785 7CFFFFF1 MOV DWORD PTR SS:[EBP-84],Reverse_.0040 UNICODE "Danke, das Passwort ist richtig !"
```

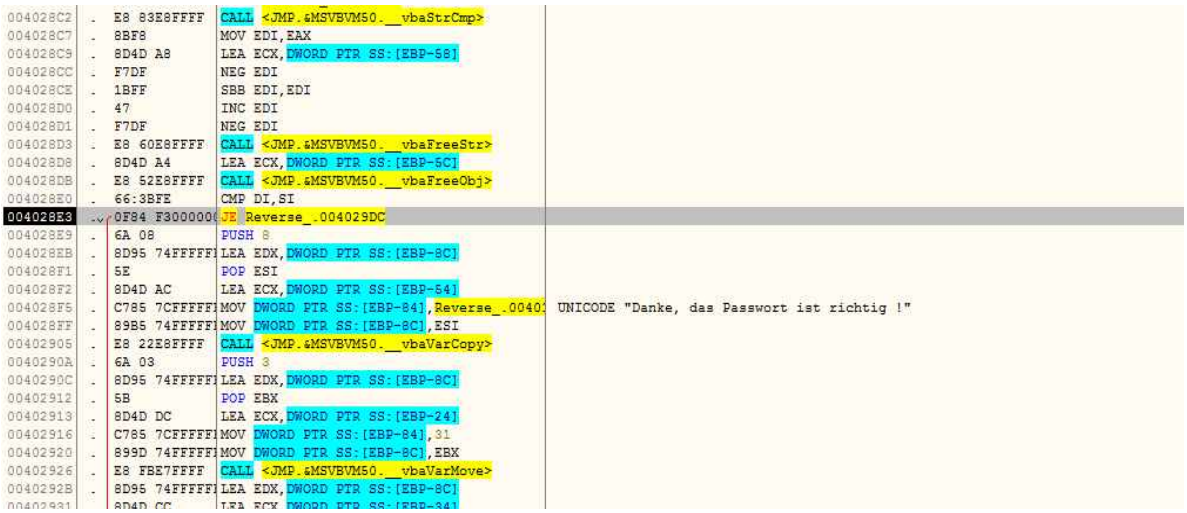
저렇게 성공했을때의 메시지로 추정되는 “Danke, das Passwort ist richtig !"가 보인다.
그리고 그 위에는 JE Reverse_.004029DC 가 보인다.
그리고 좀더 위를 보면은

```
004028BA  >  FF75 A8       PUSH DWORD PTR SS:[EBP-58]
004028BD  .  68 DC1D4000   PUSH Reverse_.00401DDC                UNICODE "2G83G35Hs2"
004028C2  .  E8 83E8FFFF   CALL <JMP.&MSVBVM50.__vbaStrCmp>
004028C7  .  8BF8          MOV EDI,EAX
004028C9  .  8D4D A8       LEA ECX,DWORD PTR SS:[EBP-58]
004028CC  .  F7DF          NEG EDI
004028CE  .  1BFF          SBB EDI,EDI
004028D0  .  47            INC EDI
004028D1  .  F7DF          NEG EDI
004028D3  .  E8 60E8FFFF   CALL <JMP.&MSVBVM50.__vbaFreeStr>
004028D8  .  8D4D A4       LEA ECX,DWORD PTR SS:[EBP-5C]
004028DB  .  E8 52E8FFFF   CALL <JMP.&MSVBVM50.__vbaFreeObj>
004028E0  .  66:3BFE       CMP DI,SI
004028E3  .v 0F84 F3000000 JE Reverse_.004029DC
004028E9  .  6A 08         PUSH 8
004028EB  .  8D95 74FFFFF1 LEA EDX,DWORD PTR SS:[EBP-8C]
004028F1  .  5E            POP ESI
004028F2  .  8D4D AC       LEA ECX,DWORD PTR SS:[EBP-54]
004028F5  .  C785 7CFFFFF1 MOV DWORD PTR SS:[EBP-84],Reverse_.0040 UNICODE "Danke, das Passwort ist richtig !"
```

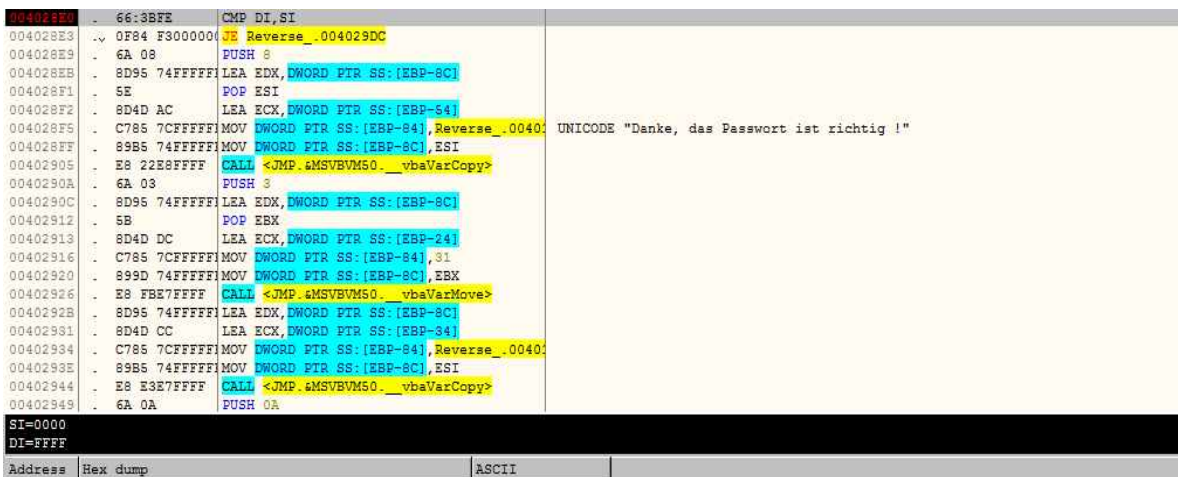
004028BA PUSH DWORD PTR SS:[EBP-58]
004028BD PUSH Reverse_.00401DDC UNICODE "2G83G35Hs2"
004028C2 CALL <JMP.&MSVBVM50.__vbaStrCmp>
이 부분을 보면 PUSH 두 번을 하고 __vbaStrCmp 함수를 호출하는 코드이다.
004028BA에 BP(Break Point)(단축키 F2)를 걸고 F9를 눌러 실행을 한후 F8로 하나씩 실행 해 보면은
PUSH DWORD PTR SS:[EBP-58] 를 실행하면 스택창에 자기가 입력한 시리얼이,(bysac을 입력함)
PUSH Reverse_.00401DDC 를 실행하면 스택창에 2G83G35Hs2 가 들어간다.

Stack은 FILO(First In Last Out) 구조라서 역순으로 들어간다. 참고로 Dump창에서
Ctrl+ G를 눌러서 004010DDC를 입력하면은 2G83G35Hs2 를 볼 수 있을 것이다.
다음으로 CALL <JMP.&MSVBVM50.__vbaStrCmp> 를 실행을 한다.
딱 봐도 vbaStrCmp가 Visual Basic String Compare 인 것 같다. 그러므로 답은 __vbaStrCmp 일 것이다. 구글링으로 검색을 해봐도 알 수 있을 것이다. 답을 입력해보자. 성공이다.

CALL <JMP.&MSVBVM50.__vbaStrCmp> 뒤에 여러 코드가 있는데 아마도 입력한 값이
2G83G35Hs2가 아니면은 CMP DI,SI 까지 여러 코드를 거친 후
CMP DI,SI 에서 DI와 SI가 같아서 점프를 해 실패 메시지가 뜨게 하는 것 같다.



그래서 004028E0 에 BP를 걸고 2G83G35Hs2를 입력하고 실행시켜보면 코드창과 Dump창 사이에
SI=0000
DI=FFFF



라고 뜬다. 두 값이 서로 다르므로 점프를 하지 않는다. 그러므로 성공메시지가 뜬다.
(2G83G35Hs2 말고 다른 값을 입력하면 DI의 값이 0000이 되어 점프함.)

2011.08.16

Made by hypen1117
hypen1117@daum.net
http://hypen1117.tistory.com

리버싱을 그렇게 잘하지 못해 구체적이지 못하고 잘못 될 수 있습니다.
잘못되거나 부족한 부분 지적해 주시길 바랍니다.