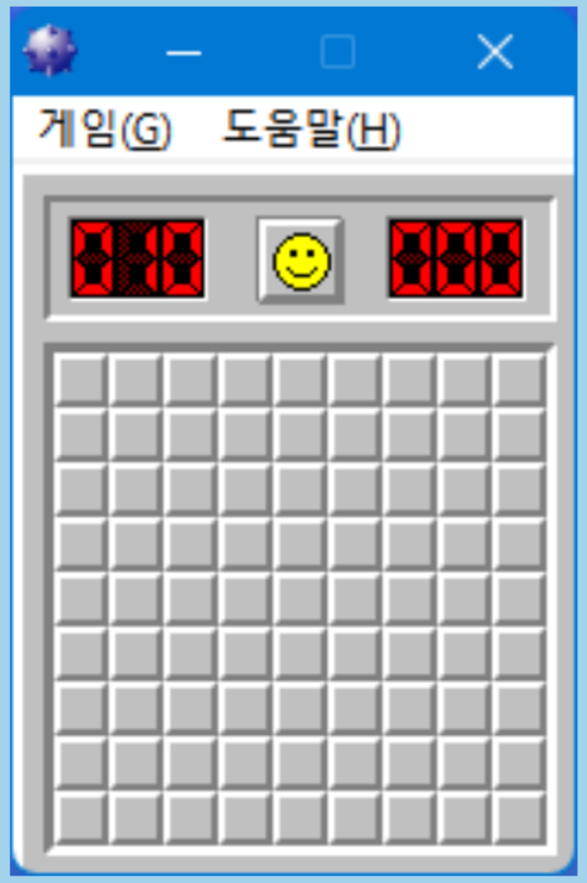
Mine Sweeper

지뢰찾기 리버싱

SCP | 이지훈 2024.09.30
2024-2학기 SCP 내부 세미나 B조

index

01 | 지뢰찾기 : 소개



MineSweeper

수많은 블록들중에 숨겨진 지뢰들을 주변의 숫자 단서를 통해 유추해 내어 지뢰를 피해 모든 지뢰를 찾아내는 게임.

마우스마저 익숙하지 않던 시절(...)에 사용자들이 마우스에 익숙해지도록 만든 게임들 중 하나

02 | 지뢰찾기 : 분석 준비

Dynamic Analysis

게임 시작 후 타이머가 흘러가는 모습

02 | 지뢰찾기 : 분석 준비

Dynamic Analysis

지뢰를 밟을 경우 게임 오버가 되면서 숨어있던 지뢰가 보이는 모습

02 | 지뢰찾기 : 분석 준비

MineSweeper

스마일 버튼을 누르면 새로운 게임이 시작되는 모습

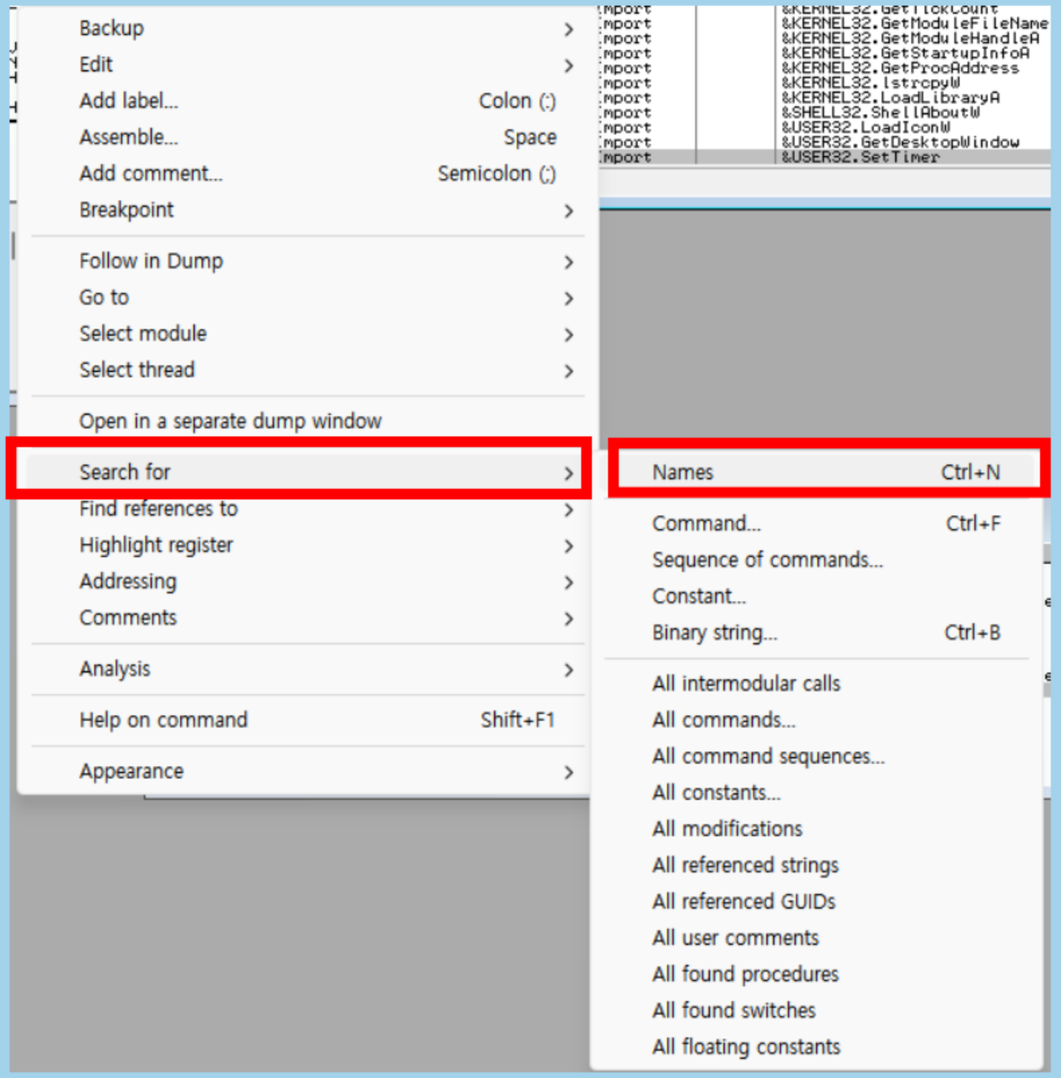
02 I 지뢰찾기 : 분석 준비

01 I 타이머 조작
지뢰찾기의 랭킹은 시간순으로 정해진다.

02 I 지뢰개수 조작
찾아야 할 지뢰의 개수가 적어진다면?

03 I 지뢰 위치 보이기
지뢰를 찾을 필요 없이 화면에 바로 보인다면?

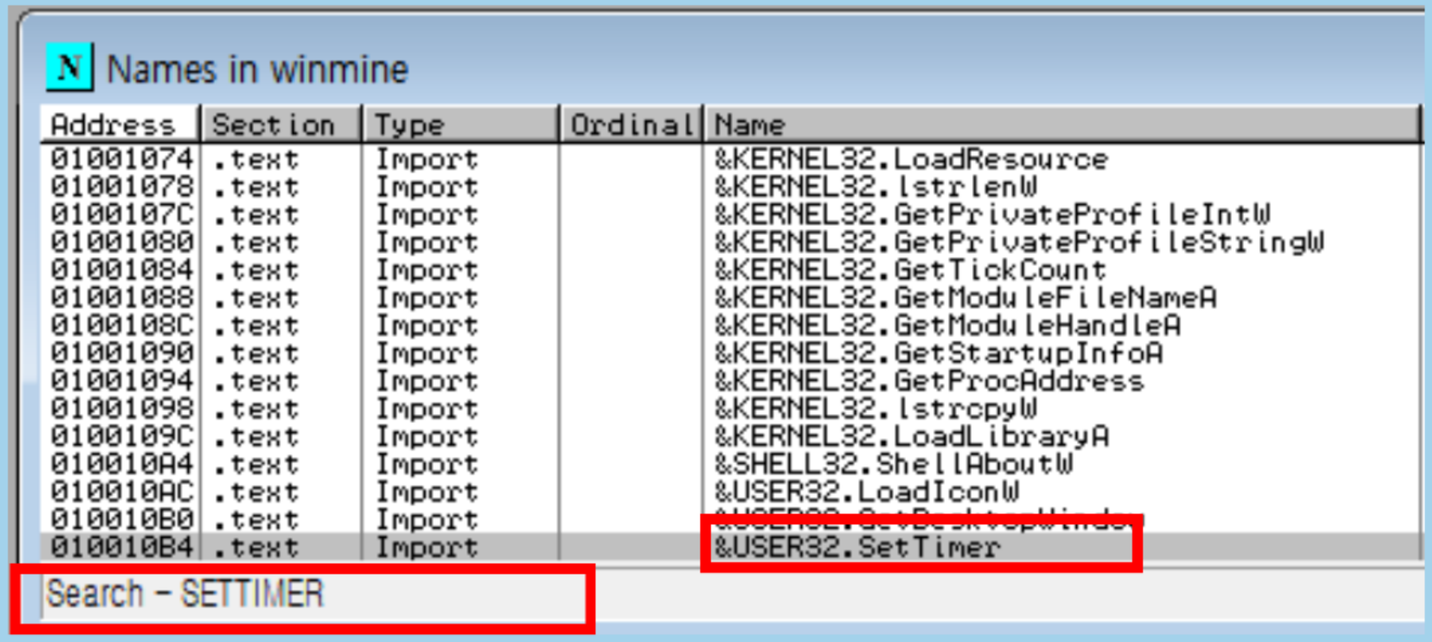
03 I 지뢰 찾기 : 타이머 조작



SetTimer()

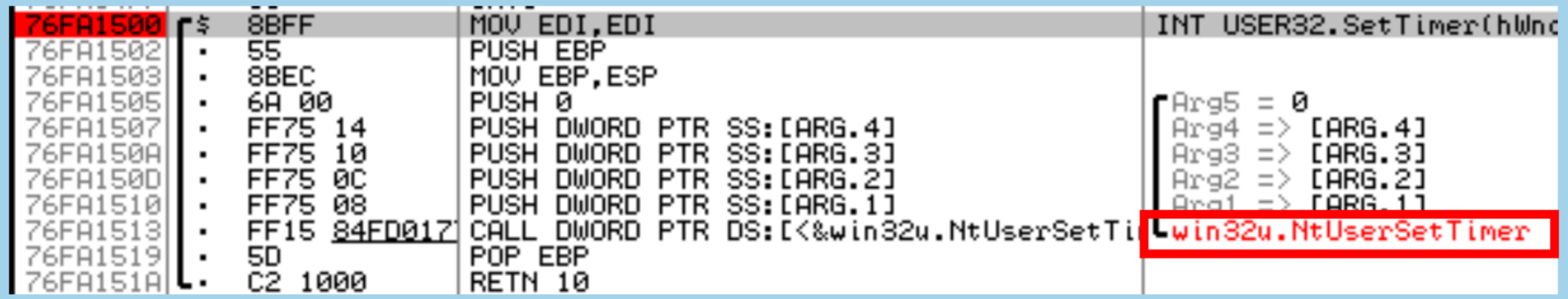
어셈블리 코드 우클릭 -> Search for -> Names

03 I 지뢰 찾기 : 타이머 조작



Names in winmine 창에서 SetTimer 검색
-> 더블클릭 하여 해당 주소로 이동

03 I 지뢰 찾기 : 타이머 조작



SetTimer() 구조체에 따라 값들을 push 후 api를 실행시키는 부분
진입점에 BreakPoint 후 게임 실행하니 BP에서 멈추는 것 확인

03 I 지뢰 찾기 : 타이머 조작

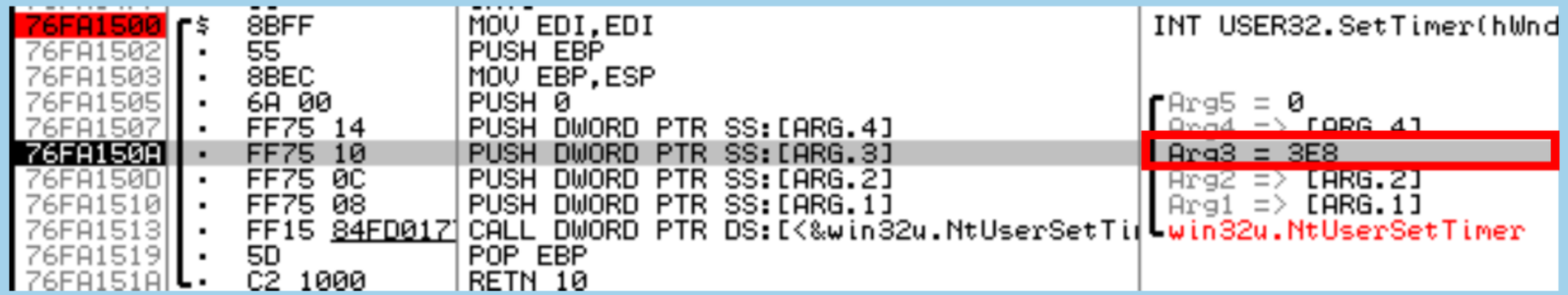
```
UINT_PTR SetTimer(
  [in, optional] HWND      hWnd,
  [in]           UINT_PTR  nIDEvent,
  [in]           UINT      uElapse,
  [in, optional] TIMERPROC lpTimerFunc
);
```

SetTimer()

[in] uElapse
형식: UINT
제한 시간 값(밀리초)입니다.

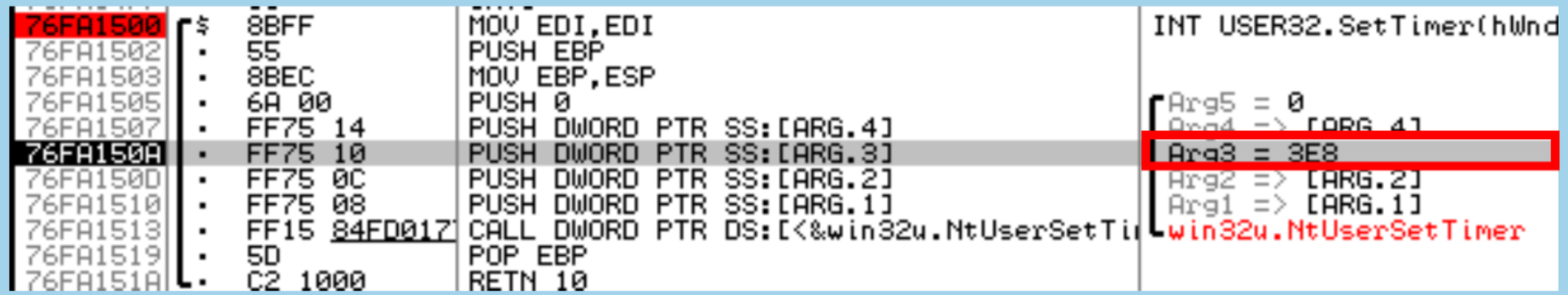
-> 시간 간격을 조절해주는 변수!!

03 I 지뢰 찾기 : 타이머 조작



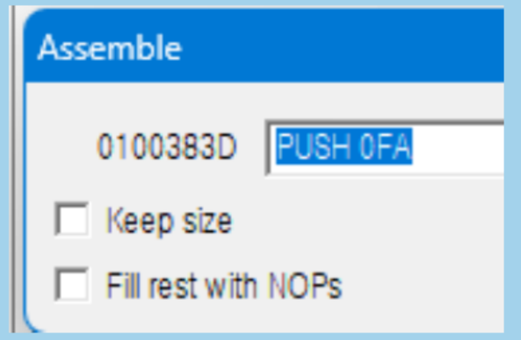
0x3E8 = 1000 (1000ms = 1초)
메모리 값 0xFA (250ms = 0.25초) 로 변경후 함수 CALL

03 I 지뢰 찾기 : 타이머 조작



하지만 이 경우 매번 게임을 시작할 때마다 메모리 변경을 해줘야함..
-> 해당 api call 이전에 인자값을 넘겨주는 부분 탐색!

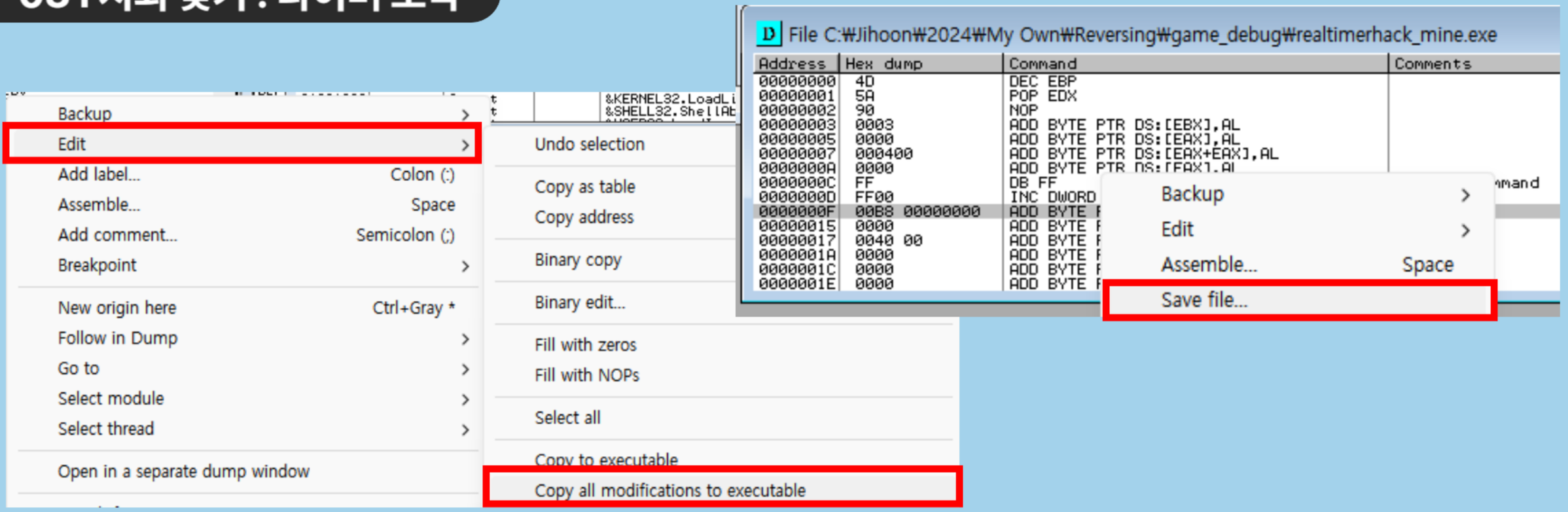
03 I 지뢰 찾기 : 타이머 조작




```
01003836  ·  E8 7AF0FFFF   CALL 010028B5                  winmine.010028B5
0100383B  ·  6A 00         PUSH 0                        TimerFunc = 00000000
0100383D  ·  68 E8030000   PUSH 3E8                      Timeout = 1000. ms
01003842  ·  53            PUSH EBX                      TimerID
01003843  ·  FF35 245B0000 PUSH DWORD PTR DS:[1005B24]   hWnd = 01620870, class = 지뢰 찾기, text = 지뢰 찾기
01003849  ·  891D 6451000  MOV DWORD PTR DS:[1005164],EBX
0100384F  ·  FF15 B410000  CALL DWORD PTR DS:[<&USER32.SetTimer>]  USER32.SetTimer
01003855  ·  85C0          TEST EAX,EAX
```

api CALL 이전 구간에서 push 0x3E8 발견
원하는 값으로 변경후 저장

03 I 지뢰 찾기 : 타이머 조작



api CALL 이전 구간에서 push 0x3E8 발견
원하는 값으로 변경후 저장

04 | 지뢰 찾기 : 지뢰 개수 조작

In Memory

메모리 덤프에서 지뢰판으로 보이는 메모리 구간을 찾음

0x01005330 값의 0A 는 지뢰 화면에서 지뢰의 개수를 나타내는 곳과 일치
-> 난이도에 따라 지뢰 개수가 변경됨

해당 메모리 값을 통해 처음 지뢰 생성 개수를 조정하는 것을 알 수 있음

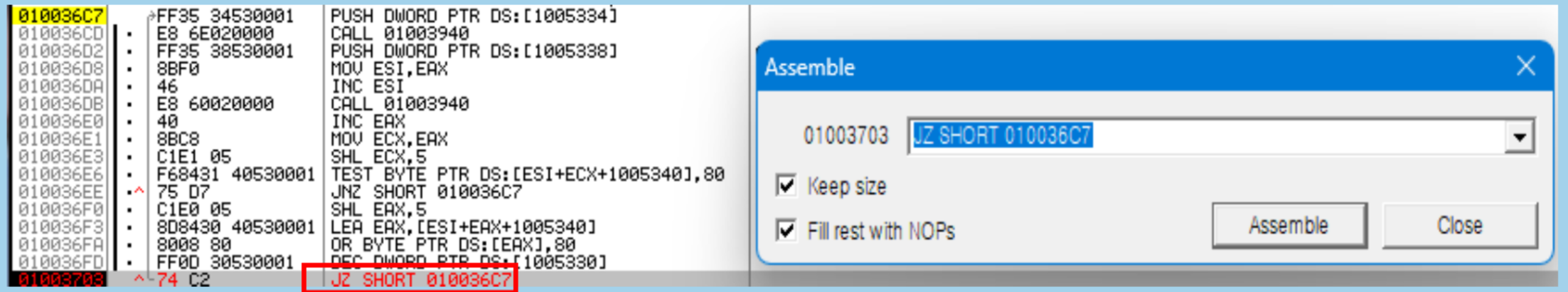
04 | 지뢰 찾기 : 지뢰 개수 조작

```
010036C7      FF35 34530001   PUSH DWORD PTR DS:[1005334]
010036CD  .   E8 6E020000     CALL 01003940
010036D2  .   FF35 38530001   PUSH DWORD PTR DS:[1005338]      Arg1 = 9
010036D8  .   8BF0            MOV ESI,EAX
010036DA  .   46              INC ESI
010036DB  .   E8 60020000     CALL 01003940                   winmine.01003940
010036E0  .   40              INC EAX
010036E1  .   8BC8            MOV ECX,EAX
010036E3  .   C1E1 05         SHL ECX,5
010036E6  .   F68431 40530001 TEST BYTE PTR DS:[ESI+ECX+1005340],80
010036EE  .^  75 D7           JNZ SHORT 010036C7
010036F0  .   C1E0 05         SHL EAX,5
010036F3  .   8D8430 40530001 LEA EAX,[ESI+EAX+1005340]
010036FA  .   8008 80         OR BYTE PTR DS:[EAX],80
010036FD  .   FF0D 30530001   DEC DWORD PTR DS:[1005330]
01003703      ^ 75 C2         JNZ SHORT 010036C7
```

```
01005330 0A
```

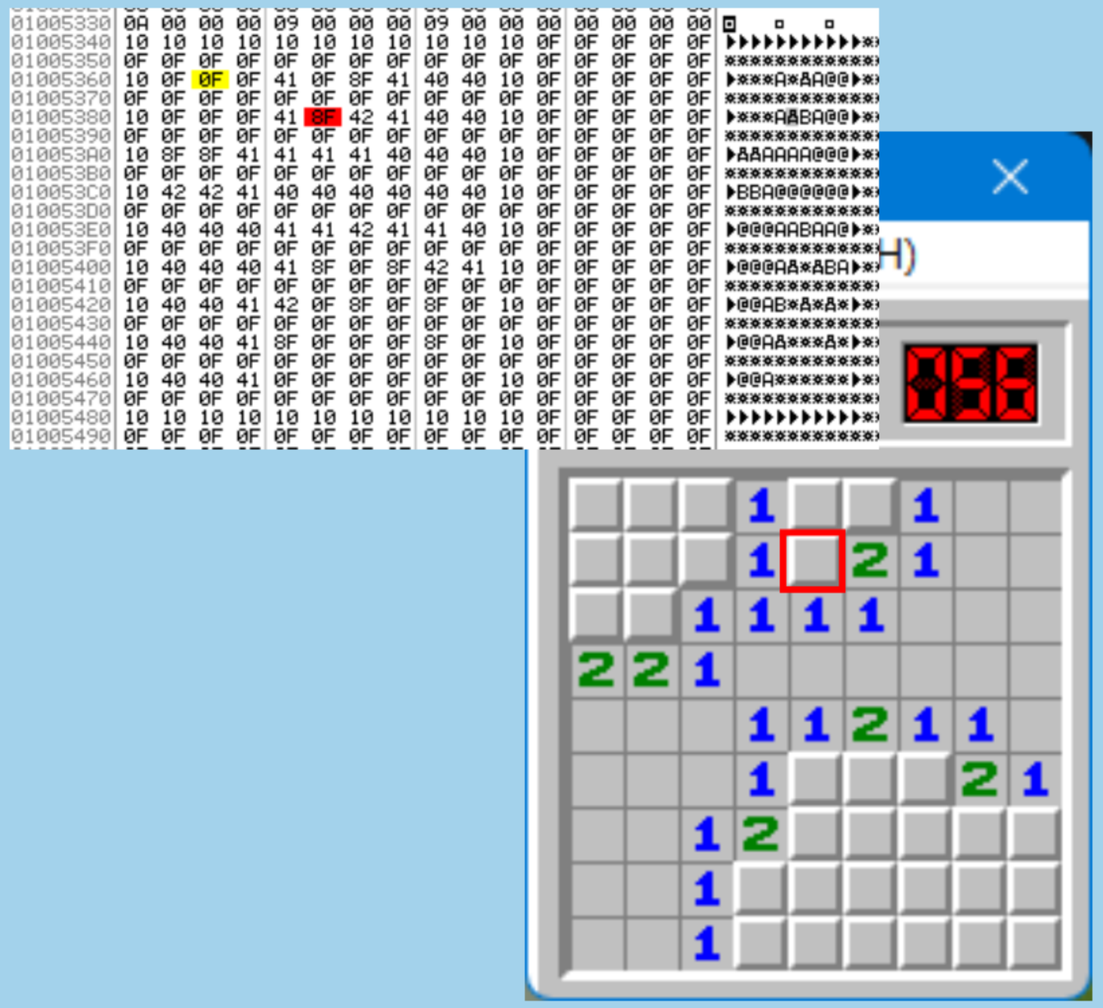
지뢰 랜덤 생성 후 메모리에 저장하는 루틴
0x010036FD 에서 메모리 값을 1씩 감소시키고,
JNZ (ZF가 0이면, 즉 윗 값이 0이 아니면 JMP) 로 윗 코드로 되돌아감

04 | 지뢰 찾기 : 지뢰 개수 조작



해당 부분을 JZ 로 변경하여 1회만 수행 후 바로 루틴 탈출

05 I 지뢰 찾기 : 지뢰 표시 조작



In Memory

게임 오버 후 남은 지뢰가 표시된다는 게임의 특성을 이용하여

지뢰가 있는 곳에 하드웨어 브레이크 포인트 설정 후
고의로 지뢰 클릭

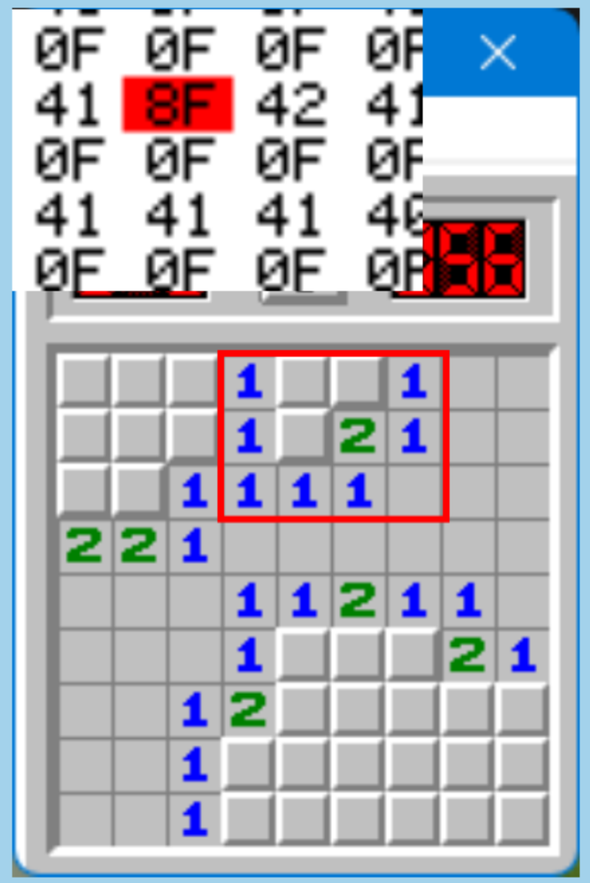
05 I 지뢰 찾기 : 지뢰 표시 조작

In Memory : 분석 결과

00 -> 지뢰 없는 곳이 클릭된 상태
80 -> 지뢰 있는 곳이 클릭된 상태
CC -> 빨강 지뢰(클릭해서 터진 곳의 지뢰)
8a -> 하얀 지뢰(보여지지 않은 곳의 지뢰가 밝혀진 모습)
8f -> 보여지지 않은 곳의 지뢰
0f -> 보여지지 않은 지뢰가 없는 곳

04 | 지뢰 찾기 : 지뢰 개수 조작

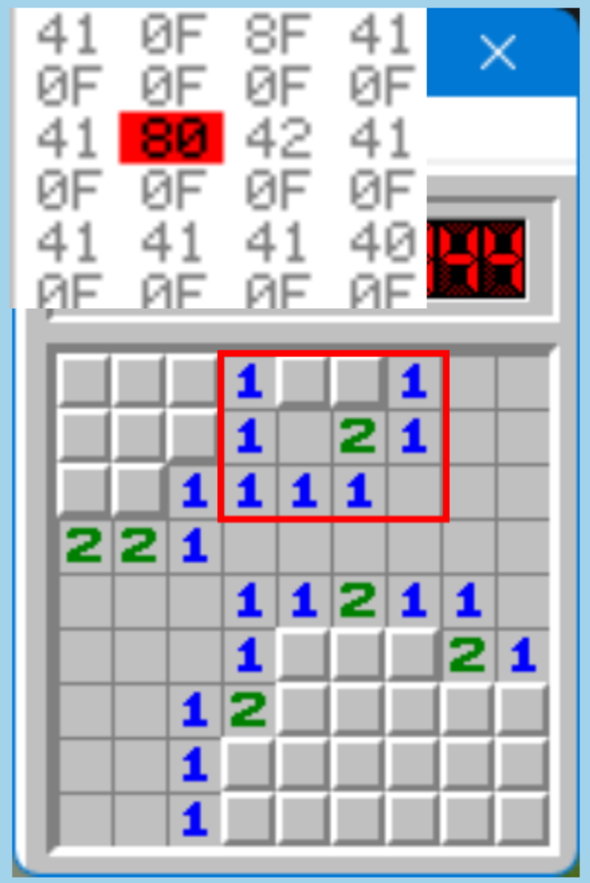
```
01003401  .  8B35 18510001   MOV ESI,DWORD PTR DS:[1005118]
01003407  .  57              PUSH EDI                          Arg2 => [100511C] = 2
01003408  .  56              PUSH ESI                          Arg1 => [1005118] = 5
01003409  .  E8 5DFDFFFF     CALL 0100316B                     winmine.0100316B
0100340E  .  57              PUSH EDI                          Arg2 = 2
0100340F  .  56              PUSH ESI                          Arg1 => [1005118] = 5
01003410  .  E8 31F2FFFF     CALL 01002646                     winmine.01002646
01003415  >  5E              POP ESI
01003416  .  5B              POP EBX
01003417  >  5F              POP EDI
01003418  .  C9              LEAVE
01003419  L. C2 0800         RETN 8
```



첫 하드웨어 BP 구간 : 8f -> 80, 즉 클릭에 대한 반응을 보이는 루틴 구간이었음

04 | 지뢰 찾기 : 지뢰 개수 조작

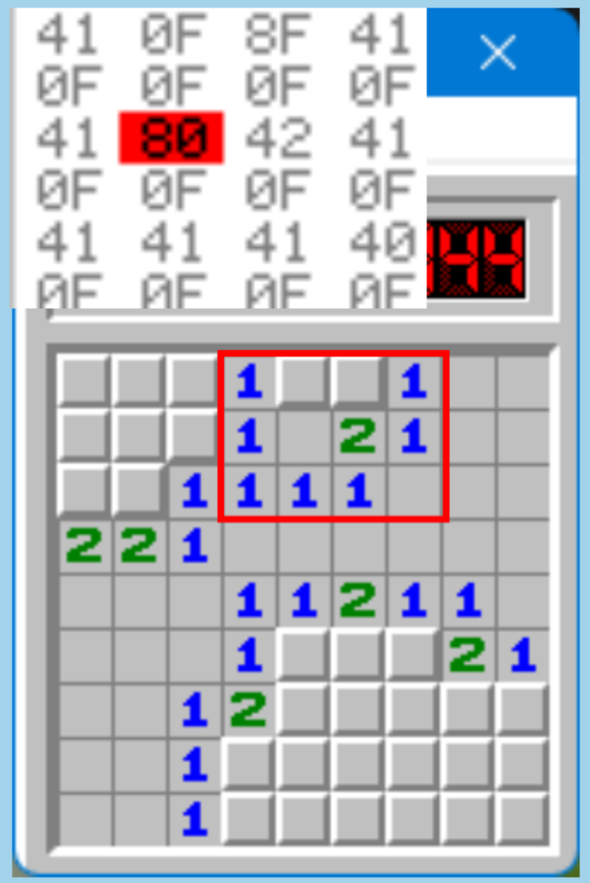
```
01003401  .  8B35 18510001   MOV ESI,DWORD PTR DS:[1005118]
01003407  .  57              PUSH EDI                          Arg2 => [100511C] = 2
01003408  .  56              PUSH ESI                          Arg1 => [1005118] = 5
01003409  .  E8 5DFDFFFF     CALL 0100316B                     winmine.0100316B
0100340E  .  57              PUSH EDI                          Arg2 = 2
0100340F  .  56              PUSH ESI                          Arg1 => [1005118] = 5
01003410  .  E8 31F2FFFF     CALL 01002646                     winmine.01002646
01003415  >  5E              POP ESI
01003416  .  5B              POP EBX
01003417  >  5F              POP EDI
01003418  .  C9              LEAVE
01003419  L. C2 0800         RETN 8
```



첫 하드웨어 BP 구간 : 8f -> 80, 즉 클릭에 대한 반응을 보이는 루틴 구간이었음

04 | 지뢰 찾기 : 지뢰 개수 조작

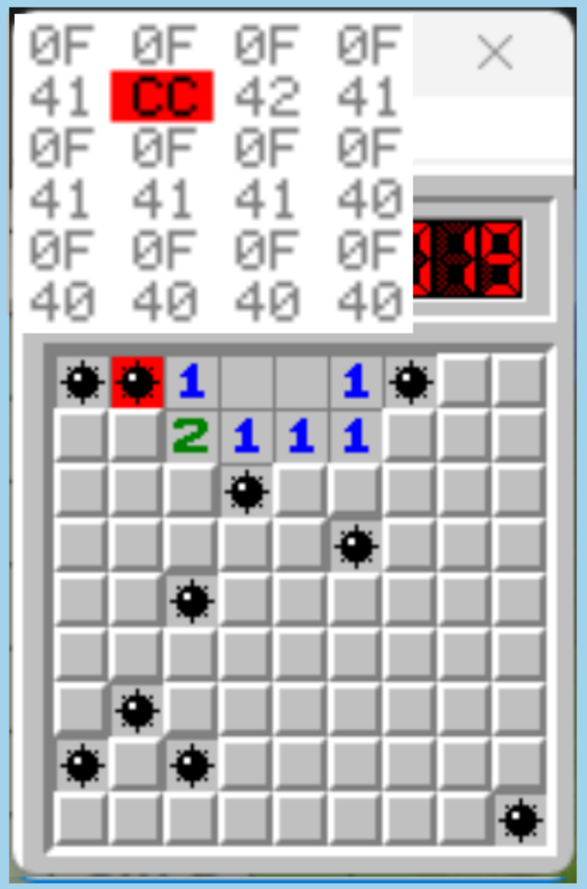
```
01003476  $  8325 64510001 00  AND DWORD PTR DS:[1005164],00000000   winmine.0100347C(guessed Arg
01003483  .  56                PUSH ESI
01003484  .  8B7424 08         MOV ESI,DWORD PTR SS:[ARG.1]
01003488  .  33C0              XOR EAX,EAX
0100348A  .  85F6              TEST ESI,ESI
0100348C  .  0F95C0            SETNZ AL
0100348F  .  40                INC EAX
01003490  .  40                INC EAX
01003491  .  50                PUSH EAX                              Arg1
01003492  .  A3 60510001       MOV DWORD PTR DS:[1005160],EAX
01003497  .  E8 77F4FFFF       CALL 01002913                         winmine.01002913
0100349C  .  33C0              XOR EAX,EAX
0100349E  .  85F6              TEST ESI,ESI
010034A0  .  0F95C0            SETNZ AL
010034A3  .  8D0485 0A000000   LEA EAX,[EAX*4+0A]
010034AA  .  50                PUSH EAX                              Arg1
010034AB  .  E8 D0FAFFFF       CALL 01002F80                         winmine.01002F80
010034B0  .  85F6              TEST ESI,ESI
```



두번재 하드웨어 BP 구간 : 80 -> CC, 지뢰임을 판별하고 게임오버,
이후에 다른 지뢰를 보여주는 함수(0x01002F80)를 CALL!!!
-> 해당 함수를 내가 원할 때 CALL 할 수 있다면?

04 | 지뢰 찾기 : 지뢰 개수 조작

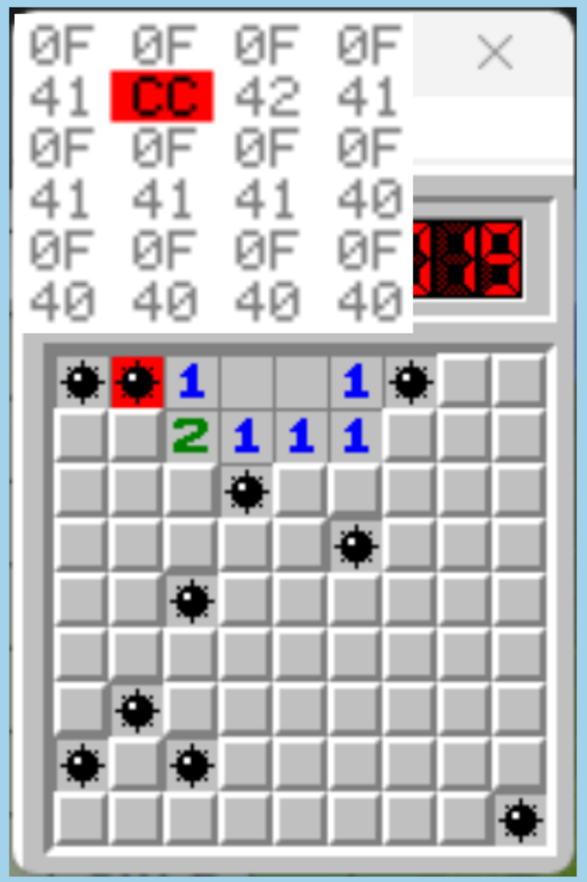
```
0100347C  $  8325 64510001 00  AND DWORD PTR DS:[1005164],00000000   winmine.0100347C(guessed Arg
01003483  .  56                PUSH ESI
01003484  .  8B7424 08         MOV ESI,DWORD PTR SS:[ARG.1]
01003488  .  33C0              XOR EAX,EAX
0100348A  .  85F6              TEST ESI,ESI
0100348C  .  0F95C0            SETNZ AL
0100348F  .  40                INC EAX
01003490  .  40                INC EAX
01003491  .  50                PUSH EAX                             Arg1
01003492  .  A3 60510001       MOV DWORD PTR DS:[1005160],EAX
01003497  .  E8 77F4FFFF       CALL 01002913                        winmine.01002913
0100349C  .  33C0              XOR EAX,EAX
0100349E  .  85F6              TEST ESI,ESI
010034A0  .  0F95C0            SETNZ AL
010034A3  .  8D0485 0A000000   LEA EAX,[EAX*4+0A]
010034AA  .  50                PUSH EAX                             Arg1
010034AB  .  E8 D0FAFFFF       CALL 01002F80                        winmine.01002F80
010034B0  .  85F6              TEST ESI,ESI
```



두번재 하드웨어 BP 구간 : 80 -> CC, 지뢰임을 판별하고 게임오버,
이후에 다른 지뢰를 보여주는 함수(0x01002F80)를 CALL!!!
-> 해당 함수를 내가 원할 때 CALL 할 수 있다면?

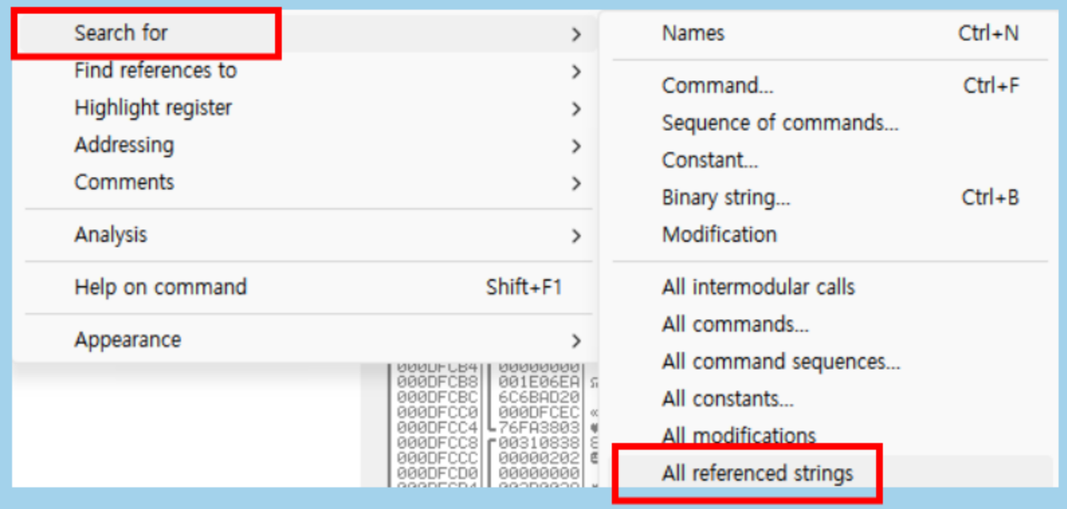
04 | 지뢰 찾기 : 지뢰 개수 조작

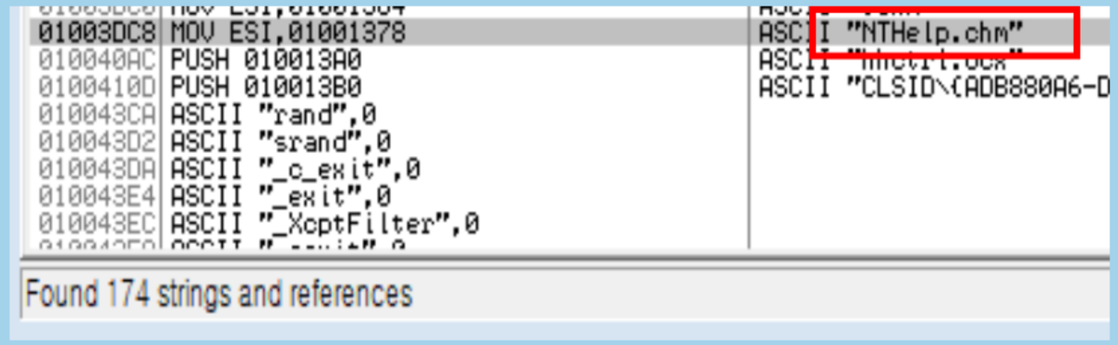
```
01002F80  r$  A1 38530001     MOV EAX,DWORD PTR DS:[1005338]
01002F85  .   83F8 01         CMP EAX,1
01002F88  .v  7C 4E           JL SHORT 01002FD8
01002F8A  .   53              PUSH EBX
01002F8B  .   56              PUSH ESI
01002F8C  .   8B35 34530001   MOV ESI,DWORD PTR DS:[1005334]
01002F92  .   57              PUSH EDI
01002F93  .   BF 60530001     MOV EDI,OFFSET 01005360
01002F98  .   8BD0            MOV EDX,EAX
01002F9A  >   33C9            XOR ECX,ECX
01002F9C  .   41              INC ECX
01002F9D  .   3BF1            CMP ESI,ECX
01002F9F  .v  7C 2E           JL SHORT 01002FCF
01002FA1  >   8A040F          MOV AL,BYTE PTR DS:[ECX+EDI]
01002FA4  .   A8 40           TEST AL,40
01002FA6  .v  75 22           JNZ SHORT 01002FCA
01002FA8  .   8AD8            MOV BL,AL
01002FAA  .   80E3 1F         AND BL,1F
01002FAD  .   84C0            TEST AL,AL
01002FAF  .v  79 0D           JNS SHORT 01002FBE
01002FB1  .   80FB 0E         CMP BL,0E
01002FB4  .v  74 14           JE SHORT 01002FCA
01002FB6  .   24 E0           AND AL,E0
01002FB8  .   0A4424 10       OR AL,BYTE PTR SS:[ARG.1]
01002FBC  .v  EB 09           JMP SHORT 01002FC7
01002FBE  >   80FB 0E         CMP BL,0E
01002FC1  .v  75 07           JNE SHORT 01002FCA
01002FC3  .   24 EB           AND AL,EB
01002FC5  .   0C 0B           OR AL,0B
01002FC7  >   88040F          MOV BYTE PTR DS:[ECX+EDI],AL
01002FCA  >   41              INC ECX
01002FCB  .   3BCE            CMP ECX,ESI
01002FCD  .^  7E D2           JLE SHORT 01002FA1
01002FCF  >   83C7 20         ADD EDI,20
01002FD2  .   4A              DEC EDX
01002FD3  .^  75 C5           JNZ SHORT 01002F9A
01002FD5  .   5F              POP EDI
01002FD6  .   5E              POP ESI
01002FD7  .   5B              POP EBX
01002FD8  >   E8 51F7FFFF     CALL 0100272E
01002FDD  L.  C2 0400         RETN 4
```



두번재 하드웨어 BP 구간 : 80 -> CC, 지뢰임을 판별하고 게임오버,
이후에 다른 지뢰를 보여주는 함수(0x01002F80)를 CALL!!!
-> 해당 함수를 내가 원할 때 CALL 할 수 있다면?

05 I 지뢰 찾기 : 지뢰 표시 조작





Search for

도움말 버튼을 클릭했을 때 실행되는 함수 부분에 코드 인젝션

해당 함수를 찾기 위해 NTHelp.chm 검색

05 | 지뢰 찾기 : 지뢰 표시 조작

```
01003D76  $  55              PUSH EBP                                winmine.01003D76(guessed Arg1,Arg
01003D77  .  8BEC            MOV EBP,ESP
01003D79  .  81EC FC000000   SUB ESP,0FC
01003D7F  .  66:837D 08 04   CMP WORD PTR SS:[ARG.1],4
01003D84  .  56              PUSH ESI
01003D85  .  57              PUSH EDI
01003D86  .v 74 40           JE SHORT 01003DC8
01003D88  .  68 FA000000     PUSH 0FA                                Count = 250.
01003D8D  .  8D85 04FFFFFF   LEA EAX,[LOCAL.63]
01003D93  .  50              PUSH EAX                                Buffer => OFFSET LOCAL.63
01003D94  .  FF35 305B0001   PUSH DWORD PTR DS:[1005B30]             hModule = 01000000 ('winmine')
01003D9A  .  FF15 88100001   CALL DWORD PTR DS:[<&KERNEL32.GetModule KERNEL32.GetModuleFileNameA
01003DA0  .  8DBC05 03FFFFFF LEA EDI,[EAX+EBP-0FD]
01003DA7  .  8BC7            MOV EAX,EDI
01003DA9  .  8D8D 04FFFFFF   LEA ECX,[LOCAL.63]
01003DAF  .  2BC1            SUB EAX,ECX
01003DB1  .  83F8 04         CMP EAX,4
01003DB4  .v 7E 0A           JLE SHORT 01003DC0
01003DB6  .  8D47 FD         LEA EAX,[EDI-3]
01003DB9  .  8038 2E         CMP BYTE PTR DS:[EAX],2E
01003DBC  .v 75 02           JNE SHORT 01003DC0
01003DBE  .  8BF8            MOV EDI,EAX
01003DC0  >  BE 84130001     MOV ESI,01001384                        ASCII ".chm"
01003DC5  .  A5              MOVS DWORD PTR ES:[EDI],DWORD PTR DS:[E
01003DC6  .v EB 0F           JMP SHORT 01003DD7
01003DC8  >  BE 78130001     MOV ESI,01001378                        ASCII "NTHelp.chm"
01003DCD  .  8DBD 04FFFFFF   LEA EDI,[LOCAL.63]
01003DD3  .  A5              MOVS DWORD PTR ES:[EDI],DWORD PTR DS:[E
01003DD4  .  A5              MOVS DWORD PTR ES:[EDI],DWORD PTR DS:[E
01003DD5  .  66:A5           MOVS WORD PTR ES:[EDI],WORD PTR DS:[ESI
01003DD7  >  6A 00           PUSH 0                                  Arg4 = 0
01003DD9  .  FF75 0C         PUSH DWORD PTR SS:[ARG.2]               Arg3 => [ARG.2]
01003DDC  .  8D85 04FFFFFF   LEA EAX,[LOCAL.63]
01003DE2  .  50              PUSH EAX                                Arg2 => OFFSET LOCAL.63
01003DE3  .  A4              MOVS BYTE PTR ES:[EDI],BYTE PTR DS:[ESI
01003DE4  .  FF15 B0100001   CALL DWORD PTR DS:[<&USER32.GetDesktopW USER32.GetDesktopWindow
01003DEA  .  50              PUSH EAX                                Arg1
01003DEB  .  E8 72020000     CALL 01004062                           winmine.01004062
01003DF0  .  5F              POP EDI
01003DF1  .  5E              POP ESI
01003DF2  .  C9              LEAVE
01003DF3  L  C2 0800         RETN 8
```

In Function Call

도움말 함수 시작 주소 : 0x01003D76
도움말 함수 리턴 주소 : 0x01003DF3

시작 주소에 파라미터 push
원하는 지뢰 보여주는 함수 CALL
이후 리턴 주소로 JMP
이때 도움말 함수의 루틴을 실행하지 않고 JMP 했으니
RETN8 -> RETN 으로 변경

05 | 지뢰 찾기 : 지뢰 표시 조작

```
01003D76    6A 0A           PUSH 0A                                 winmine.01003D76(guessed Arg1,Arg2)
01003D78    E8 03F2FFFF     CALL 01002F80
01003D7D  v EB 73           JMP SHORT 01003DF2
01003D7F  .  83.7D 08 04    CMP WORD PTR SS:[ARG.1],4
01003D84  .  56             PUSH ESI
01003D85  .  57             PUSH EDI
01003D86  .v 74 40          JE SHORT 01003DC8
01003D88  .  68 FA000000    PUSH 0FA                                ┌Count = 250.
01003D8D  .  8D85 04FFFFFF  LEA EAX,[LOCAL.63]                      │
01003D93  .  50             PUSH EAX                                │Buffer => OFFSET LOCAL.63
01003D94  .  FF35 305B0001  PUSH DWORD PTR DS:[1005B30]             │hModule = 01000000 ('winmine')
01003D9A  .  FF15 88100001  CALL DWORD PTR DS:[<&KERNEL32.GetModule└KERNEL32.GetModuleFileNameA
01003DA0  .  8DBC05 03FFFFFF LEA EDI,[EAX+EBP-0FD]
01003DA7  .  8BC7           MOV EAX,EDI
01003DA9  .  8D8D 04FFFFFF  LEA ECX,[LOCAL.63]
01003DAF  .  2BC1           SUB EAX,ECX
01003DB1  .  83F8 04        CMP EAX,4
01003DB4  .v 7E 0A          JLE SHORT 01003DC0
01003DB6  .  8D47 FD        LEA EAX,[EDI-3]
01003DB9  .  8038 2E        CMP BYTE PTR DS:[EAX],2E
01003DBC  .v 75 02          JNE SHORT 01003DC0
01003DBE  .  8BF8           MOV EDI,EAX
01003DC0  >  BE 84130001    MOV ESI,01001384                        ASCII ".chm"
01003DC5  .  A5             MOVS DWORD PTR ES:[EDI],DWORD PTR DS:[E
01003DC6  .v EB 0F          JMP SHORT 01003DD7
01003DC8  >  BE 78130001    MOV ESI,01001378                        ASCII "NTHelp.chm"
01003DCD  .  8DBD 04FFFFFF  LEA EDI,[LOCAL.63]
01003DD3  .  A5             MOVS DWORD PTR ES:[EDI],DWORD PTR DS:[E
01003DD4  .  A5             MOVS DWORD PTR ES:[EDI],DWORD PTR DS:[E
01003DD5  .  66:A5          MOVS WORD PTR ES:[EDI],WORD PTR DS:[ESI
01003DD7  >  6A 00          PUSH 0                                  ┌Arg4 = 0
01003DD9  .  FF75 0C        PUSH DWORD PTR SS:[ARG.2]               │Arg3 => [ARG.2]
01003DDC  .  8D85 04FFFFFF  LEA EAX,[LOCAL.63]                      │
01003DE2  .  50             PUSH EAX                                │Arg2 => OFFSET LOCAL.63
01003DE3  .  A4             MOVS BYTE PTR ES:[EDI],BYTE PTR DS:[ESI
01003DE4  .  FF15 B0100001  CALL DWORD PTR DS:[<&USER32.GetDesktopW │┌USER32.GetDesktopWindow
01003DEA  .  50             PUSH EAX                                ││Arg1
01003DEB  .  E8 72020000    CALL 01004062                           │└winmine.01004062
01003DF0  .  5F             POP EDI
01003DF1  .  5E             POP ESI
01003DF2  .  C9             LEAVE
01003DF3    C3              RETN
```

In Function Call

도움말 함수 시작 주소 : 0x01003D76
도움말 함수 리턴 주소 : 0x01003DF3

시작 주소에 파라미터 push
원하는 지뢰 보여주는 함수 CALL
이후 리턴 주소로 JMP
이때 도움말 함수의 루틴을 실행하지 않고 JMP 했으니
RETN8 -> RETN 으로 변경

05 I 지뢰 찾기 : 지뢰 표시 조작

In Function Call

실행후 도움말 클릭시 모습!!!