

```

shellcode1.cpp
shellcode1 (Global Scope)

#include "stdafx.h"
#include "windows.h"

int main(int argc, char* argv[])
{
    char buf[8] = { 'n','o','t','e','p','a','d','\x0' };
    WinExec(buf, SW_SHOW);
    ExitProcess(1);
}

```

```

0041178F mov     byte ptr [buf],6Eh
00411793 mov     byte ptr [ebp-0Fh],6Fh
00411797 mov     byte ptr [ebp-0Eh],74h
0041179B mov     byte ptr [ebp-0Dh],65h
0041179F mov     byte ptr [ebp-0Ch],70h
004117A3 mov     byte ptr [ebp-0Bh],61h
004117A7 mov     byte ptr [ebp-0Ah],64h
004117AB mov     byte ptr [ebp-9],0
        WinExec(buf, SW_SHOW);
004117AF mov     esi,esp
004117B1 push    5
004117B3 lea     eax,[buf]
004117B6 push    eax
004117B7 call    dword ptr [__imp__WinExec@8 (041B
004117BD cmp     esi,esp
004117BF call    __RTC_CheckEsp (0411235h)
        ExitProcess(1);
004117C4 mov     esi,esp
004117C6 push    1

```

다음과 같이 winexec() 함수를 이용해 메모장을 출력해주는 코드를 짰 후, 비주얼 스튜디오의 디버깅 기능을 이용해 디스어셈블리를 통해 어셈블리어를 추출한다.

그 후 디버깅을 통해 구한 exe 실행 파일을 이뮤니티 디버거를 통해 winexec와 exitptocess() 함수의 실제 주소를 구해 어셈블리어에 넣는다.

```

shellcode2
shellcode2 (Global Scope)

#include "stdafx.h"
#include "windows.h"

int main(int argc, char* argv[])
{
    __asm {
        // char buf[8] = { 'n','o','t','e','p','a','d','\x0' };
        mov byte ptr [ebp-8],6Eh
        mov byte ptr [ebp-7],6Fh
        mov byte ptr [ebp-6],74h
        mov byte ptr [ebp-5],65h
        mov byte ptr [ebp-4],70h
        mov byte ptr [ebp-3],61h
        mov byte ptr [ebp-2],64h
        mov byte ptr [ebp-1],0
        // xor ebx, ebx
        // mov[ebp - 1], ebx

        //WinExec(buf, SW_SHOW);
        push 5
        lea     eax, [ebp - 8]
        push    eax
        mov     eax, 0x7712F57E
        call    eax

        //ExitProcess(1);
        push 1
        mov     eax, 0x770F8ED2
        call    eax
    }
}

```

```

        mov byte ptr [ebp-4],70h
C6 45 FC 70 mov     byte ptr [ebp-3],61h
C6 45 FD 61 mov     byte ptr [ebp-2],64h
C6 45 FE 64 mov     byte ptr [ebp-1],0
C6 45 FF 00 mov     byte ptr [ebp-1],0
        // xor ebx, ebx
        // mov[ebp - 1], ebx

        //WinExec(buf, SW_SHOW);
        push 5
6A 05      push    5
8D 45 F8   lea     eax, [ebp - 8]
        lea     eax, [ebp-8]
50        push    eax
        push    eax
        mov     eax, 0x7712F57E
B8 7E F5 12 77 mov     eax, 7712F57Eh
        call    eax
FF D0     call    eax

        //ExitProcess(1);
        push 1

```

위에서 구한 어셈블리어를 브레이크 포인트를 걸고 디버깅 해서 기계어 코드를 얻는다.

