리버싱 핵심원리 따라잡기

이지훈

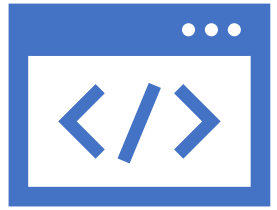
목차

리버싱 핵심원리

About this book

Reverse Engineering

프로그램 설치



Hello World!

예제

Next week

리버싱 핵심원리

이승원 지음

AhnLab에서 악성코드 분석가로 활동하며 얻은 지식들을 바탕으로 리버싱에 대해 소개한 책

Reverse Engineering

RE : 역공학. 물건이나 기계장치, 혹은 시스템 등의 구조, 기능, 동작 등을 분석하여
그 원리를 이해하며 단점을 보완하고 새로운 아이디어를 추가하는 일련의 작업

RCE (Reverse Code Engineering) : 소프트웨어 분야의 역공학. 주로 실행 파일의
분석을 진행하며, 정적 분석과 동적 분석으로 나뉜다.

Reverse Engineering

정적 분석 : 파일을 실행하지 않고 겉모습을 관찰하여 분석하는 방법
: 파일의 종류, 크기, 헤더정보, Import/Export API, 내부 문자열, 등록 정보

동적 분석 : 정적 분석에서 얻은 정보를 바탕으로 파일을 직접 실행하여 분석하는 방법
: 파일 실행, 디버깅, 레지스트리, 네트워크 등을 관찰

Reverse Engineering

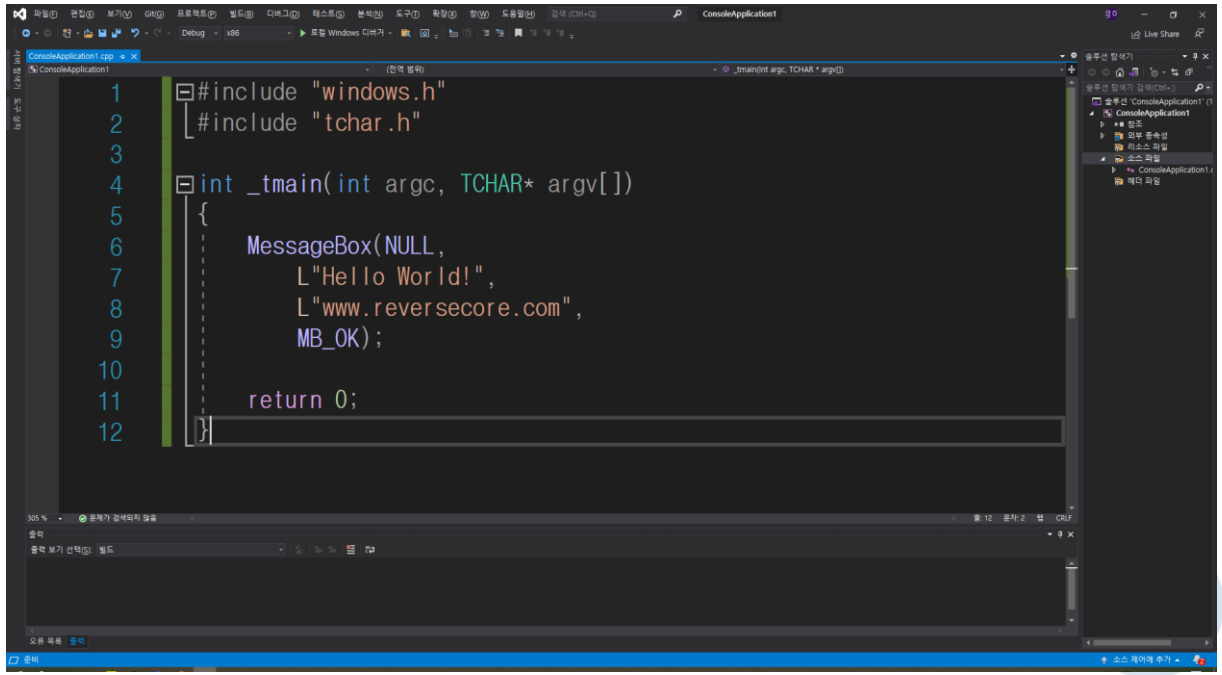
패치(Patch) : 프로그램의 파일 혹은 실행 중인 프로세스 메모리의 내용을 변경하는 작업
Ex. Windows 업데이트

크랙(Crack) : 패치의 의도가 비합법적이고 비도덕적인 경우
Ex. 불법 복제 및 사용

Source Code
Visual Studio

개발 도구에서 소스코드를 빌드하여 실행 파일을 생성

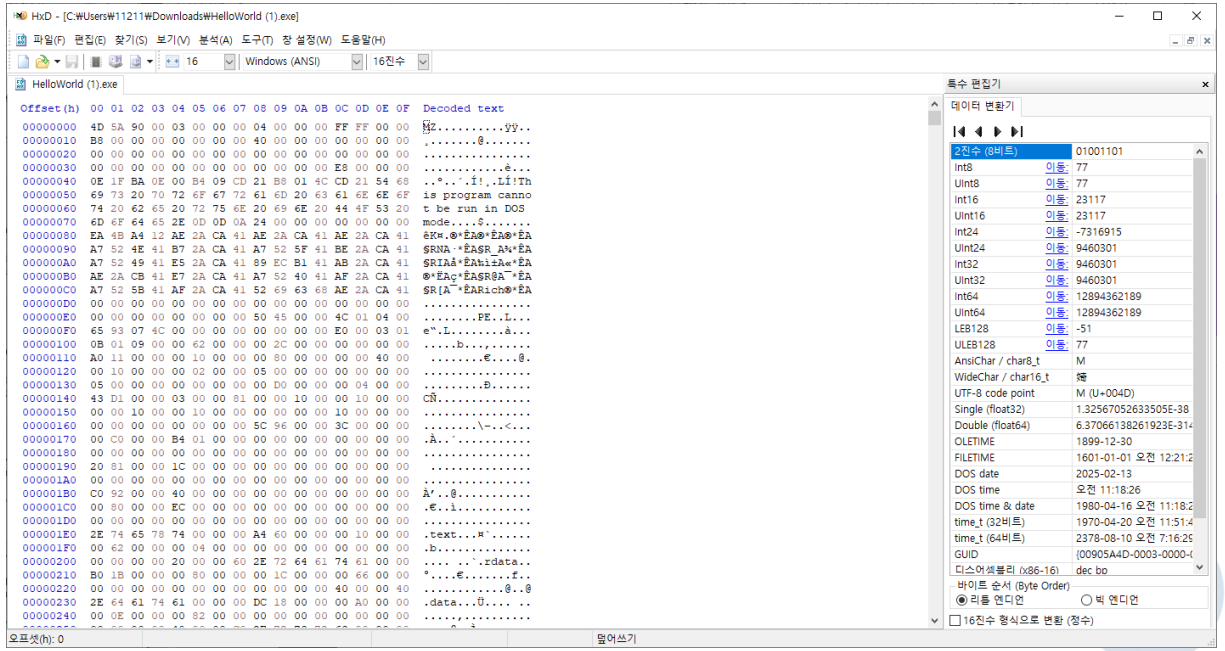
Ex. Visual C++ -> .cpp -> .exe



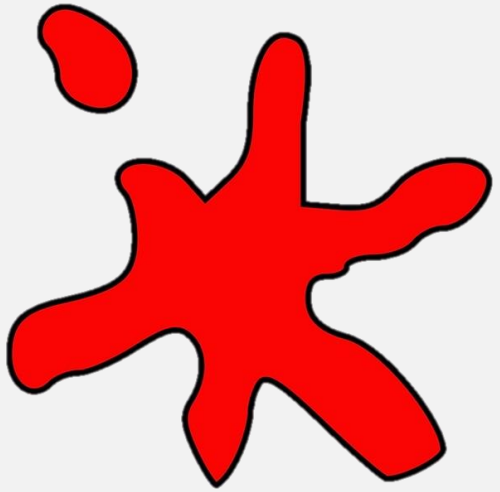
Hex Code
HxD Hex Editor

실행파일은 2진수(Binary) 형식으로 되어있다.

2진수를 16진수(Hex) 형식으로 변환시켜주는 프로그램

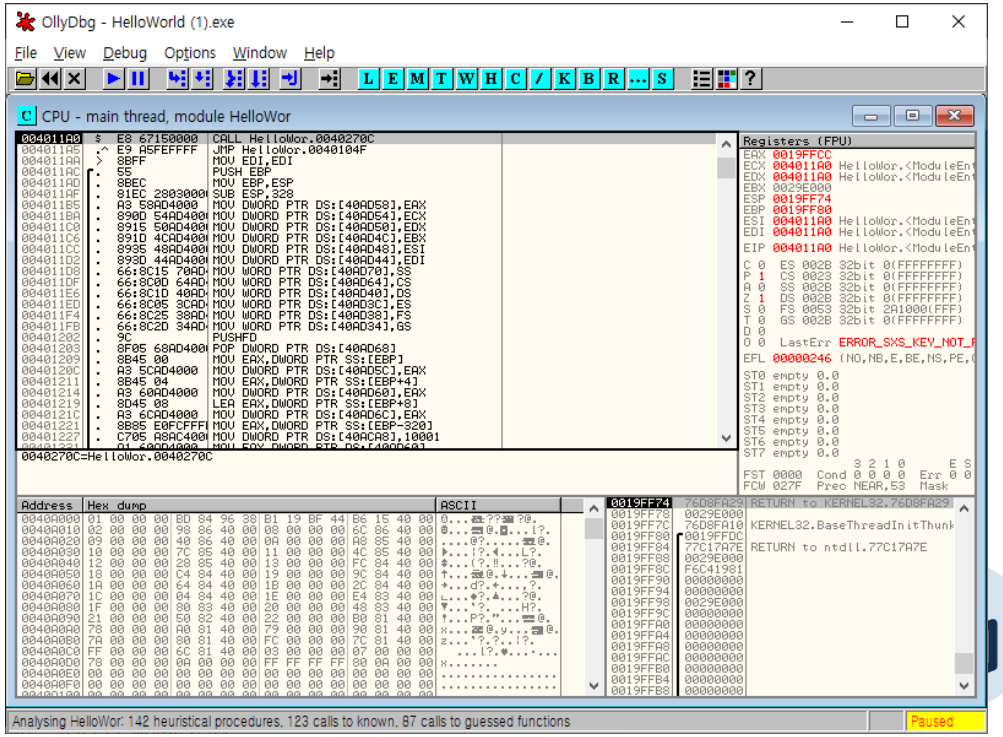


Assembly Code
Olly Dbg

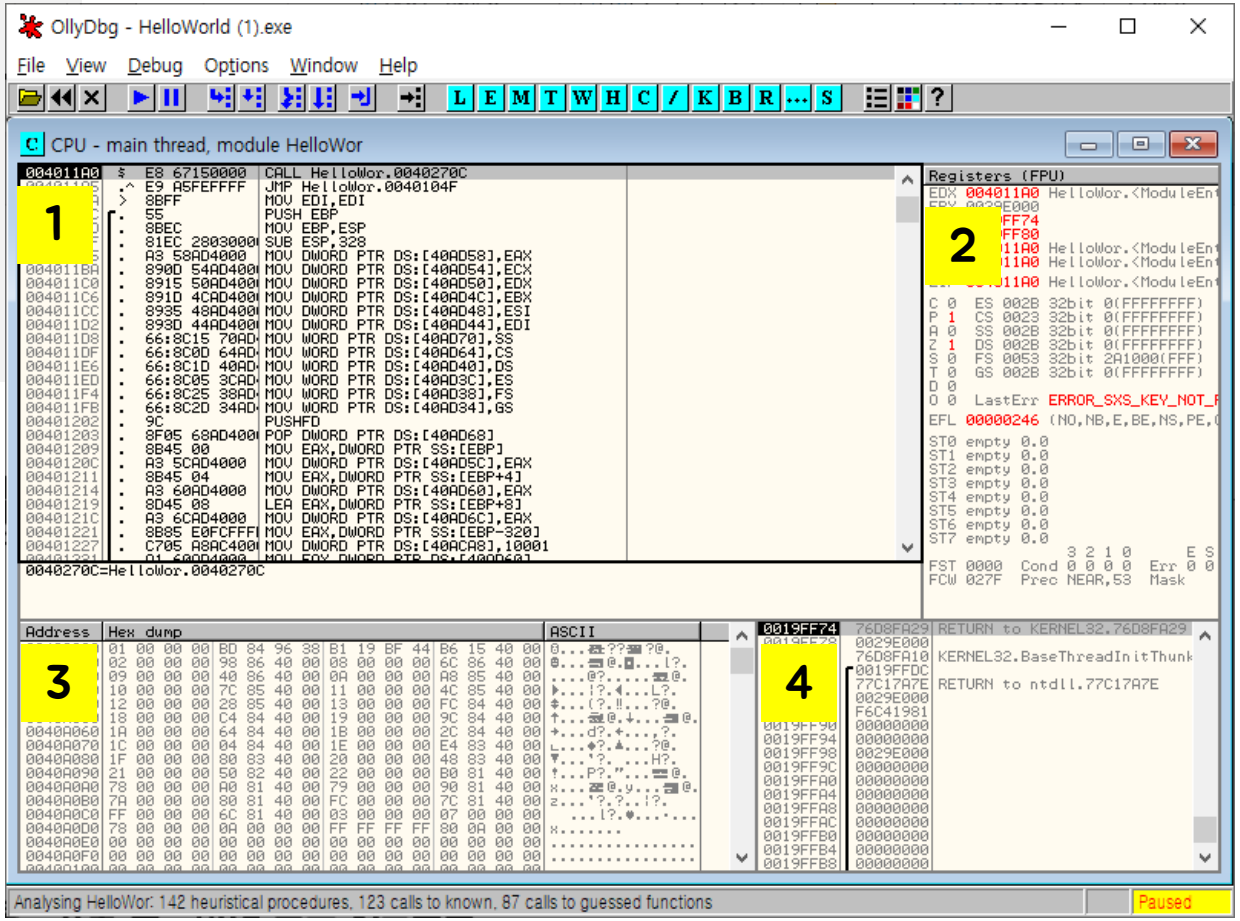


헥스 코드를 디스어셈블(Disassemble)하여 어셈블리 코드로 변환

일반적인 리버싱 과정에서 이러한 어셈블리 코드를 분석한다.



HelloWorld.exe



1. Code Window : 디스어셈블리 코드를 표시해주는 구역
   loop, jump 위치 등의 정보를 표시

2. Register Window : 레지스터 값을 표시, 수정 가능

3. Dump Window : 메모리 주소 위치를 Hex, ASCII,
   UNICODE 값으로 표시, 수정 가능

4. Stack Window : ESP Register가 가리키는
   스택 메모리를 실시간으로 표시, 수정 가능

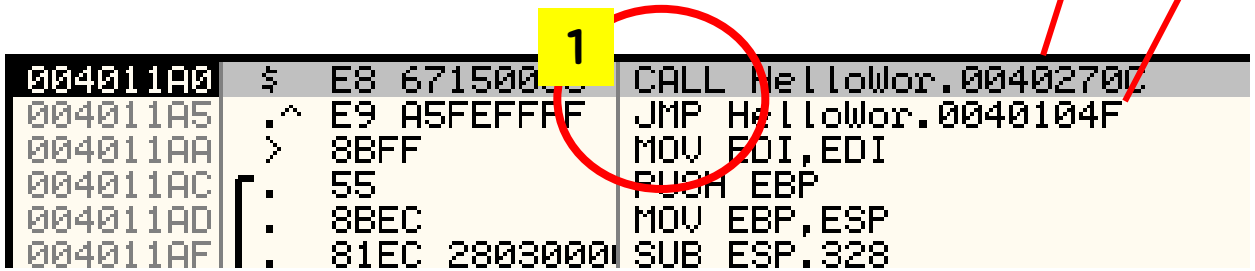
HelloWorld.exe

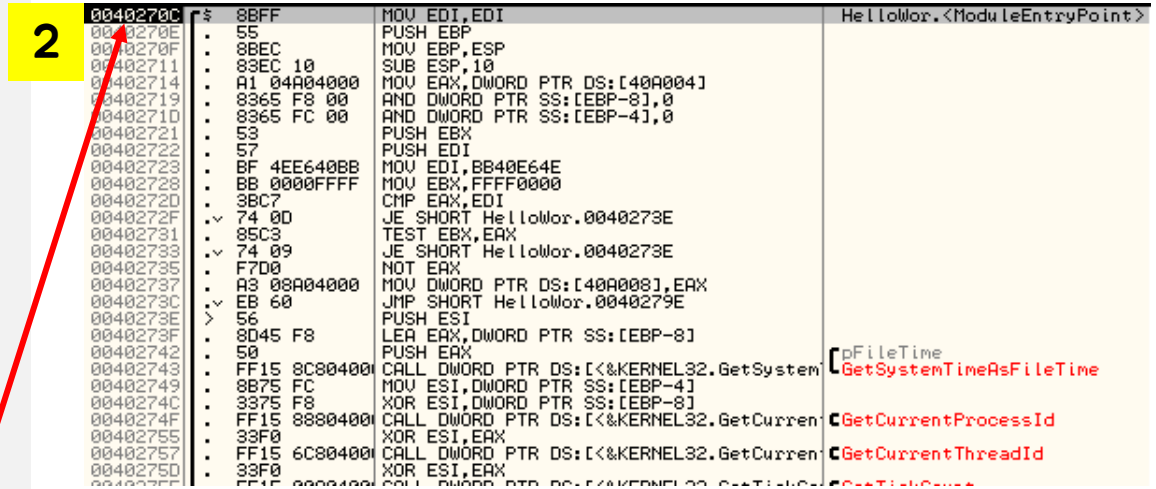
```
1   #include "windows.h"
2   #include "tchar.h"
3
4   int _tmain(int argc, TCHAR* argv[])
5   {
6       MessageBox(NULL,
7           L"Hello World!",
8           L"www.reversecore.com",
9           MB_OK);
10
11      return 0;
12  }
```

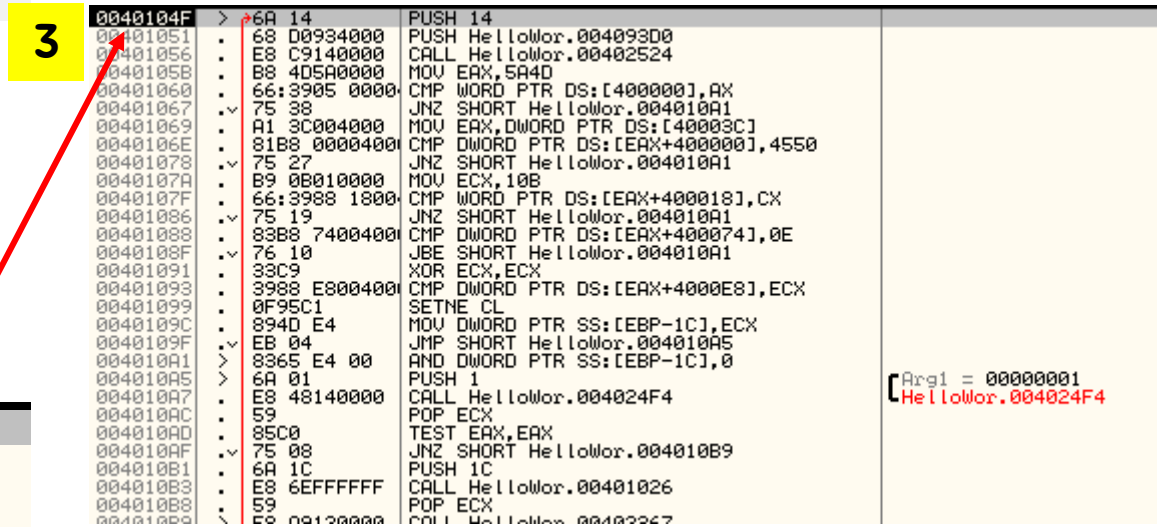
HelloWorld.exe

1. EP(EntryPoint) : 실행 시작 주소 (004011A0)

2. 0040270C 주소의 함수를 호출(CALL)
3. 0040104F 주소로 점프(JMP)







HelloWorld.exe

```
00401051   .   68 D0934000      PUSH HelloWor.004093D0
00401056   .   E8 C9140000      CALL HelloWor.00402524
0040105B   .   B8 4D5A0000      MOV EAX,5A4D
00401060   .   66:3905 0000     CMP WORD PTR DS:[400000],AX
```

```
00402524   $   68 80254000      PUSH HelloWor.00402580
00402529   .   64:FF35 0000     PUSH DWORD PTR FS:[0]
00402530   .   8B4424 10        MOV EAX,DWORD PTR SS:[ESP+10]
00402534   .   896C24 10        MOV DWORD PTR SS:[ESP+10],EBP
00402538   .   8D6C24 10        LEA EBP,DWORD PTR SS:[ESP+10]
0040253C   .   2BE0             SUB ESP,EAX
0040253E   .   53               PUSH EBX
0040253F   .   56               PUSH ESI
00402540   .   57               PUSH EDI
00402541   .   A1 04A04000      MOV EAX,DWORD PTR DS:[40A004]
00402546   .   3145 FC          XOR DWORD PTR SS:[EBP-4],EAX
00402549   .   33C5             XOR EAX,EBP
0040254B   .   50               PUSH EAX
0040254C   .   8965 E8          MOV DWORD PTR SS:[EBP-18],ESP
0040254F   .   FF75 F8          PUSH DWORD PTR SS:[EBP-8]
00402552   .   8B45 FC          MOV EAX,DWORD PTR SS:[EBP-4]
00402555   .   C745 FC FEFFI    MOV DWORD PTR SS:[EBP-4],-2
0040255C   .   8945 F8          MOV DWORD PTR SS:[EBP-8],EAX
0040255F   .   8D45 F0          LEA EAX,DWORD PTR SS:[EBP-10]
00402562   .   64:A3 000000     MOV DWORD PTR FS:[0],EAX
00402568   .   C3               RETN
```

계속 Step Into / Step Over 기능을 사용하여 함수 호출 명령어를 따라가고 있음

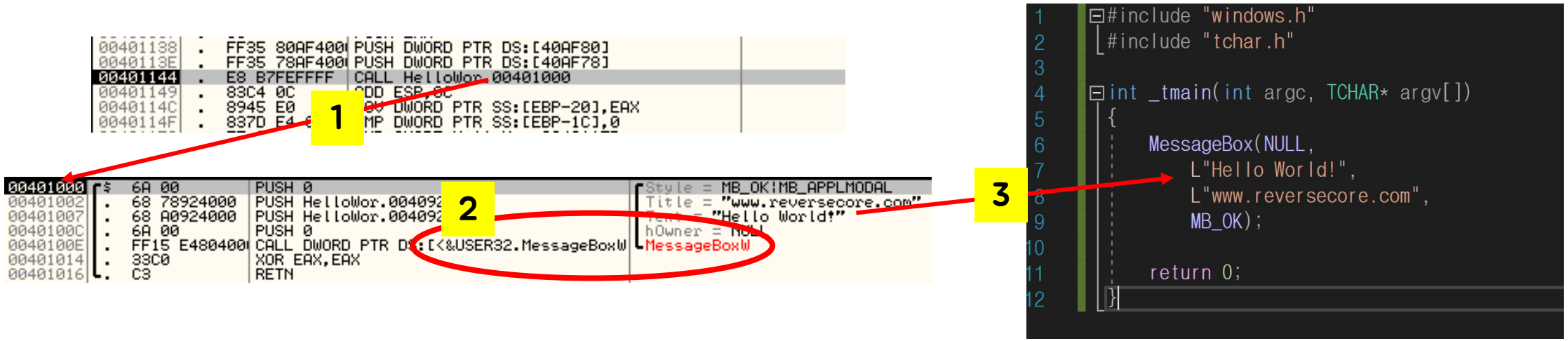
HelloWorld.exe

1. CALL 00401000 명령어를 따라가니

2. MessageBox API를 호출하는 코드를 찾았다

3. 문자열을 비교해보면 일치하는 것을 확인

따라서 00401000 함수가 _tmain 함수이다.



Next Week

Hello World 문자열 패치

Or

Beacon Flooding 과제 리뷰..

Q & A