Message cần chèn: 2052197-20522016-20521862

**Bước 1:** Dùng HxD để mở file NOTEPAD.exe và tạo thêm 1000 bytes vùng nhớ ở cuối file.

**Bước 2:** Tạo chương trình để chèn MSSV và biên dịch chương trình ở chế độ Release, Not Using Precompiled Headers

Dùng IDA Pro để mở file và xen hợp ngữ của chương trình vừa biên dịch

Về cơ bản chương trình gồm 5 dòng lệnh

push 0 ; 6a 00

push Caption ; 68 X

push Text ; 68 Y

push 0 ; 6a 00

call [MessageBoxW] ; ff15 Z

Với Z là địa chỉ của hàm MessageBoxW được import từ thư viện USER32.dll. Trong IDA Pro, mở Notepad.exe, chọn View -> Open Subviews -> Imports và ta thấy địa chỉ của hàm MessageBoxW chính là **Z = 0x01001268**

Trong HxD, ta chọn

- địa chỉ **0x00011000** trong vùng nhớ đã được mở rộ̣ng để lưu trữ mã hợp ngữ

- địa chỉ **0x00011040** để lưu trữ Caption

- địa chỉ **0x00011060** để lưu trữ text

Ta dùng công thức:

**Offset = RA - Section RA = VA - Section VA (1)**

Dùng CFF Explore mở file NOTEPAD.exe để xem section VA và section RA

+ Section RA = 0x00008400

+ Section VA = 0x0000B000

Áp dụng công thức 1 tính địa chỉ caption và text (X và Y)

00011040 - 00008400 = X - 0000B000 => X = 0x00013C40

Cộng thêm ImageBase ta có **X = 0x01013C40**

Tương tự **Y = 0x01013C60**

Như vậy đoạn code thực hiện chức năng mà ta mong đợi và có địa chỉ mới là:

**New\_entry\_point** = 0x00011000 – 0x00008400 + 0x000B000 = **0x00013C00**

**Bước 3:** Thiết lập lệnh quay về AddressOfEntryPoint ban đầu

Ta có công thức:

**Old\_emtry\_point = jmp\_instruction\_VA + 5 + relative\_VA (2)**

Nếu đât lệnh jump sau 5 câu lệnh ở bước 2 thì

j**mp\_instruction\_VA = 0x010113C14**

**old\_entry\_point = 0x0100739D** chính là giá trị AddressOfEntryPoint ban đàu đã cong ImageBase

=> Relative\_VA để chèn vào sau câu lệnh ta muốn thực thi để trở về old\_entry\_point

Suy ra, relative\_VA = 0x0100739D – 5 – 0x01013C14 = 0xFFFF3784

Đển đây ta đã có một đoạn hợp ngữ hoàn chỉnh, các địa chỉ được biểu diễn theo thứ tự little endian:

push 0 ; 6a 00

push Caption ; 68 403C0101

push Text ; 68 603C0101

push 0 ; 6a 00

call [MessageBoxW] ; ff15 68120001

mp Origianl\_Entry\_Point ; e9 8437FFFF

**Bước 4:** Dùng HxD để chèn đoạn mã cùng với caption và text vào notepad.exe

Lưu lại file

6A 00 68 40 3C 01 01 68 60 3C 01 01 6A 00 FF 15 68 12 00 01 E9 84 37 FF FF

49 00 6E 00 66 00 6F 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 32 00 30 00 35 00 32 00 31 00 39 00 37 00 38 00 2D 00 32 00 30 00 35 00 32 00 32 2E 30 00 31 00 36 00 2D 00 32 00 30 00 35 00 32 00 31 00 38 00 35 00 36

**Bước 5:** Dùng CFF Explore để chỉnh sửa các giá trị VituarSize, RawSize vì ta đã insert vào cuối file notepad.exe

Trong Optional Headers, tăng SizeOfImage le n 0x1000.

Trong Optional Headers, chỉnh sử a AddressOfEntryPoint thành 0x00013C00

BÀI 5

a, NOTEPAD.exe

***Bước 1:***

Ta có đoạn hợp ngữ cần chèn như sau:

push 0 ; 6a 00

push Caption ; 68 X

push Text ; 68 Y

push 0 ; 6a 00

call [MessageBoxW] ; ff15 Z

Z chính là địa chỉ của hàm MessageBoxW, dùng IDA Pro để hàm MessageBoxW trong notepad.exe

=> **Z = 0x01001268**

***Bước 2:***

Trong HxD, ta chọn:

+ 0x00010D40 để lưu mã hợp ngữ

+ 0x00010D60 để lưu trữ caption

+ 0x00010DA0 để lưu trữ text

Trong CFF ta có:

Section RA = 0x00008400

Section VA = 0x0000B000

RA - Section RA = VA - Section VA

Tìm X và Y từ công thức trênn

0x00010D60- 0x00008400= X - 0x0000B000

=> X = 0x00013960

Cộng thêm ImageBase, **X = 0x01013960**

Tương tự **Y = 0x010139A0**

Như vậy đoạn code thực hiện chức năng như mong đợi và có địa chỉ mới là:

**new\_entry\_point =** 0x00010D40 **-** 0x00008400 **+** 0x0000B000**= 0x00013940**

***Bước 3:***

Thiết lập lệnh quay về AddressOfEntryPoint ban đầu

**Old\_emtry\_point = jmp\_instruction\_VA + 5 + relative\_VA**

Nếu đặt lệnh Jump sau 5 câu lệnh ở bước 2 thì:

jmp\_intrustion\_VA = 0x00013940+ 0x14 = **0x00013954**

old\_entry\_point = AddressOfEntryPoint + ImageBase = 0x0000739D + 0x01000000 = 0x0100739D

=> **relative\_VA = old\_entry\_point - jmp\_intrustion\_VA - 5 = 0xFFFF3A44**

Đoạn mã hợp ngữ hoàn chỉnh:

push 0 ; 6a 00

push Caption ; 68 60390101

push Text ; 68 A0390101

push 0 ; 6a 00

call [MessageBoxW] ; ff15 68120001

mp Origianl\_Entry\_Point ; e9 443AFFFF

6A 00 68 60 39 01 01 68 A0 39 01 01 6A 00 FF 15 68 12 00 01 E9 44 3A FF FF

49 00 6E 00 66 00 6F

32 00 30 00 35 00 32 00 31 00 39 00 37 00 38 00 2D 00 32 00 30 00 35 00 32 00 32 00 30 00 31 00 36 00 2D 00 32 00 30 00 35 00 32 00 31 00 38 00 35 00 36

***Bước 4:***

Dùng HxD để chèn mã Hex vào file calc.exe

c/ calc,exe

***Bước 1:***

Ta có đoạn hợp ngữ cần chèn như sau:

push 0 ; 6a 00

push Caption ; 68 X

push Text ; 68 Y

push 0 ; 6a 00

call [MessageBoxW] ; ff15 Z

Z chính là địa chỉ của hàm MessageBoxW, dùng IDA Pro để hàm MessageBoxW trong calc.exe

=> **Z = 0x010011A8**

***Bước 2:***

Trong HxD, ta chọn:

+ 0x0001BF50 để lưu mã hợp ngữ

+ 0x0001BF70 để lưu trữ caption

+ 0x0001BFB0 để lưu trữ text

Trong CFF ta có:

Section RA = 0x00013600

Section VA = 0x00016000

RA - Section RA = VA - Section VA

Tìm X và Y từ công thức trênn

0x0001BF70 - 0x00013600 = X - 0x00016000

=> X = 0x0001E970

Cộng thêm ImageBase, **X = 0x0101E970**

Tương tự **Y = 0x0101E9B0**

Như vậy đoạn code thực hiện chức năng như mong đợi và có địa chỉ mới là:

**new\_entry\_point = 0x0001BF50 - 0x00013600 + 0x00016000 = 0x0001E950**

***Bước 3:***

Thiết lập lệnh quay về AddressOfEntryPoint ban đầu

Nếu đặt lệnh Jump sau 5 câu lệnh ở bước 2 thì:

jmp\_intrustion\_VA = 0x0101E950 + 0x14 = 0x0101E964

old\_entry\_point = AddressOfEntryPoint + ImageBase = 0x00012475 + 0x01000000 = 0x01012475

=> **relative\_VA = old\_entry\_point - jmp\_intrustion\_VA - 5 = 0xFFFF3B0C**

Đoạn mã hợp ngữ hoàn chỉnh:

push 0 ; 6a 00

push Caption ; 68 70E90101

push Text ; 68 B0E90101

push 0 ; 6a 00

call [MessageBoxW] ; ff15 A8110001

mp Origianl\_Entry\_Point ; e9 0C3BFFFF

6A 00 68 70 E9 01 01 68 B0 E9 01 01 6A 00 FF 15 A8 11 00 01 E9 0C 3B FF FF

49 00 6E 00 66 00 6F

32 00 30 00 35 00 32 00 31 00 39 00 37 00 38 00 2D 00 32 00 30 00 35 00 32 00 32 00 30 00 31 00 36 00 2D 00 32 00 30 00 35 00 32 00 31 00 38 00 35 00 36

***Bước 4:***

Dùng HxD để chèn mã Hex vào file calc.exe