

Môn học: CƠ CHẾ HOẠT ĐỘNG CỦA MÃ ĐỘC  
Bài 1: Virus đơn giản trên tập tin PE

GVHD: Phan Thế Duy

**Nhóm: Binary**

**1. THÔNG TIN CHUNG:**

Lớp: NT230.L21.ANTT

STT	Họ và tên	MSSV	Email
1	Nguyễn Trần Hùng Vĩ	18520191	18520191@gm.uit.edu.vn
2	Trương Tấn Sang	18521336	18521336@gm.uit.edu.vn
3	Huỳnh Gia Huy	18520829	18520829@gm.uit.edu.vn

## Mục Lục

<b>Bài tập 01:</b>	2
a. Viết 1 chương trình bằng ngôn ngữ Python có khả năng thêm mới một Section trong file PE. Kiểm tra lại chức năng của file PE sau khi thao tác để đảm bảo chương trình hoạt động bình thường	2
b. Viết một virus máy tính đơn giản bằng cách thực hiện thao tác chèn 1 shellcode đơn giản (hiển thị Popup chào mừng có nội dung: “MSSV1-MSSV2-MSSV3” – và thanh tiêu đề của popup có nội dung là: “NT230”) vào phần Section mới thêm & thay đổi Entry Point (để đảm bảo chức năng chương trình gốc hoạt động bình thường). Kiểm tra lại chức năng sau khi thao tác (nếu có lỗi lúc chạy, thử tìm lí do). Quan sát kết quả thực thi và giải thích.	3
c. Thực nhiệm lây nhiễm tất cả các file PE khác trong cùng một thư mục khi tập tin virus ở câu b được thực thi (mang file virus sang một thư mục khác bất kỳ và mở lên).	7

# BÁO CÁO CHI TIẾT

## Bài tập 01:

a. Viết 1 chương trình bằng ngôn ngữ Python có khả năng thêm mới một Section, xóa một Section trong file PE. Kiểm tra lại chức năng của file PE sau khi thao tác để đảm bảo chương trình hoạt động bình thường.

(Giải thích code được trình bày qua comments )

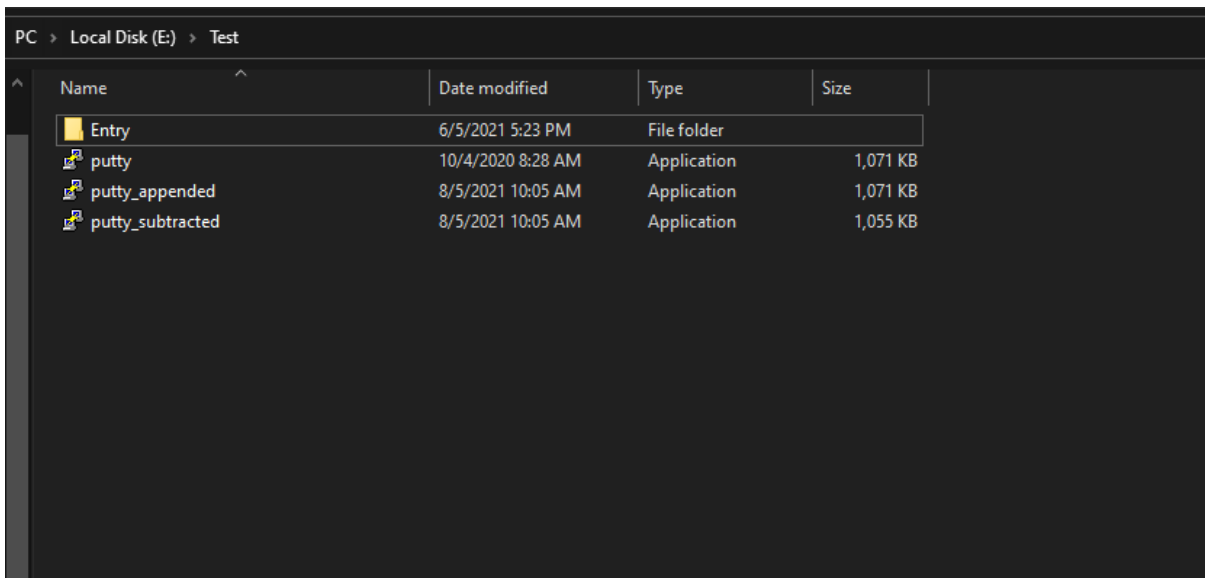
Triển khai:

Tạo 2 function có chức năng lần lượt là

- "Push": ý tưởng là dựa vào các trường *offset* và *size* của section và section header cuối cùng, từ đó tạo đó tính ra vị trí thích hợp để nối thêm section
- "pop": xóa section và data của section thông qua các property được cung cấp của pefile object

Kết quả

- Chức năng của chương trình vẫn được đảm bảo. t
- Kích thước: (H. A1)
  - Putty: 1071 KB (putty)
  - Putty thêm section: 1071 KB (putty\_appended)
  - Putty xóa section: 1055 KB (putty\_substracted)
- Kiểm tra section từ PEStudio:
  - Putty (H. A2)
  - Putty thêm section (H. A3)
  - Putty xóa section (H. A4)



Name	Date modified	Type	Size
Entry	6/5/2021 5:23 PM	File folder	
putty	10/4/2020 8:28 AM	Application	1,071 KB
putty_appended	8/5/2021 10:05 AM	Application	1,071 KB
putty_substracted	8/5/2021 10:05 AM	Application	1,055 KB

Hình. A.1

property	value	value	value	value	value	value	value
name	.text	.rdata	.data	.00cf	.gids	.rsrc	.reloc
md5	C2A8F0F2DF948E32017DA4E...	65B60216C5CCCA9452DD0D...	B4B7EC6B6BF4DFE41972C4B...	3DF28295FCF4F22F0619DBB...	C32633F1600732D51CD8162...	D2AE732E2833BEFE28CDEBF...	3F538E9CEE3F455A7E538BE3...
entropy	6.632	5.799	2.139	0.061	1.916	7.828	6.722
file-ratio (98.47%)	52.88 %	14.53 %	0.23 %	0.05 %	0.05 %	28.07 %	2.66 %
raw-address	0x0000400	0x0008DC00	0x000B4A00	0x000B5400	0x000B5600	0x000B5800	0x00100A00
raw-size (1079296 bytes)	0x0008D800 (579584 bytes)	0x00026E00 (159232 bytes)	0x00000A00 (2560 bytes)	0x00000200 (512 bytes)	0x00000200 (512 bytes)	0x0004B200 (307712 bytes)	0x00007200 (29184 bytes)
virtual-address	0x00401000	0x0048F000	0x004B6000	0x004B8000	0x004BC000	0x004BD000	0x00509000
virtual-size (1094462 bytes)	0x0008D65E (579166 bytes)	0x00026CAC (158892 bytes)	0x00004E30 (20016 bytes)	0x00000004 (4 bytes)	0x000000B4 (180 bytes)	0x0004B030 (307248 bytes)	0x0000711C (28956 bytes)
entry-point	0x0006FE96	-	-	-	-	-	-
characteristics	0x60000020	0x40000040	0xC0000040	0x40000040	0x40000040	0x40000040	0x40000040
writable	-	-	x	-	-	-	-
executable	x	-	-	-	-	-	-
shareable	-	-	-	-	-	-	-
discardable	-	-	-	-	-	-	x
initialized-data	-	x	x	x	x	x	x
uninitialized-data	-	-	-	-	-	-	-
unreadable	-	-	-	-	-	-	-
self-modifying	-	-	-	-	-	-	-
virtualized	-	-	-	-	-	-	-
file	-	-	-	-	-	Compiled-HTML, offset: 0x0...	-

Hình. A2

property	value	value	value	value	value	value	value
name	.text	.rdata	.data	.00cf	.gids	.rsrc	.reloc
md5	C2A8F0F2DF948E32...	65B60216C5CCCA9452DD0D...	B4B7EC6B6BF4DFE41972C4B...	3DF28295FCF4F22F0619DBB...	C32633F1600732D51...	D2AE732E2833BEFE28...	3F538E9CEE3F455A7E...
entropy	6.632	5.799	2.139	0.061	1.916	7.828	6.722
file-ratio (98.84%)	52.88 %	14.53 %	0.23 %	0.05 %	0.05 %	28.07 %	2.66 %
raw-address	0x0000400	0x0008DC00	0x000B4A00	0x000B5400	0x000B5600	0x000B5800	0x00100A00
raw-size (1083392 bytes)	0x0008D800 (579584 bytes)	0x00026E00 (159232 bytes)	0x00000A00 (2560 bytes)	0x00000200 (512 bytes)	0x00000200 (512 bytes)	0x0004B200 (307712 bytes)	0x00007200 (29184 bytes)
virtual-address	0x00401000	0x0048F000	0x004B6000	0x004B8000	0x004BC000	0x004BD000	0x00509000
virtual-size (1098558 bytes)	0x0008D65E (579166 bytes)	0x00026CAC (158892 bytes)	0x00004E30 (20016 bytes)	0x00000004 (4 bytes)	0x000000B4 (180 bytes)	0x0004B030 (307248 bytes)	0x0000711C (28956 bytes)
entry-point	0x0006FE96	-	-	-	-	-	-
characteristics	0x60000020	0x40000040	0xC0000040	0x40000040	0x40000040	0x40000040	0xE0000020
writable	-	-	x	-	-	-	x
executable	x	-	-	-	-	-	x
shareable	-	-	-	-	-	-	-
discardable	-	-	-	-	-	-	-
initialized-data	-	x	x	x	x	x	-
uninitialized-data	-	-	-	-	-	-	-
unreadable	-	-	-	-	-	-	-
self-modifying	-	-	-	-	-	-	x
virtualized	-	-	-	-	-	-	-
file	-	-	-	-	-	Compiled-HTML, offs...	-

Hình. A3

property	value	value	value	value	value	value	value
name	.text	.rdata	.data	.00cf	.gids	.rsrc	.reloc
md5	C2A8F0F2DF948E32017DA4E...	65B60216C5CCCA9452DD0D...	B4B7EC6B6BF4DFE41972C4B...	3DF28295FCF4F22F0619DBB...	C32633F1600732D51CD8162...	D2AE732E2833BEFE28CDEBF...	3F538E9CEE3F455A7E...
entropy	6.632	5.799	2.139	0.061	1.916	7.828	6.722
file-ratio (97.22%)	53.66 %	14.74 %	0.24 %	0.05 %	0.05 %	28.49 %	2.66 %
raw-address	0x0000400	0x0008DC00	0x000B4A00	0x000B5400	0x000B5600	0x000B5800	0x00100A00
raw-size (1050112 bytes)	0x0008D800 (579584 bytes)	0x00026E00 (159232 bytes)	0x00000A00 (2560 bytes)	0x00000200 (512 bytes)	0x00000200 (512 bytes)	0x0004B200 (307712 bytes)	0x00007200 (29184 bytes)
virtual-address	0x00401000	0x0048F000	0x004B6000	0x004B8000	0x004BC000	0x004BD000	0x00509000
virtual-size (1065506 bytes)	0x0008D65E (579166 bytes)	0x00026CAC (158892 bytes)	0x00004E30 (20016 bytes)	0x00000004 (4 bytes)	0x000000B4 (180 bytes)	0x0004B030 (307248 bytes)	0x0000711C (28956 bytes)
entry-point	0x0006FE96	-	-	-	-	-	-
characteristics	0x60000020	0x40000040	0xC0000040	0x40000040	0x40000040	0x40000040	0x40000040
writable	-	-	x	-	-	-	-
executable	x	-	-	-	-	-	-
shareable	-	-	-	-	-	-	-
discardable	-	-	-	-	-	-	-
initialized-data	-	x	x	x	x	x	x
uninitialized-data	-	-	-	-	-	-	-
unreadable	-	-	-	-	-	-	-
self-modifying	-	-	-	-	-	-	-
virtualized	-	-	-	-	-	-	-
file	-	-	-	-	-	Compiled-HTML, offset: 0x0...	-

Hình. A4

**b. Viết một virus máy tính đơn giản bằng cách thực hiện thao tác chèn 1 shellcode đơn giản (hiển thị Popup chào mừng có nội dung: “MSSV1-MSSV2-MSSV3” – và thanh tiêu đề của popup có nội dung là: “NT230”) vào phần Section mới thêm & thay đổi Entry Point (để đảm bảo chức năng chương trình gốc hoạt động bình thường). Kiểm tra lại chức năng sau khi thao tác (nếu có lỗi lúc chạy, thử tìm lí do). Quan sát kết quả thực thi và giải thích.**

Bước 1: Tạo shellcode sử dụng Metasploit (H. B1)

Dùng payload windows/messagebox

- Sử dụng câu lệnh generate (-b để chỉ định bad characters không sử dụng trong shellcode)
- Set các options:
  - EXITFUNC = thread/process
  - TEXT = "18520191-18521336-1852-209"
  - TITLE = NT230

```
msf6 payload(windows/messagebox) > generate -b "\x00"
# windows/messagebox - 278 bytes
# https://metasploit.com/
# VERBOSE=false, PrependMigrate=false, EXITFUNC=thread,
# TITLE=NT230, TEXT=18520191-18521336-18520829,
# ICON=INFORMATION
buf =
"\xd9\xeb\x9b\xd9\x74\x24\xf4\x31\xd2\xb2\x77\x31\xc9\x64" +
"\x8b\x71\x30\x8b\x76\x0c\x8b\x76\x1c\x8b\x46\x08\x8b\x7e" +
"\x20\x8b\x36\x38\x4f\x18\x75\xf3\x59\x01\xd1\xff\xe1\x60" +
"\x8b\x6c\x24\x24\x8b\x45\x3c\x8b\x54\x28\x78\x01\xe8\x8b" +
"\x4a\x18\x8b\x5a\x20\x01\xeb\xe3\x34\x49\x8b\x34\x8b\x01" +
"\xee\x31\xff\x31\xc0\xfc\xac\x84\xc0\x74\x07\xc1\xcf\x0d" +
"\x01\xc7\xeb\xf4\x3b\x7c\x24\x28\x75\xe1\x8b\x5a\x24\x01" +
"\xeb\x66\x8b\x0c\x4b\x8b\x5a\x1c\x01\xeb\x8b\x04\x8b\x01" +
"\xe2\x89\x44\x24\x1c\x61\xc3\xb2\x08\x29\xd4\x89\xe5\x89" +
"\xc2\x68\x8e\x4e\x0e\xec\x52\xe8\x9f\xff\xff\xff\x89\x45" +
"\x04\xbb\xef\xce\xe0\x60\x87\x1c\x24\x52\xe8\x8e\xff\xff" +
"\xff\x89\x45\x08\x68\x6c\x6c\x20\x41\x68\x33\x32\xe6\x64" +
"\x68\x75\x73\x65\x72\x30\xdb\x88\x5c\x24\x0a\x89\xe6\x56" +
"\xff\x55\x04\x89\xc2\x50\xbb\xa8\xa2\x4d\xbc\x87\x1c\x24" +
"\x52\xe8\x5f\xff\xff\xff\x68\x30\x58\x20\x20\x68\x4e\x54" +
"\x32\x33\x31\xdb\x88\x5c\x24\x05\x89\xe3\x68\x32\x39\x58" +
"\x20\x68\x35\x32\x30\x38\x68\x36\x2d\x31\x38\x68\x32\x31" +
"\x33\x33\x68\x2d\x31\x38\x35\x68\x30\x31\x39\x31\x68\x31" +
"\x38\x35\x32\x31\x9c\x88\x4c\x24\x1a\x89\xe1\x31\xd2\x6a" +
"\x40\x53\x51\x52\xff\xd0\x31\xc0\x50\xff\x55\x08"
msf6 payload(windows/messagebox) > generate
# windows/messagebox - 278 bytes
# https://metasploit.com/
```

Hình. B1

Bước 2: Tinh chỉnh shellcode

- Các dòng assembly (được tô) chỉ ra rằng thanh ghi eax sẽ được restore về giá trị 0, sau đó thoát chương trình (H. B2). Điều cần thiết ở đây là thay các đoạn code này, để gọi về chương trình gốc, sau khi shellcode được thực thi
  - Lấy giá trị ImageBase (pefile.OTIONAL\_HEADER.ImageBase) và EntryPoint (pefile.FILE\_HEADER.AddressOfEntryPoint) (H. B3)
  - Khai thác EntryPoint khi chạy chương trình: Entry point address = ImageBase + EntryPoint address. Vì khi chương trình được load vào memory, các đoạn code sẽ được truy vấn thông qua relative virtual address. Giá trị này bằng tổng của Virtual address và Imagebase.
- Tạo đoạn code thực hiện gọi về chương trình gốc (Các lệnh bao gồm: mv, sub, call thay vì chỉ mv, call là để tránh \x00-nullbyte) (H. B4)
- Thay các dòng shellcode vừa tạo vào vị trí tương ứng trong đoạn shellcode từ Metasploit (H. B5)

```

dc: 68 32 39 58 20      push    0x20583932
e1: 68 35 32 30 38      push    0x38303235
e6: 68 36 2d 31 38      push    0x38312d36
eb: 68 32 31 33 33      push    0x33333132
f0: 68 2d 31 38 35      push    0x3538312d
f5: 68 30 31 39 31      push    0x31393130
fa: 68 31 38 35 32      push    0x32353831
ff: 31 c9               xor     ecx,ecx
101: 88 4c 24 1a          mov     BYTE PTR [esp+0x1a],cl
105: 89 e1                mov     ecx,esp
107: 31 d2                xor     edx,edx
109: 6a 40                push    0x40
10b: 53                  push    ebx
10c: 51                  push    ecx
10d: 52                  push    edx
10e: ff d0                call    eax
110: 31 c0                xor     eax,eax
112: 50                  push    eax
113: ff 55 08             call    DWORD PTR [ebp+0x8]

```

Hình. B2

```

[*] STEP 0x01 - Add the New Section Header
[+] Section Name = b'.axc\x00\x00\x00\x00'
[+] Virtual Size = 0x1000
[+] Virtual Offset = 0x111000
[+] Raw Size = 0x1000
[+] Raw Offset = 0x107c00
[+] Characteristics = 0xe0000020

[*] STEP 0x02 - Modify the Main Headers
[+] Number of Sections = 8
[+] Size of Image = 1122304 bytes
[+] New Entry Point = 0x111000
[+] ImageBase = 0x400000
[+] Original Entry Point = 0x6fe9d

[*] STEP 0x04 - Inject the Shellcode in the New Section
[+] Shellcode wrote in the new section

Process finished with exit code 0

```

Hình.

B3

**String Literal:**

```
"\xB8\xA7\xff\x57\x11\x2D\x11\x01\x11\x11\xff\xD0"
```

**Array Literal:**

```
{ 0xB8, 0xA7, 0xFF, 0x57, 0x11, 0x2D, 0x11, 0x01, 0x11, 0x11, 0xFF, 0xD0 }
```

**Disassembly:**

```

0: b8 a7 ff 57 11      mov     eax,0x1157ffa7
5: 2d 11 01 11 11      sub     eax,0x11110111
a: ff d0                call    eax

```

Hình. B4

```

dc: 68 32 39 58 20      push  0x20583932
e1: 68 35 32 30 38      push  0x38303235
e6: 68 36 2d 31 38      push  0x38312d36
eb: 68 32 31 33 33      push  0x33333132
f0: 68 2d 31 38 35      push  0x3538312d
f5: 68 30 31 39 31      push  0x31393130
fa: 68 31 38 35 32      push  0x32353831
ff: 31 c9               xor    ecx,ecx
101: 88 4c 24 1a          mov    BYTE PTR [esp+0x1a],cl
105: 89 e1                mov    ecx,esp
107: 31 d2                xor    edx,edx
109: 6a 40                push  0x40
10b: 53                   push  ebx
10c: 51                   push  ecx
10d: 52                   push  edx
10e: ff d0                call  eax
110: b8 a7 ff 57 11        mov    eax,0x1157ffa7
115: 2d 11 01 11 11        sub    eax,0x11110111
11a: ff d0                call  eax

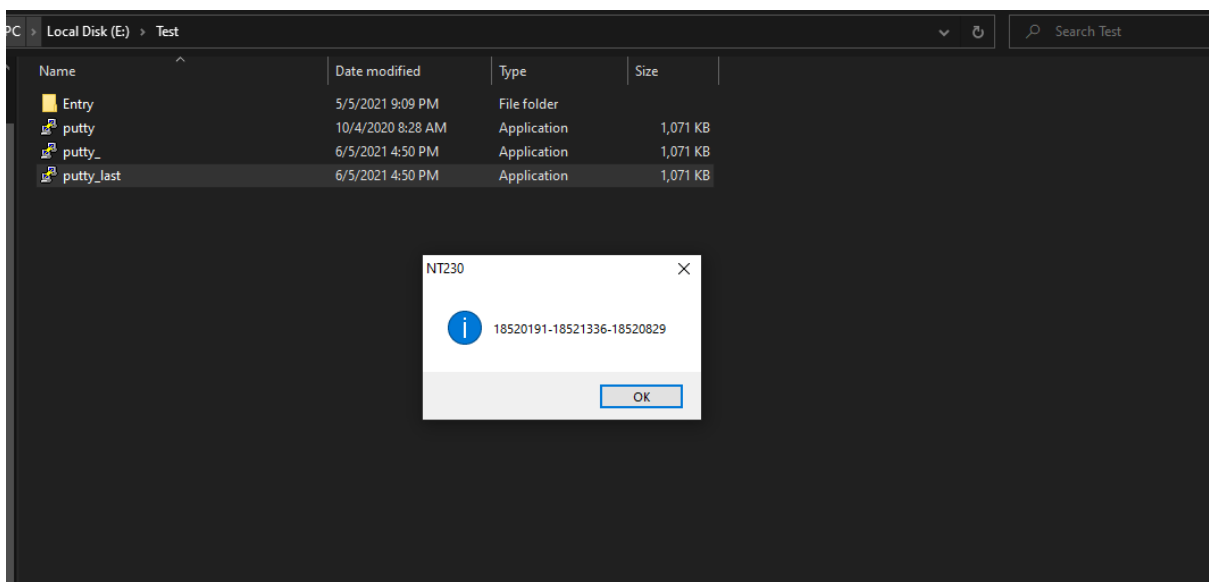
```

Hình. B5

Bước 3: Nhúng shellcode vào chương trình  
(Giải thích code được trình bày qua comments )

\*Kết quả chạy chương trình:

- Chương trình hiển thị được PopUp, sau khi hộp thoại đóng, putty.exe không chạy (H. B6)
- Trên "lý thuyết", chương trình gốc không hoạt động là do đoạn shellcode sẽ đóng tiến trình sau khi chạy hoàn tất, giải pháp là dẫn luồng thực thi về đoạn code gốc thông qua các câu lệnh jmp hoặc call đến EntryPoint ban đầu
- Dựa vào cơ sở này, áp dụng lên putty.exe (Trình bày triển khai phía trên), tuy vậy, kết quả không thành công



Hình. B6



**c. Thực nghiệm lây nhiễm tất cả các file PE khác trong cùng một thư mục khi tập tin virus ở câu b được thực thi (mang file virus sang một thư mục khác bất kỳ và mở lên).**