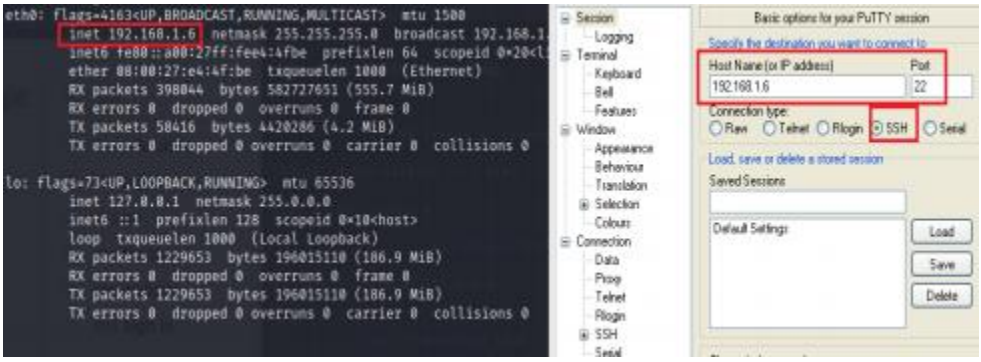
Name –Shruti Mahajan Roll No-101983046 Class-COE21

ASSIGNMENT-4

Implement bind shell, reverse shell bind and meterpreter as payload in the new code caved section. Give step by step method with appropriate screen shots to justify your claims. OR Trojan creation by virtue of PE Code Injection

Connect with our target machine after opening putty.exe

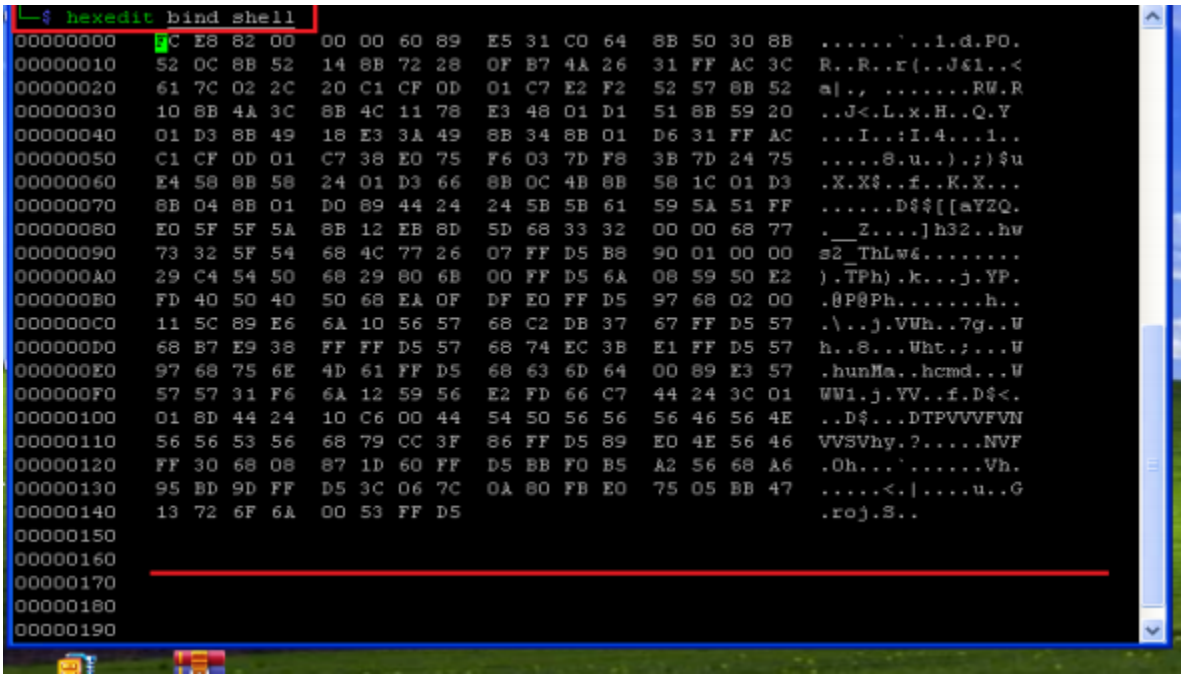


Firstly check SSH status and then if it is inactive then use following commands

Login as user after clicking on putty.exe and then enter the passwords.

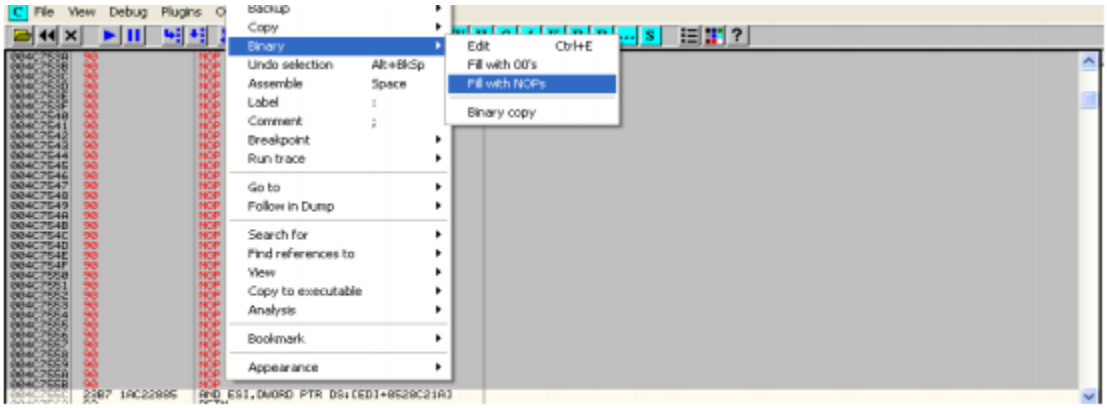
Use msfvenom command to generate bind shell payload and store it as raw data in bind_shell.

Use hexedit command to see the contents of bind_shell file and contents.



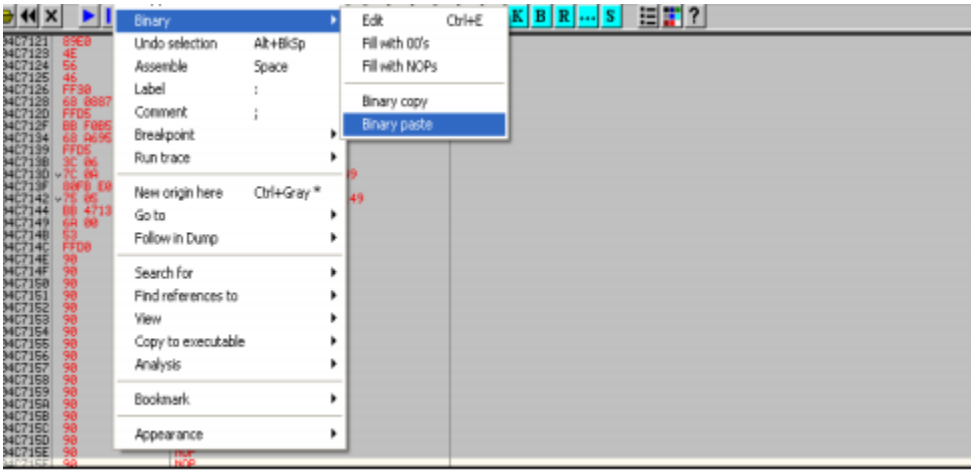
Now we will use hexdump command. Save this code in a txt file .Open putty.exe in ollydbg.

As our payload size is 328 bytes and we need free space below and above our code so we will select minimum 500 bytes in debugger and fill them with NOPs.



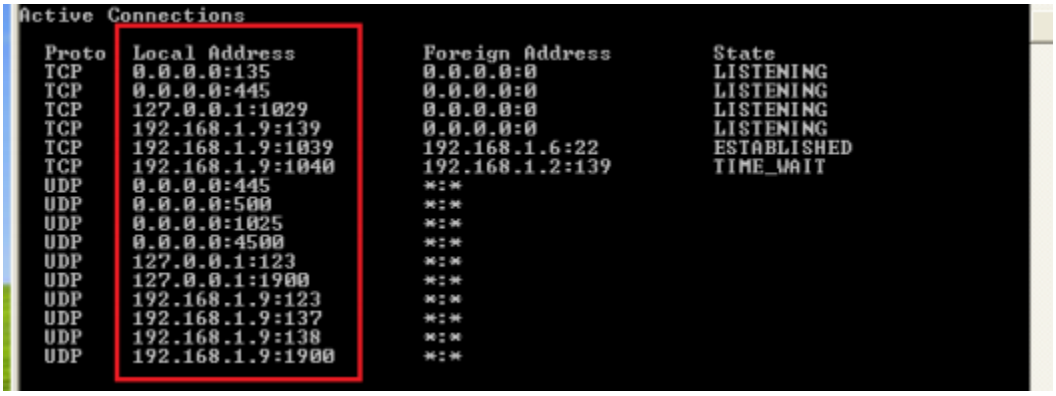
copy

Copy the code and paste it as binary paste in between NOPs filled by us.



After doing this save this in a new file

Using netstat check port 4444 status.



```
Active Connections

  Proto  Local Address          Foreign Address        State
  TCP    0.0.0.0:135            0.0.0.0:0              LISTENING
  TCP    0.0.0.0:445            0.0.0.0:0              LISTENING
  TCP    127.0.0.1:1029         0.0.0.0:0              LISTENING
  TCP    192.168.1.9:139        0.0.0.0:0              LISTENING
  TCP    192.168.1.9:1039       192.168.1.6:22         ESTABLISHED
  TCP    192.168.1.9:1040       192.168.1.2:139        TIME_WAIT
  UDP    0.0.0.0:445            *:*
  UDP    0.0.0.0:500            *:*
  UDP    0.0.0.0:1025           *:*
  UDP    0.0.0.0:4500           *:*
  UDP    127.0.0.1:123          *:*
  UDP    127.0.0.1:1900         *:*
  UDP    192.168.1.9:123        *:*
  UDP    192.168.1.9:137        *:*
  UDP    192.168.1.9:138        *:*
  UDP    192.168.1.9:1900       *:*
```

If its not open then double click on our new file and check if now Port 4444 opens.

Now try to connect from our Kali machine.

REVERSE SHELL

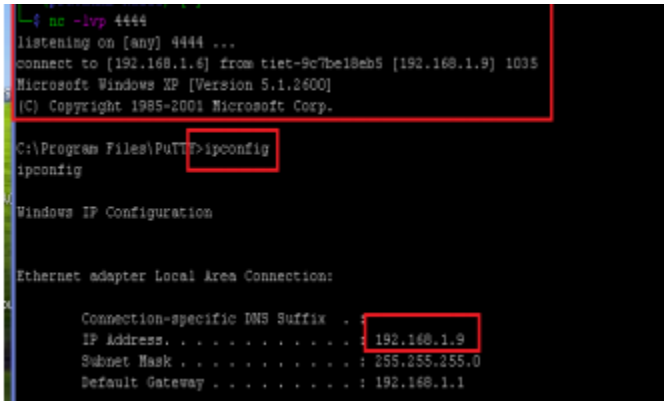
Use msfvenom command for reverse shell and then generate hexdump of it without spacing and all and then copy that in a safe place.

Now we will use putty file that was incorporated with bind shell.

Again fill NOPs where we have code of bind shell after opening putty file in ollydbg.

Replace it with the code of reverse shell and then save it with same name.

Start listening on Port 4444

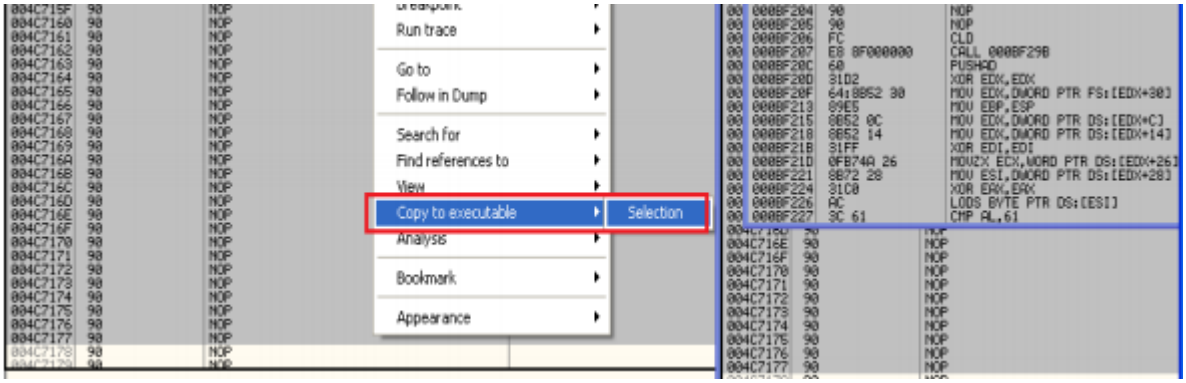


Meterpreter Shell

use msfvenom command for meterpreter reverse shell and then generate hexdump and then copy that in safe space.

Open that reverse shell Putty file in ollydbg.

Again, fill NOPs in place of reverse shell code and paste the code of meterpreter reverse shell and then binary save it.



Start the listener on Port 4444 and then double click on newly generated Putty file. One more thing we want to return to our original Putty functionality also so let's inject the addresses of original entry points saved by us such that after opening Port 4444 it starts the Putty program thus preventing victim from knowing what happened at the backend. Select these bytes and save them in a different file .

Again, start the listener on Port 4444 and double click on the newly created file, it will give us shell and also start the Putty program preventing the user from knowing if something bad happened at the backend.