Computer Network Defence Assignment 2 – Rajat Vij

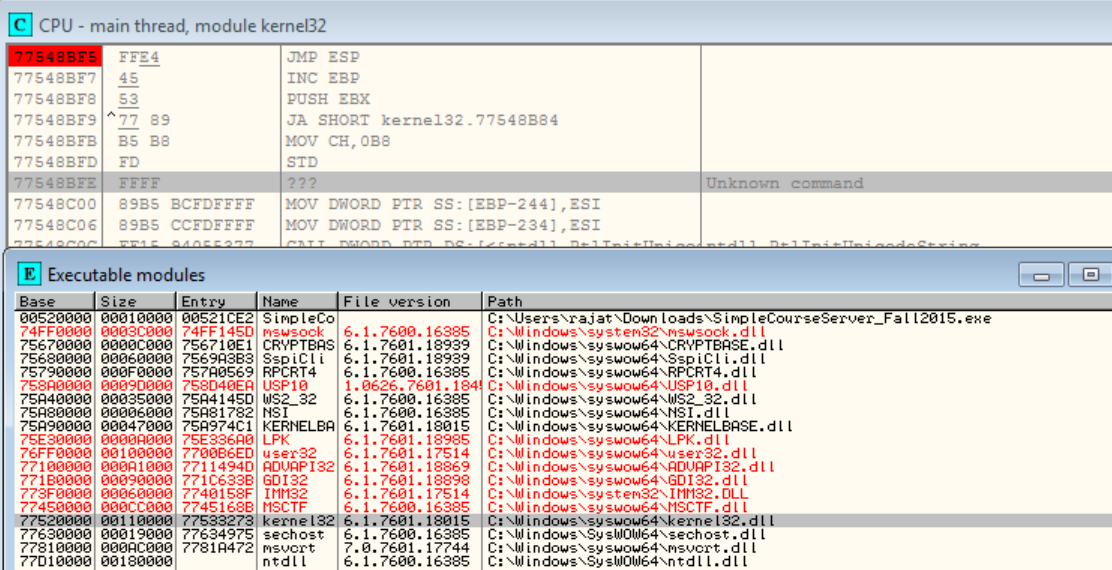
Option 1 : Exploit SimpleServer (used in assignment 1) : ie., exploit stack buffer overflow in the program to gain command line access

Step 1.

Run the program in ollydbg. Put breakpoints on the appropriate locations found in assignment 1.

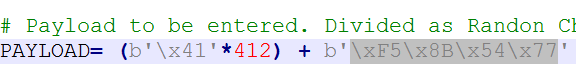
Step 2.

Find the address of JMP ESP of any associated dll by check F6.



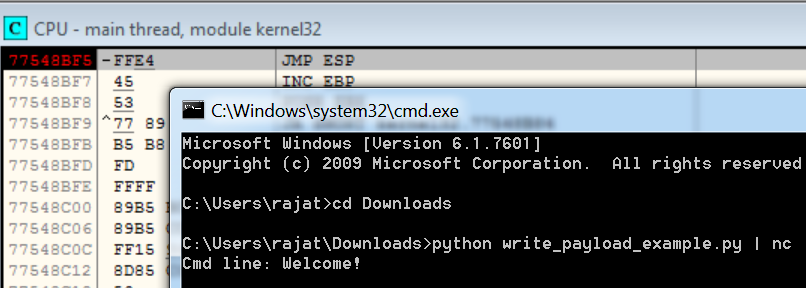
Step 3.

Put the address of JMP ESP instruction in payload’s last 4 character : 77548BF5



Step 4.

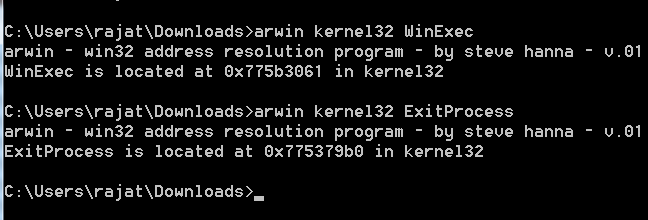
Put a breakpoint at the address to check whether your program is going into kernel or not to execute the shellcode we would inject in following steps.

Program Stops at the breakpoint : 

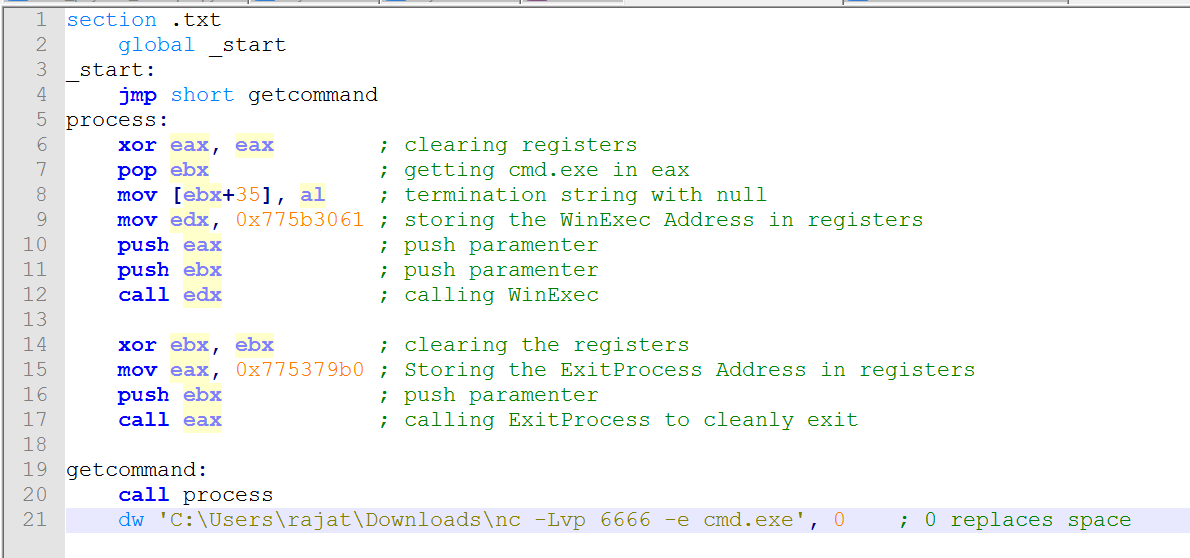
Step 5 Write the Assembly Code and append shellcode to payload:

Step 5.1

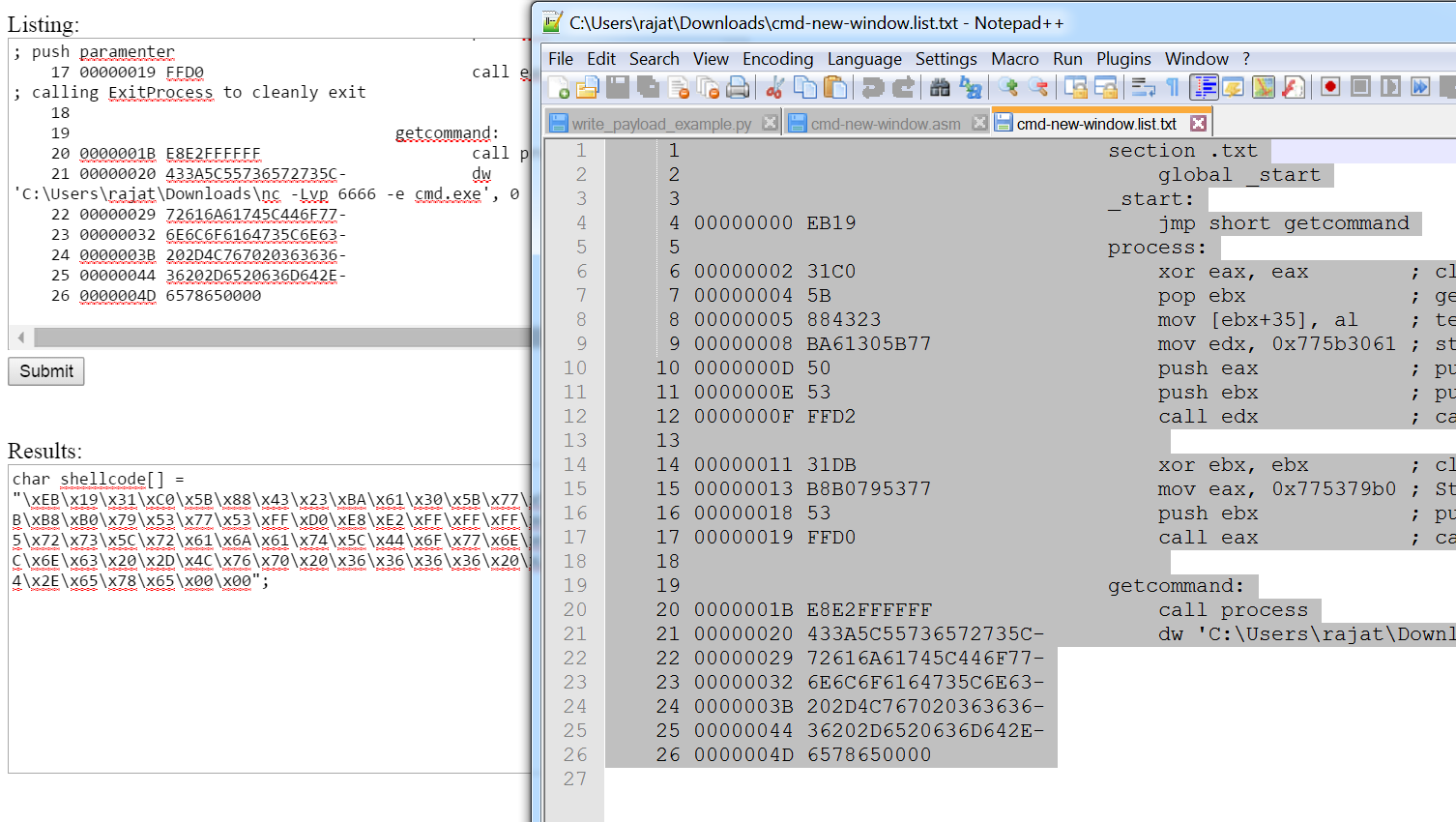
Check the address of WinExec and ExitProcess



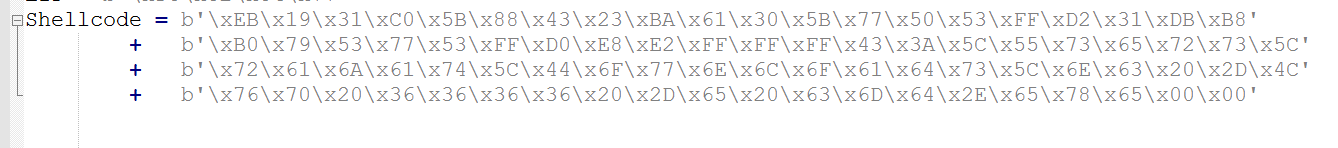
Step 5.2 Write Assembly code



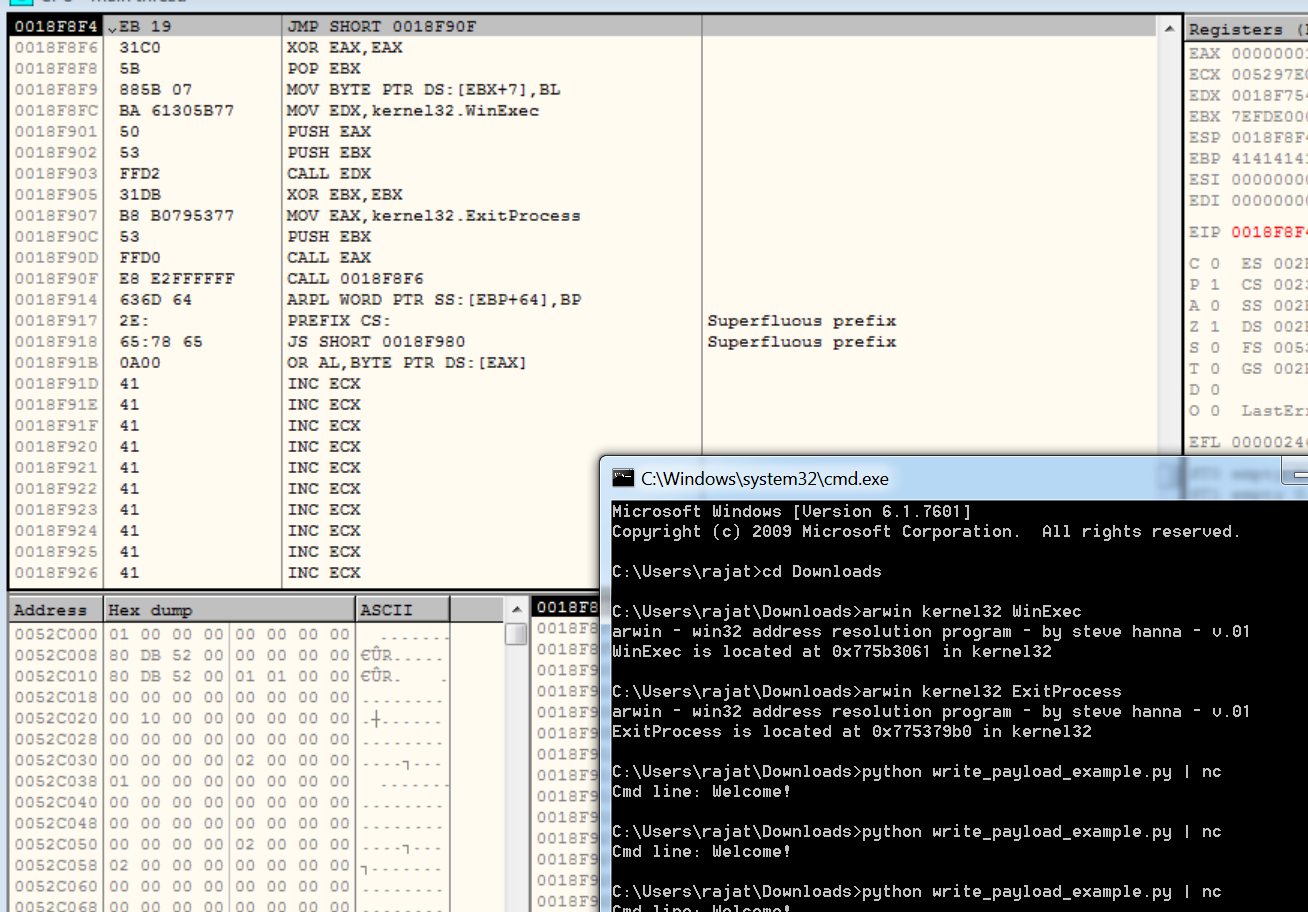
Step 5.3 Compile Assembly code and make a list file to get shellcodeScreen%20Shot%202015-09-30%20at%202.32.45%20PM.png

Step 5.4 Copy contents of list file and get shellcode from the given link.

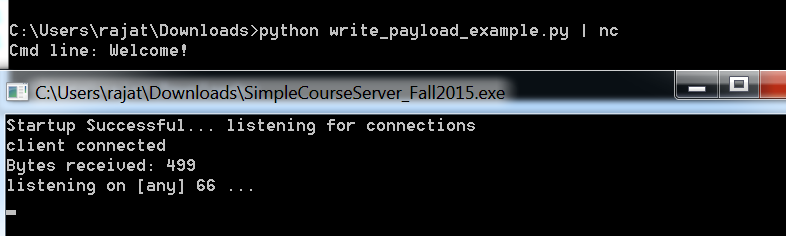
Step 5.5 Append the shellcode in the Payload:

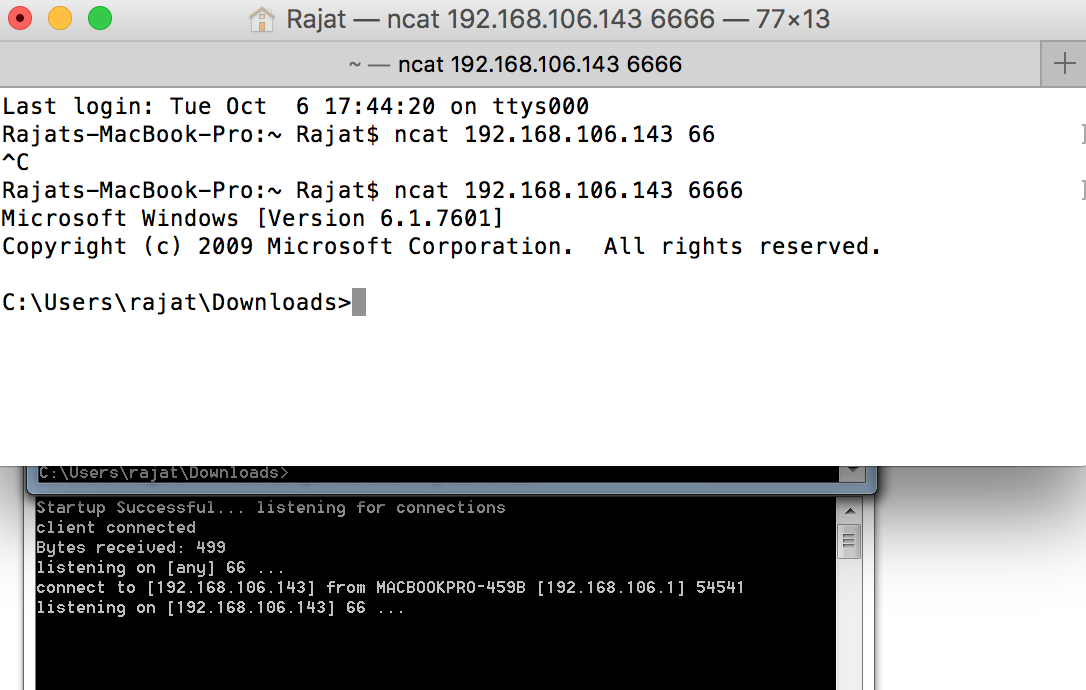


Step 6. Run the program again in ollydbg and pipe python file with payload to netcat.

Our shellcode is injected to the stack: 

Step 7. Run the program

Netcat starts Listening at our specified port: 

Connect to netcat from your machine and gain commandline access.

NOTE:

At first I wasn’t able to get a command prompt because I wasn’t giving correct number of parameters in my assembly code so I tried MessageBox using user32.dll and that worked.

Then I corrected and was able to get the command Prompt and calculator.

Then I tried to do option 2. Where at first I created a code to call nc to hard listen to a port as mentioned in above screenshots.

Option 2 Try failure Issue’s : Can Ignore the below :

Although it beat’s the purpose as we are assuming that netcat is already present on the system and is open at a port. (I am not sure but I think we can send a span mail to host and hope he opens the mail and download netcat with a bat file to run hard listen command. That’s what I thought I would say if I had completed Option 2)

But I wasn’t able to covert my code to dynamically finding the address of different API’s.

I understood the process but I think I am making a mistake somewhere in calling the API’s. I tried to do it on some IDE’s but most of them were not working properly and I didn’t have space to work on visual studio.

References :

<http://www.vividmachines.com/shellcode/shellcode.html>

<http://althing.cs.dartmouth.edu/local/shellcode.html>

<https://blackboard.gwu.edu/bbcswebdav/pid-6857353-dt-content-rid-13862019_2/courses/63195_201503/skape-win32-shellcode.pdf>

http://resources.infosecinstitute.com/stack-based-buffer-overflow-tutorial-part-1-introduction/ http://resources.infosecinstitute.com/stack-based-buffer-overflow-tutorial-part-2-exploiting-the-stack-overflow/ <http://resources.infosecinstitute.com/stack-based-buffer-overflow-tutorial-part-3-—-adding-shellcode/>

I tried using this to do option 2 but I couldn’t get it to work. Maybe I wasn’t able to convert my code properly.

<http://www.codeproject.com/Articles/325776/The-Art-of-Win-Shellcoding#ch4>

And I also refered PEB method to get Kernelbase image in

<https://blackboard.gwu.edu/bbcswebdav/pid-6857353-dt-content-rid-13862019_2/courses/63195_201503/skape-win32-shellcode.pdf>