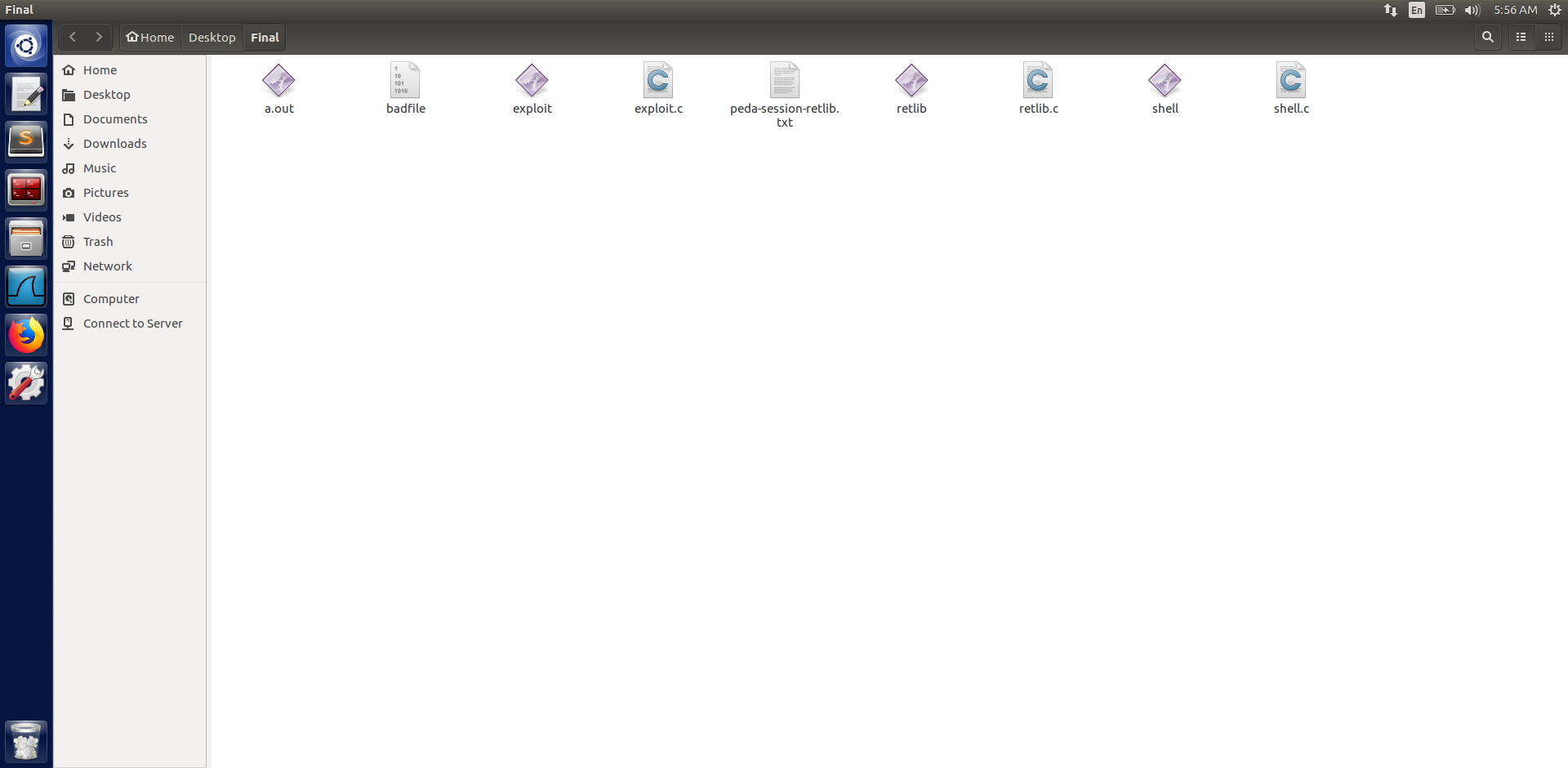
**Proiect SI-Return to lib attack**

Primul pas este dezactivarea randomizari spatiului pe care linuxul o foloseste in prezent cu comanda **$ sudo sysctl -w kernel.randomize\_va\_space=0**

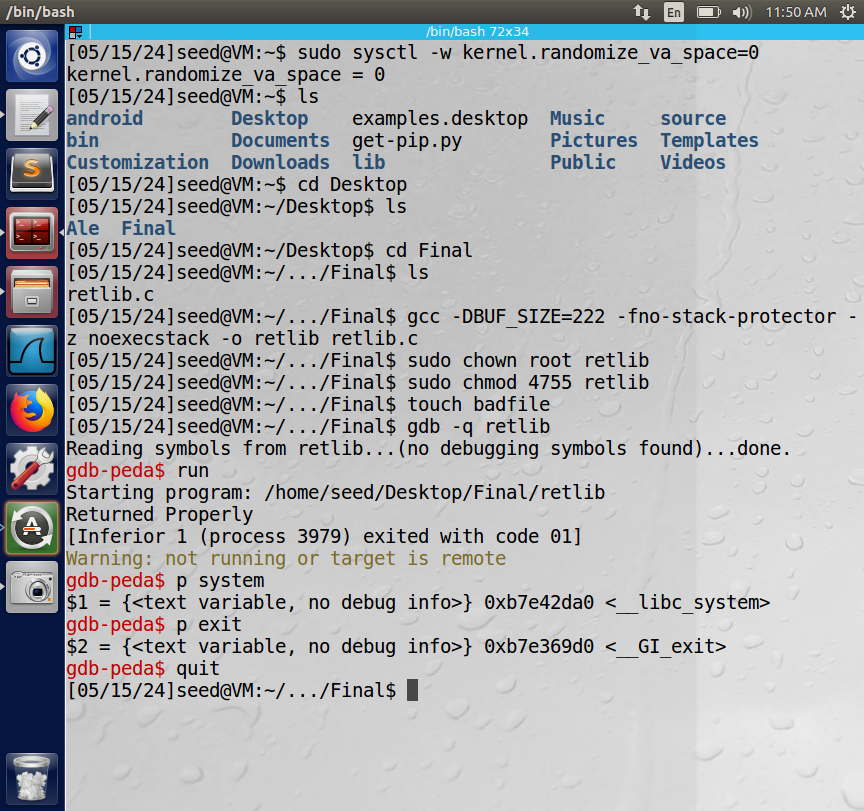
Am luat ca si BUF\_SIZE=222. Vom crea urmatoarele documente pe care le vom executa: retlib.c si exploit.c unde vom face schimbarile necesare conform laboratorului. Pentru a complica retlib.c avem nevoie de comanda **$ gcc -DBUF\_SIZE=N -fno-stack-protector -z noexecstack -o retlib retlib.c**, unde DBUF\_SIZE este cum am precizat anterior 222. După aceea vom rula **$ sudo chown root retlib , $ sudo chmod 4755 retlib** care schimbă ownership-ul fișierului retlib în root, un lucru relevant în lansarea atacului.



După aceea vom încerca să găsim adresele funțiilor system() și exit() cu ajutorul gdb-ului.

gdb-peda$ p system ne va da adresa lui system()

gdb-peda$ p exit ne va da adresa lui exit()

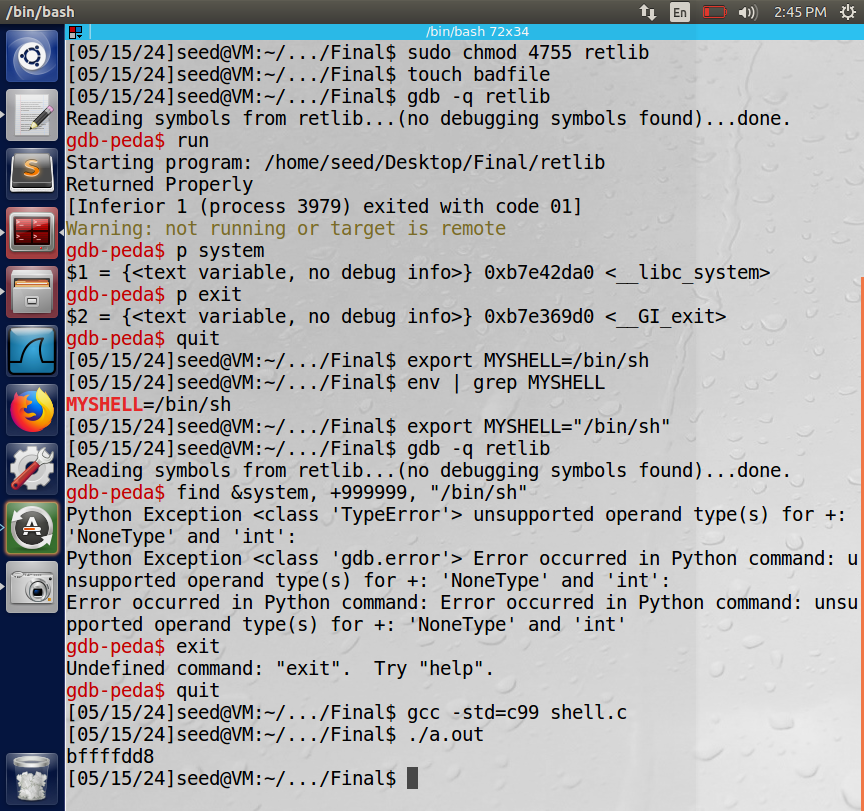


În această captură de ecran se observă pașii parcurși pentru a ajunge la adresele respective.

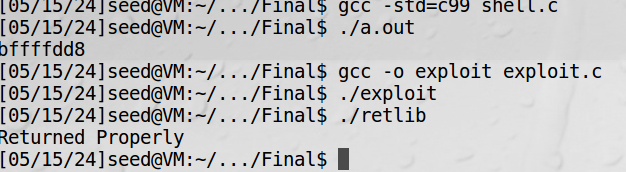
Adresa system este 0xb7e42da0

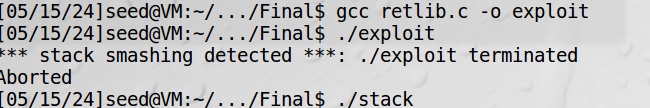
Adresa exit() 0xb7e369d0

Ne mai trebuie acum adresa lui /bin/sh/, acest lucru este arătat în captura de mai jos.



Adresa găsită este bffffdd8, iar acum vom înlocui ceea ce am găsit în codul lui exploit, compilându-l.





X,Y,Z se obin din faptul că știm cum sunt stocate adresele. Adresa lui /bin/sh/ obtinuita ca ebp+12, a lui exit() ebp+8 si a lui system ebp+4.

Trebuia să afișeze # sugerând dreptul de root.