Full Name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Homework 3: ENPM 691 – Secure Programming in C**

* + - Please replicate the following programs that are vulnerable.
    - Please do NOT use preferred stack boundary.
    - These programs are part of the slides and can be referred.
    - You will have to upload exploits scripts (e.g. Perl scripts) and evidence screenshots that you were able to execute shell code of your choice.
    - Take each program and develop an exploit, similar to the perl scripts we used in the class.

1. jmp to bss

#include <string.h>

#include <unistd.h>

char globalbuf[256];

int main(int argc,char\*\* argv) {

char localbuf[256];

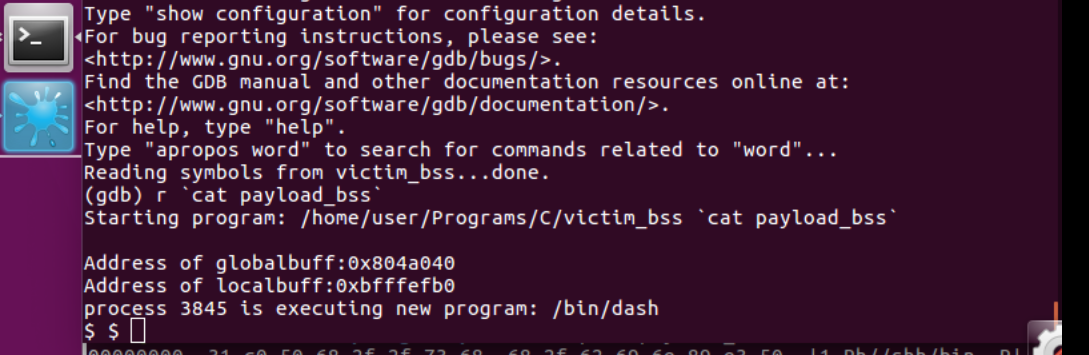
strcpy(localbuf, argv[1]);

strcpy(globalbuf,localbuf);

printf(“Address of global buff is :%p\n”, globalbuff);

return 0;

}



 2. jmp to esp

#include <string.h>

int main(int argc,char\*\* argv) {

char localbuf[256];

strcpy(localbuf, argv[1]);

return 1;

}

3. Ret2EAX

#include <string.h>

void function(char\* str)

{

char buffer[80];

strcpy(buffer, str);

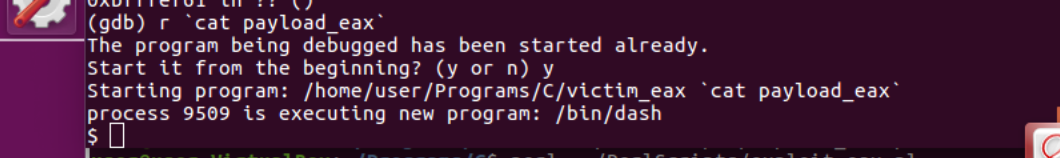
}

int main(int argc, char\*\* argv)

{

function(argv[1]);

}



4. Ret2PopRet

#include <string.h>

void function(int x, char\* str)

{

char buffer[4];

strcpy(buffer, str);

}

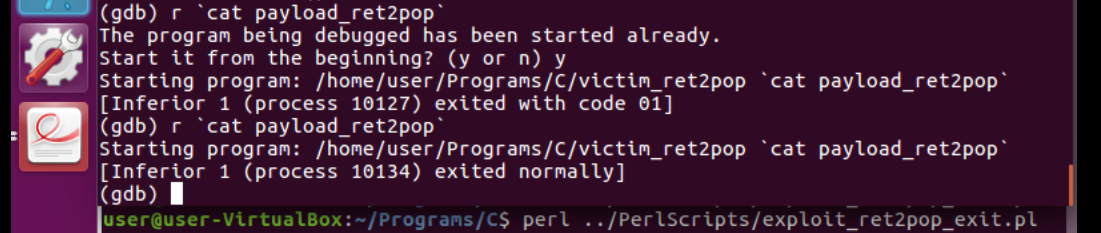
int main(int argc, char\*\* argv)

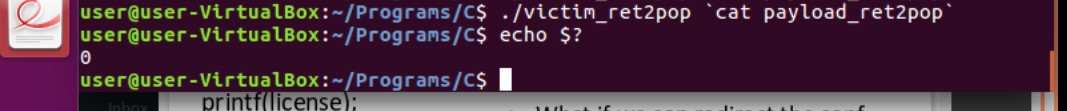
{

function(10, argv[1]);

return 1;

}





5. String pointers

#include <stdio.h>

#include <string.h>

int main(int argc, char\* args[]){

char input[256];

char \*conf = "test -f ~/.progrc";

char \*license = "THIS SOFTWARE IS ...";

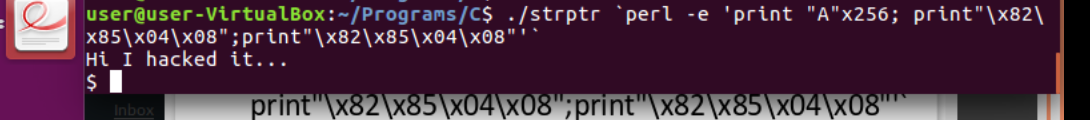
printf(license);

strcpy(input, args[1]);

if (system(conf))

printf("Missing . progrc");

}



6. Function pointers

#include <stdio.h>

#include <string.h>

void function(char\* str) {

printf("%s\n", str);

system("any command");

}

int main(int argc, char\*\* argv)

{

void (\*ptr)(char\* str);

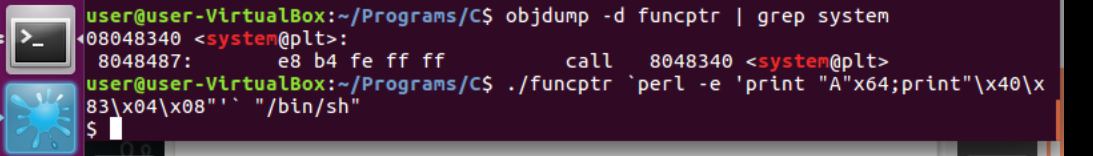
ptr = &function;

char buff[64];

strcpy(buff, argv[1]);

(\*ptr)(argv[2]);

}



7. ret2got.c

#include <stdlib.h>

int main (int argc, char \*\*argv) {

char \*ptr= NULL;

char array[10] ;

ptr = array ;

strcpy ( ptr , argv[1]) ;

printf ("Array has %s at %p\n " ,ptr, &ptr) ;

strcpy ( ptr, argv[2] ) ;

printf ( "Array has %s at %p\n", ptr , &ptr ) ;

return EXIT\_SUCCESS;

}

8. Jump to libc

#include <string.h>

int main(int argc,char\*\* argv) {

char buf[5];

strcpy(buf, argv[1]);

return 1;

}

9. Capture the Flag Challenge:

