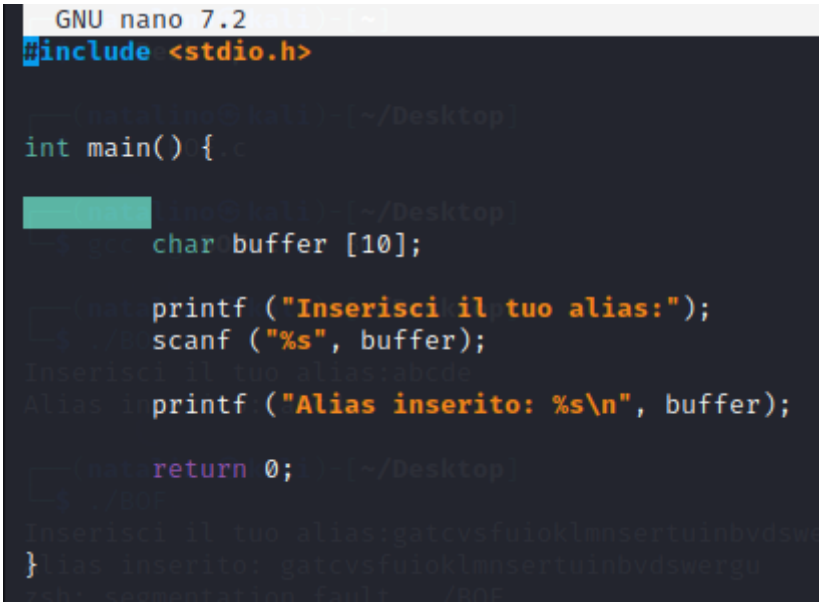
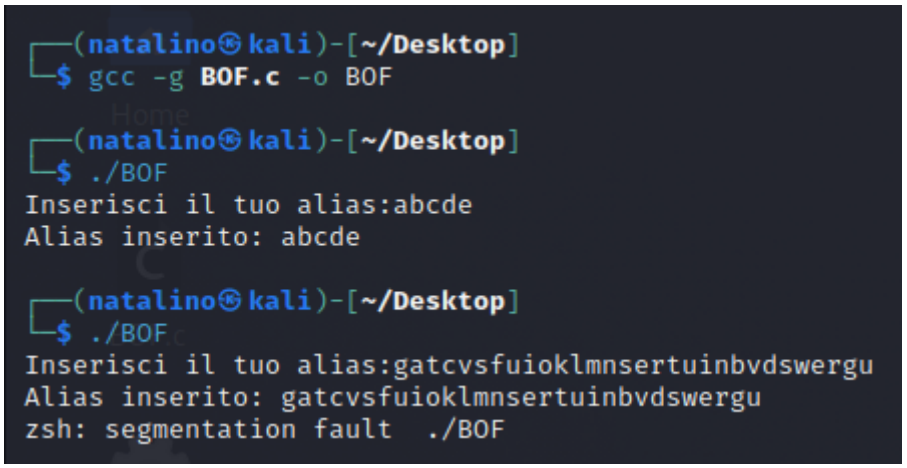
Il seguente programma scritto in C



includendo la libreria *<stdio.h>* che contiene le definizioni di input/output standard, innanzitutto dichiara una variabile *buffer* di tipo *char* con dimensione definita (10). Questa variabile verrà utilizzata per memorizzare l'alias inserito dall'utente. Il programma parte infatti invitando l'utente ad inserire il proprio alias, alla riga successiva legge il suo input memorizzandolo nella variabile *buffer*, e infine restituisce il messaggio che l'alias è stato inserito seguito dallo stesso.
In sintesi, questo codice chiede all'utente di inserire il proprio alias e poi lo stampa a video.
Lo compilo e lo eseguo inserendo un alias di 5 caratteri. Poi faccio la stessa cosa con un alias di 30, ma, ovviamente, sorge un problema



Mi ritorna l'errore *segmentation fault*, e cioè un errore di segmentazione. Esso si verifica quando un programma tenta di scrivere contenuti su una porzione di memoria alla quale non ha accesso. Nel mio caso specifico, infatti, ho inserito 30 caratteri in un buffer che ne può contenere solamente 10, e di

conseguenza alcuni di essi stanno sovrascrivendo aree di memoria inaccessibili.
Questo tipo di errore di programmazione, in base al quale i dati vengono scritti in un buffer oltre i suoi confini, rappresenta un cosiddetto *buffer overflow*. E quest'ultimo, a livello di sicurezza, può causare la compromissione del programma da parte di un utente malintenzionato. Infatti, un simile individuo potrebbe sfruttare un buffer overflow per eseguire codice arbitrario sul sistema in cui è in esecuzione il programma vulnerabile. Questo potrebbe consentirgli di accedere ai dati, eseguire attività dannose o persino prendere il controllo del sistema.
Se infatti vado a modificare la lunghezza di *buffer* e la metto a *30*, l'errore non si verifica

E' opportuno però piazzare un controllo all'interno del programma che impedisca all'utente di inserire alias superiori a quella lunghezza.

Ecco perché aggiungo la libreria *string.h* che fornisce le funzioni per la manipolazione delle stringhe e la libreria *stdlib.h* che fornisce funzioni di utilità generali, come la funzione *exit()* che sono andato ad utilizzare. Poi sono andato a definire la variabile *len* che calcola la lunghezza dell'alias dell'utente utilizzando la funzione *strlen()*, e sotto ho piazzato una condizione che controlla se la lunghezza dell'alias dell'utente è superiore a 30; se lo è, il programma stampa un messaggio di errore e termina.

```
 GNU nano 7.2
#include <stdio.h>
#include <string.h>
#include <stdlib.h>

int main() {

        char buffer [30];
        int len;

        printf ("Inserisci il tuo alias:");
        scanf ("%s", buffer);

        len = strlen(buffer);

        if (len > 30) {
                printf ("Per favore, il tuo alias è troppo lungo\n");
                exit(1);
        }

        printf ("Alias inserito: %s\n", buffer);

        return 0;


}
```

Difatti, l'output che ottengo è il seguente

```
┌──(natalino㉿kali)-[~/Desktop]
└─$ gcc -g BOF.c -o BOF

┌──(natalino㉿kali)-[~/Desktop]
└─$ ./BOF
Inserisci il tuo alias:abcde
Alias inserito: abcde

┌──(natalino㉿kali)-[~/Desktop]
└─$ ./BOF
Inserisci il tuo alias:poiuytrewqasdfghjklmnbvcxzaqwe
Alias inserito: poiuytrewqasdfghjklmnbvcxzaqwe

┌──(natalino㉿kali)-[~/Desktop]
└─$ ./BOF
Inserisci il tuo alias:iopqwertyuiopasdfghjklzxcvbnmkj
Per favore, il tuo alias è troppo lungo

┌──(natalino㉿kali)-[~/Desktop]
└─$
```