

TASK:

Creazione di un codice in C vulnerabile al Buffer Overflow andando poi a causare il “segmentation fault”, poi andare a modificare il parametro buffer portandolo ad un massimo di 30 e vedere come cambia la situazione

Seguendo le indicazioni delle slide sono andato a creare il file BOF.c:

```
GNU nano 7.0
#include <stdio.h>

int main () {

char buffer [10];

printf ("Si prega di inserire il nome utente:");
scanf ("%s", buffer);

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

return 0;

}
```

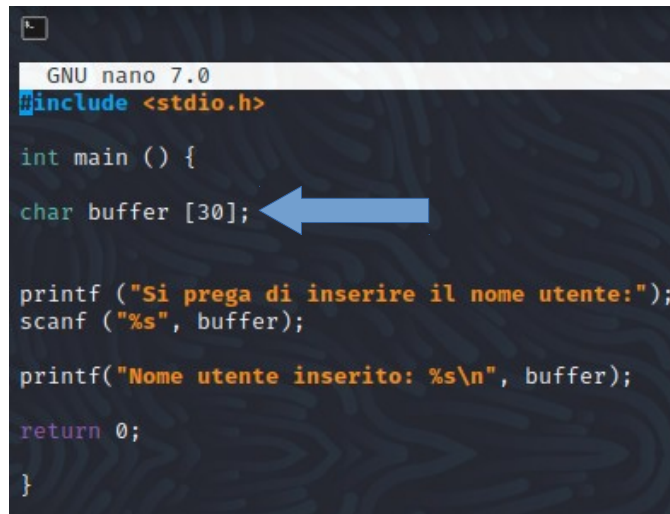
Sono andato quindi a compilarlo e poi l'ho lanciato:

```
kali-linux-2022.3-virtualbox-amd64 [In esecuzione] - Oracle VM
File Macchina Visualizza Inserimento Dispositivi Aiuto
(kali@kali)-[~/Desktop]
$ nano BOF.c
(kali@kali)-[~/Desktop]
$ gcc -g BOF.c -o BOF
(kali@kali)-[~/Desktop]
$ ./BOF
Si prega di inserire il nome utente:prova
Nome utente inserito: prova
```

A questo punto sono andato a verificare come si comporta il programma se inserisce un input con più di 10 caratteri:

```
(kali@kali)-[~/Desktop]
$ ./BOF
Si prega di inserire il nome utente:12345678910111213141516171819
Nome utente inserito: 12345678910111213141516171819
zsh: segmentation fault ./BOF
```

Seguendo le indicazioni della traccia sono andato a modificare la lunghezza massima della variabile buffer:



```
GNU nano 7.0
#include <stdio.h>

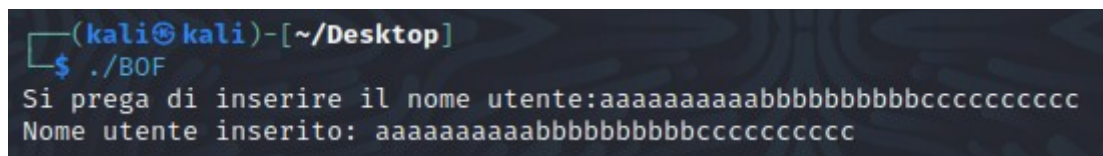
int main () {
char buffer [30];

printf ("Si prega di inserire il nome utente:");
scanf ("%s", buffer);

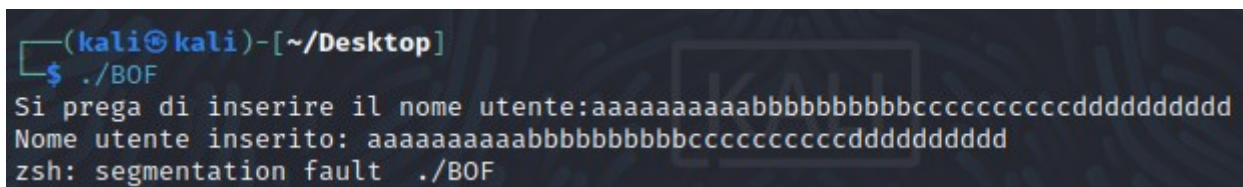
printf("Nome utente inserito: %s\n", buffer);

return 0;
}
```

Dopo averlo nuovamente compilato ho quindi fatto nuovamente le prove per vedere come si comporta il programma quando si va a mettere in input una stringa con meno di 30 caratteri e poi una stringa con più di 30 caratteri:



```
(kali㉿kali)-[~/Desktop]
$ ./BOF
Si prega di inserire il nome utente:aaaaaaaaabbbbbbbbbbccccccccc
Nome utente inserito: aaaaaaaaaabbbbbbbbbbccccccccc
```



```
(kali㉿kali)-[~/Desktop]
$ ./BOF
Si prega di inserire il nome utente:aaaaaaaaabbbbbbbbbbcccccccccccccccccc
Nome utente inserito: aaaaaaaaaabbbbbbbbbbcccccccccccccccccc
zsh: segmentation fault ./BOF
```