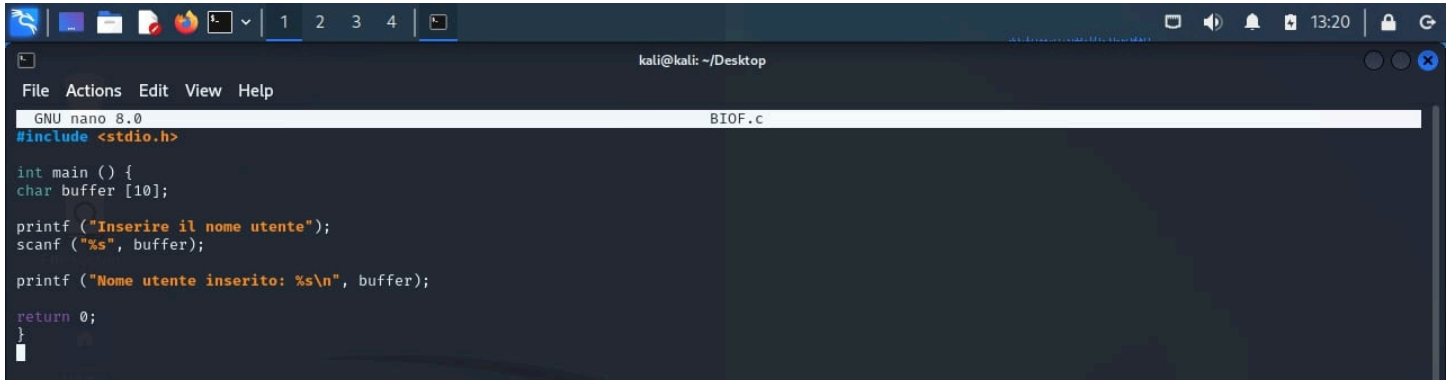


Buffer Overflow

Creazione del file “**BOF.c**” e trascrizione del codice da eseguire in una fase successiva.



```
GNU nano 8.0 BOF.c
#include <stdio.h>

int main () {
char buffer [10];

printf ("Inserire il nome utente");
scanf ("%s", buffer);

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

return 0;
}
```

Esecuzione del comando “**gcc -g BOF.c -o BOF**” che nello specifico sta ad indicare che compila il file sorgente BOF.c includendo le informazioni di debug e genera un eseguibile chiamato BOF e che andremo ad eseguire ogni qualvolta verranno effettuate delle modifiche nel codice. Successivamente, è stato utilizzato il comando “**./BOF**” per eseguire il file precedentemente creato e dopodiché sono state effettuate prove inserendo inizialmente una stringa da 5 caratteri che non ha riportato nessun errore, mentre, provando per la seconda volta ad inserire una stringa superiore ai 10 caratteri, come si può ben vedere dallo screen, il sistema ci riporta un errore di “**Buffer Overflow**”.

Nello specifico, dissezionando il codice in questione, si evince che:

gcc: È il compilatore GNU per il linguaggio C.

-g: Opzione per includere le informazioni di debug nel file eseguibile generato. Questo è utile per eseguire il debugging del programma usando un debugger come gdb.

BOF.c: È il file sorgente C da compilare.

-o BOF: Specifica il nome dell'output, ovvero il file eseguibile generato. In questo caso, l'eseguibile sarà chiamato BOF.

```
kali@kali: ~/Desktop
File Actions Edit View Help
(kali@kali)~[~/Desktop]
$ sudo nano BOF.c
(kali@kali)~[~/Desktop]
$ gcc -g BOF.c -o BOF
(kali@kali)~[~/Desktop]
$ sudo ./BOF
Inserire il nome utente guest
Nome utente inserito: guest
(kali@kali)~[~/Desktop]
$ sudo ./BOF
Inserire il nome utente guestguest
Nome utente inserito: guestguest
(kali@kali)~[~/Desktop]
$ sudo ./BOF
Inserire il nome utente guestguestguestguestguestguest
Nome utente inserito: guestguestguestguestguestguest
zsh: segmentation fault sudo ./BOF
(kali@kali)~[~/Desktop]
$
```

Come si può ben vedere da questo screen, in questo caso sono stati effettuati i medesimi comandi utilizzati anche nella prima prova dove il limite di caratteri era pari a 10. Mentre, nella successiva prova, è stato impostato il limite dei caratteri da inserire a "30" e, visionando lo screen, possiamo vedere come durante il primo tentativo con un inserimento di 30 caratteri, il sistema non abbia mostrato nessun errore. Diversamente nel secondo tentativo dove sono stati inseriti 40 caratteri e dove difatti si può notare l'errore di "**Buffer Overflow**" stampato dal sistema.

```
(kali@kali)~[~/Desktop]
$ sudo nano BOF.c
(kali@kali)~[~/Desktop]
$ gcc -g BOF.c -o BOF
(kali@kali)~[~/Desktop]
$ sudo ./BOF
Inserire il nome utente guestguestguestguestguestguest
Nome utente inserito: guestguestguestguestguestguest
(kali@kali)~[~/Desktop]
$ sudo ./BOF
Inserire il nome utente guestguestguestguestguestguest
Nome utente inserito: guestguestguestguestguestguest
(kali@kali)~[~/Desktop]
$ sudo ./BOF
Inserire il nome utente guestguestguestguestguestguest
Nome utente inserito: guestguestguestguestguestguest
zsh: segmentation fault sudo ./BOF
(kali@kali)~[~/Desktop]
$
```