BUFFER OVERFLOW

kali@kalikali: ~/Scrivania

File Azioni Modifica Visualizza Aiuto

GNU nano 7.2 BOF1.c *

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

#define MAX_CHARACTERS 30

int main() {
    char username[MAX_CHARACTERS + 1];  // +1 per il terminatore di stringa '\0'

    printf("Inserisci il nome utente (massimo %d caratteri): ", MAX_CHARACTERS);
    scanf("%s", username);

    if (strlen(username) > MAX_CHARACTERS) {
        printf("Errore: il nome utente inserito supera i %d caratteri consentiti.\n", MAX_CHARACTERS);
    } else {
        printf("Nome utente inserito correttamente: %s\n", username);
    int *address; // Dichiarazione di un puntatore a un intero

    int value = 30; // Valore da memorizzare

    // Assegna all'indirizzo di memoria il puntatore a "value"
    address = &value;

    // Stampa il valore e l'indirizzo di memoria
    printf("Valore: %d\n", *address);
    printf("Indirizzo di memoria: %p\n", address);

    }

    return 0;
}
```

La prima parte del codice gestisce l'input del nome utente e controlla se la lunghezza del nome utente supera una determinata costante ***MAX_CHARACTERS***

La seconda parte assegna al puntatore l'indirizzo di memoria di una variabile intera e successivamente stampa il valore e l'indirizzo di memoria

Utilizziamo la libreria ***<stdio.h>*** per utilizzare le funzioni di input output standard

Utilizziamo la libreria ***<string.h>*** per utilizzare la funzione ***strlen*** per calcolare la lunghezza della stringa

Costante ***MAX_CHARACTERS*** con valore di 30

Viene dichiarato un array di caratteri "username" di dimensione ***MAX_CHARACTERS + 1***

Con la funzione ***printf*** mostra un messaggio all'utente, indicando il numero massimo di caratteri consentiti

La funzione ***scanf*** per leggere l’input dell’utente e memorizzare nell’array “username”.

%s Indica che verrà letto una stringa di caratteri

Viene utilizzata la funzione ***strlen*** per controllare la lunghezza del nome inserito (Se la lunghezza supera 30 caratteri, viene visualizzato un messaggio di errore)

Se la lunghezza e valida (compresa nei 30 caratteri, viene visualizzato un messaggio di conferma con il nome inserito correttamente

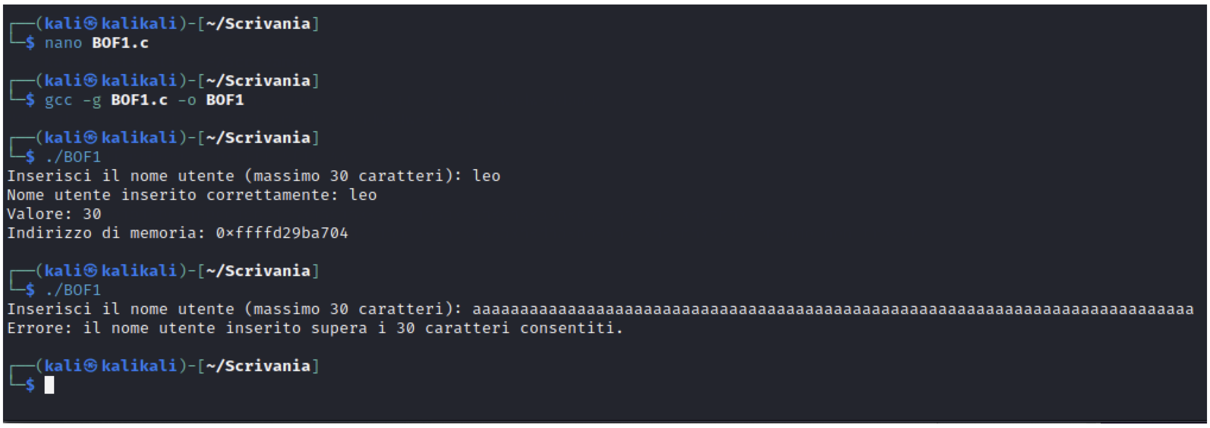
Viene dichiarato un puntatore “***address***” a un intero

Viene dichiarata una variabile “***value***” di tipo intero e viene assegnato il valore 30

Viene assegnato al puntatore “***address***” l’indirizzo di memoria della variabile “***value***” utilizzando l’operatore “***&***”

Viene utilizzata la funzione ***“printf”*** per stampare il valore puntato da ***“address”*** e l’indirizzo di memoria ***“value”***

Infine ***return 0*** per indicare che l’esecuzione è avvenuta con successo.



Il valore dell’indirizzo di memoria può variare ad ogni esecuzione del programma

ciandri l