

ESERCITAZIONE W17D4

Traccia:

Nella lezione dedicata agli attacchi di sistema, abbiamo parlato dei buffer overflow, una vulnerabilità che è conseguenza di una mancanza di controllo dei limiti dei buffer che accettano input utente.

Nelle prossime slide vedremo un esempio di codice in C volutamente vulnerabile ai BOF, e come scatenare una situazione di errore particolare chiamata «segmentation fault», ovvero un errore di memoria che si presenta quando un programma cerca inavvertitamente di scrivere su una posizione di memoria dove non gli è permesso scrivere (come può essere ad esempio una posizione di memoria dedicata a funzioni del sistema operativo).

Creazione e test del codice della traccia



The screenshot shows a terminal window with the nano text editor open. The title bar at the top indicates the user is 'kali' at 'kali' in the directory '~/Desktop'. The editor's status line shows 'GNU nano 7.2' and the filename 'BOF.c *'. The code being edited is a C program that prompts the user for a name and prints it back. The code is as follows:

```
#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;

}
```

```
kali@kali: ~/Desktop
```

File Actions Edit View Help

```
zsh: corrupt history file /home/kali/.zsh_history  
[kali@kali]~  
$ cd /home/kali/Desktop  
  
[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:utente  
Nome utente inserito: utente  
  
[kali@kali]-~/Desktop  
$ ./BOF  
Si prega di inserire il nome utente:jhgjkidghgdhjgdghjdijhgddjhjgjdhdgjhgjdhdgjdhgjdhdjdhdjkfjdkfkghjkfhdfkhkjhfdhkjfghdghjggoidgoideghhfghdrugfhdhguerhfdgudhjfgfjdkjghkdjfhgfddjjfjdkjgkjkgdkjhgdhjsjgdjgkfddhjdjgsdjkkdhoigoisorhnjnvhvuruhsrvhgrlkdghvu  
hivhhuvrvuhvruidhvrlruhduidhguhshsfhoighreihohfhfosourutyhbvcncsf  
Nome utente inserito: jhgjkidghgdhjgdghjdijhgddjhjgjdhdgjhgjdhdgjdhgjdhdjdhdjkfjdkfkghjkfhdfkhkjhfdhvjggoidgoideghhfghdrugfhdhguerhfdgudhjfgfjdkjghkdjfhgfddjjfjdkjgkjkgdkjhgdhjsjgdjgkfddhjdjgsdjkkdhoigoisorhnjnvhvuruhsrvhgrlkdghvuhivhhuvrvuhvru  
idhvrlruhduidhguhshsfhoighreihohfhfosourutyhbvcncsf  
zsh: segmentation fault ./BOF  
  
[kali@kali]-~/Desktop  
$ █
```

Modifica e test del codice aumentando le dimensioni del vettore a 30

```
kali@kali: ~/Desktop
File Actions Edit View Help
GNU nano 7.2 BOF.c
#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;

}
```

```
(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:nomeutentedipiuditrentacaratteridbfjdujsjfksd
fbskjdfbjksbfjkdjsbfjbbjdfbfjdbksfbsbdkbsfkdsjbdbseljg
Nome utente inserito: nomeutentedipiuditrentacaratteridbfjdujsjfksdfbskjdfbjksbfj
kdjsbfjbbjdfbfjdbksfbsbdkbsfkdsjbdbseljg
zsh: segmentation fault ./BOF

(kali@kali)-[~/Desktop]
$ ./BOF
Si prega di inserire il nome utente:nomeutentemenoditrentacaratteri
Nome utente inserito: nomeutentemenoditrentacaratteri

(kali@kali)-[~/Desktop]
$
```

Conclusioni

Con la modifica, abbiamo risolto i problemi relativi all'inserimento di un numero di caratteri inferiore a 30. Tuttavia, quando inseriamo un numero di caratteri superiore a 30, riscontriamo un errore di segmentazione. Questo indica che stiamo tentando di scrivere una parte dell'input su una porzione di memoria non accessibile.