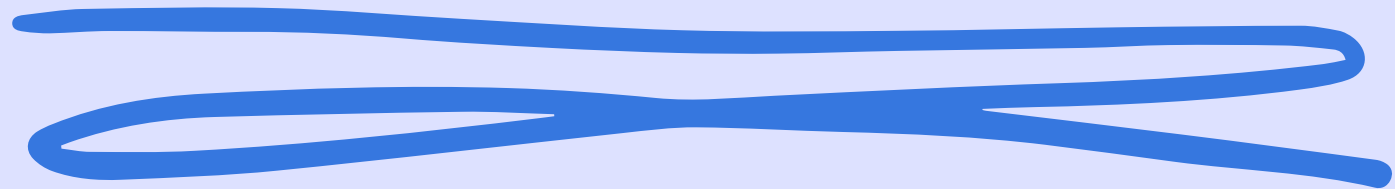


S7 L4

Buffer Overflow



Daniele Zizzi

Buffer Overflow

Il buffer overflow, è un tipo di vulnerabilità che si verifica quando i dati inviati al buffer, ne superano la grandezza. Quindi comporta la sovrascrittura di altre porzioni di memoria e il danneggiamento dei dati contenuti in esso. Le altre porzioni di memoria, posso essere sovrascritte con del codice malevolo, quindi, ogni qualvolta il sistema accederà a tale indirizzo, eseguirà il codice, compromettendo il sistema.



Codice affetto da problemi di buffer overflow, poichè, non ci sono controlli sulla lunghezza dell'input dell'utente. Possiamo notare l'errore quando nell'output riceviamo "segmentation fault"

```
#include <stdio.h>\n\nint main(){\n    char buffer[30];\n\n    printf("Si prega di inserire il nome utente:");\n\n    scanf("%s", buffer);\n\n    printf("Nome utente inserito: %s\\n", buffer);\n\n    return 0;\n}\n
```

```
(daniele@kali)~$ cd Desktop\n(daniele@kali)~/Desktop$ gcc -g bof.c -o bof1\n(daniele@kali)~/Desktop$ ./bof1\nSi prega di inserire il nome utente:oemmicowieifmowefmowieerignaeiurgnaeriugniegnueangijerngfjernfjdfsnfjd fngldafnjngldafnmglfdafngkda fngkdafngkadfnkgkf\nNome utente inserito: oemmicowieifmowefmowieerignaeiurgnaeriugniegnueangijerngfjernfjdfsnfjd fngldafnjngldafnmglfdafngkda fngkdafngkadfnkgkf\nzsh: segmentation fault ./bof1\n(daniele@kali)~/Desktop$ \n
```

Con l'utilizzo di `fgets`, al posto di `scanf`, posso restringere l'input in base alla grandezza dell'array "buffer". Qualsiasi input dell'utente, viene ristretto a 31 caratteri, in questo caso. Quindi evito il buffer overflow.

```
~/Desktop/bufferoverflow.c - Mousepad  
File Edit Search View Document Help  
+ [ ] [ ] [ ] [ ] [ ] [ ] [ ] [ ] [ ]  
1 #include <stdio.h>  
2  
3 int main(){  
4  
5 char buffer[30];  
6  
7 printf("Si prega di inserire il nome utente:");  
8  
9 fgets(buffer, sizeof(buffer), stdin);  
10  
11 printf("Nome utente inserito: %s\n", buffer);  
12  
13 return 0;  
14 }  
15 }  
16
```

```
daniele@kali: ~/Desktop  
File Actions Edit View Help  
  
(daniele@kali)-[~]  
$ cd Desktop  
  
(daniele@kali)-[~/Desktop]  
$ gcc -g bof.c -o bof1  
  
(daniele@kali)-[~/Desktop]  
$ ./bof1  
Si prega di inserire il nome utente:oemicowieifmowefmowieerignaeiurgnaeriugniegnueangijerng  
fjernfjdfsnfjd fngldafnjgl dafnmgl dfanglkda fnmgkdaf ngkadfn gkdf  
Nome utente inserito: oemicowieifmowefmowieerignaeiurgnaeriugniegnueangijerngfjernfjdfsnfjd  
fngldafnjgl dafnmgl dfanglkda fnmgkdaf ngkadfn gkdf  
zsh: segmentation fault ./bof1  
  
(daniele@kali)-[~/Desktop]  
$ ./BOF  
Si prega di inserire il nome utente:hjefbhjwrebfnhernbfkjnrjkfnerjkgnrtnknjrtnhkjrtnhjkrfnh  
jkrfnhkjrtngjkerngjkenrgkjnerkgnerknjgekrjngejkrngkejrngejrngjekrng  
Nome utente inserito: hjefbhjwrebfnhernbfkjnrjkfne  
  
(daniele@kali)-[~/Desktop]  
$ █
```

Per poter verificare un buffer overflow e quindi stamparne il risultato.
Dovremmo conoscere gli indirizzi di memoria, dove le informazioni vengono memorizzate, quindi assegnare al secondo array, un indirizzo di memoria adiacente al primo, in modo che i dati sovrascritti siano quelli del secondo array.
Poi stamparne il risultato.