

TAG - Segurança Ofensiva

Nome: Mariane Ferreira

Stack 0:

Como a memória é alocada: O %esp indica o topo da pilha e esta como endereço 0xbfffc58 e o próximo endereço é 0xc0000000 (3221224536 3221225472 em decimais), e como o topo da pilha começa em baixo e vai subindo, significa que a pilha começa de baixo para cima.

Olhando o mapa da memória, dá pra perceber que uma parte da pilha está vazia, o que não deveria acontecer, já que criamos um char para armazenamento. Logo, isso prejudica o funcionamento do programa, pois o programa vai para o próximo endereço de memória subsequente para continuar.

```
(gdb) info proc mapping
process 1492
cmdline = '/opt/protostar/bin/stack0'
cwd = '/home/user'
exe = '/opt/protostar/bin/stack0'
Mapped address spaces:

   Start Addr   End Addr       Size     Offset objfile
   -----
stack0          0x8048000   0x8049000     0x1000         0 /opt/protostar/bin/stack0
stack0          0x8049000   0x804a000     0x1000         0 /opt/protostar/bin/stack0
               0xb7e96000 0xb7e97000     0x1000         0 /lib/ld-2.11.2.so
               0xb7e97000 0xb7fd5000    0x13e000        0 /lib/libc-2.11.2.so
               0xb7fd5000 0xb7fd6000     0x1000    0x13e000 /lib/libc-2.11.2.so
               0xb7fd6000 0xb7fd8000     0x2000    0x13e000 /lib/libc-2.11.2.so
               0xb7fd8000 0xb7fd9000     0x1000    0x140000 /lib/libc-2.11.2.so
               0xb7fd9000 0xb7fdc000     0x3000         0 /lib/libc-2.11.2.so
               0xb7fdf000 0xb7fe2000     0x3000         0 /lib/libc-2.11.2.so
               0xb7fe2000 0xb7fe3000     0x1000         0 [vdso]
               0xb7fe3000 0xb7ffe000    0x1b000         0 /lib/ld-2.11.2.so
               0xb7ffe000 0xb7fff000     0x1000    0x1a000 /lib/ld-2.11.2.so
               0xb7fff000 0xb8000000     0x1000    0x1b000 /lib/ld-2.11.2.so
               0xbfffeb000 0xc0000000    0x15000         0 [stack]
(gdb)
```

Stack 1:

Little endian é quando a transferência de dados entre o processador e a memória vai de trás para frente. Exemplo, se eu passar um alfabeto, ele será passado como começando do z até o a.

Se eu passar como argumento a letra "a" e der uma cadeia de strings "cba", ele vai ler a primeira letra (que no caso é a letra a porque é little endian) e vai alocar para o espaço de memória designado. Com isso, aparece a mensagem "you have correctly got the variable to the right value"

