

```
1 /* https://sploitfun.wordpress.com/2015/02/10/understanding-glibc-malloc/ */
2
3 /* Per thread arena example. */
4 #include <stdio.h>
5 #include <stdlib.h>
6 #include <pthread.h>
7 #include <unistd.h>
8 #include <sys/types.h>
9
10 void* threadFunc(void* arg) {
11     printf("Before malloc in thread 1\n");
12     getchar();
13     char* addr = (char*) malloc(1000);
14     printf("After malloc and before free in thread 1\n");
15     getchar();
16     free(addr);
17     printf("After free in thread 1\n");
18     getchar();
19 }
20
21 int main() {
22     pthread_t t1;
23     void* s;
24     int ret;
25     char* addr;
26
27     printf("Welcome to per thread arena example::%d\n",getpid());
28     printf("Before malloc in main thread\n");
29     getchar();
30     addr = (char*) malloc(1000);
31     printf("After malloc and before free in main thread\n");
32     getchar();
33     free(addr);
34     printf("After free in main thread\n");
35     getchar();
36     ret = pthread_create(&t1, NULL, threadFunc, NULL);
37     if(ret)
38     {
39         printf("Thread creation error\n");
40         return -1;
41     }
42     ret = pthread_join(t1, &s);
43     if(ret)
44     {
```

```
45         printf("Thread join error\n");
46         return -1;
47     }
48     return 0;
49 }
```