

并行与分布式作业

“ 黄炜钊”
第三次作业

姓名：黄炜钊
班级：行政三班
学号：18340066

一、 问题描述

利用 LLVM（C、C++）或者 Soot（Java）等工具检测多线程程序中潜在的数据竞争以及是否存在不可重入函数，给出案例程序并提交分析报告。

二、 解决方案

编写下面的程序：

```
#include <pthread.h>
#include <stdio.h>

int Global;

void *Thread1(void *x) {
    Global++;
    return NULL;
}

void *Thread2(void *x) {
    Global--;
    return NULL;
}

int main() {
    pthread_t t[2];
    pthread_create(&t[0], NULL, Thread1, NULL);
    pthread_create(&t[1], NULL, Thread2, NULL);
    pthread_join(t[0], NULL);
    pthread_join(t[1], NULL);
}
```

然后使用 ThreadSanitizer 检测数据竞争，得到实验结果如下图所示。

为了解决数据冲突的现象，定义一个互斥量 mutex，使得同一时间只有一个线程在执行临界区的代码，其他线程只能等

待。等当前线程执行完之后，会从等待的线程中随机选一个进入临界区。修改代码如下：

```
#include <pthread.h>
#include <stdio.h>

int Global;
pthread_mutex_t mutex;

void *Thread1(void *x) {
    pthread_mutex_lock(&mutex);
    Global++;
    pthread_mutex_unlock(&mutex);
    return NULL;
}

void *Thread2(void *x) {
    pthread_mutex_lock(&mutex);
    Global--;
    pthread_mutex_unlock(&mutex);
    return NULL;
}

int main() {
    pthread_t t[2];
    pthread_create(&t[0], NULL, Thread1, NULL);
    pthread_create(&t[1], NULL, Thread2, NULL);
    pthread_join(t[0], NULL);
    pthread_join(t[1], NULL);
}
```

实验结果见图二。

三、 实验结果

图一

```

condor@condor-virtual-machine:~$ clang -fsanitize=thread -g tiny_race.c
condor@condor-virtual-machine:~$ ./a.out
=====
WARNING: ThreadSanitizer: data race (pid=2502)
  Write of size 4 at 0x0000014ca9b8 by thread T2:
    #0 Thread2 /home/condor/tiny_race.c:12 (a.out+0x4b877e)

  Previous write of size 4 at 0x0000014ca9b8 by thread T1:
    #0 Thread1 /home/condor/tiny_race.c:7 (a.out+0x4b871e)

  Location is global '<null>' at 0x000000000000 (a.out+0x0000014ca9b8)

  Thread T2 (tid=2505, running) created by main thread at:
    #0 pthread_create ??? (a.out+0x426ff6)
    #1 main /home/condor/tiny_race.c:19 (a.out+0x4b87fc)

  Thread T1 (tid=2504, finished) created by main thread at:
    #0 pthread_create ??? (a.out+0x426ff6)
    #1 main /home/condor/tiny_race.c:18 (a.out+0x4b87d6)

SUMMARY: ThreadSanitizer: data race /home/condor/tiny_race.c:12 in Thread2
=====
ThreadSanitizer: reported 1 warnings
condor@condor-virtual-machine:~$ ./a.out
=====
WARNING: ThreadSanitizer: data race (pid=2514)
  Write of size 4 at 0x0000014ca9b8 by thread T2:
    #0 Thread2 /home/condor/tiny_race.c:12 (a.out+0x4b877e)

  Previous write of size 4 at 0x0000014ca9b8 by thread T1:
    #0 Thread1 /home/condor/tiny_race.c:7 (a.out+0x4b871e)

  Location is global '<null>' at 0x000000000000 (a.out+0x0000014ca9b8)

  Thread T2 (tid=2517, running) created by main thread at:
    #0 pthread_create ??? (a.out+0x426ff6)
    #1 main /home/condor/tiny_race.c:19 (a.out+0x4b87fc)

  Thread T1 (tid=2516, finished) created by main thread at:
    #0 pthread_create ??? (a.out+0x426ff6)
    #1 main /home/condor/tiny_race.c:18 (a.out+0x4b87d6)

SUMMARY: ThreadSanitizer: data race /home/condor/tiny_race.c:12 in Thread2
=====
ThreadSanitizer: reported 1 warnings
condor@condor-virtual-machine:~$

```

可以发现，多线程对全局变量的值的修改确实引起了数据竞争。

图二

```

=====
WARNING: ThreadSanitizer: data race (pid=2502)
  Write of size 4 at 0x0000014ca9b8 by thread T2:
    #0 Thread2 /home/condor/tiny_race.c:12 (a.out+0x4b877e)

  Previous write of size 4 at 0x0000014ca9b8 by thread T1:
    #0 Thread1 /home/condor/tiny_race.c:7 (a.out+0x4b871e)

  Location is global '<null>' at 0x000000000000 (a.out+0x0000014ca9b8)

  Thread T2 (tid=2505, running) created by main thread at:
    #0 pthread_create ??? (a.out+0x426ff6)
    #1 main /home/condor/tiny_race.c:19 (a.out+0x4b87fc)

  Thread T1 (tid=2504, finished) created by main thread at:
    #0 pthread_create ??? (a.out+0x426ff6)
    #1 main /home/condor/tiny_race.c:18 (a.out+0x4b87d6)

SUMMARY: ThreadSanitizer: data race /home/condor/tiny_race.c:12 in Thread2
=====
ThreadSanitizer: reported 1 warnings
condor@condor-virtual-machine:~$ ./a.out
=====
WARNING: ThreadSanitizer: data race (pid=2514)
  Write of size 4 at 0x0000014ca9b8 by thread T2:
    #0 Thread2 /home/condor/tiny_race.c:12 (a.out+0x4b877e)

  Previous write of size 4 at 0x0000014ca9b8 by thread T1:
    #0 Thread1 /home/condor/tiny_race.c:7 (a.out+0x4b871e)

  Location is global '<null>' at 0x000000000000 (a.out+0x0000014ca9b8)

  Thread T2 (tid=2517, running) created by main thread at:
    #0 pthread_create ??? (a.out+0x426ff6)
    #1 main /home/condor/tiny_race.c:19 (a.out+0x4b87fc)

  Thread T1 (tid=2516, finished) created by main thread at:
    #0 pthread_create ??? (a.out+0x426ff6)
    #1 main /home/condor/tiny_race.c:18 (a.out+0x4b87d6)

SUMMARY: ThreadSanitizer: data race /home/condor/tiny_race.c:12 in Thread2
=====
ThreadSanitizer: reported 1 warnings
condor@condor-virtual-machine:~$ clang -fsanitize=thread -g tiny_race.c
condor@condor-virtual-machine:~$ ./a.out
condor@condor-virtual-machine:~$

```

可以看到数据竞争的问题已经解决了。