

Homework 4

EunJong Lee, 21700556, 21700556@handong.edu

1. Introduction

HW4's main topic is that detect dead lock status of target program without fixing the target program. To implement this, I used dynamic linking concept, and I used fifo to communicate between target program and checker program. To find deadlock, I used lock graph and check the graph has cycle.

2. Approach

Designed protocol to communication.

To get information to make lock graph, checker part get information from probe part by using fifo. To send information, probe part send information to checker part by using fifo.

messages that sent and received between checker part and probe part are as follows. Message contains (lock_info (lock or unlock) , mutex id, current thread id, memory address about location of pthread_mutex_lock). Information can be distinguished by ' ' (space).

I used O_SYNC option to handle synchronization between two parts.

Ddmon.c part

Ddmon.c is shared library. In ddmon.c, I overrode pthread_mutex_lock function and pthread_mutex_unlock function. To get original functions, I used dlsym function.

Workflow of ddmon.c

1. Open fifo with WRONLY and O_SYNC option (.ddtrace).
2. Get backtrace information(using backtrace function).
3. Get current pthread_id(using pthread_self()).
4. Write message to fifo.
5. Get original pthread_mutex_lock function's declaration(using dlsym).
6. Execute original pthread_mutex_lock function and return.

Workflow of ddcheck.c

1. Open fifo(.ddtrace) with O_RDONLY and O_SYNC options.
2. Read information from fifo.
3. Check isLock variable.
4. If isLock is 1(target program get lock).
 - A. draw edge.
 - B. Check whether the graph has cycle, and if cycle exist. Print "dead lock occurred" and print mutex&pthread information.
5. If isLock is 0(target program release lock).
 - A. Delete edge.

How to make lock graph?

I used threads, t_info, node, edge array to make lock graph.

Threads have thread id.

T_info have mutex lock information which is thread have.

Node have mutex's address.

Edge have edge information.

I used lock graph algorithm.

How to check cycle?

I used dfs to check cycle.

3. Evaluation

I used only one programs to check my program works well because I close to the deadline.

The program is ABBA dead lock program.

In this program, my program works well.

Limitation

I used target program that used no more than 2 threads.

If deadlock is not occurred, checker program go infinite loop.

4. Discussion

I usually fixed target program's code to debug the program, but sometimes fixed code make another problem. during do this homework, I learned debugging method which is not change original code. Also, I think backtrace and addr2line is good debugging tools. I will use those technics to debug the program.

I understand more about how deadlock occurs during implement lock graph algorithm.

I didn't know how to use core dump file, but while I do this homework, I learned how to use core dump file. I used core dump file to find source code line.

I think if ddchck fork new process and excute ddmon, it will be more similar to an actual debugging tool.

5. Conclusion

I implement debugging tool to detect deadlock occur in this homework. I leared how to use dynamic linking and how trace source code line number. Also, I learned more about why deadlock occurred.