

Forum:[Hack Forums](#)

Thread: [Is there any way to safely use a mutex in a detached thread?](#)

Goodies - September 10, 2016, 9:16 pm

This code has a fairly obvious race condition. Since the stack of detached threads is not unwound, objects' destructors may well not be called.

About 15-20% of the time I run this code in debug mode, it throws an error saying that a mutex was destroyed while in use. Fairly obviously.

This is what the error looks like: [\(Click to View\)](#)

PHP Code

```
<?php
#include <thread>
#include <memory>
#include <iostream>
#include <numeric>
#include <mutex>

auto mu = std::make_shared<std::mutex>();

struct Test {
    int nums[10];
    int sum(void) const {
        int s = 0;
        for (int i = 0; i < 10; ++i)
            s += nums[i];
        return s;
    }
};

void bar(std::shared_ptr<Test> test) {
    std::lock_guard<std::mutex> lock(*mu);
    std::cout << test->sum() << std::endl;
}

void foo(int i) {
    for (int ctr = 0; ctr < i; ++ctr) {
        auto ptr = std::make_shared<Test>(Test());
        // perhaps alter ptr->nums...
        auto t = std::make_shared<std::thread>(std::bind(&bar, ptr));
        t->detach();
    }
}

int main(int argc, char **argv) {
```

```
std::thread t(std::bind(&foo, 10));  
t.join();  
return 0;  
}
```

Now, in a project I'm working on, I have an async event loop that needs to send data to a user-provided callback. Since the event loop itself cannot be blocked, I need to spawn the callback in a new thread and detach it so I can continue the event loop, and simply assume the callback does not viciously violate thread safety. It is similar to the setup I have: a thread other than the main thread runs an event loop. Functions called by the event loop will spawn a thread for the callback.

Assuming the user uses a mutex, it seems there is no way to guarantee safety here...

Any tips? Do I have some major design flaws? Help is appreciated.

sqsgkrldwg - September 11, 2016, 7:26 pm

Hi,

Is it really necessary to use a pointer on the mutex ?

Also : Detached threads are not joinable.

Goodies - September 11, 2016, 11:53 pm

(Sep 11, 2016 07:26 PM)Leffe Wrote: [\[CITE\]](#) Hi,

Is it really necessary to use a pointer on the mutex ?

Also : Detached threads are not joinable.

Well I had it set up differently at first, fyi, but no, not necessarily.