

---

Subject: Re: Problem breaking loop (with close button) in main thread

Posted by [Lance](#) on Wed, 19 Oct 2022 23:04:39 GMT

[View Forum Message](#) <> [Reply to Message](#)

---

Hi Awksed

I am sorry to hear that you have to satisfy with a compromise.

I manage to create the following test. It runs as expected on Linux, but would not work on Windows.

Regards,  
Lance

---

-----

Correction: No, it's not a problem with ProcessEvents(). After I change the thread lambda to not log on GUI but to debug output file, it runs like a charm. It's interesting. I probably should not modify GUI from within other than the GUI thread. Seems Linux+GTK are more tolerant to this kind of error.

```
void Start(){
    static int cnt;
    if( cnt!= 0 )
        return;
    ++ cnt;
    stop_requested = false;
    log.Clear();
    badguy.Run([this]{
        LOG("Badguy running...\n"); // CHANGE THIS
        Mutex::Lock _(mut);
        LOG("Badguy now owns the Mutex.\n"); // CHANGE THIS
        while(!stop_requested && !IsShutdownThreads() ){
            Sleep(50);
        }
        LOG("Stop requested, Badguy exiting and releasing the Mutex...\n"); // AND THIS
    });

    // wait for badguy to acquire the mutex
    while(mut.TryEnter()){
        mut.Leave();
        ProcessEvents();
        Sleep(100);
    }

    log.Append("Trying to acquire the mutex...\n");
    GuiAcquireMutex();
    --cnt;
}
```

## File Attachments

---

1) [DeadMutex.zip](#), downloaded 194 times

---