

writing files

ryantm.com

https://github.com/ryantm/lvrug_atomic

```
File.write('atomic1.txt', 'Hello world')
```

strace (Linux)

```
$ strace ruby atomic1.rb
```

try dtruss on OS X

```
$ strace ruby atomic1.rb
```

```
....
```

```
open("atomic1.txt", O_WRONLY|O_CREAT|O_TRUNC|O_CLOEXEC, 0666) =  
7  
fcntl(7, F_GETFD) = 0x1 (flags FD_CLOEXEC)  
fstat(7, {st_mode=S_IFREG|0644, st_size=0, ...}) = 0  
ioctl(7, SNDCTL_TMR_TIMEBASE or SNDRV_TIMER_IOCTL_NEXT_DEVICE or  
TCGETS, 0x7fffcafe2b60) = -1 ENOTT  
Y (Inappropriate ioctl for device)  
write(7, "Hello world", 11) = 11  
close(7) = 0
```

```
...
```

between open and write

reading might read nothing (truncated)

writes to file erased

```
File.open('atomic2.txt', 'w') do |f|  
  1111.times do  
    f.write 'Hello world'  
  end  
end
```

```
$ strace ruby atomic2.rb
```

```
open("atomic2.txt", O_WRONLY|O_CREAT|O_TRUNC|O_CLOEXEC, 0666) =  
7  
fcntl(7, F_GETFD) = 0x1 (flags FD_CLOEXEC)  
fstat(7, {st_mode=S_IFREG|0644, st_size=0, ...}) = 0  
ioctl(7, SNDCTL_TMR_TIMEBASE or SNDRV_TIMER_IOCTL_NEXT_DEVICE or  
TCGETS, 0x7fff12593520) = -1 ENOTT  
Y (Inappropriate ioctl for device)  
write(7, "Hello worldHello worldHello worl"..., 8184) = 8184  
write(7, "Hello world", 11) = 11  
write(7, "Hello worldHello worldHello worl"..., 4026) = 4026  
close(7)
```

between write and write

reader reads partial (corrupted?) data

other writers

- truncation
- interleaving

solutions

use temporary files - simple

use journaling - used by most databases

transactional file systems

- NTFS has but deprecated

- otherwise academic

```
require 'fileutils'  
File.write('temp.txt', 'hello  
world')  
FileUtils.rm_f('temp.txt')  
"atomic5.txt"
```

```
$ strace ruby atomic5.rb
```

```
open("temp.txt", O_WRONLY|O_CREAT|O_TRUNC|O_CLOEXEC, 0666) = 7
```

```
...
```

```
write(7, "hello world", 11)          = 11
```

```
close(7)                             = 0
```

```
stat("atomic5.txt", {st_mode=S_IFREG|0644, st_size=11, ...}) = 0
```

```
stat("temp.txt", {st_mode=S_IFREG|0644, st_size=11, ...}) = 0
```

```
stat("atomic5.txt", {st_mode=S_IFREG|0644, st_size=11, ...}) = 0
```

```
stat("temp.txt", {st_mode=S_IFREG|0644, st_size=11, ...}) = 0
```

```
stat("atomic5.txt", {st_mode=S_IFREG|0644, st_size=11, ...}) = 0
```

```
rename("temp.txt", "atomic5.txt")    = 0
```

requirements

no one read temp file

only one writer of temp file

only one file updater (file write locking)

further research

how a rollback journal works

<http://www.sqlite.org/atomiccommit.html>

further research

how a rollback journal works

<http://www.sqlite.org/atomiccommit.html>

references.txt in

https://github.com/ryantm/lvrug_atomic

questions

ryantm.com

https://github.com/ryantm/lvrug_atomic