



## Learn, Share, Build

Each month, over 50 million developers come to Stack Overflow to learn, share their knowledge, and build their careers.

Google

Facebook

OR

Join the world's largest developer community.

## What generates the "text file busy" message in Unix?

What operation generates the error "text file busy"? I am unable to tell exactly.

I think it is related to the fact that I'm creating a temporary python script (using tempfile) and using execl from it, but I think that execl changes the file being run.

unix

edited May 27 '13 at 2:25

asked May 27 '13 at 0:25



Penz

1,364 3 15 23

### 9 Answers

This error means some other process or user is accessing your file. Use `ls -lsof` to check what other processes are using it. You can use `kill` command to kill it if needed.

answered May 27 '13 at 0:30



jaypal singh

50.8k 13 71 107

53 The Text file busy error in specific is about trying to modify an executable while it is executing. The "Text" here refers to the fact that the file being modified is the [text segment](#) for a running program. This is a very special case, and not the generic one that your answer seems to suggest. Even so, your answer isn't entirely incorrect. – [ArjunShankar](#) Jan 23 '14 at 21:14

The answer with the comment seems complete. – [Penz](#) Dec 6 '16 at 14:26

The OP asked which operation generates the error, not for an explanation of what the error means. – [Knickerless-Noggins](#) Jul 13 at 7:06

It's a while since I've seen that message, but it used to be prevalent in System V R3 or thereabouts a good couple of decades ago. Back then, it meant that you could not change a program executable while it was running.

For example, I was building a `make` workalike called `rmk`, and after a while it was self-maintaining. I would run the development version and have it build a new version. To get it to work, it was necessary to use the workaround:

```
gcc -g -Wall -o rmk1 main.o -L. -lrmk -L/Users/jleffler/lib/64 -lj1
if [ -f rmk ] ; then mv rmk rmk2 ; else true; fi ; mv rmk1 rmk
```

So, to avoid problems with the 'text file busy', the build created a new file `rmk1`, then moved the old `rmk` to `rmk2` (rename wasn't a problem; `unlink` was), and then moved the newly built `rmk1` to `rmk`.

I haven't seen the error on a modern system in quite a while...but I don't all that often have programs rebuilding themselves.

answered May 27 '13 at 4:09



Jonathan Leffler

497k 75 576 921

2 Here's a super quick reproducer: `echo -e '#include <unistd.h>\nint main(void){sleep`

```
(5);return 0;}' > slowprog.c && cc slowprog.c && cp a.out b.out && (. /a.out &) ; sleep
1 && cp b.out a.out . Produced the error message "cp: cannot create regular file 'a.out': Text file busy"
on my newish Fedora. – ArjunShankar Jan 23 '14 at 21:07
```

- 2 Of course, this answer is correct and gets a +1. You might want to remove the "It has been a while" disclaimer. – ArjunShankar Jan 23 '14 at 21:10

Don't know the cause but I can contribute a quick and easy work around.

I just experienced this this oddity on CentOS 6 after "cat > shScript.sh" (paste, ^Z) then editing the file in KWrite. Oddly there was no discernible instance (ps -ef) of the script executing.

My quick work around was simply to "cp shScript.sh shScript2.sh" then I was able to execute shScript2.sh. Then I deleted both. Done!

answered Jul 26 '14 at 14:10



Scott Welker  
444 4 10

You may find this to be more common on CIFS/SMB network shares. Windows doesn't allow for a file to be written when something else has that file open, and even if the service is not Windows (it might be some other NAS product), it will likely reproduce the same behaviour. Potentially, it might also be a manifestation of some underlying NAS issue vaguely related to locking/replication.

answered Apr 18 '16 at 22:02



Cameron Kerr  
546 2 11

One of my experience:

I always change the default keyboard shortcut of Chrome through reverse engineering. After modification, I forgot to close Chrome and ran the following:

```
sudo cp chrome /opt/google/chrome/chrome
cp: cannot create regular file '/opt/google/chrome/chrome': Text file busy
```

Using strace, you can find the more details:

```
sudo strace cp ./chrome /opt/google/chrome/chrome 2>&1 |grep 'Text file busy'
open("/opt/google/chrome/chrome", O_WRONLY|O_TRUNC) = -1 ETXTBSY (Text file busy)
```

answered Mar 17 at 12:53



firo  
303 4 15

This occurs when you try and write to a file that is currently being executed by the kernel, or execute a file that is currently open for writing.

Source: <http://wiki.wlug.org.nz/ETXTBSY>

answered Jul 28 at 11:13



Messa  
13.2k 1 21 37

I came across this in PHP when using `fopen()` on a file and then trying to `unlink()` it before using `fclose()` on it.

No good:

```
$handle = fopen('file.txt');
// do something
unlink('file.txt');
```

Good:

```
$handle = fopen('file.txt');
// do something
fclose($handle);
unlink('file.txt');
```

edited May 4 at 9:17



Lưu Vĩnh Phúc  
10.7k 6 33 113

answered Nov 9 '15 at 20:09



dtbarne  
5,867 2 29 42

On windows I guess? On linux the system usually allows us to delete open files - the reference in the

directory is eliminated, but the data (inode) is freed only when the number of references reaches 0. – [Penz](#) Nov 10 '15 at 22:43

No, this was on Centos. – [dtbarne](#) Nov 18 '15 at 23:17

Tested it on Linux 4.7.10 with ext4 filesystem and it haven't produced any error, worked as Penz mentioned. File deleted successfully. Maybe dtbarne is using some special filesystem. – [k3a](#) Mar 10 at 20:03

Was running this on vagrant - might be due to it being a shared folder. – [dtbarne](#) Mar 17 at 19:44

If trying to build `phpredis` on a Linux box you might need to give it time to complete modifying the file permissions, with a `sleep` command, before running the file:

```
chmod a+x /usr/bin/php/scripts/phpize \  
&& sleep 1 \  
&& /usr/bin/php/scripts/phpize
```

answered Oct 1 at 14:15

 [Stephane](#)

1,621 7 37 64

I don't think `chmod` would return before the permissions were set. That might be a filesystem issue. – [Penz](#) Oct 2 at 15:02

This occurred inside a Docker image being built. – [Stephane](#) Oct 3 at 8:07


1 Docker has multiple storage drivers, I guess not all of them are perfect. – [Penz](#) Oct 3 at 20:18

In my case, I was trying to execute a shell file (with an extension `.sh`) in a `csh` environment, and I was getting that error message.

just running with `bash` it worked for me. For example

```
bash file.sh
```

answered Oct 2 at 12:27

 [Rafayel Paremuzyan](#)

1 1

Did it have a `#!/bin/bash` header? – [Penz](#) Oct 2 at 14:59

It has the following header `#!/bin/sh` – [Rafayel Paremuzyan](#) Oct 2 at 19:01

You might want to try using `#!/usr/bin/csh` or equivalent. – [Penz](#) Oct 3 at 20:19