Segfaults potentially related to 32-bit libraries

Thu, 02/20/2020 - 17:17

#1

Hello Tim -

I've had many, many problems keeping cisTEM running. We have an SBGrid installation on a data server, and source it from our workstations. This is relevant because 2D classification can run on one workstation, but dies very frequently on the other. Often we switch workstations just for that step, even though both are using the **exact same pre-compiled binaries.**

I've spent a good amount of time troubleshooting this, and I think I have found something that may be useful to you guys:

dmesg reports lines like this after it crashes:

```
[1032090.216520] merge2d[223764]: segfault at
5400000053 ip 0000005400000053 sp 00007ffe776a68f8
error 14 in UTF-32.so[7fb74d9ca000+2000]
[1120935.158934] merge2d[165929]: segfault at
5400000053 ip 0000005400000053 sp 00007ffcd1d70578
error 14 in UTF-32.so[7f3a565ea000+2000]
[1464120.322251] merge2d[131094]: segfault at
5400000053 ip 0000005400000053 sp 00007ffe2d3687f8
error 14 in UTF-32.so[7fe4e0859000+2000]
```

Segfautls aren't *always* in UTF-32.so, often we have observed errors in glibc as well. Crashes in 3D occur but much, much less frequently.

Both workstations give identical output for:
```
$ ldd --version
ldd (GNU libc) 2.17
Copyright (C) 2012 Free Software Foundation, Inc.
This is free software; see the source for copying
conditions.  There is NO
warranty; not even for MERCHANTABILITY or FITNESS
FOR A PARTICULAR PURPOSE.
Written by Roland McGrath and Ulrich Drepper.
```

BUT - I found that I had 32-bit libraries on one system and not the other.

```
[STABLE System]$ yum list glibc
...
Installed Packages
glibc.x86_64
2.17-292.el7
@base
Available Packages
glibc.i686                                              2.17-
292.el7                                                 base
```

vs

```
[UNSTABLE System]$ yum list glibc
...

Installed Packages
glibc.i686                                              2.17-
292.el7                                                 @base
glibc.x86_64
```

I uninstalled the 32-bit libraries on the unstable system, and it seems to be running 2D classification without problems now. I'll report back if anything changes.

Does cisTEM use the system-level glibc/UTF-32? Does it check for x86 vs 64-bit presence?

Thank you for your time and efforts. Just sharing what I've observed, hopefully it's useful to you all as you wrap up cisTEM2.

D. John Lee
Fraser Lab
UCSF

djohnlee

Sigh. OK so it made it

Sigh. OK so it made it through our smaller test dataset, but then it threw another UTF-32.so segfault on the very large dataset. You probably should disregard this, I'm still troubleshooting.

timgrant

Thanks for looking into this!

I had thought that cisTEM always uses the 64-bit libraries. In this case, I think UTF-32 corresponds to the length of the UTF encoding rather than whether the library is 32 bit or 64-bit?

Tim

djohnlee

Hello Tim,

You're right, UTF-32 is the encoding, not the 64bit/32bit difference. I was excited because I've been trying to figure out why one system is stable (and the other isn't) and thought I had found the key difference between them.

They are both Skylake-SP platforms, they both have over 256 GB RAM, they both run CentOS7, they both connect to the same data node with the same copy of cisTEM. I'm 99% sure it's a problem with a system-level package that is either different or corrupt on the unstable system. I've been sort of pulling my hair out trying to figure out why I can't run 2D on one. And especially since it's always the same few segfaults - I don't have any other instability. If I had general/random segfaults or reboots I would suspect bad RAM or maybe some other hardware failure.

Thanks for your help/time. If I figure it out, I'll definitely post back (even though, as I've said, it's almost certainly a OS or package problem on the unstable system)

John

Source URL:https://cistem.org/segfaults-potentially-related-32-bit-libraries?page=0