GROMACS - Bug #2803
segmentation fault in LINCS
12/12/2018 02:19 AM - Szilárd Páll

<table>
<thead>
<tr>
<th>Status:</th>
<th>Closed</th>
</tr>
</thead>
<tbody>
<tr>
<td>Priority:</td>
<td>Normal</td>
</tr>
<tr>
<td>Assignee:</td>
<td></td>
</tr>
<tr>
<td>Category:</td>
<td>mdrun</td>
</tr>
<tr>
<td>Target version:</td>
<td>2019-rc1</td>
</tr>
<tr>
<td>Affected version</td>
<td>2019-beta3</td>
</tr>
<tr>
<td>Difficulty:</td>
<td>uncategorized</td>
</tr>
</tbody>
</table>

Description
Running the adh_dodec_vsites input with 64 ranks x 4 threads on an ARMv8 node; the repro of the segv takes anywhere between tens to thousands of millions of steps but it has been reproduced several times.

(gdb) bt
#0 0x0000000000061dc5c in gmx::constrain_lincs (computeRmsd=<optimized out>, ir=..., step=step=entry=1555720,
lincsd=0xfffbe0278ee0, md=..., cr=0xfffbe054f80, ms=..., x=x8entry=0xfffbe087b700, xprime=<optimized out>,
, xprime=entry=0xfffbe08a3e00, min_proj=<optimized out>, min_proj=entry=0x0, box=<optimized out>,
, box=entry=0x0,
, pbc=<optimized out>, pbc=entry=0x0, lambda=lambda=entry=0, dvdlambda=<optimized out>, dvdlambda=entry=0xfff5e7fca0,
, invdt=invdt=entry=200, v=optimized out, v=entry=0xfffbe0623000, bCalcVir=<optimized out>, bCalcVir=entry=false,
, vir_r_m_dr=<optimized out>, vir_r_m_dr=entry=0xfff5e7fc8e8, econq=<optimized out>, econq=entry=gmx::Positions,
rnr=optimized out, maxwarn=optimized out, warncount=entry=0xfffbe027606c)
at /home/pszilard/gromacs-19/src/gromacs/mdlib/lincs.cpp:2565
#1 0x00000000000614b60 in gmx::Constraints::Impl::apply (this=<optimized out>,
, bLog=optimized out, step=1555720, delta_step=65535, delta_step=entry=1, step_scaling=step_scaling=entry=1, x=0xfff
be087b700, xprime=0xfffbe08a3e00,
, min_proj=min_proj=entry=0x0, box=optimized out, lambda=0, dvdlambda=optimized out, v=<optimized out>,
, econq=optimized out) at /home/pszilard/gromacs-19/src/gromacs/mdlib/constr.cpp:449
#2 0x00000000000615560 in gmx::Constraints::apply (this=<optimized out>, bLog=optimized out), bEn
er=optimized out, step=<optimized out>, delta_step=delta_step=entry=1, step_scaling=step_scaling=entry=1, x=<optimized out>,
, xprime=<optimized out>, min_proj=min_proj=entry=0x0, box=box=entry=0xfffbe02cd2e4, lambda=optimized out,
dvdlambda=dvdlambda=entry=0xfff5e7fca0, v=0xfffbe0623000, vir=vir=entry=0x0, econq=econq=entry=gmx::Positions)
at /home/pszilard/gromacs-19/src/gromacs/mdlib/constr.cpp:317
#3 0x00000000000640578 in constrain_coordinates (step=<optimized out>, dvdlambda=dvdlambda=entry=0
fff5e7fca0, state=state=entry=0xfffbe02cd2b0, vir_part=vir_part=entry=0xfff5e7fcbc0, upd=<optimized out>,
, constr=<optimized out>,
, bCalcVir=bCalcVir=entry=false, do_log=do_log=entry=false, do_ene=do_ene=entry=false)
at /home/pszilard/gromacs-19/src/gromacs/math/vec.h:570
#4 0x0000000000095b178 in gmx::Integrator::do_md (this=<optimized out>, this=entry=0xfff5e7fd7a0, this=entry=0xfff5e7ff100)
at /home/pszilard/gromacs-19/src/gromacs/mdrun/md.cpp:1156
#5 0x0000000000095941c in gmx::Integrator::run (this=entry=0xfff5e7ff100, ei=<optimized out>,
, doRerun=doRerun=entry=true)
at /home/pszilard/gromacs-19/src/gromacs/mdrun/integrator.cpp:72
#6 0x00000000000683b94 in gmx::Mdrunner::mdrunner (this=entry=0xfff5e7fe7d0)
at /home/pszilard/gromacs-19/src/gromacs/mdrun/runner.cpp:1434
#7 0x000000000006854d8 in gmx::mdrunner_start_fn (arg=<optimized out>) at /home/pszilard/gromacs-19/src/gromacs/mdrun/runner.cpp:219
#8 0x00000000006ea028 in tMPI_Thread_starter (arg=0x1e6e3930) at /home/pszilard/gromacs-19/src/external/thread_mpi/src/tmpi_init.cpp:399
#9 0x0000ffff88a97bb0 in start_thread () from /lib64/libpthread.so.0
#10 0x0000ffff886eb4c0 in thread_start () from /lib64/libc.so.6
(gdb) l
2560     {
2561         cconerr(lincsd, xprime, pbc,
2562                 &ncons_loc, &p_ssd, &p_max, &p_imax);
2563         if (isMultiSim(&ms))
2564             {               
2565                 sprintf(buf3, " in simulation %d", ms.sim);
2566             }
2567         else
2568         {
2569             buf3[0] = 0;
(gdb)

bash 2-$ bash (3*$bash)

Associated revisions
Revision b23d1234 - 12/14/2018 02:25 PM - Berk Hess
Fixed nullptr dereference in LINCS error
Sometimes a dereferenced nullptr could be passed as a reference
to gmx_multisim_t to the Constraint factory function.
Changed the reference in the Constraint object to a pointer.
Fixed #2803
Change-Id: i4806069973067d27078a1324d18a406c7b3e227d

History
#1 - 12/12/2018 10:07 AM - Berk Hess
I think the issue is that a dereferenced nullptr to gmx_multisim_t is passed to the constructor for Constraints. I thought C++ would not allow this, but
maybe the implicit arg list somehow circumvents the check. Anyhow, the isMultiSim check checks for nullptr, so we should pass a pointer not a
reference.

#2 - 12/12/2018 10:27 AM - Gerrit Code Review Bot
Gerrit received a related patchset '1' for Issue #2803.
Uploader: Berk Hess (hess@kth.se)
Change-Id: gromacs~release-2019~I4806069973067d27078a1324d18a406c7b3e227d
Gerrit URL: https://gerrit.gromacs.org/8812

#3 - 12/12/2018 10:27 AM - Berk Hess
- Status changed from New to Fix uploaded
- Target version set to 2019-rc1

#4 - 12/12/2018 10:31 AM - Berk Hess
Unrelated to the segv, but do we expect LINCS warnings in this system?

#5 - 12/12/2018 11:11 AM - Szilárd Páll
- Description updated

Berk Hess wrote:
I think the issue is that a dereferenced nullptr to gmx_multisim_t is passed to the constructor for Constraints. I thought C++ would not allow this, but
maybe the implicit arg list somehow circumvents the check. Anyhow, the isMultiSim check checks for nullptr, so we should pass a pointer not a
reference.

Interesting -- I also thought isMultiSim should not allow the 2565 to execute, that's why I included the line. I am still unsure why passing by reference
makes isMultiSim return true.
Unrelated to the segv, but do we expect LINCS warnings in this system?

Not that I know of.

What's also weird is why does this happen once so rarely? `ms == NULL` throughout the entire run.

ms is only used during LINCS warnings. You don't get those more often, or do you?

NO, I've never seen one -- at least not during these tests I've been doing on ARMv8.

ms is only used during LINCS warnings. You don't get those more often, or do you?

NO, I've never seen one -- at least not during these tests I've been doing on ARMv8.

- Status changed from Fix uploaded to Resolved

going to close this now

- Status changed from Resolved to Closed