Running preprocessing-tools-test compiled with gcc 4.9 with asan gives:

```
==508==ERROR: AddressSanitizer: heap-buffer-overflow on address 0x6240000060d0 at pc 0x7ffff5703143 bp 0x7fffffffb4a0 sp 0x7fffffffb498
READ of size 4 at 0x6240000060d0 thread T0
 #0 0x7ffff5703142 in copy_rvec /mnt/workspace/roland-temp/gromacs/src/gromacs/math/vec.h:330
 #1 0x7ffff5703142 in add_conf /mnt/workspace/roland-temp/gromacs/src/gromacs/gmxpreprocess/addconf.c:437
 #2 0x7ffff57c3355 in add_solv /mnt/workspace/roland-temp/gromacs/src/gromacs/gmxpreprocess/solvate.cpp:473
 #3 0x7ffff57c3355 in gmx_solvate /mnt/workspace/roland-temp/gromacs/src/gromacs/gmxpreprocess/solvate.cpp:780
 #4 0x55555558d3fb in TestBody /mnt/workspace/roland-temp/gromacs/src/gromacs/gmxpreprocess/tests/solvate.cpp:86
 #5 0x55555556ab9a in HandleSehExceptionsInMethodIfSupported::testing::Run()
 /mnt/workspace/roland-temp/gromacs/src/external/gmock-1.7.0/gtest/src/gtest.cc:2078
 #6 0x55555556ab9a in void testing::internal::HandleExceptionsInMethodIfSupported::testing::Run()
 /mnt/workspace/roland-temp/gromacs/src/external/gmock-1.7.0/gtest/src/gtest.cc:2326
 #9 0x5555555fa493 in testing::internal::UnitTestImpl::Run()
 /mnt/workspace/roland-temp/gromacs/src/external/gmock-1.7.0/gtest/src/gtest.cc:3929
 #14 0x555555570194 in RUN_ALL_TESTS
 /mnt/workspace/roland-temp/gromacs/src/external/gmock-1.7.0/gtest/include/gtest/gtest.h:2288
 #15 0x555555570194 in main /mnt/workspace/roland-temp/gromacs/src/testutils/unittest_main.cpp:63
 #16 0x7fffff2d064e in __libc_start_main (/lib/x86_64-linux-gnu/libc.so.6+0x21de4)
 #17 0x555555572a38 in __libc_start_main (/lib/x86_64-linux-gnu/libc.so.6+0x21de4)
```

Running it in gdb (also when compiled with gcc 4.8.2) with "break addconf.c:437 if jnr<0" shows that this gets negatives which it shouldn't. It might be better if this code were using the simple NS from analysis-tools instead of how do_nsgrid is currently implemented.

This is with current master (ee53bf96df)

Related issues:

Related to Gromacs - Bug #1393: genbox segfaults with after verlet scheme bec... Closed 12/05/2013

Associated revisions

11/19/2015
Revision d77dddb8 - 05/23/2014 12:16 AM - Roland Schulz

Fix memory issue in solvate

Caused by SIMD padding introduced by new group kernels.

Fixes #1499

Change-Id: i5126217c9b752f1c1fd04d01e26449871dc52d5b

---

**History**

#1 - 05/12/2014 11:31 AM - Mark Abraham

- Related to Bug #1393: genbox segfaults with after verlet scheme become default added

#2 - 05/12/2014 02:27 PM - Mark Abraham

Agree we should use simpler code here.

I suspect the jjnr<0 issue relates to the SIMD padding of lists introduced for the new group scheme kernels in 4.6, but I'd have to ask Erik. Probably a work-around is to skip the loop body if jjnr<0, and if so, the fix should go in release-4.6.

#3 - 05/22/2014 04:40 PM - Erik Lindahl

Correct - jjnr<0 is always padding.

#4 - 05/23/2014 12:17 AM - Gerrit Code Review Bot

Gerrit received a related patchset '1' for Issue #1499.
Uploader: Roland Schulz (roland@rschulz.eu)
Change-Id: i5126217c9b752f1c1fd04d01e26449871dc52d5b
Gerrit URL: https://gerrit.gromacs.org/3487

#5 - 05/23/2014 12:57 AM - Gerrit Code Review Bot

Gerrit received a related patchset '1' for Issue #1499.
Uploader: Erik Lindahl (erik@kth.se)
Change-Id: i133115817166988a31623bbfe12b31290cf021c4
Gerrit URL: https://gerrit.gromacs.org/3488

#6 - 05/23/2014 12:57 AM - Erik Lindahl

- Status changed from New to Fix uploaded
- Affected version changed from git master to 5.0

#7 - 05/24/2014 11:45 PM - Roland Schulz

- Status changed from Fix uploaded to Resolved
- % Done changed from 0 to 100

Applied in changeset d77dddb83f64f0265b81f226ff1061b37bc10877b.

#8 - 05/26/2014 01:48 PM - Erik Lindahl

---
- Status changed from Resolved to Closed

#9 - 07/15/2014 04:25 PM - Teemu Murtola
- Category set to preprocessing (pdb2gmx,grompp)
- Target version set to 4.6.x