Description
I found these codes will crash because of Ractor:

```ruby
e = (1..1000).to_a
arr = [e,e,e,e]

arr.map { |sub_arr|
  Ractor.new(sub_arr) do |sub_arr|
    sub_arr.map { Digest::MD5.hexdigest(rand.to_s) }
  end
}.map{|r| r.take}.reduce(:+)
```

If change e to (1..100), no crashes any more. If we don't use Ractor, it's also good.

Environment: ruby3.0.0p0 + rails6.1.1.

Related issues:
Related to Ruby master - Bug #17742: Random.rand gets wrong after using Ractor... Closed

Associated revisions
Revision 3acc81d9 - 02/10/2021 10:44 AM - nobu (Nobuyoshi Nakada)
Fixed race in dtoa [Bug #17612]

Fixed the race condition when replacing freelist entry with its chained next element. At acquiring an entry, hold the entry once with the special value, then release by replacing it with the next element again after acquired. If another thread is holding the same entry at that time, spinning until the entry gets released.

Co-Authored-By: Koichi Sasada ko1@atdot.net

Revision 6bf32cbe - 03/12/2021 08:11 PM - naruse (Yui NARUSE)
merge revision(s)
63abb5c227e5c20d18d0deb1699f251da93ca6cb534d02631e71209b12ab269d011701027e85bc9.2addb01ae14c0a4cf190b7c969b91726966a0e0f, 3acc81d9e41b18380b9e0168fe2b5e5e0c727256: [Backport #17612]
dtoa.c: make compilable independently

```diff
- dtoa.c: constified
  clang seems to locate never modified local data in the const segment implicitly.
- missing/dtoa.c | 2 ++
  1 file changed, 1 insertion(+), 1 deletion(-)
```
dtoa.c: make thread-safe by using atomic CAS
Fixed race in dtoa [Bug #17612]

Fixed the race condition when replacing 'freelist' entry with its chained next element. At acquiring an entry, hold the entry once with the special value, then release by replacing it with the next element again after acquired. If another thread is holding the same entry at that time, spinning until the entry gets released.

Co-Authored-By: Koichi Sasada <ko1@atdot.net>

History

#1 - 02/09/2021 02:31 AM - ko1 (Koichi Sasada)
- Assignee set to ko1 (Koichi Sasada)
- Status changed from Open to Assigned

Could you give me an error log?

2.times.map { Ractor.new { loop { rand.to_s } } }.each(&:take)

I can find an issue on the simpler example, but I also observed different error on the your original example. I want to confirmed what error you got.

#2 - 02/09/2021 01:05 PM - tomanderson (Thomas Ender)

ko1 (Koichi Sasada) Here is my error log when running your simpler code:

irb(main):036:0> 2.times.map{ Ractor.new{ loop{ rand.to_s } } }.each(&:take)

internal:ractor:267: warning: Ractor is experimental, and the behavior may change in future versions of Ruby! Also there are many implementation issues.
*** Error in `irb': free(): invalid next size (fast): 0x00007f69a0001490 ***

======== Backtrace: =========
/lib/x86_64-linux-gnu/libc.so.6(+0x777f5)[0x7f69aaea37f5]
/lib/x86_64-linux-gnu/libc.so.6(+0x8038a)[0x7f69aaeac38a]
/lib/x86_64-linux-gnu/libc.so.6(cfree+0x4c)[0x7f69aaeb058c]
/home/thomas/.rbenv/versions/3.0.0/lib/libruby.so.3.0(ruby_sized_xfree+0x41)[0x7f69ab2df621]
/home/thomas/.rbenv/versions/3.0.0/lib/libruby.so.3.0(+0x165c2a)[0x7f69ab2f35b2a]
/home/thomas/.rbenv/versions/3.0.0/lib/libruby.so.3.0(+0x28457e)[0x7f69ab47a57e]
/home/thomas/.rbenv/versions/3.0.0/lib/libruby.so.3.0(+0x28886c)[0x7f69ab478e6c]
/home/thomas/.rbenv/versions/3.0.0/lib/libruby.so.3.0(+0x29008e)[0x7f69ab48608e]
/home/thomas/.rbenv/versions/3.0.0/lib/libruby.so.3.0(rb_vm_exec+0xc8)[0x7f69ab48bc38]
/home/thomas/.rbenv/versions/3.0.0/lib/libruby.so.3.0(+0x2a0c67)[0x7f69ab496c7]
/home/thomas/.rbenv/versions/3.0.0/lib/libruby.so.3.0(rb_vrescue2+0x10a)[0x7f69ab2dc561a]
/home/thomas/.rbenv/versions/3.0.0/lib/libruby.so.3.0(rb_rescue2+0x8a)[0x7f69ab2c589a]
/home/thomas/.rbenv/versions/3.0.0/lib/libruby.so.3.0(+0x28457e)[0x7f69ab47a57e]
/home/thomas/.rbenv/versions/3.0.0/lib/libruby.so.3.0(+0x298d31)[0x7f69ab48ed31]
/home/thomas/.rbenv/versions/3.0.0/lib/libruby.so.3.0(+0x2993b3)[0x7f69ab48f3b3]
/home/thomas/.rbenv/versions/3.0.0/lib/libruby.so.3.0(+0x28886c)[0x7f69ab478e6c]
/home/thomas/.rbenv/versions/3.0.0/lib/libruby.so.3.0(+0x290a5a)[0x7f69ab48ba5a]
/home/thomas/.rbenv/versions/3.0.0/lib/libruby.so.3.0(rb_vm_exec+0xc8)[0x7f69ab48bc38]
/home/thomas/.rbenv/versions/3.0.0/lib/libruby.so.3.0(+0x247735)[0x7f69ab44f735]
/home/thomas/.rbenv/versions/3.0.0/lib/libruby.so.3.0(+0x2516f4)[0x7f69ab4476f4]
/lib/x86_64-linux-gnu/libthread.so.0(+0x76ba)[0x7f6999a9f6ba]
/lib/x86_64-linux-gnu/libthread.so.0(clone=0x6d)[0x7f6999a134dd]

======== Memory map: =========
00400000-00401000 r-xp 00000000 08:01 6563405 /home/thomas/.rbenv/versions/3.0.0/bin/ruby
00600000-00601000 r--p 00000000 08:01 6563405 /home/thomas/.rbenv/versions/3.0.0/bin/ruby
00601000-00602000 rw-p 00001000 08:01 6563405 /home/thomas/.rbenv/versions/3.0.0/lib/ruby
0234d000-0234e000 rw-p 00000000 00:00 0
76998000000-76998021000 rw-p 00000000 00:00 0
76998021000-7699c000000 ---p 00000000 00:00 0
7699c000000-7699c021000 rw-p 00000000 00:00 0
FYI, my environment is ruby 3.0.0 installed by rbenv, OS is ubuntu 16.04 desktop x64. My example was firstly found crashed in Rails, but later I put it into .rb and run script with ruby, it crashed in the same way. I also test it on windows 10 with ruby 3.0 installed by rubyinstaller.org version and seems to be the same.
If there's a ruby build with debug symbols or ASAN, this should be pretty easy to scope out.

#4 - 02/10/2021 12:18 PM - nobu (Nobuyoshi Nakada)
- Status changed from Assigned to Closed

Applied in changeset git|3acc81d9e41b18380b9e0168fe2b5e5e0e727256.

Fixed race in dtoa [Bug #17612]

Fixed the race condition when replacing freelist entry with its chained next element. At acquiring an entry, hold the entry once with the special value, then release by replacing it with the next element again after acquired. If another thread is holding the same entry at that time, spinning until the entry gets released.

Co-Authored-By: Koichi Sasada ko1@atdot.net

#5 - 02/19/2021 08:17 PM - ko1 (Koichi Sasada)
- Backport changed from 2.5: UNKNOWN, 2.6: UNKNOWN, 2.7: UNKNOWN, 3.0: UNKNOWN to 2.5: UNKNOWN, 2.6: UNKNOWN, 2.7: UNKNOWN, 3.0: REQUIRED

#6 - 03/13/2021 08:54 AM - naruse (Yui NARUSE)
- Backport changed from 2.5: UNKNOWN, 2.6: UNKNOWN, 2.7: UNKNOWN, 3.0: REQUIRED to 2.5: UNKNOWN, 2.6: UNKNOWN, 2.7: UNKNOWN, 3.0: DONE

#7 - 03/24/2021 09:46 PM - wanabe (_ wanabe)
- Related to Bug #17742: Random.rand gets wrong after using Ractor and requiring benchmark added