Ruby - Bug #1341

pthread_cond_timedwait failing in 1.9.1-p0 thread tests on HP-UX 11i v2

04/01/2009 01:52 AM - graza (Graham Agnew)

Status:	Rejected		
Priority:	Normal		
Assignee:			
Target version:	2.6		
ruby -v:	ruby 1.9.1p0 (2009-01-30 revision 21907) [ia64-hpux11.23]	**Backport:**	

Description

=begin
I have been trying to compile and test 1.9.1-p0 on HP-UX 11i v2. When running the tests, the threads tests crash with the following bug:

> [BUG] pthread_cond_timedwait: 22
> ruby 1.9.1p0 (2009-01-30 revision 21907) [ia64-hpux11.23]
>
> ## -- control frame ----------
>
> -- Ruby level backtrace information---
>
> [NOTE]
> You may encounter a bug of Ruby interpreter. Bug reports are welcome.
> For details: http://www.ruby-lang.org/bugreport.html

The errno 22 means EINVAL. I put some print statements into the thread_pthread.c file to work out what was going on and it looks like there's a condition variable that is being initialised twice.
=end

History

#1 - 07/14/2009 12:14 AM - yugui (Yuki Sonoda)

- Priority changed from Normal to 3

- Target version changed from 1.9.1 to 2.0.0

=begin

=end

#2 - 11/13/2009 01:00 AM - docwhat (Christian Höltje)

=begin
This also happens when running "make test" in solaris but not "env RUBYLIB=./lib ./ruby test/ruby/test_thread.rb"

test_thread.rb ....bootstraptest.tmp.rb:6: [BUG] pthread_cond_timedwait: 22
ruby 1.9.1p243 (2009-07-16 revision 24175) [sparc-solaris2.8]

-- control frame ----------
c:0010 p:---- s:0028 b:0028 l:000027 d:000027 CFUNC :join
c:0009 p:0013 s:0024 b:0024 l:0018b8 d:000023 BLOCK bootstraptest.tmp.rb:6
c:0008 p:---- s:0020 b:0020 l:000019 d:000019 FINISH
c:0007 p:---- s:0018 b:0018 l:000017 d:000017 CFUNC :each
c:0006 p:0018 s:0015 b:0015 l:0018b8 d:002430 BLOCK bootstraptest.tmp.rb:3
c:0005 p:---- s:0012 b:0012 l:000011 d:000011 FINISH
c:0004 p:---- s:0010 b:0010 l:000009 d:000009 CFUNC :times

c:0003 p:0013 s:0007 b:0006 l:0018b8 d:001fd8 EVAL   bootstraptest.tmp.rb:2
c:0002 p:---- s:0004 b:0004 l:000003 d:000003 FINISH
c:0001 p:0000 s:0002 b:0002 l:0018b8 d:0018b8 TOP

-- Ruby level backtrace information--
bootstraptest.tmp.rb:6:in join' bootstraptest.tmp.rb:6:in block (2 levels) in '
bootstraptest.tmp.rb:3:in each' bootstraptest.tmp.rb:3:in block in '
bootstraptest.tmp.rb:2:in times' bootstraptest.tmp.rb:2:in '

[NOTE]
You may encounter a bug of Ruby interpreter. Bug reports are welcome.
For details: http://www.ruby-lang.org/bugreport.html

E......

It then hangs at this test.
=end

#3 - 04/22/2010 02:31 AM - mame (Yusuke Endoh)

- Assignee set to mame (Yusuke Endoh)

=begin
Hi,

I guess this is the limitation of Solaris:

- http://bugs.opensolaris.org/view_bug.do?bug_id=4038480
- http://docs.sun.com/app/docs/doc/806-0630/6j9vkb8ct?a=view

  EINVAL
  ...
  For cond_timedwait(), the specified number of seconds, abstime,
  is greater than current_time + 100,000,000, where current_time
  is the current time, or the number of nanoseconds is greater
  than or equal to 1,000,000,000.

Maybe, HP-UX has the same limitation, though I cannot find the
evidence.

I wrote a workaround patch:

diff --git a/thread_pthread.c b/thread_pthread.c
index e6295db..7387724 100644
--- a/thread_pthread.c
+++ b/thread_pthread.c
@@ -633,6 +633,35 @@ native_sleep(rb_thread_t *th, struct timeval *tv)
(unsigned long)ts.tv_sec, ts.tv_nsec);
r = pthread_cond_timedwait(&th->native_thread_data.sleep_cond,
&th->interrupt_lock, &ts);

- [redacted]

+#define MEGA_SEC 1000000

- [redacted]

- [redacted]
- [redacted]
- [redacted]
- [redacted]
- [redacted]
- [redacted]
- [redacted]
- [redacted]
- [redacted]
- [redacted]
- [redacted]
- [redacted]
- [redacted]
- [redacted]
- [redacted]
- [redacted]
- [redacted]
- [redacted]
- [redacted]

  ```
   if (r && r != ETIMEDOUT) rb_bug_errno("pthread_cond_timedwait", r);
  ```

  ```
   thread_debug("native_sleep: pthread_cond_timedwait end (%d)\n", r);
  ```

--
Yusuke Endoh mame@tsg.ne.jp
=end

#4 - 05/06/2010 06:01 AM - graza (Graham Agnew)

=begin
Today I downloaded, patched, and compiled the latest snapshot.

On HP-UX, that patch stopped the rb_bug_errno happening, although test/ruby/test_thread.rb scripts blocked indefinitely. On being interrupted it was the following test:

#874 test_thread.rb:34:in `<top (required)>':

The code in your patch doesn't look right to me. Shouldn't the code re-fetch the time using gettimeofday each time through the loop, and then add the MEGA_SEC to that? As it is, it's wrong because it adds a MEGA_SEC to tvn, so after one MEGA_SEC it will enter a hard loop.

Cheers,
Gra.
=end

#5 - 05/06/2010 09:23 AM - mame (Yusuke Endoh)

=begin
Hi,

2010/5/6 Graham Agnew redmine@ruby-lang.org:

> On HP-UX, that patch stopped the rb_bug_errno happening, although the test/ruby/test_io.rb and test/ruby/test_thread.rb scripts both blocked indefinitely.
>
> The code in your patch doesn't look right to me. ?Shouldn't the code re-fetch the time using gettimeofday each time through the loop, and then add the MEGA_SEC to that? ?As it is, it's wrong because it adds a MEGA_SEC to tvn, so after one MEGA_SEC it will enter a hard loop.

Thank you for your testing! How about the following patch?

To tell the truth, I'm writing a patch without test because I
don't have HP-UX. If this is wrong again, It is really helpful
for you to correct the patch by yourself.

diff --git a/thread_pthread.c b/thread_pthread.c
index e6295db..3c13f72 100644
--- a/thread_pthread.c
+++ b/thread_pthread.c
@@ -631,8 +631,29 @@ native_sleep(rb_thread_t *th, struct timeval *tv)
int r;
thread_debug("native_sleep: pthread_cond_timedwait start (%ld, %ld)\n",
(unsigned long)ts.tv_sec, ts.tv_nsec);

- [redacted]
  ```
  r = pthread_cond_timedwait(&th->native_thread_data.sleep_cond,
        &th->interrupt_lock, &ts);
  ```
- [redacted]
- [redacted]
- [redacted]
- [redacted]
- [redacted]
- [redacted]
- [redacted]

+#define MEGA_SEC 1000000

- [redacted]
- [redacted]
- [redacted]
- [redacted]
- [redacted]
- [redacted]
- [redacted]
- [redacted]
- [redacted]
- [redacted]
- [redacted]
- [redacted]
  ```
  if (r && r != ETIMEDOUT) rb_bug_errno("pthread_cond_timedwait", r);
  ```
  ```
  thread_debug("native_sleep: pthread_cond_timedwait end (%d)\n", r);
  ```

--
Yusuke Endoh mame@tsg.ne.jp

=end

#6 - 05/08/2010 01:19 AM - graza (Graham Agnew)

=begin
Hi Yusuke,

That code looks better although I'm still getting test/ruby/test_thread.rb blocking indefinitely. I will see if I can attach to the process with a debugger and figure out what it's blocked on.

Thanks,
Gra.
=end

#7 - 09/14/2010 04:35 PM - shyouhei (Shyouhei Urabe)

- *Status changed from Open to Assigned*

=begin

=end

#8 - 06/11/2011 02:25 PM - ko1 (Koichi Sasada)

Endo-san,

Can we close this issue?

#9 - 06/11/2011 04:28 PM - mame (Yusuke Endoh)

I guess it still reproduces on Solaris.
I have no idea about HP-UX.

--
Yusuke Endoh mame@tsg.ne.jp

#10 - 06/11/2011 05:22 PM - kosaki (Motohiro KOSAKI)

Now, following two links are dead. Do anyone know new URLs?

  - http://docs.sun.com/app/docs/doc/806-0630/6j9vkb8ct?a=view
  - http://bugs.opensolaris.org/view_bug.do?bug_id=4038480

#11 - 07/05/2011 01:35 AM - mame (Yusuke Endoh)

- *Status changed from Assigned to Open*
- *Assignee deleted (mame (Yusuke Endoh))*

Hello,

> Now, following two links are dead. Do anyone know new URLs?
>
>   - http://docs.sun.com/app/docs/doc/806-0630/6j9vkb8ct?a=view
>   - http://bugs.opensolaris.org/view_bug.do?bug_id=4038480

Here.

http://download.oracle.com/docs/cd/E19683-01/816-0216/6m6ngupgv/index.html

EINVAL
Invalid argument. For cond_init(), type is not a recognized type. For cond_timedwait(), the specified number of seconds, abstime, is greater than current_time + 100,000,000, where current_time is the current time, or the number of nanoseconds is greater than or equal to 1,000,000,000.

The problem that I now focus on is that pthread_cond_timedwait may fail
with EINVAL if an argument is greater than current_time + 100,000,000
on Solaris.
The patch of [ruby-core:29702] is too old and cannot be applied, so
I rewrote and committed a new patch at r32409. Now, "make test" passes
on Solaris. Congrats.

Unfortunately, the original issue that OP reported was a different problem.
But I guess that there is no hope of fixing the issue. At least I cannot.
So I resign the assignee of this ticket. Sorry for late action.

--
Yusuke Endoh mame@tsg.ne.jp

#12 - 07/25/2011 08:19 PM - naruse (Yui NARUSE)

- Status changed from Open to Feedback

Feedback about HP-UX is welcome

#13 - 08/08/2011 09:10 PM - kosaki (Motohiro KOSAKI)

- Subject changed from pthread_cond_timedwait failing in 1.9.1-p0 thread tests to pthread_cond_timedwait failing in 1.9.1-p0 thread tests on HP-UX 11i v2

#14 - 10/30/2012 09:31 AM - ko1 (Koichi Sasada)

- Target version changed from 2.0.0 to 2.6

Please tell us if you have HP-UX.

#15 - 11/05/2012 10:56 PM - mame (Yusuke Endoh)

- Status changed from Feedback to Rejected

HP-UX is not supported. I'm sorry, but please create a patch that works yourself.
If you provide us the patch and it looks benign to other platforms, we may apply it to trunk.

--
Yusuke Endoh mame@tsg.ne.jp