Ruby master - Bug #5883

thread bootstrap tests fail with clang 3.0 on Snow Leopard

01/12/2012 06:46 AM - luislavena (Luis Lavena)

Status: Rejected
Priority: Normal
Assignee:
Target version: 2.0.0

ruby -v:
ruby 2.0.0dev (2012-01-12 trunk 34277) [x86_64-darwin10.8.0]

Description

Hello,

Attempted to use clang to build latest trunk (r34277), using clang-3.0 provided by MacPorts:

$ clang --version
clang version 3.0 (tags/RELEASE_30/final)
Target: x86_64-apple-darwin10.8.0
Thread model: posix

And generated the following output:

test_thread.rb ........................................F.............

#927 test_thread.rb:307:in `<top (required)>':
g = "abc".enum_for(:scan, /./)
loop { g.next }
#=> killed by SIGABRT (signal 6)
| bootstraptest.tmp.rb:3: [BUG] Bus Error
| ruby 2.0.0dev (2012-01-12 trunk 34277) [x86_64-darwin10.8.0]
|
| -- Control frame information -------------------------------
| c:0004 p:---- s:0011 b:0011 l:000010 d:000010 CFUNC :loop
c:0003 p:0029 s:0008 b:0007 l:002078 d:000ee0 EVAL bootstraptest.tmp.rb:3
c:0002 p:---- s:0004 b:0004 l:000003 d:000003 FINISH
c:0001 p:0000 s:0002 b:0002 l:002078 d:002078 TOP

| -- Ruby level backtrace information -------------------------
| bootstraptest.tmp.rb:3:in`'
| bootstraptest.tmp.rb:3:in `loop`

| -- C level backtrace information -----------------------------

| See Crash Report log file under ~/Library/Logs/CrashReport or
| /Library/Logs/CrashReporter, for the more detail of.

| -- Other runtime information ---------------------------------

| * Loaded script: bootstraptest.tmp.rb

| * Loaded features:

| 0 enumerator.so
| 1 /Users/luis/projects/oss/ruby/clangbuild/.ext/x86_64-darwin10.8.0/enc/encdb.bundle
| 2 /Users/luis/projects/oss/ruby/clangbuild/.ext/x86_64-darwin10.8.0/enc/trans/transdb.bundle
| 4 /Users/luis/projects/oss/ruby/clangbuild/rbconfig.rb
| 6 /Users/luis/projects/oss/ruby/lib/rubygems/exceptions.rb
| 7 /Users/luis/projects/oss/ruby/lib/rubygems/custom_require.rb
| 8 /Users/luis/projects/oss/ruby/lib/rubygems.rb

[NOTE]
You may have encountered a bug in the Ruby interpreter or extension libraries.
Bug reports are welcome.
For details: http://www.ruby-lang.org/bugreport.html

[ruby-dev:34128]
FAIL 1/943 tests failed
make: *** [yes-btest-ruby] Error 1

Only reference is [ruby-dev:34128]

No special flag were used for compilation. Make reported the following prior start compiling:

$ make
CC = clang
LD = ld
LDSHARED = clang -dynamiclib
CFLAGS = -O3 -ggdb -Wall -Wextra -Wno-unused-parameter -Wno-parentheses -Wno-long-long -Wno-missing-field-initializers
-Wunused-variable -Werror=pointer-arithmetic -Werror=write-strings -Werror=declaration-after-statement -Werror=shorten-64-to-32
-Werror=implicit-function-declaration -fno-common -pipe
XCFLAGS = -include ruby/config.h -include ruby/missing.h -D_FORTIFY_SOURCE=2 -fstack-protector -fno-strict-overflow
-ffvisibility=hidden -DRUBY_EXPORT
CPPFLAGS = -D_XOPEN_SOURCE -D_DARWIN_C_SOURCE -D_DARWIN_UNLIMITED_SELECT -I.
-Xincludel/x86_64-darwin10.8.0 -I./include -I..
DLDFLAGS = -Wl,-undefined,dynamic_lookup -Wl,-multiply_defined,suppress -Wl,-flat_namespace -install_name
/Users/luis/local/ruby2/lib/libruby.2.0.0.dylib -current_version 2.0.0 -compatibility_version 2.0.0 -Wl,-unexported_symbol,Init
-Wl,-unexported_symbol,threadptr -fstack-protector -Wl,-u,_objc_msgSend -fstack-protector -Wl,-u,_objc_msgSend
SOLIBS =

BASE RUBY was ruby 1.9.3p0 (2011-10-30 revision 33570) [x86_64-darwin10.8.0]

Thank you.

History

#1 - 01/12/2012 07:30 AM - naruse (Yui NARUSE)
  - Status changed from Open to Rejected
  - Assignee deleted (mame (Yusuke Endoh))

Use gcc, clang can't work for ruby yet.

#2 - 01/12/2012 07:37 AM - luislavena (Luis Lavena)

Oh bummer, where is the fun then?

#3 - 01/12/2012 09:27 AM - naruse (Yui NARUSE)

Luis Lavena wrote:

  Oh bummer, where is the fun then?

I can't understand this sentence with my poor English skill but kindly speaking,
Yes, it's not work and I know to fix it is not easy because I tried to fix and reported it
to LLVM and sometimes it works (see my tickes and commits related to LLVM/clang).
But LLVM/clang is so a developing product that it sometimes work, sometimes not work
through my experience.
Once clang-trunk works well but at another time it doesn't.

For example I confirm following works well, but on FreeBSD 9.0 it doesn't work.
ruby 2.0.0dev (2012-01-12 trunk 34277) [x86_64-darwin11.2.0]
clang version 3.1 (trunk 147813)
Target: x86_64-apple-darwin11.2.0

So I concluded that we can't support clang now and wait it stable.
Under such circumstances, what I should say to a casual user or a bummer like you
will be "Use gcc". Of course an effort to make clang works with ruby is welcome,
and patches are also welcome.
I'll constantly continue such effort.

08/06/2021 2/3
Yui NARUSE wrote:

I can't understand this sentence with my poor English skill but kindly speaking, 
Yes, it's not work and I know to fix it is not easy because I tried to fix and reported it 
to LLVM and sometimes it works (see my tickes and commits related to LLVM/clang).

Yui, was not my intention to say you didn't want to fix this.

It will be great if a clear statement is made in relation to clang, like:

"Don't use clang with Ruby yet, Ruby code does not compile properly under it."

So I concluded that we can't support clang now and wait it stable. 
Under such circumstances, what I should say to a casual user or a bummer like you 
will be "Use gcc".

The bummer was my silly attempt to make a joke, yet again, it got lost in the translation.

Of course an effort to make clang works with ruby is welcome, 
and patches are also welcome.

I'm chasing a longjmp exception right now so won't be able to work on this type of things.

I'll constantly continue such effort.

And we all appreciate it.

Once again, thank you.