

Ruby - Bug #19518

Recent Source Releases Do Not Compile on CentOS 7 Due to configure Script Error Generated By autoconf >= 2.70

03/08/2023 07:09 PM - eviljoel (evil joel)

Status:	Closed	
Priority:	Normal	
Assignee:		
Target version:		
ruby -v:	3.2.1, 3.2.0, 3.1.3, 3.0.5	Backport: 2.7: UNKNOWN, 3.0: UNKNOWN, 3.1: UNKNOWN, 3.2: UNKNOWN

Description

Source releases where the 'configure' script was generated by autoconf 2.70 or greater results in a failed Ruby compilation on CentOS 7. This is because autoconf 2.70 and 2.71 adds some code that attempts to detect C11 support resulting in a compilation with the -std11 flag instead of the -std99 flag. The gcc bundled with CentOS 7 is version 4.8.5 which only has experimental C11 support. This experimental C11 support appears to not handle 'extern' appropriately.

I've included a Dockerfile that reproduces the problem:

```
FROM centos:7

RUN yum -y install autoconf gcc libyaml-devel make
RUN curl -O https://cache.ruby-lang.org/pub/ruby/3.2/ruby-3.2.1.tar.xz
# Verify the SHA256SUM as listed here: https://www.ruby-lang.org/en/news/2023/02/08/ruby-3-2-1-released/
RUN [ "$(sha256sum ruby-3.2.1.tar.xz | sed -n 's/^([^\s]*)\s.*$/\1/p' | tr -d "\n" | tr -d ' ')" = '746c8661ae25449cbdc5297d1092702e93e66f365a75fecb740d4f292ced630c' ]
RUN tar -xf ruby-3.2.1.tar.xz
RUN cd ruby-3.2.1 && ./configure
RUN cd ruby-3.2.1 && make
RUN /ruby-3.2.1/ruby --version
```

I've also included a Dockerfile that works around the problem simply by regenerating the configure script with autoconf 2.69:

```
FROM centos:7

RUN yum -y install autoconf gcc libyaml-devel make
RUN curl -O https://cache.ruby-lang.org/pub/ruby/3.2/ruby-3.2.1.tar.xz
# Verify the SHA256SUM as listed here: https://www.ruby-lang.org/en/news/2023/02/08/ruby-3-2-1-released/
RUN [ "$(sha256sum ruby-3.2.1.tar.xz | sed -n 's/^([^\s]*)\s.*$/\1/p' | tr -d "\n" | tr -d ' ')" = '746c8661ae25449cbdc5297d1092702e93e66f365a75fecb740d4f292ced630c' ]
RUN tar -xf ruby-3.2.1.tar.xz
RUN cd ruby-3.2.1 && autoconf configure.ac > configure
RUN cd ruby-3.2.1 && ./configure
RUN cd ruby-3.2.1 && make
RUN /ruby-3.2.1/ruby --version
```

This 'configure' bug affects the following releases of Ruby:

- ruby-3.2.1
- ruby-3.2.0
- ruby-3.1.3
- ruby-3.0.5

Ruby 2.7.7's 'configure' script was generated with autoconf 2.71 but appears to be unaffected.

My experience working with autoconf is very limited, but I believe this is mostly due to an error in autoconf itself. I believe autoconf should be smart enough to not attempt to leverage gcc's experimental C11 support. Regardless, Ruby should still compile correctly on CentOS 7 especially when prior bugfix releases of 3.0 and 3.1 did so successfully. At a minimum, future releases of 3.0 and 3.1 should generate their 'configure' scripts with autoconf 2.69 to maintain backwards compatibility and to address this regression. Ruby developers should also consider supporting compilation of Ruby 3.2 and 3.3 on CentOS 7 until CentOS 7 support ends next year.

If you agree this is also an autoconf error, I'll file a bug upstream.

Thank you.

Related issues:

Is duplicate of Ruby - Bug #19161: Cannot compile 3.0.5 or 3.1.3 on Red Hat E...

[Feedback](#)

History

#1 - 03/08/2023 07:14 PM - eviljoel (evil joel)

Forgot to include the compile error:

```
compiling miniinit.c
In file included from vm_core.h:164:0,
    from iseq.h:14,
    from mini_builtin.c:3,
    from miniinit.c:51:
thread_pthread.h:109:39: error: expected '=', ',', ';', 'asm' or '__attribute__' before 'struct'
  RUBY_EXTERN RB_THREAD_LOCAL_SPECIFIER struct rb_execution_context_struct *ruby_current_ec;
                                         ^
In file included from iseq.h:14:0,
    from mini_builtin.c:3,
    from miniinit.c:51:
vm_core.h: In function 'rb_current_execution_context':
vm_core.h:1865:34: error: 'ruby_current_ec' undeclared (first use in this function)
    rb_execution_context_t *ec = ruby_current_ec;
                                ^
vm_core.h:1865:34: note: each undeclared identifier is reported only once for each function it appears in
miniinit.c: At top level:
cc1: warning: unrecognized command line option "-Wno-tautological-compare" [enabled by default]
cc1: warning: unrecognized command line option "-Wno-self-assign" [enabled by default]
cc1: warning: unrecognized command line option "-Wno-parentheses-equality" [enabled by default]
cc1: warning: unrecognized command line option "-Wno-constant-logical-operand" [enabled by default]
cc1: warning: unrecognized command line option "-Wno-cast-function-type" [enabled by default]
make: *** [miniinit.o] Error 1
```

#2 - 03/08/2023 07:23 PM - eviljoel (evil joel)

I listed the wrong compiler flags above. The correct flags are -std=gnu11 and -std=gnu99.

#3 - 03/08/2023 11:44 PM - mame (Yusuke Endoh)

I think this is a duplicate of [#19161](#). It looks like [@nobu \(Nobuyoshi Nakada\)](#) wrote a patch but he forget to commit it.

#4 - 03/09/2023 12:19 AM - hsbt (Hiroshi SHIBATA)

- Is duplicate of Bug #19161: Cannot compile 3.0.5 or 3.1.3 on Red Hat Enterprise Linux 7 added

#5 - 03/09/2023 06:20 AM - nobu (Nobuyoshi Nakada)

- Status changed from Open to Closed

Applied in changeset [git|96d1acfd6c6b42f2029f44d5b5920961d6efa92](#).

[Bug [#19161](#)] Check for TLS usability

On all platforms using GCC, even other than darwin.

#6 - 03/09/2023 10:58 AM - jaruga (Jun Aruga)

The gcc bundled with CentOS 7 is version 4.8.5 which only has experimental C11 support. This experimental C11 support appears to not handle 'extern' appropriately.

In this case, I believe when you try to build the new applications in CentOS7 / RHEL 7, a better choice is using a new version gcc by installing the RPM package in the Software Collections (SCL) rather than using the old version system gcc.

You can see the

- <https://linuxize.com/post/how-to-install-gcc-compiler-on-centos-7/#installing-multiple-gcc-versions>
- <https://www.softwarecollections.org/en/docs/>

The steps to install the new version gcc via SCL. I have not tested the steps by myself.

Install the SCL.

```
$ sudo yum install centos-release-scl
```

Search the available software collections "devtoolset-N" including the gcc. Replace the N with the actual number that you find.

```
$ yum search devtoolset
```

Install the SCL.

```
$ sudo yum install devtoolset-N
```

Enable the software collection. When you want to use

```
$ scl enable devtoolset-N bash
```

```
$ gcc --version
```

When you want to enable the SCL in the Dockerfile or a bash script without interactive mode. Maybe the script can be like this.

```
scl enable devtoolset-7 - << \EOF
set -ex
gcc --version

./configure ...
make
make install

EOF
```

#7 - 03/09/2023 11:21 AM - jaruga (Jun Aruga)

For the GCC, it seems that the devtoolset-8 and gcc-8 is the latest version on the Software Collections.
https://www.softwarecollections.org/en/scls/?search=devtoolset&policy=&repo=&order_by=-create_date&per_page=10
<https://www.softwarecollections.org/en/scls/rhscl/devtoolset-8/>

Just in case, the latest version of Ruby RPM package you can use on Software Collections on CentOS 7 is Ruby 2.6.
<https://www.softwarecollections.org/en/scls/rhscl/rh-ruby26/>

#8 - 03/09/2023 05:00 PM - eviljoel (evil joel)

Agreed that this is a duplicate.

Thanks @jaruga (Jun Aruga). I forgot about SCL. I should probably look into integrating that into my build pipeline.

Files

ruby321BuildFailure-20230308.Dockerfile	531 Bytes	03/08/2023	eviljoel (evil joel)
ruby321BuildWorkAround-20230308.Dockerfile	586 Bytes	03/08/2023	eviljoel (evil joel)