If I am using a refinement to make a private method into a public one, and I call the method, then Ruby crashes. Here's a simple example:

class Cow
  private
  def moo() end
end

module PublicCows
  refine(Cow) {
    public :moo
  }
end

using PublicCows
Cow.new.moo

It segfaults:

$ ruby scratch.rb
scratch.rb:13: [BUG] Segmentation fault at 0x007f7fffbdff8
ruby 2.4.0dev (2016-09-06 trunk 56078) [x86_64-openbsd6.0]

-- Control frame information -----------------------------------------------
c:0002 p:0049 s:0007 e:000005 EVAL   scratch.rb:13 [FINISH]
c:0001 p:0000 s:0003 E:001d50 (none) [FINISH]

-- Ruby level backtrace information ----------------------------------------
scratch.rb:13:in `<main>''

-- Other runtime information -----------------------------------------------
* Loaded script: scratch.rb
* Loaded features:
  0 enumerator.so
  1 thread.rb
  2 rational.so
  3 complex.so
  4 /home/kernigh-prefix/lib/ruby/2.4.0/x86_64-openbsd6.0/enc/encdb.so
  5 /home/kernigh-prefix/lib/ruby/2.4.0/x86_64-openbsd6.0/enc/trans/transdb.so
  6 /home/kernigh-prefix/lib/ruby/2.4.0/unicode_normalize.rb
  7 /home/kernigh-prefix/lib/ruby/2.4.0/x86_64-openbsd6.0/rbconfig.rb
  8 /home/kernigh-prefix/lib/ruby/2.4.0/rubygems/compatibility.rb
  9 /home/kernigh-prefix/lib/ruby/2.4.0/rubygems/defaults.rb
 10 /home/kernigh-prefix/lib/ruby/2.4.0/rubygems/deprecate.rb
 11 /home/kernigh-prefix/lib/ruby/2.4.0/rubygems/errors.rb
 12 /home/kernigh-prefix/lib/ruby/2.4.0/rubygems/version.rb
 13 /home/kernigh-prefix/lib/ruby/2.4.0/rubygems/requirement.rb
 14 /home/kernigh-prefix/lib/ruby/2.4.0/rubygems/platform.rb
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

Abort trap (core dumped)

There's a small chance that I get a SystemStackError instead of a segfault:

$ ruby scratch.rb
scratch.rb:13:in `<main>': stack level too deep (SystemStackError)

Feature #12697 had inspired me to try making a refinement where Module#attr_accessor and Module#define_method are public. That's how I found this bug.

Associated revisions
Revision 7bbab207 - 12/08/2016 05:16 AM - nobu (Nobuyoshi Nakada)
vm_insnhelper.c: zsuper in refinements

- vm_insnhelper.c (vm_call_zsuper): prevent infinite recursion zsuper in refinements. [ruby-core:77161] [Bug #12729]

Revision 57023 - 12/08/2016 05:16 AM - nobu (Nobuyoshi Nakada)
vm_insnhelper.c: zsuper in refinements

- vm_insnhelper.c (vm_call_zsuper): prevent infinite recursion zsuper in refinements. [ruby-core:77161] [Bug #12729]

Revision 57023 - 12/08/2016 05:16 AM - nobu (Nobuyoshi Nakada)
vm_insnhelper.c: zsuper in refinements

- vm_insnhelper.c (vm_call_zsuper): prevent infinite recursion zsuper in refinements. [ruby-core:77161] [Bug #12729]

Revision 57023 - 12/08/2016 05:16 AM - nobu (Nobuyoshi Nakada)
vm_insnhelper.c: zsuper in refinements

- vm_insnhelper.c (vm_call_zsuper): prevent infinite recursion zsuper in refinements. [ruby-core:77161] [Bug #12729]

Revision 57023 - 12/08/2016 05:16 AM - nobu (Nobuyoshi Nakada)
vm_insnhelper.c: zsuper in refinements

- vm_insnhelper.c (vm_call_zsuper): prevent infinite recursion zsuper in refinements. [ruby-core:77161] [Bug #12729]

Revision 57023 - 12/08/2016 05:16 AM - nobu (Nobuyoshi Nakada)
vm_insnhelper.c: zsuper in refinements

- vm_insnhelper.c (vm_call_zsuper): prevent infinite recursion zsuper in refinements. [ruby-core:77161] [Bug #12729]

History
#1 - 09/22/2016 05:19 PM - noahgibbs (Noah Gibbs)
On latest head of master I don't get this abort trap or core dump, but I also don't see it return. It hangs, using 100% CPU, for at least a full minute and can only be stopped with "kill 9".

#2 - 12/08/2016 04:41 AM - nobu (Nobuyoshi Nakada)
- Backport changed from 2.1: UNKNOWN, 2.2: UNKNOWN, 2.3: UNKNOWN to 2.1: REQUIRED, 2.2: REQUIRED, 2.3: REQUIRED
- Description updated

#3 - 12/08/2016 05:16 AM - nobu (Nobuyoshi Nakada)
- Status changed from Open to Closed

Applied in changeset r57023.

---

vm_insnhelper.c: zsUPER in refinements

- vm_insnhelper.c (vm_call_zsuper): prevent infinite recursion zsUPER in refinements. [ruby-core:77161] [Bug #12729]