

Ruby trunk - Bug #6228

[mingw] Errno::EBADF in ruby/test_io.rb on ruby_1_9_3

03/31/2012 01:30 AM - jonforums (Jon Forums)

| | |
|---|------------------|
| Status: Closed | |
| Priority: Normal | |
| Assignee: luislavena (Luis Lavena) | |
| Target version: 1.9.3 | |
| ruby -v: ruby 1.9.3p172 (2012-03-30 revision 35179) [i386-mingw32] | Backport: |
| Description On Win7 32bit built with MinGW 4.6.2 using the RubyInstaller build recipes, I get the following test error: sh-3.1\$ ruby --version ruby 1.9.3p172 (2012-03-30 revision 35179) [i386-mingw32] sh-3.1\$ make test-all TESTS='openssl fiddle psych zlib io json mkmf pathname stringio erb fileutils ruby/test_io.rb ruby/test_io_m17n.rb ruby/test_file.rb' 7) Error: test_lines(TestIO): Errno::EBADF: Bad file descriptor c:/Users/Jon/Documents/RubyDev/ruby-git/test/ruby/test_io.rb:33:in close' c:/Users/Jon/Documents/RubyDev/ruby-git/test/ruby/test_io.rb:33:inrescue in block in pipe' c:/Users/Jon/Documents/RubyDev/ruby-git/test/ruby/test_io.rb:30:in `block in pipe' | |

Associated revisions

Revision 32ef6114 - 05/12/2012 09:46 PM - luislavena (Luis Lavena)

- test/ruby/test_io.rb (class TestIO): Disable GC during IO tests to avoid file descriptors being GC'ed. Suggestion by Tomoyuki Chikanaga [ruby-core:43951][Bug #6228]

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@35631 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision 35631 - 05/12/2012 09:46 PM - luislavena (Luis Lavena)

- test/ruby/test_io.rb (class TestIO): Disable GC during IO tests to avoid file descriptors being GC'ed. Suggestion by Tomoyuki Chikanaga [ruby-core:43951][Bug #6228]

Revision 35631 - 05/12/2012 09:46 PM - luislavena (Luis Lavena)

- test/ruby/test_io.rb (class TestIO): Disable GC during IO tests to avoid file descriptors being GC'ed. Suggestion by Tomoyuki Chikanaga [ruby-core:43951][Bug #6228]

Revision 35631 - 05/12/2012 09:46 PM - luislavena (Luis Lavena)

- test/ruby/test_io.rb (class TestIO): Disable GC during IO tests to avoid file descriptors being GC'ed. Suggestion by Tomoyuki Chikanaga [ruby-core:43951][Bug #6228]

Revision 35631 - 05/12/2012 09:46 PM - luislavena (Luis Lavena)

- test/ruby/test_io.rb (class TestIO): Disable GC during IO tests to avoid file descriptors being GC'ed. Suggestion by Tomoyuki Chikanaga [ruby-core:43951][Bug #6228]

Revision 35631 - 05/12/2012 09:46 PM - luislavena (Luis Lavena)

- test/ruby/test_io.rb (class TestIO): Disable GC during IO tests to avoid file descriptors being GC'ed. Suggestion by Tomoyuki Chikanaga [ruby-core:43951][Bug #6228]

Revision 35631 - 05/12/2012 09:46 PM - luislavena (Luis Lavena)

- test/ruby/test_io.rb (class TestIO): Disable GC during IO tests to avoid file descriptors being GC'ed. Suggestion by Tomoyuki Chikanaga [ruby-core:43951][Bug #6228]

Revision 526a853d - 05/13/2012 02:50 PM - nagachika (Tomoyuki Chikanaga)

- test/ruby/test_io.rb (test_flush_in_finalizer1): don't use IO.for_fd to close IO objects. it create IO object with already closed fd, and cause occasional Errno::EBADF in following tests. [ruby-core:45020] [Bug #6228]

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@35637 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision 35637 - 05/13/2012 02:50 PM - nagachika (Tomoyuki Chikanaga)

- test/ruby/test_io.rb (test_flush_in_finalizer1): don't use IO.for_fd to close IO objects. it create IO object with already closed fd, and cause occasional Errno::EBADF in following tests. [ruby-core:45020] [Bug #6228]

Revision 35637 - 05/13/2012 02:50 PM - nagachika (Tomoyuki Chikanaga)

- test/ruby/test_io.rb (test_flush_in_finalizer1): don't use IO.for_fd to close IO objects. it create IO object with already closed fd, and cause occasional Errno::EBADF in following tests. [ruby-core:45020] [Bug #6228]

Revision 35637 - 05/13/2012 02:50 PM - nagachika (Tomoyuki Chikanaga)

- test/ruby/test_io.rb (test_flush_in_finalizer1): don't use IO.for_fd to close IO objects. it create IO object with already closed fd, and cause occasional Errno::EBADF in following tests. [ruby-core:45020] [Bug #6228]

Revision 35637 - 05/13/2012 02:50 PM - nagachika (Tomoyuki Chikanaga)

- test/ruby/test_io.rb (test_flush_in_finalizer1): don't use IO.for_fd to close IO objects. it create IO object with already closed fd, and cause occasional Errno::EBADF in following tests. [ruby-core:45020] [Bug #6228]

Revision 35637 - 05/13/2012 02:50 PM - nagachika (Tomoyuki Chikanaga)

- test/ruby/test_io.rb (test_flush_in_finalizer1): don't use IO.for_fd to close IO objects. it create IO object with already closed fd, and cause occasional Errno::EBADF in following tests. [ruby-core:45020] [Bug #6228]

Revision 35637 - 05/13/2012 02:50 PM - nagachika (Tomoyuki Chikanaga)

- test/ruby/test_io.rb (test_flush_in_finalizer1): don't use IO.for_fd to close IO objects. it create IO object with already closed fd, and cause occasional Errno::EBADF in following tests. [ruby-core:45020] [Bug #6228]

Revision 26e1ebb1 - 05/14/2012 04:18 AM - naruse (Yui NARUSE)

merge revision(s) 35631:

```
* test/ruby/test_io.rb (class TestIO): Disable GC during IO tests to
avoid file descriptors being GC'ed. Suggestion by Tomoyuki Chikanaga
[ruby-core:43951][Bug #6228]
```

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/branches/ruby_1_9_3@35642 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision e2aeaed6 - 05/14/2012 08:14 AM - naruse (Yui NARUSE)

merge revision(s) 35636,35637: [Backport #6429]

```
* test/ruby/test_io.rb (TestIO): revert r35631. it broke the intent of
test_flush_in_finalizer1. [ruby-core:43951] [Bug ##6228]
```

```
* test/ruby/test_io.rb (test_flush_in_finalizer1): don't use IO.for_fd
to close IO objects. it create IO object with already closed fd, and
cause occasional Errno::EBADF in following tests. [ruby-core:45020]
[Bug #6228]
```

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/branches/ruby_1_9_3@35646 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

History

#1 - 03/31/2012 02:09 AM - naruse (Yui NARUSE)

Do you know which revision break it?

#2 - 03/31/2012 03:55 AM - jonforums (Jon Forums)

Not yet.

#3 - 03/31/2012 06:38 AM - jonforums (Jon Forums)

Haven't found root cause, but these commits also have Errno::EBADF errors:

commit revision patch

```
b1bcb82 r34865 p154
1b21d03 r34811 p146
7e2e3dc r34806 p144
```

In the above commits, the failures are at a different location in ruby/test_io.rb likely because I'm not building deps in order to speed things up:

5) Error:

```
test_invalid_advise(TestIO):
Errno::EBADF: Bad file descriptor - C:/Users/Jon/AppData/Local/Temp/test_io20120330-6868-1lqi3p9
c:/Users/Jon/Documents/RubyDev/ruby-git/test/ruby/test_io.rb:1119:in make_tempfile'
c:/Users/Jon/Documents/RubyDev/ruby-git/test/ruby/test_io.rb:1848:inblock (2 levels) in test_invalid_advise'
c:/Users/Jon/Documents/RubyDev/ruby-git/test/ruby/test_io.rb:1847:in each'
c:/Users/Jon/Documents/RubyDev/ruby-git/test/ruby/test_io.rb:1847:inblock in test_invalid_advise'
c:/Users/Jon/Documents/RubyDev/ruby-git/test/ruby/test_io.rb:1846:in each'
c:/Users/Jon/Documents/RubyDev/ruby-git/test/ruby/test_io.rb:1846:inblock in test_invalid_advise'
../../../../Users/Jon/Documents/RubyDev/ruby-git/test/runner.rb:15:in ``
```

I suspect this testing difference isn't relevant since EBADF errors persist, but it is a difference.

Time permitting this weekend, I'll try again using git bisect

#4 - 04/01/2012 12:27 AM - jonforums (Jon Forums)

In my configuration, one fix is to remove the recent `-DFD_SETSEZE=32767` define from the RubyInstaller build recipe. Removing the define gives the following results on r35179:

- no deps build using WinSDK 7.1 (without RubyInstaller) has no test_lines(TestIO) nor test_invalid_advise(TestIO) Errno::EBADF failures
- no deps build using MinGW GCC 4.6.2 (with or without RubyInstaller) has no test_lines nor test_invalid_advise failures
- full deps build using MinGW GCC 4.6.2 (with RubyInstaller recipe) has no test_lines nor test_invalid_advise failures

However, removing the define is not acceptable as it was added to fix the following user issue

<https://github.com/oneclick/rubyinstaller/issues/104>

And it get's more convoluted.

Running git bisect between commits a81c766/r34649 (good) and 7e2e3dc/r34806 (bad) with `-DFD_SETSEZE=32767` enabled in the RubyInstaller recipes, the "first bad commit" is Nobu's 35bf809/r34749 tweak to Enumerator in io.c. Building the "good" fa1a2ae/r34746 parent commit (without deps) resulted in no test_invalid_advise failure when running `make test-all TESTS='ruby/test_io.rb'`. But strangely, test_lines failed while test_invalid_advise passed when running my original `make test-all TESTS='openssl ... ruby/test_io.rb ...'`

I suspect there are other lurking issues left to discover, so don't close this issue as a 3rd party issue until we know more.

#5 - 04/01/2012 12:49 AM - jonforums (Jon Forums)

of course it's the `FD_SETSIZE` macro I'm talking about...

#6 - 04/09/2012 12:38 AM - naruse (Yui NARUSE)

- Status changed from Open to Assigned

- Assignee set to luislavena (Luis Lavena)

#7 - 04/17/2012 07:23 AM - h.shirosaki (Hiroshi Shirosaki)

This issue seems to be related to GC finalizer.

GC is triggered just before Errno::EBADF occurs, and finalizer closes the fd of make_tempfile. Errno::EBADF is raised for trying to close again the closed fd of make_tempfile.

This seems due to test_flush_in_finalizer1.

```

500.times {
  f = File.open(path, "w")
  fds << f.fileno
  f.print "hoge"
}

```

f is not closed. When GC sweeps f, the fd associated with f seems to be closed.
 fds contains 3. And make_tempfile in test_invalid_advise creates fd of 3.

I confirmed the error disappeared with the following patch.
 -DFD_SETSEZE=32767 and ruby 1.9.3p184 (2012-04-15 revision 35335) [i386-mingw32] on Windows 7.

```

diff --git a/test/ruby/test_io.rb b/test/ruby/test_io.rb
index edf449d..70ca240 100644
--- a/test/ruby/test_io.rb
+++ b/test/ruby/test_io.rb
@@ -1800,6 +1800,7 @@ End
end
ensure
fds.each {|fd| IO.for_fd(fd).close rescue next}

```

- GC.start end

```
def test_flush_in_finalizer2
```

#8 - 04/17/2012 10:14 PM - h.shirosaki (Hiroshi Shirosaki)

I tried to create simplified test case of this issue.

```

cat gc_io_test.rb
f = open("test_file", "w")
p f.fileno
IO.for_fd(f.fileno).close

```

```

f = open("test_file", "w")
p f.fileno

```

```
GC.start # trigger GC
```

```

begin
f.close # => raise Errno::EBADF
rescue
p $!
end

```

```
File.unlink("test_file")
```

```

~/ruby-trunk/bin/ruby -v gc_io_test.rb
ruby 2.0.0dev (2012-04-17 trunk 35363) [x86_64-darwin11.3.0]
5
5
#

```

#9 - 04/17/2012 10:27 PM - luislavena (Luis Lavena)

- Status changed from Assigned to Open
- Assignee deleted (luislavena (Luis Lavena))

Who should be the responsible for GC? Seems to me that open file descriptor shouldn't be garbage collected while still open.

#10 - 04/17/2012 11:47 PM - phasis68 (Heesob Park)

I found that this issue is related with the auto-close flag of IO.

The following code fails with Errno::EBADF

```

f = open("test_file", "w")
IO.for_fd(f.fileno).autoclose = true # default value
GC.start
f.close

```

But, the following code works fine.

```
f = open("test_file", "w")
IO.for_fd(f.fileno).autoclose = false
GC.start
f.close
```

#11 - 04/18/2012 12:25 AM - mame (Yusuke Endoh)

- Status changed from Open to Assigned
- Assignee set to authorNari (Narihiro Nakamura)

Who should be the responsible for GC?

Nari-san.

--
Yusuke Endoh mame@tsg.ne.jp

#12 - 04/22/2012 12:44 AM - nagachika (Tomoyuki Chikanaga)

Hi,

I think the root cause is wrong usage of IO.for_fd in ensure clause of test_flush_finalizer1,
and just call GC.start works fine for recycle fds.

This test was written by nobu.

nakada-san
Is the following patch seems ok?

```
diff --git a/test/ruby/test_io.rb b/test/ruby/test_io.rb
index ebf9b76..be5a333 100644
--- a/test/ruby/test_io.rb
+++ b/test/ruby/test_io.rb
@@ -1891,7 +1891,7 @@ End
 }
end
ensure
```

- fds.each {|fd| IO.for_fd(fd).close rescue next}
- GC.start end

```
def test_flush_in_finalizer2
```

#13 - 04/22/2012 04:58 AM - jonforums (Jon Forums)

Why is the following

```
ensure
fds.each {|fd| IO.for_fd(fd).close rescue next}
```

wrong usage of IO.for_fd?

In normal usage, I would likely use this style of explicit closure (by IO wrapping the fd) but not kick off a GC cycle with GC.start.

I'd like to understand why the usage is wrong. Specifically, is there another style of explicit recycling of fds that doesn't require starting a GC cycle?

#14 - 04/22/2012 08:54 AM - mame (Yusuke Endoh)

- Assignee changed from authorNari (Narihiro Nakamura) to nobu (Nobuyoshi Nakada)

+1 for nagachika-san.

jonforums (Jon Forums) wrote:

Why is the following

```
ensure
fds.each {|fd| IO.for_fd(fd).close rescue next}
```

wrong usage of IO.for_fd?

```
f = open("...") # this first IO has a fd X which is opened now.
IO.for_fd(f.fileno) # this second IO also has the fd X.
GC.start # the second IO is GC'ed; the fd X is auto-closed.
f.close # this attempts to close the fd X, but it is already closed => Errno::EBADF
```

Specifically, is there another style of explicit recycling of fds that doesn't require starting a GC cycle?

Call IO#close explicitly.

--
Yusuke Endoh mame@tsg.ne.jp

#15 - 04/23/2012 02:01 AM - jonforums (Jon Forums)

mame (Yusuke Endoh) wrote:

jonforums (Jon Forums) wrote:

Why is the following

```
ensure
fds.each {|fd| IO.for_fd(fd).close rescue next}
```

wrong usage of IO.for_fd?

```
f = open("...") # this first IO has a fd X which is opened now.
IO.for_fd(f.fileno) # this second IO also has the fd X.
GC.start # the second IO is GC'ed; the fd X is auto-closed.
f.close # this attempts to close the fd X, but it is already closed => Errno::EBADF
```

Specifically, is there another style of explicit recycling of fds that doesn't require starting a GC cycle?

Call IO#close explicitly.

Endoh-san, thank you. I understand.

Why do you think the same incorrect test code fails (consistently, not intermittently) on windows when only using -DFD_SETSIZE, but doesn't appear to fail on *nix? On Arch, I've not been able to repro when building using -DFD_SETSIZE=32767. This may indicate a lurking weakness in an _WIN32 MRI codepath (or winsock2 peculiarity) that doesn't exist (or does a better job of hiding) on *nix.

FYI, there's another recent issue on windows in which -DFD_SETSIZE=32767 exposes a SEGV with EventMachine with a simple EM.run {}. By default, EM sets FD_SETSIZE to 1024. Rebuilding ruby on windows without the FD_SETSIZE override fixes the SEGV. The EM issue is a separate issue than this one, but may be related. I haven't started investigating, so FYI.

I will test again on ruby_1_9_3 and trunk with the suggested patch and reply if errors persist.

Jon

#16 - 04/24/2012 12:18 AM - h.shirosaki (Hiroshi Shirosaki)

Why do you think the same incorrect test code fails (consistently, not intermittently) on windows when only using -DFD_SETSIZE, but doesn't appear to fail on *nix? On Arch, I've not been able to repro when building using -DFD_SETSIZE=32767. This may indicate a lurking weakness in an _WIN32 MRI codepath (or winsock2 peculiarity) that doesn't exist (or does a better job of hiding) on *nix.

This issue depends on the timing when IO object is GCed. GC of an object with the same fd must occur between open() and close(). I guess -DFD_SETSIZE=32767 would change memory usage and GC timing. MRI implementation is different between Windows and Linux, so GC timing would differ.

FYI, there's another recent issue on windows in which -DFD_SETSIZE=32767 exposes a SEGV with EventMachine with a simple EM.run {}. By default, EM sets FD_SETSIZE to 1024. Rebuilding ruby on windows without the FD_SETSIZE override fixes the SEGV. The EM issue is a separate issue than this one, but may be related. I haven't started investigating, so FYI.

I think above issue is cause of fd_array buffer overflow.

```
typedef struct fd_set
{
  u_int fd_count;
  SOCKET fd_array[FD_SETSIZE];
} fd_set;
```

On EM, FD_SETSIZE = 1024 and fd_array[1024].
EM uses FD_SET() and FD_SET() seems rb_w32_fdset() on Windows.

<https://github.com/ruby/ruby/blob/trunk/win32/win32.c#L2458>

In rb_w32_fdset(), FD_SETSIZE = 32767 since rb_w32_fdset is compiled with -DFD_SETSIZE=32767.

```
if (i == set->fd_count) {
  if (set->fd_count < FD_SETSIZE) { // FD_SETSIZE = 32767
    set->fd_array[i] = s;           // `i` could be over 1023
    set->fd_count++;
  }
}
```

If above scenario is correct, FD_SETSIZE of Ruby should be equal or less than FD_SETSIZE of EM.

include/winsock2.h has FD_SET macro on mingw, but MRI undef FD_SET and uses rb_w32_fdset() function. It might be better that FD_SET() is macro instead of function.

#17 - 04/24/2012 12:38 AM - luislavena (Luis Lavena)

- Assignee changed from nobu (Nobuyoshi Nakada) to usa (Usaku NAKAMURA)

Usa-san, what do you think about Hiroshi suggestion about leave FD_SET() as macro instead?

#18 - 04/24/2012 11:53 AM - usa (Usaku NAKAMURA)

Hello,

In message "[ruby-core:44560] [ruby-trunk - Bug #6228] [mingw] Errno::EBADF in ruby/test_io.rb on ruby_1_9_3"
on Apr.24,2012 00:38:57, luislavena@gmail.com wrote:

Usa-san, what do you think about Hiroshi suggestion about leave FD_SET() as macro instead?

First, as Jon mentioned, the SEGV of EM is another issue.
Please split it from this ticket.

Second, I don't understand how Hiroshi want to change the code,
so I cannot say anything about this.

Regards,

--

U.Nakamura usa@garbagecollect.jp

#19 - 04/24/2012 08:24 PM - luislavena (Luis Lavena)

- Assignee changed from usa (Usaku NAKAMURA) to luislavena (Luis Lavena)

usa (Usaku NAKAMURA) wrote:

First, as Jon mentioned, the SEGV of EM is another issue.
Please split it from this ticket.

Understood, creating another one to have FD_SET() discussion.

I'll commit suggested patch from nagachika-san.

Thank you.

#20 - 04/28/2012 04:21 AM - luislavena (Luis Lavena)

- Status changed from Assigned to Feedback

luislavena (Luis Lavena) wrote:

I'll commit suggested patch from nagachika-san.

I seem unable to reproduce the exact exception on both trunk and ruby_1_9_3 branches. The error seems randomly manifest.

Right now, against ruby_1_9_3, it shows in test_invalid_advise

5) Error:

```
test_invalid_advise(TestIO):
Errno::EBADF: Bad file descriptor - C:/Users/Luis/AppData/Local/Temp/test_io20120427-1432-1xmkrxh
C:/Users/Luis/Projects/oss/ruby/test/ruby/test_io.rb:1119:in make_tempfile'
C:/Users/Luis/Projects/oss/ruby/test/ruby/test_io.rb:1872:inblock (2 levels) in test_invalid_advise'
C:/Users/Luis/Projects/oss/ruby/test/ruby/test_io.rb:1871:in each'
C:/Users/Luis/Projects/oss/ruby/test/ruby/test_io.rb:1871:inblock in test_invalid_advise'
C:/Users/Luis/Projects/oss/ruby/test/ruby/test_io.rb:1870:in each'
C:/Users/Luis/Projects/oss/ruby/test/ruby/test_io.rb:1870:in test_invalid_advise'
../test/runner.rb:15:in `'
```

#21 - 04/28/2012 06:15 AM - jonforums (Jon Forums)

I seem unable to reproduce the exact exception on both trunk and ruby_1_9_3 branches. The error seems randomly manifest.

Right now, against ruby_1_9_3, it shows in test_invalid_advise

I've seen failures in both test_invalid_advise (see comment #3) and test_lines. Both disappeared when I tested Nagachika-san's suggested patch.

I'll try again with the latest trunk and ruby_1_9_3, but IIRC, the test_invalid_advise failure occurred when I built without deps.

#22 - 04/28/2012 10:45 PM - jonforums (Jon Forums)

On Win7 32bit, using the following build recipe on both trunk@r35485 and ruby_1_9_3@r35484 with MinGW GCC 4.6.2

```
sh -c ".../configure --enable-shared --disable-install-doc CPPFLAGS='-DFD_SETSIZE=32767'"
make
make test-all TESTS="ruby/test_io.rb"
```

I get only the test_invalid_advise(TestIO) error on ruby_1_9_3, and a freeze (cleared by CTRL-C) on trunk. Nagachika-san's suggested patch fixes the errors as does adding TestIO#setup and TestIO#teardown containing GC.disable and GC.enable, respectively.

#23 - 05/13/2012 06:46 AM - luislavena (Luis Lavena)

- Status changed from Feedback to Closed

- % Done changed from 0 to 100

This issue was solved with changeset [r35631](#).

Jon, thank you for reporting this issue.

Your contribution to Ruby is greatly appreciated.

May Ruby be with you.

-
- test/ruby/test_io.rb (class TestIO): Disable GC during IO tests to avoid file descriptors being GC'ed. Suggestion by Tomoyuki Chikanaga [ruby-core:43951][Bug [#6228](#)]

#24 - 05/13/2012 12:08 PM - luislavena (Luis Lavena)

- Status changed from Closed to Assigned

- % Done changed from 100 to 0

Reopening as reported in [ruby-core:45015] this breaks chkbuid on debian.

#25 - 05/13/2012 09:22 PM - h.shirosaki (Hiroshi Shirosaki)

Hi Luis,

I think Nagachika-san's original patch as below is sufficient for this issue. GC.disable for all tests seems too far.

Did you test with this patch?

```
diff --git a/test/ruby/test_io.rb b/test/ruby/test_io.rb
index ebf9b76..be5a333 100644
--- a/test/ruby/test_io.rb
+++ b/test/ruby/test_io.rb
@@ -1891,7 +1891,7 @@ End
```

```
}
end
ensure

• fds.each {|fd| IO.for_fd(fd).close rescue next}
• GC.start end
```

```
def test_flush_in_finalizer2
```

#26 - 05/13/2012 11:31 PM - nagachika (Tomoyuki Chikanaga)

Hi Luis,

Sorry for a lack of enough explanations about my patch.

The ensure clause of test_flush_in_finalizer1 seems to try close all IO objects opened above and not GC'ed & finalized using IO.for_fd. But it's wrong usage of IO.for_fd. The scenario is as follows.

```
io = open("../")
fd = io.fileno
io = nil
IO.for_fd(fd).close # <- at this moment, io (if not GC'ed yet) has invalid fd,
# because it's already closed by IO.for_fd(fd).close
new_io = open("../") # new_io might have same fd of io, because file descriptor can be recycled.
... # => io is GC'ed and finalized here. Now new_io has closed fd!
new_io.gets # => raised Errno::EBADF
```

test_flush_in_finalizer1 is test for finalizer of IO (see [#3910](#)) and GC.disable breaks it's intent.

I'm sorry, but I'd like to revert [r35631](#).

And I'll commit my patch. Please check it.

#27 - 05/13/2012 11:50 PM - nagachika (Tomoyuki Chikanaga)

- Status changed from Assigned to Closed

- % Done changed from 0 to 100

This issue was solved with changeset [r35637](#).

Jon, thank you for reporting this issue.

Your contribution to Ruby is greatly appreciated.

May Ruby be with you.

-
- test/ruby/test_io.rb (test_flush_in_finalizer1): don't use IO.for_fd to close IO objects. it create IO object with already closed fd, and cause occasional Errno::EBADF in following tests. [ruby-core:45020] [Bug [#6228](#)]

#28 - 05/14/2012 01:53 AM - luislavena (Luis Lavena)

Hi Tomoyuki,

Please revert my change. As commented before, was unable to use your suggested patch since the invalid file descriptor error 'moved' to a different test.

I was unable to replicate the error from chkbuild.

Please revert and commit your change, will verify again later today.

Thank you.

Sorry for top posting. Sent from mobile.

On May 13, 2012 11:32 AM, "nagachika (Tomoyuki Chikanaga)" <nagachika00@gmail.com> wrote:

Issue [#6228](#) has been updated by nagachika (Tomoyuki Chikanaga).

Hi Luis,

Sorry for a lack of enough explanations about my patch.

The ensure clause of test_flush_in_finalizer1 seems to try close all IO objects opened above and not GC'ed & finalized using IO.for_fd. But it's wrong usage of IO.for_fd. The scenario is as follows.

```
io = open("../")
```

```
fd = io.fileno
io = nil
IO.for_fd(fd).close # <- at this moment, io (if not GC'ed yet) has
invalid fd,
# because it's already closed by IO.for_fd(fd).close
new_io = open(...) # new_io might have same fd of io, because file
descriptor can be recycled.
... # => io is GC'ed and finalized here. Now new_io has closed fd!
new_io.gets # => raised Errno::EBADF
```

test_flush_in_finalizer1 is test for finalizer of IO (see [#3910](#)) and GC.disable breaks it's intent.
I'm sorry, but I'd like to revert [r35631](#).

And I'll commit my patch. Please check it.

Bug [#6228](#): [mingw] Errno::EBADF in ruby/test_io.rb on ruby_1_9_3
<https://bugs.ruby-lang.org/issues/6228#change-26601>

Author: jonforums (Jon Forums)
Status: Assigned
Priority: Normal
Assignee: luislavena (Luis Lavena)
Category: test
Target version: 1.9.3
ruby -v: ruby 1.9.3p172 (2012-03-30 revision 35179) [i386-mingw32]

On Win7 32bit built with MinGW 4.6.2 using the RubyInstaller build recipes, I get the following test error:

```
sh-3.1$ ruby --version
ruby 1.9.3p172 (2012-03-30 revision 35179) [i386-mingw32]
```

```
sh-3.1$ make test-all TESTS='openssl fiddle psych zlib io json mkmf
pathname stringio erb
fileutils ruby/test_io.rb ruby/test_io_m17n.rb ruby/test_file.rb'
```

```
7) Error:
test_lines(TestIO):
Errno::EBADF: Bad file descriptor
c:/Users/Jon/Documents/RubyDev/ruby-git/test/ruby/test_io.rb:33:in
close'
c:/Users/Jon/Documents/RubyDev/ruby-git/test/ruby/test_io.rb:33:in
rescue in block in pipe'
c:/Users/Jon/Documents/RubyDev/ruby-git/test/ruby/test_io.rb:30:in
`block in pipe'
```

--
<http://bugs.ruby-lang.org/>