Ruby master - Bug #8644
valgrind error in a readline test
07/16/2013 09:13 PM - akr (Akira Tanaka)

Status: Closed
Priority: Normal
Assignee: kouji (Kouji Takao)
Target version: 2.6
ruby -v: ruby 2.1.0dev (2013-07-16 trunk 42006) [x86_64-linux]

Backport: 1.9.3: UNKNOWN, 2.0.0: UNKNOWN

Description

test-all の readline で SEGV が発生しました。
valgrind をかけてみたところ、1つの問題を指摘されました。

% ./ruby -v
ruby 2.1.0dev (2013-07-16 trunk 42006) [x86_64-linux]
% valgrind ./ruby -I.ext/x86_64-linux ../../ruby/test/runner.rb ../../ruby/test/readline -n test_closed_outstream
==27651== Memcheck, a memory error detector
==27651== Copyright (C) 2002-2011, and GNU GPL'd, by Julian Seward et al.
==27651== Using Valgrind-3.7.0 and LibVEX; rerun with -h for copyright info
==27651== Command: ./ruby -I.ext/x86_64-linux ../../ruby/test/runner.rb ../../ruby/test/readline -n test_closed_outstream
==27651== Run options: -n test_closed_outstream

# Running tests:

[1/1] TestReadline#test_closed_outstream
==27651== Invalid read of size 4
==27651== at 0x597B610: fileno (fileno.c:37)
==27651== by 0x74A3E6A: readline_readline (readline.c:397)
==27651== by 0x390B39: call_cfunc_m1 (vm_insnhelper.c:1348)
==27651== by 0x396943: vm_call_cfunc_with_frame (vm_insnhelper.c:1492)
==27651== by 0x395144: vm_call_cfunc (vm_insnhelper.c:1582)
==27651== by 0x394915: vm_call_method (vm_insnhelper.c:1774)
==27651== by 0x396A84: vm_call_general (vm_insnhelper.c:1925)
==27651== by 0x3724DC: vm_exec_core (insns.def:1017)
==27651== by 0x383DDF: vm_exec (vm.c:1198)
==27651== by 0x389DBD: invoke_block_from_c (vm.c:646)
==27651== by 0x38E299: vm_yield (vm.c:677)
==27651== by 0x380012: rb_yield_0 (vm_eval.c:937)
==27651== Address 0x847a1a0 is 0 bytes inside a block of size 568 free'd
==27651== at 0x4C27D4E: free (vg_replace_malloc.c:427)
==27651== by 0x5977CCC: fclose@@GLIBC_2.2.5 (iofclose.c:88)
==27651== by 0x190F5C: nogvl_fclose (io.c:4057)
==27651== by 0x3A3532: rb_thread_call_without_gvl (thread.c:1354)
==27651== by 0x3A3532: rb_thread_call_without_gvl (thread.c:1354)
==27651== Finished tests in 0.521803s, 1.9164 tests/s, 5.7493 assertions/s.
1 tests, 3 assertions, 0 failures, 0 errors, 0 skips

ruby -v: ruby 2.1.0dev (2013-07-16 trunk 42006) [x86_64-linux]
==27651==
==27651== HEAP SUMMARY:
==27651== in use at exit: 1,905,554 bytes in 30,452 blocks
==27651== total heap usage: 94,021 allocs, 63,569 frees, 19,535,963 bytes allocated
==27651==
LEAK SUMMARY:

- definitely lost: 447,671 bytes in 3,643 blocks
- indirectly lost: 979,312 bytes in 17,524 blocks
- possibly lost: 0 bytes in 0 blocks
- still reachable: 478,571 bytes in 9,285 blocks
- suppressed: 0 bytes in 0 blocks

Rerun with --leak-check=full to see details of leaked memory

ERROR SUMMARY: 1 errors from 1 contexts (suppressed: 4 from 4)

Related issues:
Related to Ruby master - Bug #12950: irb: 'input-method.rb:151: [BUG] Segment... 

ASSOCIATED REVISIONS

Revision d2fa5dff - 08/06/2013 12:24 PM - kouji (Kouji Takao)
- ext/readline/readline.c (Init_readline, readline_s_set_output) (clear_rl_outstream, readline_s_set_input, clear_rl_instream) (readline_readline): fix causing SEGV if closed IO object that is set Readline.input or Readline.output. Patched by akr [ruby-dev:47509] [Bug #8644]

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@42402 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision 42402 - 08/06/2013 12:24 PM - kouji (Kouji Takao)
- ext/readline/readline.c (Init_readline, readline_s_set_output) (clear_rl_outstream, readline_s_set_input, clear_rl_instream) (readline_readline): fix causing SEGV if closed IO object that is set Readline.input or Readline.output. Patched by akr [ruby-dev:47509] [Bug #8644]

Revision 42402 - 08/06/2013 12:24 PM - kouji (Kouji Takao)
- ext/readline/readline.c (Init_readline, readline_s_set_output) (clear_rl_outstream, readline_s_set_input, clear_rl_instream) (readline_readline): fix causing SEGV if closed IO object that is set Readline.input or Readline.output. Patched by akr [ruby-dev:47509] [Bug #8644]

Revision 42402 - 08/06/2013 12:24 PM - kouji (Kouji Takao)
- ext/readline/readline.c (Init_readline, readline_s_set_output) (clear_rl_outstream, readline_s_set_input, clear_rl_instream) (readline_readline): fix causing SEGV if closed IO object that is set Readline.input or Readline.output. Patched by akr [ruby-dev:47509] [Bug #8644]

Revision 42402 - 08/06/2013 12:24 PM - kouji (Kouji Takao)
- ext/readline/readline.c (Init_readline, readline_s_set_output) (clear_rl_outstream, readline_s_set_input, clear_rl_instream) (readline_readline): fix causing SEGV if closed IO object that is set Readline.input or Readline.output. Patched by akr [ruby-dev:47509] [Bug #8644]

Revision 42402 - 08/06/2013 12:24 PM - kouji (Kouji Takao)
- ext/readline/readline.c (Init_readline, readline_s_set_output) (clear_rl_outstream, readline_s_set_input, clear_rl_instream) (readline_readline): fix causing SEGV if closed IO object that is set Readline.input or Readline.output. Patched by akr [ruby-dev:47509] [Bug #8644]

Revision 42402 - 08/06/2013 12:24 PM - kouji (Kouji Takao)
- ext/readline/readline.c (Init_readline, readline_s_set_output) (clear_rl_outstream, readline_s_set_input, clear_rl_instream) (readline_readline): fix causing SEGV if closed IO object that is set Readline.input or Readline.output. Patched by akr [ruby-dev:47509] [Bug #8644]

Revision 42402 - 08/06/2013 12:24 PM - kouji (Kouji Takao)
- ext/readline/readline.c (Init_readline, readline_s_set_output) (clear_rl_outstream, readline_s_set_input, clear_rl_instream) (readline_readline): fix causing SEGV if closed IO object that is set Readline.input or Readline.output. Patched by akr [ruby-dev:47509] [Bug #8644]

Revision 42402 - 08/06/2013 12:24 PM - kouji (Kouji Takao)
- ext/readline/readline.c (Init_readline, readline_s_set_output) (clear_rl_outstream, readline_s_set_input, clear_rl_instream) (readline_readline): fix causing SEGV if closed IO object that is set Readline.input or Readline.output. Patched by akr [ruby-dev:47509] [Bug #8644]

Revision 42402 - 08/06/2013 12:24 PM - kouji (Kouji Takao)
- ext/readline/readline.c (Init_readline, readline_s_set_output) (clear_rl_outstream, readline_s_set_input, clear_rl_instream) (readline_readline): fix causing SEGV if closed IO object that is set Readline.input or Readline.output. Patched by akr [ruby-dev:47509] [Bug #8644]

Revision 42402 - 08/06/2013 12:24 PM - kouji (Kouji Takao)
- ext/readline/readline.c (Init_readline, readline_s_set_output) (clear_rl_outstream, readline_s_set_input, clear_rl_instream) (readline_readline): fix causing SEGV if closed IO object that is set Readline.input or Readline.output. Patched by akr [ruby-dev:47509] [Bug #8644]

Revision 42402 - 08/06/2013 12:24 PM - kouji (Kouji Takao)
- ext/readline/readline.c (Init_readline, readline_s_set_output) (clear_rl_outstream, readline_s_set_input, clear_rl_instream) (readline_readline): fix causing SEGV if closed IO object that is set Readline.input or Readline.output. Patched by akr [ruby-dev:47509] [Bug #8644]

Revision 874dc256 - 08/11/2013 05:58 PM - naruse (Yui NARUSE)
- ext/readline/readline.c (readline_s_set_input): on OS X with editline, Readline.readline doesn't work because readline_get doesn't use
rl_getc. The difference is introduced by r42402 [ruby-dev:47509] [Bug #8644]. Before it rb_io_stdio_file set ifp->stdio_file. Therefore add manually setting the value.

- ext/readline/readline.c (readline_s_set_onput): ditto.

Revision 42525 - 08/11/2013 05:58 PM - naruse (Yui NARUSE)

- ext/readline/readline.c (readline_s_set_input): on OS X with editline, Readline.readline doesn't work because readline_get doesn't use rl_getc. The difference is introduced by r42402 [ruby-dev:47509] [Bug #8644]. Before it rb_io_stdio_file set ifp->stdio_file. Therefore add manually setting the value.
- ext/readline/readline.c (readline_s_set_onput): ditto.

Revision 42525 - 08/11/2013 05:58 PM - naruse (Yui NARUSE)

- ext/readline/readline.c (readline_s_set_input): on OS X with editline, Readline.readline doesn't work because readline_get doesn't use rl_getc. The difference is introduced by r42402 [ruby-dev:47509] [Bug #8644]. Before it rb_io_stdio_file set ifp->stdio_file. Therefore add manually setting the value.
- ext/readline/readline.c (readline_s_set_onput): ditto.

Revision 42525 - 08/11/2013 05:58 PM - naruse (Yui NARUSE)

- ext/readline/readline.c (readline_s_set_input): on OS X with editline, Readline.readline doesn't work because readline_get doesn't use rl_getc. The difference is introduced by r42402 [ruby-dev:47509] [Bug #8644]. Before it rb_io_stdio_file set ifp->stdio_file. Therefore add manually setting the value.
- ext/readline/readline.c (readline_s_set_onput): ditto.
Readline.readline doesn't work because readline_get doesn't use rl_getc. The difference is introduced by r42402 [ruby-dev:47509] [Bug #8644]. Before it rb_io_stdio_file set ifp->stdio_file. Therefore add manually setting the value.

- ext/readline/readline.c (readline_s_set_onput): ditto.

Revision 42525 - 08/11/2013 05:58 PM - naruse (Yui NARUSE)

- ext/readline/readline.c (readline_s_set_input): on OS X with editline,
  Readline.readline doesn't work because readline_get doesn't use rl_getc. The difference is introduced by r42402 [ruby-dev:47509] [Bug #8644]. Before it rb_io_stdio_file set ifp->stdio_file. Therefore add manually setting the value.

- ext/readline/readline.c (readline_s_set_onput): ditto.

History

#1 - 07/16/2013 09:47 PM - akr (Akira Tanaka)
- File test_closed_outstream-segv.txt added

% valgrind ./ruby -rreadline -e 'r, w = IO.pipe
Readline.input = r
Readline.output = w
r.close
Readline.readline'
==10001== Memcheck, a memory error detector
==10001== Copyright (C) 2002-2010, and GNU GPL'd, by Julian Seward et al.
==10001== Using Valgrind-3.6.0.SVN-Debian and LibVEX; rerun with -h for copyright info
==10001== Command: ./ruby -rreadline -e
==10001== r: w = IO.pipe
==10001== Readline.input = r
==10001== Readline.output = w
==10001== r.close
==10001== Readline.readline'
==10001== Invalid read of size 4
==10001== at 0x59706A0: fileno (fileno.c:37)
==10001== by 0x5EDB699: readline_readline (readline.c:394)
==10001== by 0x255A5FE: call_cfunc_m1 (vm_inshhelper.c:1348)
==10001== by 0x25556EE: vm_call_cfunc_with_frame (vm_inshhelper.c:1492)
==10001== by 0x255828: vm_call_cfunc (vm_inshhelper.c:1582)
==10001== by 0x25A368: vm_call_method (vm_inshhelper.c:1774)
==10001== by 0x25A8B02: vm_call_general (vm_inshhelper.c:1925)
==10001== by 0x25E4D0: vm_exec_core (insns.def:1017)

#2 - 07/18/2013 05:41 PM - akr (Akira Tanaka)
- Description updated

#3 - 07/18/2013 07:49 PM - akr (Akira Tanaka)

% valgrind ./ruby -rreadline -e 'r, w = IO.pipe
Readline.input = r
Readline.output = w
r.close
Readline.readline'
==10001== Memcheck, a memory error detector
==10001== Copyright (C) 2002-2010, and GNU GPL'd, by Julian Seward et al.
==10001== Using Valgrind-3.6.0.SVN-Debian and LibVEX; rerun with -h for copyright info
==10001== Command: ./ruby -rreadline -e
==10001== r: w = IO.pipe
==10001== Readline.input = r
==10001== Readline.output = w
==10001== r.close
==10001== Readline.readline'
==10001== Invalid read of size 4
==10001== at 0x59706A0: fileno (fileno.c:37)
==10001== by 0x5EDB699: readline_readline (readline.c:394)
==10001== by 0x255A5FE: call_cfunc_m1 (vm_inshhelper.c:1348)
==10001== by 0x25556EE: vm_call_cfunc_with_frame (vm_inshhelper.c:1492)
==10001== by 0x255828: vm_call_cfunc (vm_inshhelper.c:1582)
==10001== by 0x25A368: vm_call_method (vm_inshhelper.c:1774)
==10001== by 0x25A8B02: vm_call_general (vm_inshhelper.c:1925)
==10001== by 0x25E4D0: vm_exec_core (insns.def:1017)
=10001== by 0x26D8C8: vm_exec (vm.c:1198)
=10001== by 0x26E53C: rb_iseq_eval_main (vm.c:1448)
=10001== by 0x12D3CC: ruby_exec_internal (eval.c:252)
=10001== by 0x12D4DF: ruby_exec_node (eval.c:317)
=10001== Address 0x6e5f970 is 0 bytes inside a block of size 568 free'd
=10001== at 0x4C240FD: free (vg_replace_malloc.c:366)
=10001== by 0x596C5FC: fclose@@GLIBC_2.2.5 (iofclose.c:88)
=10001== by 0x6EDB6EF: readline_readline (readline.c:397)
=10001== by 0x258AFE: call_cfunc_m1 (vm_insnhelper.c:1348)
=10001== by 0x25A368: vm_call_method (vm_insnhelper.c:1348)
=10001== by 0x25A0B2: vm_call_general (vm_insnhelper.c:1348)
=10001== by 0x25E4D0: vm_exec_core (insns.def:1017)
=10001== by 0x26D8C8: vm_exec (vm.c:1198)
=10001== by 0x26E53C: rb_iseq_eval_main (vm.c:1448)
=10001== by 0x12D3CC: ruby_exec_internal (eval.c:252)
=10001== by 0x12D4DF: ruby_exec_node (eval.c:317)
=10001== Address 0x6e5fbf0 is 0 bytes inside a block of size 568 free'd
=10001== at 0x4C240FD: free (vg_replace_malloc.c:366)
=10001== by 0x596C5FC: fclose@@GLIBC_2.2.5 (iofclose.c:88)
=10001== by 0x157014: fptr_finalize (io.c:4057)
=10001== by 0x15712B: rb_io_fptr_cleanup (io.c:4099)
=10001== by 0x157464: rb_io_close_m (io.c:4221)
=10001== by 0x15740C: rb_io_close (io.c:4250)
=10001== by 0x156E76: nogvl_fclose (io.c:4254)
=10001== by 0x27AB3D: call_without_gvl (thread.c:1244)
=10001== by 0x27AC6F: rb_thread_call_without_gvl (thread.c:1354)
=10001== by 0x156EBC: maygvf_fclose (io.c:4057)
=10001== by 0x157014: fptr_finalize (vg_replace_malloc.c:366)
=10001== by 0x596C5FC: fclose@@GLIBC_2.2.5 (iofclose.c:88)
=10001== by 0x15712B: rb_io_fptr_cleanup (io.c:4099)
=10001== by 0x27AB3D: call_without_gvl (thread.c:1244)
=10001== by 0x27AC6F: rb_thread_call_without_gvl (thread.c:1354)
=10001== by 0x156EBC: maygvf_fclose (io.c:4057)
=10001== by 0x157014: fptr_finalize (io.c:4057)

```
% valgrind ./ruby -rreadline -e
r, w = IO.pipe
Readline.input = r
Readline.output = w
w.close
Readline.readline'

--- Valgrind start output ---------------------------------------

`Readline.output` test ทดสอบการใช้งานของ `Readline.output`

```
% valgrind ./ruby -rreadline -e
r, w = IO.pipe
Readline.input = r
Readline.output = w
w.close
Readline.readline'
```

--- Valgrind end output ---------------------------------------

%% invalid read of size 4

```
==9994== Memcheck, a memory error detector
==9994== Copyright (C) 2002-2010, and GNU GPL'd, by Julian Seward et al.
==9994== Using Valgrind-3.6.0.SVN-Debian and LibVEX; rerun with -h for copyright info
==9994== Command: ./ruby -rreadline -e
==9994== r, w = IO.pipe
==9994== Readline.input = r
==9994== Readline.output = w
==9994== w.close
==9994== Readline.readline'
```

--- Valgrind end output ---------------------------------------

%% Invalid read of size 4

```
==9994== at 0x59706A0: fileno (fileno.c:37)
==9994== by 0x6E6B6EF: readline_readline (readline.c:397)
==9994== by 0x258AFE: call_cfunc_m1 (vm_insnhelper.c:1348)
==9994== by 0x259828: vm_call_cfunc_with_frame (vm_insnhelper.c:13492)
==9994== by 0x259828: vm_call_cfunc_with_frame (vm_insnhelper.c:1582)
==9994== by 0x25A368: vm_call_method (vm_insnhelper.c:1348)
==9994== by 0x25A0B2: vm_call_general (vm_insnhelper.c:1348)
==9994== by 0x25E4D0: vm_exec_core (insns.def:1017)
==9994== by 0x26D8C8: vm_exec (vm.c:1198)
==9994== by 0x26E53C: rb_iseq_eval_main (vm.c:1448)
==9994== by 0x12D3CC: ruby_exec_internal (eval.c:252)
==9994== by 0x12D4DF: ruby_exec_node (eval.c:317)
==9994== Address 0x6e5f970 is 0 bytes inside a block of size 568 free'd
==9994== at 0x4C240FD: free (vg_replace_malloc.c:366)
==9994== by 0x596C5FC: fclose@@GLIBC_2.2.5 (iofclose.c:88)
==9994== by 0x156E76: nogvl_fclose (io.c:4057)
==9994== by 0x27AB3D: call_without_gvl (thread.c:1244)
==9994== by 0x27AC6F: rb_thread_call_without_gvl (thread.c:1354)
==9994== by 0x156EBC: maygvf_fclose (io.c:4057)
```

```
03/20/2022
```

```
5/7
```
valgrind ./ruby -rreadline -e

% ./ruby -v
ruby 2.1.0dev (2013-07-18 trunk 42040) [x86_64-linux]

#4 - 07/18/2013 11:02 PM - akr (Akira Tanaka)
- File readline-fix-invalid-read.patch added

```bash
03/20/2022
```

```
0x157014: fptr_finalize (io.c:4099)
0x15712B: rb_io_fptr_cleanup (io.c:4130)
0x15740C: rb_io_close (io.c:4221)
0x157464: rb_io_close_m (io.c:4250)
0x258B2E: call_cfunc_0 (vm_insnhelper.c:1354)
0x2596EE: vm_call_cfunc_with_frame (vm_insnhelper.c:1492)

-e:6:in readline': closed stdout (IOError)
```

```
=9994==  by 0x157014: fptr_finalize (io.c:4099)
=9994==  by 0x15712B: rb_io_fptr_cleanup (io.c:4130)
=9994==  by 0x15740C: rb_io_close (io.c:4221)
=9994==  by 0x157464: rb_io_close_m (io.c:4250)
=9994==  by 0x258B2E: call_cfunc_0 (vm_insnhelper.c:1354)
=9994==  by 0x2596EE: vm_call_cfunc_with_frame (vm_insnhelper.c:1492)

=-e:6:in readline': closed stdout (IOError)
```
#6 - 08/01/2013 10:52 PM - akr (Akira Tanaka)
http://rbci.lakewood.privs.net/ruby-trunk/log/20130801T131001Z.diff.html.gz

... TestRange#test_bsearch_for_other_numerics = s = .
TestRange#test_bsearch_typechecks_return_values = s = .
+TestRange#test_bsearch_with_mathn = /home/chkbuild/build/ruby/test/readline/test_readline.rb:299: [BUG] Segmentation fault
ruby

+-- Control frame information -------------------------------
+c:0026 p:0013 s:0122 e:000121 METHOD /home/chkbuild/build/ruby/lib/minitest/unit.rb:362
+c:0028 p:0013 s:0127 e:000126 CFUNC :readline
...

Rubyにセグメンテーションフォールトが発生しています。
test_readline.rb:299 は test_closed_outstream で呼ばれているところです。
Signatureが不思議なのでどうでしょうか。

#7 - 08/06/2013 09:11 PM - kouji (Kouji Takao)

パッチありがとうございます。取り込みます。

#8 - 08/06/2013 09:24 PM - kouji (Kouji Takao)
- Status changed from Assigned to Closed
- % Done changed from 0 to 100

This issue was solved with changeset r42402.
Akira, thank you for reporting this issue.
Your contribution to Ruby is greatly appreciated.
May Ruby be with you.

- ext/readline/readline.c (Init_readline, readline_s_set_output) (clear_rl_outstream, readline_s_set_input, clear_rl_instream) (readline_readline): fix causing SEGV if closed IO object that is set Readline.input or Readline.output. Patched by akr [ruby-dev:47509] [Bug #8644]

#9 - 11/26/2016 08:05 AM - wanabe (_ wanabe)

- Related to Bug #12950: irb: 'input-method.rb:151: [BUG] Segmentation fault' / 'malloc(): smallbin double linked list corrupted' added

Files
| test_closed_outstream-segv.txt | 166 KB | 07/16/2013 | akr (Akira Tanaka) |
| readline-fix-invalid-read.patch | 5.14 KB | 07/18/2013 | akr (Akira Tanaka) |