Ruby master - Bug #3005
Ruby core dump - [BUG] rb_sys_fail() - errno == 0

03/24/2010 09:24 PM - syepes (Sebastian YEPES)

Status: Closed
Priority: Normal
Assignee: 
Target version: 2.0.0
ruby -v: 1.9.1p376

Backport:

Description
=begin
My ruby script is generating cores, this is the dump:

---
sv.rb:268: [BUG] rb_sys_fail() - errno == 0
ruby 1.9.1p376 (2009-12-07 revision 26041) [ia64-hpux11.31]

-- control frame ------------
c:0005 p:---- s:0016 b:0016 l:000015 d:000015 CFUNC :gets
c:0002 p:---- s:0004 b:0004 l:000003 d:000003 FINISH
c:0001 p:0000 s:0002 b:0002 l:000003 d:000003 TOP

-- Ruby level backtrace information-----------------------------
sv.rb:268:in gets'
sv.rb:268:insvcm_listener'
sv.rb:780:in ''

[NOTE]
You may encounter a bug of Ruby interpreter. Bug reports are welcome.
For details: http://www.ruby-lang.org/bugreport.html

---Type to continue, or q to quit---
#13 0x60000000d872190:0 in vm_call_method (th=, cfp=, num=, blockptr=,
**History**

### #1 - 03/25/2010 09:21 AM - nobu (Nobuyoshi Nakada)

- Status changed from Open to Feedback

```
begin
How to reproduce it?
=end
```

### #2 - 03/25/2010 06:46 PM - syepes (Sebastian YEPES)

- File sv.rb added

```
begin
Ok, I have made a reduced version of the script that always reproduces the core dumps. The problem must be related with the oci8, because if I comment out the lines 23 and 25 it does not generate cores.

Hope some one can help me solve this issue.

Environment:
HP-UX B.11.31 U ia64
ruby-oci8-2.0.4
ruby 1.9.1p376 (2009-12-07 revision 26041) [ia64-hpux11.31]
Oracle 11gR2

Reproduction:
ruby sv.rb
Send a HUP to the ruby process (kill -HUP -> Here comes the core ;)
```

# ruby sv.rb

cfgfile_load: LOADING CONFIGURATION DATA
xml_listener: Processing XML
xml_register_hup: RELOADING CONFIGURATION DATA
cfgfile_load: LOADING CONFIGURATION DATA
sv.rb:44: [BUG] rb_sys_fail() - errno == 0
ruby 1.9.1p376 (2009-12-07 revision 26041) [ia64-hpux11.31]

-- control frame ---------
c:0002 p:---- s:0004 b:0004 l:000003 d:000003 FINISH
c:0001 p:0000 s:0002 b:0002 l:0009b4 d:0009b4 TOP

-- Ruby level backtrace information----------------------------------------
sv.rb:44:in gets'
sv.rb:44:in xml_listener'
sv.rb:60:in ''

[NOTE]
You may encounter a bug of Ruby interpreter. Bug reports are welcome. For details: [http://www.ruby-lang.org/bugreport.html](http://www.ruby-lang.org/bugreport.html)

Abort(coredump)
I have just tested this problem with the last version of Ruby 1.9.1p378 and it also generates core dumps.

Anyone know what can be causing this issue?

---

If it is caused by the signal handler setting changed by Oracle client library, it may be fixed by adding BEQUEATH_DETACH=yes to sqlnet.ora.

http://download.oracle.com/docs/cd/B19306_01/network.102/b14213/sqlnet.htm#sthref374

---

Thanks for the info, I have just tested adding this setting to: /opt/oracle/client/product/11.1.0/network/admin/sqlnet.ora but I am still getting the cores ;-;

Do you know what we can to to debug this issue?

---

If I do, I'll install tusc(*1) and run the following command to trace system calls.

tusc -o tusc.log ruby sv.rb


---

The issue will be fixed the following patch.
Ok, I have made a reduced version of the script that always reproduces the core dumps. The problem must be related with the oci8, because if I comment out the lines 23 and 25 it does not generate cores.

The problem is caused by resetting errno in the signal handler. It can be reproduced by replacing line 23 and 25 with Math.log10(10), which set errno to zero.
Hi,
At Fri, 9 Apr 2010 00:51:49 +0900, Takehiro Kubo wrote in [ruby-core:29340]:

Ok, I have made a reduced version of the script that always reproduces the core dumps. The problem must be related with the oci8, because if I comment out the lines 23 and 25 it does not generate cores.

The problem is caused by resetting errno in the signal handler. It can be reproduced by replacing line 23 and 25 with Math.log10(10), which set errno to zero.

Thank you for your investigation and patch. It seems correct and I've merged it into the trunk.

--
Nobu Nakada

This issue was solved with changeset r27265. Sebastian, thank you for reporting this issue. Your contribution to Ruby is greatly appreciated. May Ruby be with you.