

## Ruby master - Bug #9820

### miniruby -e 'Process.kill(:INT, \$\$)' hang under cron

05/09/2014 08:06 AM - akr (Akira Tanaka)

<b>Status:</b> Closed	
<b>Priority:</b> Normal	
<b>Assignee:</b>	
<b>Target version:</b>	
<b>ruby -v:</b> ruby 2.2.0dev (2014-05-09 trunk 45884) [x86_64-linux]	<b>Backport:</b> 2.0.0: DONTNEED, 2.1: DONE

#### Description

TestBeginEndBlock#test\_propagate\_signaled hang  
<http://chkbuid002.hsbt.org/chkbuid/ruby-trunk/log/20140509T060013Z.fail.html.gz>

miniruby -e 'Process.kill(:INT, \$\$)'

cron hang  
hang

strace  
cron timer thread prctl poll  
context switch

strace strace.cron.log strace.term.log

?

```
% ./miniruby -v
ruby 2.2.0dev (2014-05-09 trunk 45884) [x86_64-linux]
% uname -mrsv
Linux 3.2.0-4-amd64 #1 SMP Debian 3.2.54-2 x86_64
% lsb_release -idrc
Distributor ID: Debian
Description:    Debian GNU/Linux 7.5 (wheezy)
Release:        7.5
Codename:       wheezy
```

#### Associated revisions

##### Revision 77af38d0 - 05/10/2014 04:10 PM - nobu (Nobuyoshi Nakada)

signal.c: directly enqueue

- signal.c (rb\_f\_kill): directly enqueue an ignored signal to self, except for SIGSEGV and SIGBUS. [ruby-dev:48203] [Bug #9820]

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@45911 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

##### Revision 45911 - 05/10/2014 04:10 PM - nobu (Nobuyoshi Nakada)

signal.c: directly enqueue

- signal.c (rb\_f\_kill): directly enqueue an ignored signal to self, except for SIGSEGV and SIGBUS. [ruby-dev:48203] [Bug #9820]

##### Revision 45911 - 05/10/2014 04:10 PM - nobu (Nobuyoshi Nakada)

signal.c: directly enqueue

- signal.c (rb\_f\_kill): directly enqueue an ignored signal to self, except for SIGSEGV and SIGBUS. [ruby-dev:48203] [Bug #9820]

**Revision 45911 - 05/10/2014 04:10 PM - nobu (Nobuyoshi Nakada)**

signal.c: directly enqueue

- signal.c (rb\_f\_kill): directly enqueue an ignored signal to self, except for SIGSEGV and SIGBUS. [ruby-dev:48203] [Bug #9820]

**Revision 45911 - 05/10/2014 04:10 PM - nobu (Nobuyoshi Nakada)**

signal.c: directly enqueue

- signal.c (rb\_f\_kill): directly enqueue an ignored signal to self, except for SIGSEGV and SIGBUS. [ruby-dev:48203] [Bug #9820]

**Revision 45911 - 05/10/2014 04:10 PM - nobu (Nobuyoshi Nakada)**

signal.c: directly enqueue

- signal.c (rb\_f\_kill): directly enqueue an ignored signal to self, except for SIGSEGV and SIGBUS. [ruby-dev:48203] [Bug #9820]

**Revision 45911 - 05/10/2014 04:10 PM - nobu (Nobuyoshi Nakada)**

signal.c: directly enqueue

- signal.c (rb\_f\_kill): directly enqueue an ignored signal to self, except for SIGSEGV and SIGBUS. [ruby-dev:48203] [Bug #9820]

**Revision 3771a370 - 05/10/2014 04:10 PM - nobu (Nobuyoshi Nakada)**

thread.c: always deliver signal immediately

- thread.c (ruby\_kill): always deliver signal immediately, without check for main thread. no longer called in other context. [ruby-dev:48203] [Bug #9820]

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@45912 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

**Revision 45912 - 05/10/2014 04:10 PM - nobu (Nobuyoshi Nakada)**

thread.c: always deliver signal immediately

- thread.c (ruby\_kill): always deliver signal immediately, without check for main thread. no longer called in other context. [ruby-dev:48203] [Bug #9820]

**Revision 45912 - 05/10/2014 04:10 PM - nobu (Nobuyoshi Nakada)**

thread.c: always deliver signal immediately

- thread.c (ruby\_kill): always deliver signal immediately, without check for main thread. no longer called in other context. [ruby-dev:48203] [Bug #9820]

**Revision 45912 - 05/10/2014 04:10 PM - nobu (Nobuyoshi Nakada)**

thread.c: always deliver signal immediately

- thread.c (ruby\_kill): always deliver signal immediately, without check for main thread. no longer called in other context. [ruby-dev:48203] [Bug #9820]

**Revision 45912 - 05/10/2014 04:10 PM - nobu (Nobuyoshi Nakada)**

thread.c: always deliver signal immediately

- thread.c (ruby\_kill): always deliver signal immediately, without check for main thread. no longer called in other context. [ruby-dev:48203] [Bug #9820]

**Revision 45912 - 05/10/2014 04:10 PM - nobu (Nobuyoshi Nakada)**

thread.c: always deliver signal immediately

- thread.c (ruby\_kill): always deliver signal immediately, without check for main thread. no longer called in other context. [ruby-dev:48203] [Bug #9820]

**Revision 45912 - 05/10/2014 04:10 PM - nobu (Nobuyoshi Nakada)**

thread.c: always deliver signal immediately

- thread.c (ruby\_kill): always deliver signal immediately, without check for main thread. no longer called in other context. [ruby-dev:48203] [Bug #9820]

**Revision 61cd4e9d - 05/11/2014 03:19 AM - nobu (Nobuyoshi Nakada)**

test\_process.rb: WIFEXITED and so on are system dependent

- test/ruby/test\_process.rb (test\_status\_kill): WIFEXITED and so on are available only if signal is supported by the system. [ruby-dev:48203] [Bug #9820]

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@45918 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

**Revision 45918 - 05/11/2014 03:19 AM - nobu (Nobuyoshi Nakada)**

test\_process.rb: WIFEXITED and so on are system dependent

- test/ruby/test\_process.rb (test\_status\_kill): WIFEXITED and so on are available only if signal is supported by the system. [ruby-dev:48203] [Bug #9820]

**Revision 45918 - 05/11/2014 03:19 AM - nobu (Nobuyoshi Nakada)**

test\_process.rb: WIFEXITED and so on are system dependent

- test/ruby/test\_process.rb (test\_status\_kill): WIFEXITED and so on are available only if signal is supported by the system. [ruby-dev:48203] [Bug #9820]

**Revision 45918 - 05/11/2014 03:19 AM - nobu (Nobuyoshi Nakada)**

test\_process.rb: WIFEXITED and so on are system dependent

- test/ruby/test\_process.rb (test\_status\_kill): WIFEXITED and so on are available only if signal is supported by the system. [ruby-dev:48203] [Bug #9820]

**Revision 45918 - 05/11/2014 03:19 AM - nobu (Nobuyoshi Nakada)**

test\_process.rb: WIFEXITED and so on are system dependent

- test/ruby/test\_process.rb (test\_status\_kill): WIFEXITED and so on are available only if signal is supported by the system. [ruby-dev:48203] [Bug #9820]

**Revision 45918 - 05/11/2014 03:19 AM - nobu (Nobuyoshi Nakada)**

test\_process.rb: WIFEXITED and so on are system dependent

- test/ruby/test\_process.rb (test\_status\_kill): WIFEXITED and so on are available only if signal is supported by the system. [ruby-dev:48203] [Bug #9820]

**Revision 45918 - 05/11/2014 03:19 AM - nobu (Nobuyoshi Nakada)**

test\_process.rb: WIFEXITED and so on are system dependent

- test/ruby/test\_process.rb (test\_status\_kill): WIFEXITED and so on are available only if signal is supported by the system. [ruby-dev:48203] [Bug #9820]

**Revision 06042f85 - 09/01/2014 05:16 PM - nagachika (Tomoyuki Chikanaga)**

merge revision(s) r45911,r45912,r45917,r45918,r45919: [Backport #9820]

```
* signal.c (rb_f_kill): directly enqueue an ignored signal to self,  
  except for SIGSEGV and SIGBUS. [ruby-dev:48203] [Bug #9820]
```

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/branches/ruby\_2\_1@47345 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

**Revision 47345 - 09/01/2014 05:16 PM - nagachika (Tomoyuki Chikanaga)**

merge revision(s) r45911,r45912,r45917,r45918,r45919: [Backport #9820]

```
* signal.c (rb_f_kill): directly enqueue an ignored signal to self,  
  except for SIGSEGV and SIGBUS. [ruby-dev:48203] [Bug #9820]
```

**Revision 998befe8 - 10/06/2014 07:23 AM - nobu (Nobuyoshi Nakada)**

signal.c: send signal unless handled

- signal.c (rb\_f\_kill): should not ignore signal unless the default handler is registered. [ruby-dev:48592] [Bug #9820]

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@47818 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

**Revision 47818 - 10/06/2014 07:23 AM - nobu (Nobuyoshi Nakada)**

signal.c: send signal unless handled

- signal.c (rb\_f\_kill): should not ignore signal unless the default handler is registered. [ruby-dev:48592] [Bug #9820]

**Revision 47818 - 10/06/2014 07:23 AM - nobu (Nobuyoshi Nakada)**

signal.c: send signal unless handled

- signal.c (rb\_f\_kill): should not ignore signal unless the default handler is registered. [ruby-dev:48592] [Bug #9820]

**Revision 47818 - 10/06/2014 07:23 AM - nobu (Nobuyoshi Nakada)**

signal.c: send signal unless handled

- signal.c (rb\_f\_kill): should not ignore signal unless the default handler is registered. [ruby-dev:48592] [Bug #9820]

**Revision 47818 - 10/06/2014 07:23 AM - nobu (Nobuyoshi Nakada)**

signal.c: send signal unless handled

- signal.c (rb\_f\_kill): should not ignore signal unless the default handler is registered. [ruby-dev:48592] [Bug #9820]

**Revision 47818 - 10/06/2014 07:23 AM - nobu (Nobuyoshi Nakada)**

signal.c: send signal unless handled

- signal.c (rb\_f\_kill): should not ignore signal unless the default handler is registered. [ruby-dev:48592] [Bug #9820]

**Revision 47818 - 10/06/2014 07:23 AM - nobu (Nobuyoshi Nakada)**

signal.c: send signal unless handled

- signal.c (rb\_f\_kill): should not ignore signal unless the default handler is registered. [ruby-dev:48592] [Bug #9820]

**Revision 2ebafed8 - 10/07/2014 02:40 PM - nobu (Nobuyoshi Nakada)**

signal.c: get rid of deadlock by discarded signals

- signal.c (rb\_f\_kill): get rid of deadlock as unhandled and discarded signals do not make interrupt\_cond signaled. based on the patch by Kazuki Tsujimoto at [ruby-dev:48606]. [Bug #9820]

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@47832 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

**Revision 47832 - 10/07/2014 02:40 PM - nobu (Nobuyoshi Nakada)**

signal.c: get rid of deadlock by discarded signals

- signal.c (rb\_f\_kill): get rid of deadlock as unhandled and discarded signals do not make interrupt\_cond signaled. based on the patch by Kazuki Tsujimoto at [ruby-dev:48606]. [Bug #9820]

**Revision 47832 - 10/07/2014 02:40 PM - nobu (Nobuyoshi Nakada)**

signal.c: get rid of deadlock by discarded signals

- signal.c (rb\_f\_kill): get rid of deadlock as unhandled and discarded signals do not make interrupt\_cond signaled. based on the patch by Kazuki Tsujimoto at [ruby-dev:48606]. [Bug #9820]

**Revision 47832 - 10/07/2014 02:40 PM - nobu (Nobuyoshi Nakada)**

signal.c: get rid of deadlock by discarded signals

- signal.c (rb\_f\_kill): get rid of deadlock as unhandled and discarded signals do not make interrupt\_cond signaled. based on the patch by Kazuki Tsujimoto at [ruby-dev:48606]. [Bug #9820]

**Revision 47832 - 10/07/2014 02:40 PM - nobu (Nobuyoshi Nakada)**

signal.c: get rid of deadlock by discarded signals

- signal.c (rb\_f\_kill): get rid of deadlock as unhandled and discarded signals do not make interrupt\_cond signaled. based on the patch by Kazuki Tsujimoto at [ruby-dev:48606]. [Bug #9820]

**Revision 47832 - 10/07/2014 02:40 PM - nobu (Nobuyoshi Nakada)**

signal.c: get rid of deadlock by discarded signals

- signal.c (rb\_f\_kill): get rid of deadlock as unhandled and discarded signals do not make interrupt\_cond signaled. based on the patch by Kazuki Tsujimoto at [ruby-dev:48606]. [Bug #9820]

**Revision 47832 - 10/07/2014 02:40 PM - nobu (Nobuyoshi Nakada)**

signal.c: get rid of deadlock by discarded signals

- signal.c (rb\_f\_kill): get rid of deadlock as unhandled and discarded signals do not make interrupt\_cond signaled. based on the patch by Kazuki Tsujimoto at [ruby-dev:48606]. [Bug #9820]

**Revision a732f344 - 10/15/2014 03:18 PM - nagachika (Tomoyuki Chikanaga)**

merge revision(s) r47818,r47832: [Backport #9820]

```
* signal.c (rb_f_kill): should not ignore signal unless the
  default handler is registered. [ruby-dev:48592] [Bug #9820]
```

```
* signal.c (rb_f_kill): get rid of deadlock as unhandled and
  discarded signals do not make interrupt_cond signaled.
  based on the patch by Kazuki Tsujimoto at [ruby-dev:48606].
  [Bug #9820]
```

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/branches/ruby\_2\_1@47956 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

**Revision 47956 - 10/15/2014 03:18 PM - nagachika (Tomoyuki Chikanaga)**

merge revision(s) r47818,r47832: [Backport #9820]

```
* signal.c (rb_f_kill): should not ignore signal unless the
  default handler is registered. [ruby-dev:48592] [Bug #9820]
```

```
* signal.c (rb_f_kill): get rid of deadlock as unhandled and
  discarded signals do not make interrupt_cond signaled.
  based on the patch by Kazuki Tsujimoto at [ruby-dev:48606].
  [Bug #9820]
```

**History**

---

**#1 - 05/09/2014 01:06 PM - akr (Akira Tanaka)**

```
##### cron ##### SIGINT  signal handler
SIG_IGN #####
```

```
#####
##### hang #####
```

```
./miniruby -e 'trap(:INT, "IGNORE"); Process.kill(:INT, $$)'
```

```
Ctrl-C #####SIGQUIT #####
```

```
./miniruby -e 'trap(:QUIT, "IGNORE"); Process.kill(:QUIT, $$)'
```

```
#####hang #####
workaround trap(:INT, "DEFAULT") #####
```

```
#####Ruby 2.1.0 #####
Ruby 2.0.0p481  hang #####
  rubyci.org  2.0.0  hang #####
2.0.0  hang  TestBeginEndBlock#test_propagate_signaled  TestIO_Console#test_noctty  #####
```

**#2 - 05/09/2014 02:16 PM - nobu (Nobuyoshi Nakada)**

```
r39819#####SIG_IGN#####
#####kill#####
```

**#3 - 05/09/2014 02:35 PM - nobu (Nobuyoshi Nakada)**

```
#####
```

```
diff --git i/thread.c w/thread.c
index dfa91a8..209aff1 100644
--- i/thread.c
+++ w/thread.c
@@ -5180,10 +5180,11 @@ ruby_kill(rb_pid_t pid, int sig)
     * delivered immediately and synchronously.
```



Ruby signal handler  
ruby\_kill kill

```
diff --git a/signal.c b/signal.c
index 171f5ac..7a83dbc 100644
--- a/signal.c
+++ b/signal.c
@@ -487,7 +487,11 @@ rb_f_kill(int argc, const VALUE *argv)
    default:
        t = signal_ignored(sig);
        if (t) {
-           if (t < 0) ruby_kill(pid, sig);
+           if (t < 0) {
+               if (kill(pid, sig) < 0) {
+                   rb_sys_fail(0);
+               }
+           }
            break;
        }
        signal_enqueue(sig);
```

**#14 - 10/15/2014 03:19 PM - nagachika (Tomoyuki Chikanaga)**

- Backport changed from 2.0.0: DONTNEED, 2.1: REQUIRED to 2.0.0: DONTNEED, 2.1: DONE

r47818 and r47832 were backported into ruby\_2\_1 branch at r47956.

**Files**

---

strace.cron.log	15.8 KB	05/09/2014	akr (Akira Tanaka)
strace.term.log	17.6 KB	05/09/2014	akr (Akira Tanaka)