

Ruby - Bug #1287

\$? not set after Open3::popen3

03/14/2009 05:46 PM - phasis68 (Heesob Park)

Status: Rejected Priority: Normal Assignee: Target version: 1.9.1 ruby -v: ruby 1.9.1p0 (2009-01-30 revision 21907) [i686-linux]	Backport:
Description =begin \$? is not set after popen3 in 1.9.1 Try following code: require 'open3' status = Open3::popen3("echo hello") do stdin,stdout,stderr stdout.each { line puts "stdout:"+line } end p \$? [sidns@ns ~]\$ ruby -v po.rb ruby 1.8.6 (2007-09-24 patchlevel 111) [i686-linux] stdout:hello #<Process::Status: pid=20140,exited(0)> [siweb@localhost ~]\$ ruby -v po.rb ruby 1.9.1p0 (2009-01-30 revision 21907) [i686-linux] stdout:hello nil =end	

History

#1 - 03/14/2009 06:54 PM - akr (Akira Tanaka)

- Status changed from Open to Rejected

=begin

ruby 1.8 don't have a way to know the exit status because double fork is used.

\$? doesn't reflect the actual status.

```
% ruby-1.8.6p287 -ropen3 -ve '
status = Open3::popen3("echo hello;exit 1") do |stdin,stdout,stderr|
  stdout.each { |line| puts "stdout:"+line }
end
p $?'
ruby 1.8.6 (2008-08-11 patchlevel 287) [i686-linux]
stdout:hello
#<Process::Status: pid=31667,exited(0)>
```

open3 in ruby 1.9 has the fourth argument, wait thread, to the block.

The thread wait the child process and returns the exit status as the value of the thread.

```
% ruby-1.9.1p0 -ropen3 -ve '
status = Open3::popen3("echo hello;exit 1") do |stdin,stdout,stderr,waitth|
  stdout.each { |line| puts "stdout:"+line }
p waitth.value
end
'
ruby 1.9.1p0 (2009-01-30 revision 21907) [i686-linux]
stdout:hello
#<Process::Status: pid 31847 exit 1>
```

=end

#2 - 05/19/2021 06:10 PM - schneems (Richard Schneeman)

FWIW the work around is to capture the status and re-exec with an exit. It works, but the downside is that shell invocations are not cheap:

```
irb(main):042:0> a, b, c = Open3.capture3("ls /does/not/exist")
irb(main):043:0> `exit #{c.exitstatus}`
=> ""
irb(main):044:0> $?
=> #<Process::Status: pid 78179 exit 1>
```