

Ruby trunk - Bug #14765

Arguments passed to Open3.popen3() are not interpreted as wildcards

05/16/2018 01:08 PM - Rushyanth (Rushyanth reddy)

Status: Rejected	
Priority: Normal	
Assignee:	
Target version:	
ruby -v: 2.3.1	Backport: 2.3: UNKNOWN, 2.4: UNKNOWN, 2.5: UNKNOWN
Description In console when we write <pre>stdin, stdout, stderr, wait_thr = Open3.popen4("cat *.*.yml") stdout.readlines</pre> All the files with names starting with . and ending in .yml are shown But when we do <pre>stdin, stdout, stderr, wait_thr = Open3.popen4("cat", "*.*.yml") stdout.readlines</pre> It returns an empty array stderr.readlines says ["cat: *.*.yml: No such file or directory\n"] Because it is interpreting the string arguments as it is and looking for a file with name *.*.yml and not interpreting as in the first case.	

History

#1 - 05/16/2018 01:11 PM - Rushyanth (Rushyanth reddy)

In console when we write

```
stdin, stdout, stderr, wait_thr = Open3.popen3("cat *.*.yml")
```

All the files with names starting with . and ending in .yml are shown

But when we do

```
stdin, stdout, stderr, wait_thr = Open3.popen3("cat", "*.*.yml")
```

It returns an empty array

stderr.readlines says

```
["cat: *.*.yml: No such file or directory\n"]
```

Because it is interpreting the string arguments as it is and looking for a file with name *.*.yml and not interpreting as in the first case.

#2 - 05/16/2018 01:13 PM - Rushyanth (Rushyanth reddy)

- Backport set to 2.3: UNKNOWN, 2.4: UNKNOWN, 2.5: UNKNOWN

- Tracker changed from Feature to Bug

#3 - 05/16/2018 06:41 PM - Rushyanth (Rushyanth reddy)

- ruby -v set to 2.3.1

#4 - 05/17/2018 12:21 AM - nobu (Nobuyoshi Nakada)

- Description updated

- Subject changed from Arguments passed to Open3.popen3() are not interpreted as regular expressions to Arguments passed to Open3.popen3() are not interpreted as wildcards

What's Open3.popen4?

The bundled library open3.rb does not provide such method.

Maybe a third party's method?

#5 - 05/17/2018 06:44 AM - Rushyanth (Rushyanth reddy)

Sorry my bad. That was typing error while raising the issue. Its actually `Open3.popen3()`. Its mentioned correctly in the subject

#6 - 05/17/2018 07:39 AM - akr (Akira Tanaka)

- Status changed from Open to Rejected

wildcard expansion is done at shell.

If you need shell, specify a command line as one string.

#7 - 05/17/2018 09:21 AM - nobu (Nobuyoshi Nakada)

Besides, the array form is to get rid of the expansion by the shell, which might cause security issues.

#8 - 05/17/2018 09:27 AM - Rushyanth (Rushyanth reddy)

Suppose I have a variable

`x = "abc"`

I need to cat a file `abc.yml` I write

`Open3.popen3("cat #{x}.yml")` ----- (1)

This will lead to Command Injection warning.

So we need to pass the arguments seperately like

`Open3.popen3("cat", x+".yml")` ----- (2)

The second type will not lead to command Injection warning and both cases 1 and 2 work fine

But If I need to retrieve data of many files not just a single file.

Then I need to write

`Open3.popen3("cat *.yml")` ----- (1)

or

`Open3.popen3("cat", "**.yml")` ----- (2)

The 1st case works fine but the second case fails .

Because in second case `popen3` is interpreting its second argument `*.yml` as a file name .

But `*` should actually be interpreted as regular expression character where it retrieves all the file names ending in `.yml`

#9 - 05/17/2018 09:43 AM - Hanmac (Hans Mackowiak)

`popen3` uses `Process.spawn` and this does show this in the docs:

`commandline` : command line string which is passed to the standard shell

`cmdname, arg1, ...` : command name and one or more arguments (This form does not use the shell. See below for caveats.)

#10 - 05/17/2018 11:45 AM - nobu (Nobuyoshi Nakada)

`Open3.popen3("cat", *Dir.glob("*.yml"))`