Ruby - Bug #4841

WEBrick threading leads to infinite loop

06/06/2011 11:24 PM - peakxu (Peak Xu)

Status:	Closed		
Priority:	Normal		
Assignee:	normalperson (Eric Wong)		
Target version:			
ruby -v:	run_this.rb	**Backport:**	2.3: UNKNOWN, 2.4: UNKNOWN

Description

In a multi-threaded environment, the callback in WEBrick web server does not appear to work properly. If shutdown is called right after start, the server can entire an infinite loop and need to be killed using Ctrl-C.

Reproducible on 1.9.2 p180 on Windows and 1.8.7 on Cygwin.

ruby -v run_this.rb

ruby 1.8.7 (2008-08-11 patchlevel 72) [i386-cygwin]
[2011-06-06 10:12:58] INFO WEBrick 1.3.1
[2011-06-06 10:12:58] INFO ruby 1.8.7 (2008-08-11) [i386-cygwin]
[2011-06-06 10:12:58] INFO WEBrick::HTTPServer#start: pid=4860 port=19000
run_this.rb:95:in join': Interrupt from run_this.rb:95:in stop'
from run_this.rb:102

Note: killed manually to break out of infinite loop.

Recommended fix:
In webrick/server.rb around line 85-88, there's a callback as well as initialization of status to Running. If we move the callback below the initialization, this fixes the issue on 1.8.7 Cygwin but not in 1.9.2 Windows. Code should look like below:

@logger.info
"#{self.class}#start: pid=#{$$} port=#{@config[:Port]}"

```
    thgroup = ThreadGroup.new
    @status = :Running
    call_callback(:StartCallback)
    while @status == :Running
    <snip>
```

ruby -v output now looks like:

ruby 1.8.7 (2008-08-11 patchlevel 72) [i386-cygwin]
[2011-06-06 10:05:45] INFO WEBrick 1.3.1
[2011-06-06 10:05:45] INFO ruby 1.8.7 (2008-08-11) [i386-cygwin]
[2011-06-06 10:05:45] INFO WEBrick::HTTPServer#start: pid=5228 port=19000
[2011-06-06 10:05:45] INFO going to shutdown ...
[2011-06-06 10:05:45] INFO WEBrick::HTTPServer#start done.

Associated revisions

Revision 22474d8f9636e1555d61cd4bb3d93698dc1ab2af - 12/08/2017 06:51 PM - Eric Wong

webrick: allow shutdown after StartCallback

We must to ensure the @status ivar is set to :Running before
running StartCallback, otherwise Webrick::Server#stop will not
change the @status to :Shutdown properly.

Note: I have not been able to reproduce the original issue but
understood at least part of the problem and fixed it with this
commit. However, the original reporter (Peak Xu) was still able
to reproduce the problem on 1.9.2 p180 on Windows, so I'm not
sure what else might be going on. Ruby threading and
synchronization primitives have changed a lot since 1.9.2, so

maybe that was fixed elsewhere.

- lib/webrick/server.rb: call StartCallback sooner [Bug #4841]

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@61083 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision 22474d8f - 12/08/2017 06:51 PM - Eric Wong

webrick: allow shutdown after StartCallback

We must to ensure the @status ivar is set to :Running before
running StartCallback, otherwise Webrick::Server#stop will not
change the @status to :Shutdown properly.

Note: I have not been able to reproduce the original issue but
understood at least part of the problem and fixed it with this
commit. However, the original reporter (Peak Xu) was still able
to reproduce the problem on 1.9.2 p180 on Windows, so I'm not
sure what else might be going on. Ruby threading and
synchronization primitives have changed a lot since 1.9.2, so
maybe that was fixed elsewhere.

- lib/webrick/server.rb: call StartCallback sooner [Bug #4841]

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@61083 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

History

#1 - 06/11/2011 04:48 PM - ko1 (Koichi Sasada)

- Assignee set to gotoyuzo (GOTOU Yuuzou)

#2 - 06/11/2011 04:48 PM - ko1 (Koichi Sasada)

- Status changed from Open to Assigned

#3 - 06/23/2011 04:58 PM - nahi (Hiroshi Nakamura)

I guess that WEBrick::Server#{start,stop,shutdown} is not designed to be thread-safe and there's no simple fix for your usage. When I used
WEBrick::Server for such purpose, I created my own lock outside of WEBrick::Server instances.

I'm going to ask gotoyuzo (the author) but please don't expect a response soon.

Do you think I understand the issue correctly?

#4 - 06/23/2011 04:58 PM - nahi (Hiroshi Nakamura)

- Assignee changed from gotoyuzo (GOTOU Yuuzou) to nahi (Hiroshi Nakamura)

#5 - 12/08/2017 12:38 PM - mame (Yusuke Endoh)

- Assignee changed from nahi (Hiroshi Nakamura) to normalperson (Eric Wong)

normal, could you please check this ticket?

#6 - 12/08/2017 06:52 PM - Anonymous

- Status changed from Assigned to Closed

Applied in changeset trunk|r61083.

---

webrick: allow shutdown after StartCallback

We must to ensure the @status ivar is set to :Running before
running StartCallback, otherwise Webrick::Server#stop will not
change the @status to :Shutdown properly.

Note: I have not been able to reproduce the original issue but
understood at least part of the problem and fixed it with this
commit. However, the original reporter (Peak Xu) was still able
to reproduce the problem on 1.9.2 p180 on Windows, so I'm not
sure what else might be going on. Ruby threading and
synchronization primitives have changed a lot since 1.9.2, so
maybe that was fixed elsewhere.

- lib/webrick/server.rb: call StartCallback sooner [Bug #4841]

#7 - 12/08/2017 07:21 PM - normalperson (Eric Wong)

mame@ruby-lang.org wrote:

> normal, could you please check this ticket?
>
> ───
>
> Bug #4841: WEBrick threading leads to infinite loop
> https://bugs.ruby-lang.org/issues/4841#change-68231

Sure, I think the problem is fixed in r61083.

Looking at the original run-this.rb script, it seems racy
in that the Mutex does not synchronize enough scope for
the ConditionVariable.

#8 - 12/08/2017 09:54 PM - mame (Yusuke Endoh)

Thank you for your quick response!

Files

run_this.rb		2.53 KB	06/06/2011		peakxu (Peak Xu)