

ChiliProject - Bug # 896: Enabling "Authentication required" mode returns 404s

Status:	Closed	Priority:	Normal
Author:	Yehuda Katz	Category:	Permissions
Created:	2012-02-16	Assignee:	Felix Schäfer
Updated:	2012-03-06	Due date:	
Remote issue URL:			
Affected version:	3.0.0		
Description:	When I enable "Authentication required" mode (in Settings->Authentication), users who are not logged in get a 404 error. Once I manually put in /login or /anythingtogeta404 , then the login page will load. The production.log says >Processing WelcomeController#index (for 206.196.165.121 at 2012-02-16 12:06:14) [GET] > Parameters: {"action"=>"index", "controller"=>"welcome"} >Redirected to http://domain/login?back_url=http%3A%2F%2Fdomain%2F >Filter chain halted as [:check_if_login_required] rendered_or_redirected. >Completed in 9ms (DB: 4) 302 Found [http://domain/] Reproduce: 0. Enable "Authentication Required" feature. 1. Open Chrome Incognito window (so there is not previous session info) 2. Go to http://your_installation/ -> get browser 404 3. Go to http://your_installation/issues/1 (assuming you have an issue 1) -> get blank page 4. Go to http://your_installation/abc -> get formatted 404 from ChiliProject 5. Go back to either 2 or 3 and see that the login form loads.		

History

2012-02-16 08:04 pm - Pavel Nakonechny

+1

You can also reproduce this bug by deleting *_chiliproject_session* cookie from your browser.

2012-03-01 01:17 am - Felix Schäfer

- Target version set to 3.1.0

- Assignee set to Felix Schäfer

Intriguing, I'll try to reproduce this locally.

2012-03-05 12:10 pm - Holger Just

I can not reproduce the bug using Thin or Webrick on Ruby 1.8.7p358. I always get a HTTP 302 which forwards me to @/login?back_url=...@, which is consistent to what your @production.log@ states.

Could you please give a little more details about your setup (Ruby version, App server) and verify that your ChiliProject/App server actually sends a 404, or if it might be just your browser mis-interpreting something.

If you still get a 404, could you please verify, if it comes from the original resource (e.g. @http://domain.com/@) or from the forwarded resource (e.g. @http://domain.com/login?back_url=...@)

2012-03-06 04:31 am - Yehuda Katz

ruby 1.8.7 (2011-02-18 patchlevel 334) [i686-linux]

Rails 2.3.14

mongrel (1.1.5)

(It is set up through cPanel, but all cPanel does is execute '/usr/bin/ruby /usr/bin/mongrel_rails start -p 12001 -d -e production -P log/mongrel.pid')

I am connecting directly to the mongrel instance, not through my Apache (HTTPD 2.2) proxy, to make sure the problem is not with Apache.

I am using Google Chrome 17, but I have replicated the issue in other browsers.

The browser is not getting the 302 at all even though the log says it is being sent.

I can give you access to my server if it will help.

2012-03-06 11:34 am - Holger Just

Okay, I can reproduce it with Mongrel 1.1.5. On access of @http://localhost:12001@ I always get the following stack trace:

```
<pre>
Tue Mar 06 11:56:27 +0100 2012: Error calling Dispatcher.dispatch #<NoMethodError: You have a nil object when you didn't expect it!
You might have expected an instance of Array.
The error occurred while evaluating nil.split>
/Users/hjust/.rvm/gems/ree-1.8.7-2011.03@chili/gems/actionpack-2.3.14/lib/action_controller/cgi_process.rb:52:in `dispatch CGI'
/Users/hjust/.rvm/gems/ree-1.8.7-2011.03@chili/gems/actionpack-2.3.14/lib/action_controller/dispatcher.rb:101:in `dispatch CGI'
/Users/hjust/.rvm/gems/ree-1.8.7-2011.03@chili/gems/actionpack-2.3.14/lib/action_controller/dispatcher.rb:27:in `dispatch'
/Users/hjust/.rvm/gems/ree-1.8.7-2011.03@chili/gems/mongrel-1.1.5/bin/./lib/mongrel/rails.rb:76:in `process'
/Users/hjust/.rvm/gems/ree-1.8.7-2011.03@chili/gems/mongrel-1.1.5/bin/./lib/mongrel/rails.rb:74:in `synchronize'
/Users/hjust/.rvm/gems/ree-1.8.7-2011.03@chili/gems/mongrel-1.1.5/bin/./lib/mongrel/rails.rb:74:in `process'
/Users/hjust/.rvm/gems/ree-1.8.7-2011.03@chili/gems/mongrel-1.1.5/bin/./lib/mongrel.rb:159:in `process_client'
/Users/hjust/.rvm/gems/ree-1.8.7-2011.03@chili/gems/mongrel-1.1.5/bin/./lib/mongrel.rb:158:in `each'
/Users/hjust/.rvm/gems/ree-1.8.7-2011.03@chili/gems/mongrel-1.1.5/bin/./lib/mongrel.rb:158:in `process_client'
/Users/hjust/.rvm/gems/ree-1.8.7-2011.03@chili/gems/mongrel-1.1.5/bin/./lib/mongrel.rb:285:in `run'
/Users/hjust/.rvm/gems/ree-1.8.7-2011.03@chili/gems/mongrel-1.1.5/bin/./lib/mongrel.rb:285:in `initialize'
/Users/hjust/.rvm/gems/ree-1.8.7-2011.03@chili/gems/mongrel-1.1.5/bin/./lib/mongrel.rb:285:in `new'
/Users/hjust/.rvm/gems/ree-1.8.7-2011.03@chili/gems/mongrel-1.1.5/bin/./lib/mongrel.rb:285:in `run'
/Users/hjust/.rvm/gems/ree-1.8.7-2011.03@chili/gems/mongrel-1.1.5/bin/./lib/mongrel.rb:268:in `initialize'
/Users/hjust/.rvm/gems/ree-1.8.7-2011.03@chili/gems/mongrel-1.1.5/bin/./lib/mongrel.rb:268:in `new'
/Users/hjust/.rvm/gems/ree-1.8.7-2011.03@chili/gems/mongrel-1.1.5/bin/./lib/mongrel.rb:268:in `run'
/Users/hjust/.rvm/gems/ree-1.8.7-2011.03@chili/gems/mongrel-1.1.5/bin/./lib/mongrel/configurator.rb:282:in `run'
/Users/hjust/.rvm/gems/ree-1.8.7-2011.03@chili/gems/mongrel-1.1.5/bin/./lib/mongrel/configurator.rb:281:in `each'
/Users/hjust/.rvm/gems/ree-1.8.7-2011.03@chili/gems/mongrel-1.1.5/bin/./lib/mongrel/configurator.rb:281:in `run'
/Users/hjust/.rvm/gems/ree-1.8.7-2011.03@chili/gems/mongrel-1.1.5/bin/mongrel_rails:128:in `run'
/Users/hjust/.rvm/gems/ree-1.8.7-2011.03@chili/gems/mongrel-1.1.5/bin/./lib/mongrel/command.rb:212:in `run'
/Users/hjust/.rvm/gems/ree-1.8.7-2011.03@chili/gems/mongrel-1.1.5/bin/mongrel_rails:281
/Users/hjust/.rvm/gems/ree-1.8.7-2011.03@chili/bin/mongrel_rails:19:in `load'
/Users/hjust/.rvm/gems/ree-1.8.7-2011.03@chili/bin/mongrel_rails:19
</pre>
```

It seems to stem from a bug in Rack 1.1 which was fixed there with

"f6f3c60938":<https://github.com/rack/rack/commit/f6f3c60938ea3b08f3292a2480e6753c293584e5> which is included in Rack 1.2 (which is incompatible with Rails 2.3 sadly).

During my tests I observed various similar errors which seem to stem from Mongrel using Rails' CGI interface in a way which doesn't fully work anymore. All the errors I observed doesn't stem from ChiiiProject but from either Mongrel or Rails CGI interface. So as it stands now, I'd recommend one of two options:

* Don't use Mongrel but something like Thin or Unicorn. This would be the recommended option as Mongrel 1.x doesn't seem to be maintained anymore at all.

* Use mongrel 1.2.0-pre (@gem install mongrel --pre@). That seemed to work during my tests if I apply the above mentioned rack patch, but I have no idea if it could break in other places.

2012-03-06 11:46 am - Holger Just

Another option (which most people do) is to use Passenger.

Given the age of Mongrel and the large amount of work-arounds which are required nowadays to keep it working, I see only slim chances of us fully supporting it.

We accept patches to make it work, but given that all the problems I observed stem from the (Rack, Rails, Mongrel) triumvirate, it's probably hard to get right and we would be moving way outside of our realm. Also things probably are going to get much worse once we moved to Rails 3.

2012-03-06 02:31 pm - Yehuda Katz

- *Status changed from Open to Closed*

cPanel will start supporting Passenger in the next version.

In the mean time, I will keep "Authentication required" off. Everything seems to be working besides that.

2012-03-06 05:02 pm - Yehuda Katz

As a note, another problem I have been having is caused by the same issue.

/sys/fetch_changesets appears to the browser to return a 404 no matter what, but the production.log shows that the correct response is being returned.