

## Ruby - Bug #8182

### XMLRPC request fails with "Wrong size. Was 31564, should be 1501"

03/29/2013 06:08 AM - tsagadar (Marcel Mueller)

<b>Status:</b>	Closed	
<b>Priority:</b>	Normal	
<b>Assignee:</b>	kou (Kouhei Sutou)	
<b>Target version:</b>	2.2.0	
<b>ruby -v:</b>	ruby 2.0.0p0 (2013-02-24 revision 39474) [x86_64-darwin12.2.1]	<b>Backport:</b> 1.9.3: DONTNEED, 2.0.0: DONE, 2.1: DONE
<b>Description</b> <p>Since upgrading to Ruby 2.0.0p0 we can no longer use MailChimp through the Hominid gem. Hominid relies on the Ruby XMLRPC client to access MailChimp.</p> <p>Debugging the issues revealed the following problem: net/http/response.rb transparently deflates the response body, removes the "content-encoding" response header (response.rb:255), but does not adjust the "content-length" header accordingly. This makes xmlrpc/client.rb:506 raise the error, that the response body and the declared length in "content-length" does not match.</p> <p>I propose a high priority for this issue for two reason:</p> <ul style="list-style-type: none"><li>• The problem should occur whenever Ruby XMLRPC is used to access a service that supports content encoding with "deflate", "gzip", or "x-gzip"</li><li>• I don't see a workaround to this problem that could be used</li></ul>		

#### Associated revisions

**Revision fc3b4d06ca4f5c46457983b51f844f5afcd325df - 04/06/2014 10:26 PM - naruse (Yui NARUSE)**

- lib/xmlrpc/client.rb (do\_rpc): don't check body length.  
If HTTP content-encoding is used, the length may be different.  
[Bug #8182] [ruby-core:53811]

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@45529 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

**Revision fc3b4d06 - 04/06/2014 10:26 PM - naruse (Yui NARUSE)**

- lib/xmlrpc/client.rb (do\_rpc): don't check body length.  
If HTTP content-encoding is used, the length may be different.  
[Bug #8182] [ruby-core:53811]

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@45529 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

**Revision 4f3f77b9f23ef288640d8faa5edb870a55b28937 - 05/27/2014 02:46 AM - U.Nakamura**

merge revision(s) 45529: [Backport #8182]

```
* lib/xmlrpc/client.rb (do_rpc): don't check body length.  
If HTTP content-encoding is used, the length may be different.  
[Bug #8182] [ruby-core:53811]
```

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/branches/ruby\_2\_0\_0@46153 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

**Revision 4f3f77b9 - 05/27/2014 02:46 AM - U.Nakamura**

merge revision(s) 45529: [Backport #8182]

```
* lib/xmlrpc/client.rb (do_rpc): don't check body length.  
If HTTP content-encoding is used, the length may be different.  
[Bug #8182] [ruby-core:53811]
```

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/branches/ruby\_2\_0\_0@46153 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

**Revision 909291085c87b41d6718c2630ee49210eb9d9684 - 05/27/2014 03:26 PM - nagachika (Tomoyuki Chikanaga)**

merge revision(s) r45529: [Backport #8182]

```
* lib/xmlrpc/client.rb (do_rpc): don't check body length.
  If HTTP content-encoding is used, the length may be different.
  [Bug #8182] [ruby-core:53811]
```

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/branches/ruby\_2\_1@46186 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

## Revision 90929108 - 05/27/2014 03:26 PM - nagachika (Tomoyuki Chikanaga)

merge revision(s) r45529: [Backport #8182]

```
* lib/xmlrpc/client.rb (do_rpc): don't check body length.
  If HTTP content-encoding is used, the length may be different.
  [Bug #8182] [ruby-core:53811]
```

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/branches/ruby\_2\_1@46186 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

## History

---

### #1 - 03/29/2013 06:58 AM - hsbt (Hiroshi SHIBATA)

- Category set to core
- Assignee set to drbrain (Eric Hodel)
- Priority changed from Normal to 5
- Target version set to 2.1.0

### #2 - 04/01/2013 05:34 PM - tsagadar (Marcel Mueller)

Sample code to reproduce the issue using Flickr. With 1.9.3 this gives "Invalid API key" which is fine, but 2.0.0 returns the error of the size mismatch.

```
require 'xmlrpc/client'
```

```
server = XMLRPC::Client.new('api.flickr.com', '/services/xmlrpc/', 80)
s = server.call('flickr.test.echo')
```

### #3 - 05/16/2013 06:30 PM - duncan (Duncan Mac-Vicar P.)

I sent a pull request about this bug: <https://github.com/ruby/ruby/pull/308>

### #4 - 05/20/2013 07:37 AM - zzak (zzak \_)

- File 308.patch added
- Status changed from Open to Assigned

I've attached Duncan's patch from github, so we can close the pull request.

### #5 - 06/04/2013 11:25 AM - zzak (zzak \_)

- Assignee changed from drbrain (Eric Hodel) to kou (Kouhei Sutou)

### #6 - 06/10/2013 10:09 PM - kou (Kouhei Sutou)

I confirmed the problem. I consider how to resolve this problem. Please wait.

I don't like that response header is changed in Inflator. It seems that header should be changed by response itself.

Here is a workaround:

```
require 'xmlrpc/client'
```

```
server = XMLRPC::Client.new('api.flickr.com', '/services/xmlrpc/', 80)
server.http_header_extra = {"accept-encoding" => "identity"} # <= add this line (workaround)
s = server.call('flickr.test.echo')
```

### #7 - 06/12/2013 08:12 AM - drbrain (Eric Hodel)

=begin  
Consider this example:

```
http.request req do |res|
  expected = res['Content-Length'] # size of encoded content
  actual = 0
```

```
  res.read_body { |chunk| actual += chunk.bytesize }
```

```
raise 'too short' unless expected == actual
```

```
end
```

Here the user stores the Content-Length before Net::HTTP can read the decoded body, so altering the Content-Length may not help fix this class of bug generally (but will for XMLRPC).

An alternate is to only check Content-Length when Content-Encoding is set, but Net::HTTP deletes the Content-Encoding header when it begins to decode the body. (So the user would need to retrieve the Content-Encoding before reading the body).

However this is fixed we need to document the recommended way to perform this type of integrity check.  
=end

#### **#8 - 06/12/2013 08:36 AM - drbrain (Eric Hodel)**

- File *net.http.bug8182.patch* added

This patch updates the 308 patch to update the Content-Length in Response instead of Inflater and fixes the variable name ("deflated length" is the original value, but only when deflate-encoded). It also adds documentation describing when the Content-Length will be changed so the user can test properly.

We will need Naruse's OK to commit it, he rejected changing Content-Length in [#6492: https://bugs.ruby-lang.org/issues/6492#note-10](https://bugs.ruby-lang.org/issues/6492#note-10)

#### **#9 - 06/12/2013 08:46 AM - drbrain (Eric Hodel)**

- File *xmlrpc\_client.rb.bug8182.patch* added

Here is a possible patch that only checks Content-Length if the Content-Encoding was handled by Net::HTTP. Unfortunately I don't have an XMLRPC service available to test it against.

#### **#10 - 06/13/2013 12:15 AM - kou (Kouhei Sutou)**

drbrain (Eric Hodel) wrote:

Consider this example:

```
http.request req do |res|
  expected = res['Content-Length'] # size of encoded content
  actual = 0

  res.read_body { |chunk| actual += chunk.bytesize }

  raise 'too short' unless expected == actual
end
```

Here the user stores the Content-Length before Net::HTTP can read the decoded body, so altering the Content-Length may not help fix this class of bug generally (but will for XMLRPC).

I think that we don't care about the case. In the case, the user must handle Content-Encoding by himself. The user should disable auto-decode feature.

I think that we care about the following case (this is XMLRPC::Client case):

```
http.request req do |res|
  actual = 0

  res.read_body { |chunk| actual += chunk.bytesize }

  expected = res['Content-Length'] # size of encoded content
  raise 'too short' unless expected == actual
end
```

In this case, we already deleted Content-Encoding when the user retrieves Content-Length. So the user should be able to handle the body as decoded content. In this point of view, Content-Length should be the length of decoded content. duncan's patch is this point of view.

I like duncan's approach but don't like the details as I mentioned...

#### **#11 - 06/13/2013 12:24 AM - kou (Kouhei Sutou)**

drbrain (Eric Hodel) wrote:

This patch updates the 308 patch to update the Content-Length in Response instead of Inflater and fixes the variable name ("deflated length" is the original value, but only when deflate-encoded). It also adds documentation describing when the Content-Length will be changed so the user can test properly.

I think that we should delete Content-Length between deleting Content-Encoding and yielding the first chunk when both Content-Encoding and Content-Length are specified. I think the following HERE is a good location

```
if clen
  # HERE
  @socket.read clen, dest, true # ignore EOF
  return
end
```

but we don't have information whether Content-Encoding is handled at the location... I don't like the following code because the code ignores abstraction by inflater.

```
if clen
  self.delete "content-length" is @socket.is_a?(Inflater)
  @socket.read clen, dest, true # ignore EOF
  return
end
```

Umm...

#### #12 - 06/13/2013 12:29 AM - kou (Kouhei Sutou)

drbrain (Eric Hodel) wrote:

Here is a possible patch that only checks Content-Length if the Content-Encoding was handled by Net::HTTP. Unfortunately I don't have an XMLRPC service available to test it against.

I'm sorry but I don't like it. It is a bit tricky.

I think that Net::HTTP should resolve this problem rather than XMLRPC has a workaround for this problem.

#### #13 - 06/13/2013 10:49 PM - kou (Kouhei Sutou)

- Assignee changed from kou (Kouhei Sutou) to naruse (Yui NARUSE)

I think that this problem should be fixed in net/http. So I change the assigned to naruse.

#### #14 - 06/14/2013 07:03 PM - naruse (Yui NARUSE)

Maybe I misunderstood, why is the assertion is needed?

I feel it should be simply removed.

Moreover chunked encoded response doesn't have Content-Length header.  
So it sounds nonsense.

#### #15 - 06/14/2013 08:23 PM - duerst (Martin Dürst)

Checking Content-Length can help detect cases where the connection is closed prematurely. I'm not sure how important this is in the application where the error was discovered, but it's well known as a problem in HTTP in general that premature closed connections are difficult to detect.

Regards, Martin.

On 2013/06/14 19:03, naruse (Yui NARUSE) wrote:

Issue [#8182](#) has been updated by naruse (Yui NARUSE).

Maybe I misunderstood, why is the assertion is needed?  
I feel it should be simply removed.

**Moreover chunked encoded response doesn't have Content-Length header.  
So it sounds nonsense.**

Bug [#8182](#): XMLRPC request fails with "Wrong size. Was 31564, should be 1501"  
<https://bugs.ruby-lang.org/issues/8182#change-39928>

Author: tsagadar (Marcel Mueller)

Status: Assigned  
Priority: High  
Assignee: naruse (Yui NARUSE)  
Category: core  
Target version: current: 2.1.0  
ruby -v: ruby 2.0.0p0 (2013-02-24 revision 39474) [x86\_64-darwin12.2.1]  
Backport:

Since upgrading to Ruby 2.0.0p0 we can no longer use MailChimp through the Hominid gem. Hominid relies on the Ruby XMLRPC client to access MailChimp.

Debugging the issues revealed the following problem: net/http/response.rb transparently deflates the response body, removes the "content-encoding" response header (response.rb:255), but does not adjust the "content-length" header accordingly. This makes xmlrpc/client.rb:506 raise the error, that the response body and the declared length in "content-length" does not match.

I propose a high priority for this issue for two reason:

- The problem should occur whenever Ruby XMLRPC is used to access a service that supports content encoding with "deflate", "gzip", or "x-gzip"
- I don't see a workaround to this problem that could be used

#### #16 - 06/16/2013 12:48 AM - naruse (Yui NARUSE)

duerst (Martin Dürst) wrote:

Checking Content-Length can help detect cases where the connection is closed prematurely. I'm not sure how important this is in the application where the error was discovered, but it's well known as a problem in HTTP in general that premature closed connections are difficult to detect.

By default Net::HTTP#read wait to read content of the size which is specified with CONTENT\_LENGTH, or chunked encoding. Therefore such situation does not exist.

#### #17 - 07/12/2013 07:31 PM - Eregon (Benoit Daloze)

naruse (Yui NARUSE) wrote:

Maybe I misunderstood, why is the assertion is needed?  
I feel it should be simply removed.

Moreover chunked encoded response doesn't have Content-Length header.  
So it sounds nonsense.

I agree, it does not seem XMLRPC's job to check Content-Length.

Could we please solve this issue promptly?  
Such a bad bug should not be left opened for long,  
it encourages users to use workarounds like above which is really not a proper solution.

#### #18 - 08/07/2013 04:12 PM - naruse (Yui NARUSE)

- Category changed from core to lib
- Assignee changed from naruse (Yui NARUSE) to kou (Kouhei Sutou)
- Priority changed from 5 to Normal
- Backport set to 1.9.3: DONTNEED, 2.0.0: REQUIRED

The check should be removed as following:

```
diff --git a/lib/xmlrpc/client.rb b/lib/xmlrpc/client.rb
index ced3d01..6aa7c1f 100644
--- a/lib/xmlrpc/client.rb
+++ b/lib/xmlrpc/client.rb
@@ -509,8 +509,6 @@ module XMLRPC # :nodoc:
   expected = resp["Content-Length"] || "<unknown>"
   if data.nil? or data.bytesize == 0
     raise "Wrong size. Was #{data.bytesize}, should be #{expected}"
-  elsif expected != "<unknown>" and expected.to_i != data.bytesize and resp["Transfer-Encoding"]
-    raise "Wrong size. Was #{data.bytesize}, should be #{expected}"
```

```
end
```

```
parse_set_cookies(resp.get_fields("Set-Cookie"))
```

#### #19 - 10/03/2013 12:54 AM - bbuchalter (Brian Buchalter)

Just a reminder, this fix is not yet in 2.1.0 rc1.

[https://github.com/ruby/ruby/blob/v2\\_1\\_0\\_preview1/lib/xmlrpc/client.rb?source=c#L509](https://github.com/ruby/ruby/blob/v2_1_0_preview1/lib/xmlrpc/client.rb?source=c#L509)

Would love to see it in there!

#### #20 - 10/18/2013 04:51 AM - joevandyk (Joe Van Dyk)

I'm running into this same problem on Ruby 2.0 p247.

```
RuntimeError: Wrong size. Was 157, should be 134
/usr/local/stow/ruby-2.0.0-p247/lib/ruby/2.0.0/xmlrpc/client.rb:506:in `do_rpc' /usr/local/stow/ruby-2.0.0-p247/lib/ruby/2.0.0/xmlrpc/client.rb:281:in `call2'
/usr/local/stow/ruby-2.0.0-p247/lib/ruby/2.0.0/xmlrpc/client.rb:262:in `call'
/mnt/data/tanga/current/bundler/ruby/2.0.0/gems/hominid-3.0.4/lib/hominid/api.rb:32:in `method_missing'
/mnt/data/tanga/current/programs/mailchimp/mailchimp.rb:28:in `perform'
```

What's the best way for me to fix this problem now?

#### #21 - 01/04/2014 02:10 AM - mat (Mathieu Arnold)

Today I upgraded a ruby app that uses xmlrpc/client from 1.9.3-p374 to 2.0.0-p353, still not fixed in it...

#### #22 - 01/30/2014 06:16 AM - hsbt (Hiroshi SHIBATA)

- Target version changed from 2.1.0 to 2.2.0

#### #23 - 03/31/2014 01:14 AM - naruse (Yui NARUSE)

As I wrote, the check is useless. It should be removed.

#### #24 - 04/06/2014 10:08 PM - naruse (Yui NARUSE)

- Backport changed from 1.9.3: DONTNEED, 2.0.0: REQUIRED to 1.9.3: DONTNEED, 2.0.0: REQUIRED, 2.1: REQUIRED

#### #25 - 04/06/2014 10:27 PM - naruse (Yui NARUSE)

- Status changed from Assigned to Closed

- % Done changed from 0 to 100

Applied in changeset r45529.

- 
- lib/xmlrpc/client.rb (do\_rpc): don't check body length.  
If HTTP content-encoding is used, the length may be different.  
[Bug #8182] [ruby-core:53811]

#### #26 - 05/27/2014 02:47 AM - usa (Usaku NAKAMURA)

- Backport changed from 1.9.3: DONTNEED, 2.0.0: REQUIRED, 2.1: REQUIRED to 1.9.3: DONTNEED, 2.0.0: DONE, 2.1: REQUIRED

Backported into ruby\_2\_0\_0 at r46153.

#### #27 - 05/27/2014 03:27 PM - nagachika (Tomoyuki Chikanaga)

- Backport changed from 1.9.3: DONTNEED, 2.0.0: DONE, 2.1: REQUIRED to 1.9.3: DONTNEED, 2.0.0: DONE, 2.1: DONE

r45529 was backported into ruby\_2\_1 branch at r46186.

#### Files

308.patch	3.37 KB	05/20/2013	zzak (zzak _)
net.http.bug8182.patch	2.98 KB	06/12/2013	drbrain (Eric Hodel)
xmlrpc_client.rb.bug8182.patch	903 Bytes	06/12/2013	drbrain (Eric Hodel)