Backport200 - Backport #7831

Net::HTTP does not allow users to handle Content-Encoding in responses like 1.x

02/12/2013 08:57 AM - drbrain (Eric Hodel)

Status:	Closed	
Priority:	Normal	
Assignee:	drbrain (Eric Hodel)	

Description
I added a feature to always add a feature to always add Accept-Encoding to HTTP requests and always decode HTTP responses with Content-Encoding.

On Ruby 1.9 and older you could handle Content-Encoding for yourself.

Now Ruby always handles Content-Encoding for you, but does not give you a good indicator that this has taken place. In mechanize this leads to an incompatibility as the Content-Length header is not updated from the original value.

This also disallows handling of bad server responses that browsers handle.

The attached patch (upcoming) addresses this by only handling Content-Decoding in a response if the user did not override the Accept-Encoding header.

Since this is an incompatibility I would like this fixed for Ruby 2.0. Sorry I did not catch it sooner, I was too busy with RubyGems and RDoc.

Related issues:			
Related to Backport200 - Backport #7852: Backport r39238 to 2.0.0		**Closed**	**02/14/2013**
Related to Ruby master - Bug #7924: r39232 □□ net/http □□□□ reponse □□□□□□□□□...		**Closed**	**02/24/2013**

Associated revisions

Revision ddddf044 - 02/14/2013 03:51 PM - naruse (Yui NARUSE)

merge revision(s) 39232,39233,39238: [Backport #7831][Backport #7852]

```
    * lib/net/http:  Do not handle Content-Encoding when the user sets
      Accept-Encoding.  This allows users to handle Content-Encoding for
      themselves.  This restores backwards-compatibility with Ruby 1.x.

    * lib/net/http/generic_request.rb:  ditto.

    * lib/net/http/response.rb:  ditto

    * test/net/http/test_http.rb:  Test for the above.

    * test/net/http/test_http_request.rb:  ditto.

    * test/net/http/test_httpresponse.rb:  ditto.
      [ruby-trunk - Bug #7831]

    * lib/net/http.rb:  Removed OpenSSL dependency from Net::HTTP.

    * test/net/http/test_http.rb:  Remove Zlib dependency from tests.

    * test/net/http/test_http_request.rb:  ditto.
```

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/branches/ruby_2_0_0@39244 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision 39244 - 02/14/2013 03:51 PM - naruse (Yui NARUSE)

merge revision(s) 39232,39233,39238: [Backport #7831][Backport #7852]

```
* lib/net/http:  Do not handle Content-Encoding when the user sets
  Accept-Encoding.  This allows users to handle Content-Encoding for
  themselves.  This restores backwards-compatibility with Ruby 1.x.

* lib/net/http/generic_request.rb:  ditto.

* lib/net/http/response.rb:  ditto
```

```
* test/net/http/test_http.rb:  Test for the above.

* test/net/http/test_http_request.rb:  ditto.

* test/net/http/test_httpresponse.rb:  ditto.
  [ruby-trunk - Bug #7831]

* lib/net/http.rb:  Removed OpenSSL dependency from Net::HTTP.

* test/net/http/test_http.rb:  Remove Zlib dependency from tests.

* test/net/http/test_http_request.rb:  ditto.
```

History

#1 - 02/12/2013 09:10 AM - drbrain (Eric Hodel)

- File net.http.user_handled_content_encoding.patch added

#2 - 02/14/2013 04:48 AM - drbrain (Eric Hodel)

- Status changed from Open to Assigned

- Priority changed from Normal to 6

Naruse-San, I would like feedback for ruby 2.0.0 inclusion of this patch.

#3 - 02/14/2013 09:53 AM - naruse (Yui NARUSE)

ok, commit and backport to ruby_2_0_0 it.

#4 - 02/14/2013 10:12 AM - drbrain (Eric Hodel)

- Assignee changed from naruse (Yui NARUSE) to mame (Yusuke Endoh)

- Priority changed from 6 to Normal

Committed at r39232

mame, may I backport to 2_0_0?

#5 - 02/14/2013 01:56 PM - ko1 (Koichi Sasada)

I got failures because of no zlib and no open-ssl.

Could you skip tests if we don't have such libraries.

---

test-results:

Running tests:

[30/98] HTTPRequestTest#test_header_set = 0.00 s
1) Failure:
test_header_set(HTTPRequestTest) [C:/ko1/src/ruby/trunk/test/net/http/test_http_
request.rb:70]:
Bug #7831 - automatically decode content

[34/98] HTTPRequestTest#test_initialize_accept_encoding = 0.00 s
2) Failure:
test_initialize_accept_encoding(HTTPRequestTest) [C:/ko1/src/ruby/trunk/test/net
/http/test_http_request.rb:59]:
Bug #7831 - automatically decode content

[77/98] TestNetHTTP_v1_2#test_request = 0.14 s
3) Error:
test_request(TestNetHTTP_v1_2):
LoadError: cannot load such file -- openssl
C:/ko1/src/ruby/trunk/lib/net/http.rb:1427:in rescue in transport_request'
C:/ko1/src/ruby/trunk/lib/net/http.rb:1407:intransport_request'
C:/ko1/src/ruby/trunk/lib/net/http.rb:1382:in request'
C:/ko1/src/ruby/trunk/test/net/http/test_http.rb:435:intest_request_GET'

C:`/ko1/src/ruby/trunk/test/net/http/test_http.rb:421:in `block in test_reque
```

st'
C:/ko1/src/ruby/trunk/lib/net/http.rb:851:in start'
C:/ko1/src/ruby/trunk/test/net/http/utils.rb:11:instart'
C:/ko1/src/ruby/trunk/test/net/http/test_http.rb:420:in `test_request'

[81/98] TestNetHTTP_v1_2#test_set_formDL is deprecated, please use Fiddle
[92/98] TestNetHTTP_v1_2_chunked#test_request = 0.41 s
4) Error:
test_request(TestNetHTTP_v1_2_chunked):
LoadError: cannot load such file -- openssl
C:/ko1/src/ruby/trunk/lib/net/http.rb:1427:in rescue in transport_request'
C:/ko1/src/ruby/trunk/lib/net/http.rb:1407:intransport_request'
C:/ko1/src/ruby/trunk/lib/net/http.rb:1382:in request'
C:/ko1/src/ruby/trunk/test/net/http/test_http.rb:435:intest_request_GET'

```
C:/ko1/src/ruby/trunk/test/net/http/test_http.rb:421:in `block in test_reque
```

st'
C:/ko1/src/ruby/trunk/lib/net/http.rb:851:in start'
C:/ko1/src/ruby/trunk/test/net/http/utils.rb:11:instart'
C:/ko1/src/ruby/trunk/test/net/http/test_http.rb:420:in `test_request'

Finished tests in 14.122794s, 6.9391 tests/s, 30.8013 assertions/s.
98 tests, 435 assertions, 2 failures, 2 errors, 0 skips

ruby -v: ruby 2.0.0dev (2013-02-14 trunk 39236) [i386-mswin32_100]

#6 - 02/15/2013 12:15 AM - mame (Yusuke Endoh)

- Assignee changed from mame (Yusuke Endoh) to naruse (Yui NARUSE)

drbrain (Eric Hodel) wrote:

> Committed at r39232
>
> mame, may I backport to 2_0_0?

The issue that ko1 reported is fixed by #7852, right?

It is difficult for me to determine if they should be backported or not, so I leave it up to net/http maintainer.
Naruse-san, if you think it is appropriate, could you please backport them to ruby_2_0_0 or ask drbrain to do so?

Thanks!

--
Yusuke Endoh mame@tsg.ne.jp

#7 - 02/15/2013 12:24 AM - mame (Yusuke Endoh)

- Assignee changed from naruse (Yui NARUSE) to drbrain (Eric Hodel)

Oops, naruse-san has already said ok. Then, drbrain, please go ahead. Thanks!

--
Yusuke Endoh mame@tsg.ne.jp

#8 - 02/15/2013 12:50 AM - naruse (Yui NARUSE)

- Tracker changed from Bug to Backport

- Project changed from Ruby master to Backport200

- Category deleted (lib)

- Target version deleted (2.0.0)

#9 - 02/15/2013 12:51 AM - naruse (Yui NARUSE)

- Status changed from Assigned to Closed

- % Done changed from 0 to 100

This issue was solved with changeset r39244.
Eric, thank you for reporting this issue.

Your contribution to Ruby is greatly appreciated.
May Ruby be with you.

---

merge revision(s) 39232,39233,39238: [Backport #7831][Backport #7852]

```
* lib/net/http:  Do not handle Content-Encoding when the user sets
  Accept-Encoding.  This allows users to handle Content-Encoding for
  themselves.  This restores backwards-compatibility with Ruby 1.x.

* lib/net/http/generic_request.rb:  ditto.

* lib/net/http/response.rb:  ditto

* test/net/http/test_http.rb:  Test for the above.

* test/net/http/test_http_request.rb:  ditto.

* test/net/http/test_httpresponse.rb:  ditto.
  [ruby-trunk - Bug #7831]

* lib/net/http.rb:  Removed OpenSSL dependency from Net::HTTP.

* test/net/http/test_http.rb:  Remove Zlib dependency from tests.

* test/net/http/test_http_request.rb:  ditto.
```

#10 - 02/24/2013 07:55 AM - kazuhiko (Kazuhiko Shiozaki)

hsbt (Hiroshi SHIBATA) found a serious problem with this change.

require 'net/http'
r = Net::HTTP.start('www.iana.org') {|http|
response = http.get('/domains/example')
}
p r.header.decode_content
puts r.body

Even this small code will display 'false' for r.header.decode_content and gzip'ed binary for body (if the remote http server supports compression).

This issue happens because :
1) http.rb : get() sets accept-encoding = "gzip;q=1.0,deflate;q=0.6,identity;q=0.3" if not specified
2) then http/generic_request.rb : initialize() tries to set decode_content=true ONLY IF accept-encoding does not exist
thus decode_content remains in false and the body is not deflated.

What is not good is that we have 'setting accept-encoding if not specified' code twice, i.e. http.rb : get() and http/generic_request.rb : initialize(). And the latter code does not take into consideration the former code.

AFAIK, there is no other code that sets accept-encoding. So removing 'setting accept-encoding if not specified' code from http.rb : get() should solve this issue.

#11 - 02/24/2013 08:07 AM - kazuhiko (Kazuhiko Shiozaki)

> thus decode_content remains in false and the body is not deflated.

oups, "thus decode_content remains in false and the body is not UNCOMPRESSED".

Files

net.http.user_handled_content_encoding.patch	6.72 KB	02/12/2013	drbrain (Eric Hodel)