

Ruby - Bug #12126

[PATCH] openssl: accept moving write buffer for write_nonblock

02/29/2016 01:02 PM - normalperson (Eric Wong)

Status:	Closed	Backport: 2.1: UNKNOWN, 2.2: UNKNOWN, 2.3: UNKNOWN
Priority:	Normal	
Assignee:		
Target version:		
ruby -v:		
Description		
By setting the <code>SSL_MODE_ACCEPT_MOVING_WRITE_BUFFER</code> flag. This flag was introduced at the same time as <code>SSL_MODE_ENABLE_PARTIAL_WRITE</code> in OpenSSL 0.9.4 and makes usage with non-blocking sockets much easier. Before this, a Rubyist would need to remember the exact object which failed to write and reuse it later when the socket became writable again. This causes problems when the buffer is given by another layer of the application (e.g. a buffer is given by a Rack middleware or application to a Rack web server). <pre>* ext/openssl/ossl_ssl.c (ossl_sslctx_s_alloc): enable SSL_MODE_ACCEPT_MOVING_WRITE_BUFFER by default</pre> Related to https://bugs.ruby-lang.org/issues/12085 I'm not sure if this can be considered a bugfix which should be backported. I have found <code>write_nonblock</code> usability to be severely hampered by the lack of this flag.		
Related issues:		
Related to Ruby - Feature #12085: [PATCH] openssl: document `exception: false...		Closed

Associated revisions

Revision 280f7322151655b23c11a673b1adaab8796a358f - 03/31/2016 08:33 PM - Eric Wong

openssl: accept moving write buffer for write_nonblock

By setting the `SSL_MODE_ACCEPT_MOVING_WRITE_BUFFER` flag. This flag was introduced at the same time as `SSL_MODE_ENABLE_PARTIAL_WRITE` in OpenSSL 0.9.4 and makes usage with non-blocking sockets much easier.

Before this, a Rubyist would need to remember the exact object which failed to write and reuse it later when the socket became writable again. This causes problems when the buffer is given by another layer of the application (e.g. a buffer is given by a Rack middleware or application to a Rack web server).

- ext/openssl/openssl.c (openssl_ctx_s_alloc):
enable `SSL_MODE_ACCEPT_MOVING_WRITE_BUFFER` by default
[Bug #12126]

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@54466 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision 280f7322 - 03/31/2016 08:33 PM - Eric Wong

openssl: accept moving write buffer for write_nonblock

By setting the `SSL_MODE_ACCEPT_MOVING_WRITE_BUFFER` flag. This flag was introduced at the same time as `SSL_MODE_ENABLE_PARTIAL_WRITE` in OpenSSL 0.9.4 and makes usage with non-blocking sockets much easier.

Before this, a Rubyist would need to remember the exact object which failed to write and reuse it later when the socket became writable again. This causes problems when the buffer is given by another layer of the application (e.g. a buffer is given by a Rack middleware or application to a Rack web server).

- ext/openssl/openssl.c (ossl_sslctx_s_alloc):
enable SSL_MODE_ACCEPT_MOVING_WRITE_BUFFER by default
[Bug #12126]

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@54466 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

History

#1 - 03/01/2016 01:41 AM - normalperson (Eric Wong)

normalperson@yhbt.net wrote:

Before this, a Rubyist would need to remember the exact object which failed to write and reuse it later when the socket became writable again. This causes problems when the buffer is given by another layer of the application (e.g. a buffer is given by a Rack middleware or application to a Rack web server).

I should note the content of the buffer still needs to remain the unchanged. And IMHO, any reasonable non-blocking server working with stream sockets is expected to maintain identical content when retrying writes.

The main difference with this patch is the memory addresses passed to SSL_write() C function can be safely changed when retrying. IMHO it's unreasonable for a Ruby programmer to care about memory addresses passed to C functions when writing in Ruby.

#2 - 03/17/2016 07:00 AM - shyouhei (Shyouhei Urabe)

No objection.

#3 - 03/31/2016 08:33 PM - Anonymous

- Status changed from Open to Closed

Applied in changeset r54466.

openssl: accept moving write buffer for write_nonblock

By setting the SSL_MODE_ACCEPT_MOVING_WRITE_BUFFER flag. This flag was introduced at the same time as SSL_MODE_ENABLE_PARTIAL_WRITE in OpenSSL 0.9.4 and makes usage with non-blocking sockets much easier.

Before this, a Rubyist would need to remember the exact object which failed to write and reuse it later when the socket became writable again. This causes problems when the buffer is given by another layer of the application (e.g. a buffer is given by a Rack middleware or application to a Rack web server).

- ext/openssl/openssl.c (ossl_sslctx_s_alloc):
enable SSL_MODE_ACCEPT_MOVING_WRITE_BUFFER by default
[Bug #12126]

#4 - 06/07/2016 07:17 AM - naruse (Yui NARUSE)

- Related to Feature #12085: [PATCH] openssl: document `exception: false' for *_nonblock added

#5 - 06/07/2016 07:18 AM - naruse (Yui NARUSE)

- Related to Bug #12309: Segmentation fault caused by OpenSSL added

#6 - 06/07/2016 07:20 AM - naruse (Yui NARUSE)

- Related to Bug #6168: Segfault in OpenSSL bindings added

#7 - 10/23/2016 03:19 PM - rhenium (Kazuki Yamaguchi)

- Related to deleted (Bug #6168: Segfault in OpenSSL bindings)

#8 - 10/23/2016 03:23 PM - rhenium (Kazuki Yamaguchi)

- Related to deleted (Bug #12309: Segmentation fault caused by OpenSSL)

Files

0001-openssl-accept-moving-write-buffer-for-write_nonbloc.patch	2.53 KB	02/29/2016	normalperson (Eric Wong)
---	---------	------------	--------------------------