

Ruby trunk - Bug #15219

Backport: Ruby 2.5.X to support OpenSSL 1.1.1 and TLS 1.3

10/09/2018 03:20 PM - jaruga (Jun Aruga)

Status: Closed	
Priority: Normal	
Assignee:	
Target version:	
ruby -v:	Backport: 2.4: DONTNEED, 2.5: DONE
Description I would be happy that the coming Ruby 2.5.2 would support OpenSSL 1.1.1 and TLS 1.3 [1]. To do that, it seems at least below patch has to be backported to Ruby 2.5. net/http, net/ftp: fix session resumption with TLS 1.3 https://github.com/ruby/ruby/commit/1dfc377 And new ruby/openssl 2.2.2 has to be bundled in the Ruby 2.5.2. Possible? Thank you. [1] OpenSSL 1.1.1 release note: https://www.openssl.org/blog/blog/2018/09/11/release111/	

Associated revisions

Revision 64234 - 08/08/2018 02:13 PM - rhenium (Kazuki Yamaguchi)

net/http, net/ftp: fix session resumption with TLS 1.3

When TLS 1.3 is in use, the session ticket may not have been sent yet even though a handshake has finished. Also, the ticket could change if multiple session ticket messages are sent by the server. Use `SSLContext#session_new_cb` instead of calling `SSLSocket#session` immediately after a handshake. This way also works with earlier protocol versions.

Revision 64252 - 08/09/2018 10:00 AM - rhenium (Kazuki Yamaguchi)

net/http, net/ftp: skip SSL/TLS session resumption tests

Due to a bug in OpenSSL 1.1.0h1, the callback set by `SSLContext#session_new_cb=` does not get called for clients, making net/http and net/ftp not attempt session resumption.

Let's disable the affected test cases for now. Another option would be to fallback to using `SSLSocket#session` as we did before r64234. But since only a single version is affected and hopefully a new stable version containing the fix will be released in near future, I chose not to add such workaround code to lib/.

[1] <https://github.com/openssl/openssl/pull/5967>

Revision 0fd238c7 - 03/12/2019 11:23 PM - nagachika (Tomoyuki Chikanaga)

merge revision(s) 64234,64252: [Backport #15219]

```
net/http, net/ftp: fix session resumption with TLS 1.3
```

```
When TLS 1.3 is in use, the session ticket may not have been sent yet
even though a handshake has finished. Also, the ticket could change if
multiple session ticket messages are sent by the server. Use
SSLContext#session_new_cb instead of calling SSLSocket#session
immediately after a handshake. This way also works with earlier protocol
versions.
```

```
net/http, net/ftp: skip SSL/TLS session resumption tests
```

Due to a bug in OpenSSL 1.1.0h[1] (it's only in this specific version; it was introduced just before the release and is already fixed in their stable branch), the callback set by `SSLContext#session_new_cb=` does not get called for clients, making `net/http` and `net/ftp` not attempt session resumption.

Let's disable the affected test cases for now. Another option would be to fallback to using `SSLSocket#session` as we did before r64234. But since only a single version is affected and hopefully a new stable version containing the fix will be released in near future, I chose not to add such workaround code to `lib/`.

[1] <https://github.com/openssl/openssl/pull/5967>

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/branches/ruby_2_5@67237 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision 67237 - 03/12/2019 11:23 PM - nagachika (Tomoyuki Chikanaga)

merge revision(s) 64234,64252: [Backport #15219]

`net/http, net/ftp: fix session resumption with TLS 1.3`

When TLS 1.3 is in use, the session ticket may not have been sent yet even though a handshake has finished. Also, the ticket could change if multiple session ticket messages are sent by the server. Use `SSLContext#session_new_cb` instead of calling `SSLSocket#session` immediately after a handshake. This way also works with earlier protocol versions.

`net/http, net/ftp: skip SSL/TLS session resumption tests`

Due to a bug in OpenSSL 1.1.0h[1] (it's only in this specific version; it was introduced just before the release and is already fixed in their stable branch), the callback set by `SSLContext#session_new_cb=` does not get called for clients, making `net/http` and `net/ftp` not attempt session resumption.

Let's disable the affected test cases for now. Another option would be to fallback to using `SSLSocket#session` as we did before r64234. But since only a single version is affected and hopefully a new stable version containing the fix will be released in near future, I chose not to add such workaround code to `lib/`.

[1] <https://github.com/openssl/openssl/pull/5967>

History

#1 - 10/09/2018 03:20 PM - jaruga (Jun Aruga)

- Backport deleted (2.3: UNKNOWN, 2.4: UNKNOWN, 2.5: UNKNOWN)

- Tracker changed from Bug to Feature

#2 - 10/09/2018 06:08 PM - shevegen (Robert A. Heiler)

This would be nice indeed. I have a small gem that collects information about the host-system (on the target computer platform; usually linux) available, and notifies when there are more recent versions of a software available, e. g. a new gcc release, a new m4 release, a new bison release and so forth.

I am a bit wary of upgrading openssl from openssl-1.1.0i to openssl-1.1.1 mostly because I am never absolutely sure how well a more recent openssl may work on ruby. And the primary reason for me to use openssl (and have ruby support it, too) is so that I can push new gem releases of my code, actually. This was also a major reason why I used to open issues about both openssl and readline, and I think it was nobu who then added the "+" commandline flag to configure, to allow compilation to proceed only if all that has been wanted, been found too (as otherwise I may have to re-compile ruby or at the least work on this in the `ext/` subdirectory, such as for readline or openssl or zlib).

So naturally, I think it would be nice if more recent openssl versions could be supported on the ruby 2.5.x branch too, if this will retain backwards-compatible behaviour.

Having said that, I think after x-mas, I will be using ruby 2.6.x so it would not

be of a massive benefit to me personally.

On a side note, if it were possible, it may be helpful to notify on the ruby-doc website which versions of a particular software is supported.

Take:

<https://ruby-doc.org/stdlib/libdoc/openssl/rdoc/OpenSSL.html>

This page could list which version is compatible - or at the least has been tested. I don't know of a good way to have this automatically for all versions, but I think it may be useful for quite a few people. (Openssl, zlib and Readline are usually what I need to have in the local ruby version, since it is very convenient or necessary for other things.)

I think naruse is in charge of handling both 2.6.x and 2.5.x release so perhaps he should be asked.

#3 - 10/09/2018 06:34 PM - jaruga (Jun Aruga)

- Subject changed from Ruby 2.5.X supporting OpenSSL 1.1.1 and TLS 1.3 to Ruby 2.5.X to support OpenSSL 1.1.1 and TLS 1.3

#4 - 10/19/2018 12:51 PM - jaruga (Jun Aruga)

To do that, it seems at least below patch has to be backported to Ruby 2.5.

net/http, net/ftp: fix session resumption with TLS 1.3
<https://github.com/ruby/ruby/commit/1dfc377>

Maybe this patch too.

config: support .include directive
<https://github.com/ruby/openssl/pull/216>

And optionally this patch.

test: use larger keys for SSL tests
<https://github.com/ruby/openssl/pull/217>

#5 - 10/23/2018 01:48 PM - jaruga (Jun Aruga)

- Subject changed from Ruby 2.5.X to support OpenSSL 1.1.1 and TLS 1.3 to Backport: Ruby 2.5.X to support OpenSSL 1.1.1 and TLS 1.3

#6 - 12/16/2018 09:57 PM - naruse (Yui NARUSE)

- Backport set to 2.4: DONTNEED, 2.5: UNKNOWN
- Status changed from Open to Closed
- Tracker changed from Feature to Bug

Close to be on tracking on backport process.

#7 - 01/08/2019 02:50 AM - nagachika (Tomoyuki Chikanaga)

- Backport changed from 2.4: DONTNEED, 2.5: UNKNOWN to 2.4: DONTNEED, 2.5: REQUIRED

#8 - 01/14/2019 11:27 AM - nagachika (Tomoyuki Chikanaga)

Maybe this patch too.

config: support .include directive
<https://github.com/ruby/openssl/pull/216>

And optionally this patch.

test: use larger keys for SSL tests
<https://github.com/ruby/openssl/pull/217>

Hmm, these two pull requests are not merged yet in ruby/openssl and neither committed into ruby trunk.

We can backport them only after they are committed into trunk according to our stable branch management policy.

[rhenium \(Kazuki Yamaguchi\)](#) Could you handle these pull requests?

#9 - 03/12/2019 11:23 PM - nagachika (Tomoyuki Chikanaga)

- Backport changed from 2.4: *DONTNEED*, 2.5: *REQUIRED* to 2.4: *DONTNEED*, 2.5: *DONE*

ruby_2_5 r67237 merged revision(s) 64234,64252.