

Ruby trunk - Feature #12077

Consolidate SSLSocket interface with TCPSocket

02/16/2016 11:33 AM - tonci (Tonči Damjanić)

Status:	Open
Priority:	Normal
Assignee:	
Target version:	
Description	
Hello,	
Is there a reason why SSLSocket#read doesn't work the same as SSLSocket#sysread and why there is no SSLSocket#recv?	
Right now it is impossible to have code agnostic in regards to the socket type.	
One solution is to monkey-patch OpenSSL::SSL::SSLSocket (see lower) to share the same interface as TCPSocket, but given the generic names of monkey-patched functions, I'm not sure if they could result with some leaks or strange behavior.	
<pre>class OpenSSL::SSL::SSLSocket def read(*args) sysread(*args) end def write(*args) syswrite(*args) end def close sysclose end end</pre>	
Thoughts?	
Related issues:	
Is duplicate of Ruby trunk - Bug #8126: OpenSSL::SSL::SSLSocket does not defi...	Assigned

History

#1 - 02/16/2016 11:34 AM - tonci (Tonči Damjanić)

- Description updated

#2 - 02/16/2016 11:35 AM - tonci (Tonči Damjanić)

- Description updated

#3 - 02/24/2016 04:08 PM - tonci (Tonči Damjanić)

- Assignee set to ruby-core

#4 - 03/06/2016 01:02 AM - normalperson (Eric Wong)

tonci.damjanic@gmail.com wrote:

Is there a reason why SSLSocket#read doesn't work the same as SSLSocket#sysread

Because IO#read and IO#sysread are different, too. The former is read-in-full behavior (outside of EOF) while sysread matches the low-level C interface and allows for partial reads.

Likewise with write vs syswrite regarding write-in-full behavior.

and why there is no SSLSocket#recv?

Currently, OpenSSL does not provide an interface to `recv(2)` which handles flag arguments :<

I too wish for `recv(2)` and `send(2)` interfaces which lets me use `MSG_MORE|MSG_DONTWAIT` flags.

#5 - 05/17/2016 06:37 AM - naruse (Yui NARUSE)

- Assignee deleted (*ruby-core*)

#6 - 04/27/2017 01:47 PM - naruse (Yui NARUSE)

- Is duplicate of Bug #8126: *OpenSSL::SSL::SSLSocket* does not define `#recv` and `#send` messages added