Subject: Re: A problem with TcpSocket::GetLine()
Posted by Oblivion on Tue, 03 Dec 2013 17:12:18 GMT

nlneilson wrote on Tue, 03 December 2013 18:03First: I suggest you create a server on your own machine.
There is an example of a server and client in the SDK.

Second: End your String with \0

Then post with what you are getting.

I use a socket as client in C/C++ to communicate with a java server. Trying to jump on the web until you get the bugs out with the client and server on your own machine is like trying to run before learning to walk.

Hello nineilson, thank you for your reply.

I already examined server & client example, I am afraid it was not very helpful for my case.

The code snippet I posted above was actually the simplest test case I could come up with, to show the actual behaviour.
(by the way, this is tested under Linux)

Just to clarify the situation in detail: I wrote a POP3 class, which actually works as expected and my intention was to upload it to the Upp bazaar. But then I decided to simplfy the code a bit. Because;

1) A standard POP3 server produces two types of well defined and predictable responses which can be read by a client. A single line respone (terminates with a "\r\n") and a multi-line response (terminates with a "\r\n.\r\n").

3) So, to increase the code clarity I decided to use the TcpSocket::GetLine() method to read single-line server responses (such as the pop3 server "hello" messages).

In theory, this should work (or I am getting something wrong?). But in practice, as with the above test case, it simply and immediately fails with a timeout error and increasing the timeout value with GlobalTimeout() or Timeout() does not help at all -- It does not read anything. Now, the thing is, as I've mentioned on my previous post, TcpSocket::Get() method works where TcpSocket::GetLine() reads nothing.

Any other ideas?

Regards.