

Re: xmlrpc, httplib and SSL

Source: <http://coding.derkeiler.com/Archive/Python/comp.lang.python/2004-03/4462.html>

From: Roger Binns (rogerb_at_rogerbinns.com)

Date: 03/27/04

Date: Sat, 27 Mar 2004 11:21:30 -0800

Skip Montanaro wrote:

> *[snip summary of changes]*

>

> Roger> I now have code that works for me and my project. However it

> Roger> wouldn't really be appropriate for going back into the standard

> Roger> library because it spends most of its time having to subvert the

> Roger> design and implementation of the existing classes.

>

> Roger> However I was wondering if anyone was working on fixing the (IMHO

> Roger> horrible) mess and wants any moral support?

>

> Sounds like you've done most of the work already. Why not post a few

> patches to SF for the various affected modules?

The standard modules need to be redesigned! And M2Crypto would have to be the standard SSL. And as I state in the first paragraph you quote, it has to subvert the existing classes. Think of my code as the result of using chainsaws and band-aids all over the existing code. The end result works fine for me and my project. But I wouldn't remotely want to change the existing library to anything like it.

Here were the goals I was trying to meet. The current Python library does not remotely meet them.

XML-RPC client:

- Connections are made over SSL
- A callback to verify the certificate if there is no CA or other defined mechanism
- HTTP/1.1 is used, and the connection is reused if still open
- HTTP authentication is used
- Automatic reopen of connection if above one doesn't work (eg remote end closed it after timeout)
- Be thread safe (eg one connection per thread - don't reuse across threads)

XML-RPC server:

comp.lang.python: Re: xmlrpc, httplib and SSL

- Connections are accepted over SSL
- Callbacks to verify incoming connections, certificates and credentials of the connections
- Verification of HTTP authentication information
- Connections are kept open/keep-alive (HTTP/1.1)
- Use of a bounded thread pool that handles the connections

Roger