Subject: Re: WebSSL memory leak problem.
Posted by rylek on Wed, 26 May 2010 10:00:51 GMT
View Forum Message <> Reply to Message

Hello there!

I'm afraid that commenting out CRYPTO_set_mem_functions doesn't remove the memory leaks, it just hides them because OpenSSL then uses standard C allocator instead of the one in U++. In a similar vein, getting rid of SSL_library_init() is in my opinion a bad idea either. Some time ago when debugging a https client using the Web/SSL package I also started to receive some memory leaks, but after recompiling an updated version of OpenSSL they disappeared.

In order to be able to debug this kind of leaks, a long time ago I added a simple memory allocation logging feature to Web/SSL (just set LOG_UPP_SSL_MALLOC to 1). The last time I was tracing the leaks, they seemed related to certain global tables allocated during the SSL library initialization (which might explain why removing the initialization call seemingly removed the problem). However, as I said, after recompiling the OpenSSL they disappeared altogether so I honestly didn't continue to pursue the problem.

Please also note that serious havoc would result from linking your application to OpenSSL compiled with a different build style (MT vs ST, shared vs static libraries). When recompiling OpenSSL it is good to note which /M flag the make sends to the compiler (at least this holds when compiling under 32-bit MSVC 7.1, which is what I did). In any case if you fail to get rid of the leaks, I can at least try to post somewhere the compiled OpenSSL libraries (libeay / ssleay) which do work for me so that you can try them out. I hope the Web/SSL package is not discontinued, it's just that getting rid of memory leaks in code you didn't write is sometimes quite complicated. Which of course doesn't mean we should ignore them.

Regards

Tomas