Building libcurl+ssl using VS Express 2013 For Windows Desktop

Aron P. Dobos 28 October 2013

Instructions for 32-bit Version

1. Download software:
   1. ActivePerl: <http://www.activestate.com/activeperl/downloads>
   2. OpenSSL 1.0.1c <http://www.openssl.org/source/openssl-1.0.1c.tar.gz>
   3. Libssl 1.4.2 <http://www.libssh2.org/download/libssh2-1.4.2.tar.gz>
   4. Curl 7.28.0 <http://curl.haxx.se/download/curl-7.28.0.zip>
   5. Netwide assembler <http://www.nasm.us/pub/nasm/releasebuilds/2.10.05/win32/nasm-2.10.05-win32.zip>
2. Install ActivePerl if you don’t already have it
3. Create a folder c:\curl and un-tar openssl, libssl, and curl into this folder
4. Start menu -> run -> cmd.exe
5. cd C:\Program Files (x86)\Microsoft Visual Studio 12.0\VC\bin
6. Enter the command and press enter: vcvars32.bat
7. Test by typing cl.exe. Output should look like this:

c:\Program Files (x86)\Microsoft Visual Studio 12.0\VC\bin>cl.exe

Microsoft (R) C/C++ Optimizing Compiler Version 18.00.21005.1 for x86

Copyright (C) Microsoft Corporation. All rights reserved.

1. Unzip the netwide assembler binaries to c:\nasm-2.10.05
2. Add it to the path path = %PATH%;C:\nasm-2.10.055
3. cd c:\curl\openssl-1.0.1c
4. perl Configure VC-WIN32 –-prefix=c:/curl/openssl\_dll
5. ms\do\_nasm
6. nmake –f ms\ntdll.mak
7. nmake –f ms\ntdll.mak test
8. nmake –f ms\ntdll.mak install
9. Open VS 2013 Express.
10. Open c:\curl\libssh2-1.4.2\win32\libssh2.dsw
11. Allow VS2013 to upgrade the project file
12. On the toolbar, select the “DLL Release” configuration.
13. Right click on ‘libssh2’ in the solution explorer, and select ‘Properties’
    1. Under ‘C/C++/General/Additional Include Directories’, add c:\curl\openssl\_dll\include;
    2. Under ‘Linker/General/Additional Library Directories’, add c:\curl\openssl\_dll\lib;
    3. Under ‘Linker/Input’, remove zlib.lib
14. Build the solution C:\curl\libssh2-1.4.2\win32\Release\_dll should have the files libssh2.dll and libssh2.lib. You might get errors about the “tests” project not finding libeay32.lib, but you can probably ignore this.
15. Close the project. Open C:\curl\curl-7.28.0\lib\libcurl.vcproj – allow VC2013 to upgrade the project file.
16. Select the ‘Release’ build.
17. In solution explorer, select properties of ‘libcurl’
18. In General, select Configuration Type = ‘Dynamic Library (.dll)’ , then click “Apply” in the dialog
19. Under C/C++/General/Additional Include Directories, add C:\curl\openssl-1.0.1c\inc32\;C:\curl\libssh2-1.4.2\include; c:\curl\openssl\_lib\include\; c:\curl\openssl\_lib\include\openssl
20. Under C/C++/Preprocessor/Preprocessor Definitions/ add USE\_OPENSSL;USE\_LIBSSH2;CURL\_DISABLE\_LDAP;HAVE\_LIBSSH2;HAVE\_LIBSSH2\_H;LIBSSH2\_WIN32 ;LIBSSH2\_LIBRARY;USE\_SSLEAY
21. Under Linker/Input/Additional Dependencies, add libssh2.lib;libeay32.lib;ssleay32.lib;ws2\_32.lib
22. Under Linker/General/Additional Library Directories, add C:\curl\libssh2-1.4.2\win32\Release\_dll;c:\curl\openssl\_dll\lib
23. Build. It should succeed.
24. Now you have all the dlls, libs, and relevant header files to include in other VC2013 projects that use the v120 C runtime.
25. Aggregate all the headers and binaries into a single “redistributable” with these commands:
    1. >> copy C:\curl\curl-7.28.0\lib\Release\libcurl.dll c:\curl\openssl\_dll\bin
    2. >> copy C:\curl\curl-7.28.0\lib\Release\libcurl.lib c:\curl\openssl\_dll\lib
    3. >> copy C:\curl\curl-7.28.0\lib\Release\libcurl.exp c:\curl\openssl\_dll\lib
    4. >> copy C:\curl\libssh2-1.4.2\win32\Release\_dll\libssh2.dll c:\curl\openssl\_dll\bin
    5. >> copy C:\curl\libssh2-1.4.2\win32\Release\_dll\libssh2.lib c:\curl\openssl\_dll\lib
    6. >> copy C:\curl\libssh2-1.4.2\win32\Release\_dll\libssh2.exp c:\curl\openssl\_dll\lib
    7. >> mkdir c:\curl\openssl\_dll\include\ssh
    8. >> copy c:\curl\libssh2-1.4.2\include\\* c:\curl\openssl\_dll\include\ssh
    9. >> mkdir c:\curl\openssl\_dll\include\curl
    10. >> copy c:\curl\curl-7.28.0\include\curl\\* c:\curl\openssl\_dll\include\curl
26. Rename the c:\curl\openssl\_dll folder to libcurl\_ssl\_vc2013\_win32

Instructions for 64-bit Version

1. If you’re doing this right after having finished up the 32 bit build above, create a subfolder in c:\curl called “win32-build” and move all the current contents of c:\curl (except the untouched tar.gz files) into c:\curl\win32-build.
2. Close the command prompt and open a new one to refresh the environment variables.
3. Change to c:\curl and untar the three libraries: curl-7.28.0, libssh2-1.4.2, openssl-1.0.1c
4. cd C:\Program Files (x86)\Microsoft Visual Studio 12.0\VC
5. Enter the command and press enter>> vcvarsall.bat x86\_amd64
6. Test by typing cl.exe. Output should look like this:

C:\Program Files (x86)\Microsoft Visual Studio 12.0\VC>cl

Microsoft (R) C/C++ Optimizing Compiler Version 18.00.21005.1 for x64

Copyright (C) Microsoft Corporation. All rights reserved.

1. Unzip the netwide assembler binaries to c:\nasm-2.10.05
2. cd c:\curl\openssl-1.0.1c
3. perl Configure VC-WIN64A –-prefix=c:/curl/openssl\_dll
4. ms\do\_win64a
5. nmake –f ms\ntdll.mak
6. nmake –f ms\ntdll.mak test
7. nmake –f ms\ntdll.mak install
8. Open VS 2013 Express.
9. Open c:\curl\libssh2-1.4.2\win32\libssh2.dsw
10. Allow VS2013 to upgrade the project file
11. Click on the dropdown button next to “Win32” and open the “Configuration Manager”
12. Under “Active solution platform”, select “<New…>” and click OK to create x64 configuration.
13. Under “Active solution configuration”, select “DLL Release”
14. Right click on ‘libssh2’ in the solution explorer, and select ‘Properties’
    1. Under ‘C/C++/General/Additional Include Directories’, add c:\curl\openssl\_dll\include;
    2. Under ‘Linker/General/Additional Library Directories’, add c:\curl\openssl\_dll\lib;
    3. Under ‘Linker/Input’, remove zlib.lib
15. Build the solution C:\curl\libssh2-1.4.2\win32\Release\_dll should have the files libssh2.dll and libssh2.lib. You might get errors about the “tests” project not finding libeay32.lib, but you can probably ignore this.
16. Close the project. Open C:\curl\curl-7.28.0\lib\libcurl.vcproj – allow VC2013 to upgrade the project file.
17. In the configuration manager, create a new “x64” platform, and select the ‘Release’ build.
18. In solution explorer, select properties of ‘libcurl’
19. In General, select Configuration Type = ‘Dynamic Library (.dll)’, then click “Apply” in the dialog
20. Under C/C++/General/Additional Include Directories, add C:\curl\openssl-1.0.1c\inc32\;C:\curl\libssh2-1.4.2\include; c:\curl\openssl\_lib\include\; c:\curl\openssl\_lib\include\openssl
21. Under C/C++/Preprocessor/Preprocessor Definitions/ add USE\_OPENSSL;USE\_LIBSSH2;CURL\_DISABLE\_LDAP;HAVE\_LIBSSH2;HAVE\_LIBSSH2\_H;LIBSSH2\_WIN32 ;LIBSSH2\_LIBRARY;USE\_SSLEAY
22. Under Linker/Input/Additional Dependencies, add libssh2.lib;libeay32.lib;ssleay32.lib;ws2\_32.lib
23. Under Linker/General/Additional Library Directories, add C:\curl\libssh2-1.4.2\win32\Release\_dll;c:\curl\openssl\_dll\lib
24. Build. It should succeed.
25. Now you have all the 64-bit dlls, libs, and relevant header files to include in other VC2013 projects that use the v120 C runtime.
26. Aggregate all the headers and binaries into a single “redistributable” with these commands:
    1. >> copy C:\curl\curl-7.28.0\lib\x64\Release\libcurl.dll c:\curl\openssl\_dll\bin
    2. >> copy C:\curl\curl-7.28.0\lib\x64\Release\libcurl.lib c:\curl\openssl\_dll\lib
    3. >> copy C:\curl\curl-7.28.0\lib\x64\Release\libcurl.exp c:\curl\openssl\_dll\lib
    4. >> copy C:\curl\libssh2-1.4.2\win32\Release\_dll\libssh2.dll c:\curl\openssl\_dll\bin
    5. >> copy C:\curl\libssh2-1.4.2\win32\Release\_dll\libssh2.lib c:\curl\openssl\_dll\lib
    6. >> copy C:\curl\libssh2-1.4.2\win32\Release\_dll\libssh2.exp c:\curl\openssl\_dll\lib
    7. >> mkdir c:\curl\openssl\_dll\include\ssh
    8. >> copy c:\curl\libssh2-1.4.2\include\\* c:\curl\openssl\_dll\include\ssh
    9. >> mkdir c:\curl\openssl\_dll\include\curl
    10. >> copy c:\curl\curl-7.28.0\include\curl\\* c:\curl\openssl\_dll\include\curl
27. Rename the c:\curl\openssl\_dll folder to libcurl\_ssl\_vc2013\_x64