# 在Windows上编译最新的CURL，含有zlib，openssl

1. 准备工作

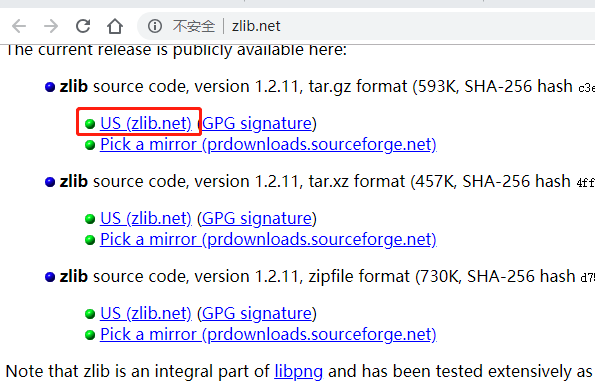
  a. 下载zlib

          zlib可以使得HTTP请求支持gzip压缩，其地址如下：

          官网：http://zlib.net/

          下载地址：<http://zlib.net/zlib-1.2.11.tar.gz>

本地：E:\程序人生\开源库\curl\20181130\



      b. 下载openssl

          opensll使用与支持SSL，例如HTTPS的请求。

          官网：http://www.openssl.org

          下载地址：http://www.openssl.org/source/openssl-1.0.1f.tar.gz

本地：E:\程序人生\开源库\curl\20181130\

      c. 下载curl

          官网：http://curl.haxx.se

          下载地址：在http://curl.haxx.se/download.html选择你所想要的版本，我这儿选用 curl-7.35.0.tar.gz版本 [http://curl.haxx.se/download/curl-7.35.0.tar.gz]

注意：curl-7.62.0.tar.gz版本我按以下方式编译失败，原因待查

本地：E:\程序人生\开源库\curl\20181130\

    d. 下载ActivePerl

          官网：http://www.activestate.com/activeperl/downloads

          请注意根据自己的系统版本【32,64】选择不同的安装包【链接失效时用迅雷可下 载】

本地：E:\程序人生\开源库\curl\20181130\

IMG_256

2、编译

a. 安装activePerl

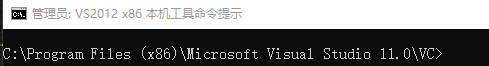
        直接双击安装，用默认设置就可以了。

     b. 解压所有的源码文件，全部放在一个目录下，比如我的目录结 构如下：



     c. 编译zlib

          从开始菜单，选择Microsoft Visual Studio 2012 > Visual Studio Tools > VS2012 X86本机 工具命令提示，如下图：



      弹出一个CMD窗口，切换到zlib的目录下E:\程序人生\开源库 \curl\20181130\zlib-1.2.11，然后输入：

nmake -f win32/Makefile.msc OBJA="inffast.obj"

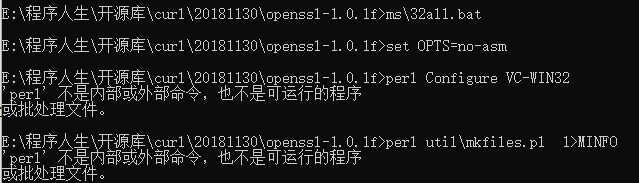
过了一会儿就编译成功。

   d. 编译openssl

     将当前目录切换到E:\程序人生\开源库\curl\20181130\openssl-1.0.1f，执行下面的命令：

ms\32all.bat

报错：



解决办法：set path=C:\Perl64\bin;%path%;

再执行ms\32all.bat

然后就开始了漫长的编译，我这等了10分钟才编译好= =$

  e. 编译curllib

    将当前目录切换到curl的的目录下C:\curl\curl-7.35.0\lib，然后依次输入一下的命令：

set OPENSSL\_PATH=..\..\openssl-1.0.1f

set ZLIB\_PATH=..\..\zlib-1.2.11

nmake -f Makefile.vc10 CFG=release-dll-zlib-dll

注意 Makefile.vc6是VC6.0的版本，VC8是VC2005，VC9是VS2008(我安装的是vs2012)，依次类推吧。

CFG是配置编译结果选项，其具体内容如下：

Usage: nmake /f makefile.vc10 CFG=<config> <target>

where <config> is one of:

release - release static library

release-ssl - release static library with ssl

release-zlib - release static library with zlib

release-ssl-zlib - release static library with ssl and zlib

release-ssl-ssh2-zlib - release static library with ssl, ssh2 and zlib

release-ssl-dll - release static library with dynamic ssl

release-zlib-dll - release static library with dynamic zlib

release-ssl-dll-zlib-dll - release static library with dynamic ssl and dynamic zlib

release-dll - release dynamic library

release-dll-ssl-dll - release dynamic library with dynamic ssl

release-dll-zlib-dll - release dynamic library with dynamic zlib

release-dll-ssl-dll-zlib-dll - release dynamic library with dynamic ssl and dynamic zlib

debug - debug static library

debug-ssl - debug static library with ssl

debug-zlib - debug static library with zlib

debug-ssl-zlib - debug static library with ssl and zlib

debug-ssl-ssh2-zlib - debug static library with ssl, ssh2 and zlib

debug-ssl-dll - debug static library with dynamic ssl

debug-zlib-dll - debug static library with dynamic zlib

debug-ssl-dll-zlib-dll - debug static library with dynamic ssl and dynamic zlib

debug-dll - debug dynamic library

debug-dll-ssl-dll - debug dynamic library with dynamic ssl

debug-dll-zlib-dll - debug dynamic library with dynamic zlib1

debug-dll-ssl-dll-zlib-dll - debug dynamic library with dynamic ssl and dynamic zlib

<target> can be left blank in which case all is assumed

Makefile.vc8(501) : fatal error U1050: please choose a valid configuration "rele

ase-dll-ssl-dll-zlib-dll "

Stop.

大家可以根据自己的需要编译不同的DLL和lib，我就选择编译了两个静态库：

nmake -f Makefile.vc8 CFG=release-ssl-zlib

nmake -f Makefile.vc8 CFG=debug-ssl-zlib

然后就是开始华丽丽的刷屏等待啦~~~

    过了几分钟之后，就编译好了，将C:\curl\curl-7.35.0\lib\会有debug-ssl-zlib和release-ssl-zlib目录，拷贝到VS的工程目下，然后配置一下就可以使用了。

KevinJPeng

2018/11/30

---------------------

来源：CSDN

原文：https://blog.csdn.net/hujkay/article/details/18986153