**LibCurl编译**

# LibCurl依赖库

## OpenSSL

## Zlib

## SSH2

## NGHTTP2

# OpenSSL编译

## 源码信息

源码位置：<https://github.com/openssl/openssl>

版本：3.0.1

CURL本身支持HTTPS，但只支持SSL，不支持TLS；加入此库可以支持TLS

## 编译要求

In addition to the requirements and instructions listed in INSTALL.md, these are required as well:

### Perl

We recommend Strawberry Perl, available from <http://strawberryperl.com/> Please read NOTES.PERL for more information, including the use of CPAN. An alternative is ActiveState Perl, <https://www.activestate.com/ActivePerl> for which you may need to explicitly build the Perl module Win32/Console.pm via <https://platform.activestate.com/ActiveState> and then download it.

### Netwide Assembler (NASM)

NASM is the only supported assembler. It is available from [https://www.nasm.us](https://www.nasm.us/).

## 编译步骤

1. Install Perl
2. Install NASM
3. Make sure both Perl and NASM are on your %PATH%
4. From the root of the OpenSSL source directory enter perl Configure VC-WIN32 if you want 32-bit OpenSSL or perl Configure VC-WIN64A if you want 64-bit OpenSSL or perl Configure to let Configure figure out the platform
5. nmake
6. nmake test
7. nmake install

备注：编译64位时需要先执行nmake clean，清除32位相关信息，第1步骤直接使用Perl可能会报错，需要找到perl对应安装目录将Perl替换为PerlX.XX.X.exe。Debug版本使用perl Configure debug-VC-WIN32，64位类似

# Zlib编译

## 源码信息

源码位置：https://github.com/madler/zlib

版本：1.2.11

此库主要用于压缩

## 编译步骤

打开目录中[zlib](https://github.com/madler/zlib)/[contrib](https://github.com/madler/zlib/tree/master/contrib)/**vstudio**/对应的SLN编译即可

# SSH2编译

## 源码信息

源码位置：https://github.com/libssh2/libssh2

版本：1.10.0

默认的curl支持ftp和ftps，但不支持sftp，加入此库可以支持sftp

## 编译要求

### cmake

从官网下载并安装<https://cmake.org/download/>

## 编译步骤

1. mkdir bin
2. cd bin
3. cmake ..
4. cmake --build .

注意上述中的”.”必须带上，cmake默认编译为win64，可以找到cmake安装目录下对应的GUI界面，配置为WIN32程序，下述中NGHTTP2也类似。

# NGHTTP2编译

## 源码信息

源码位置：https://github.com/nghttp2/nghttp2

版本：1.46.0

此库可以让curl支持HTTP2.0

## 编译要求

### cmake

从官网下载并安装<https://cmake.org/download/>

## 编译步骤

1. cmake .
2. cmake --build .

注意上述中的”.”必须带上

# LibCurl编译

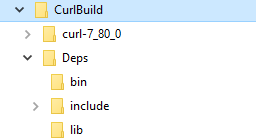
## 源码信息

源码位置：https://github.com/curl/curl

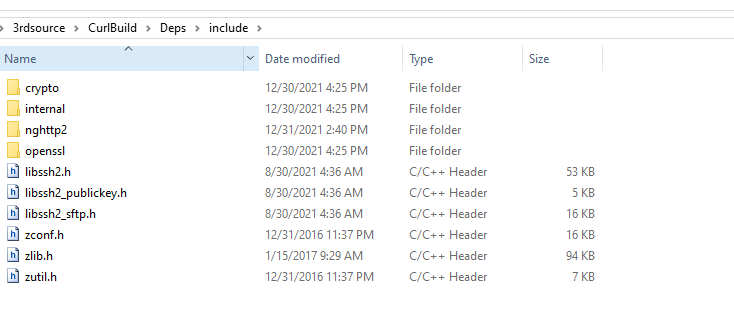
版本：7.80.0

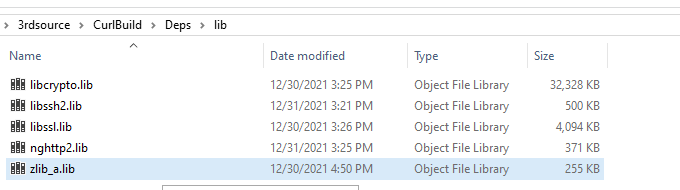
## 编译步骤

1. 目录结构



1. 将上面依赖库相关lib和头文件拷贝，结构如下2图





1. 在Curl目录下的WinBuild下执行编译命令，如下

nmake /f makefile.vc mode=dll WITH\_SSL=static WITH\_ZLIB=static WITH\_SSH2=static WITH\_NGHTTP2=static vc=16 DEBUG=yes



相关输出文件保存在curl-7\_80\_0\builds下

# 总结

命令行工具统一使用x64 Native Tools Command Prompt for VS 2019或x86 Native Tools Command Prompt for VS 2019

使用CmakeGUI生成的，统一生成VS2019工程

OPENSSL和Curl编译64bit时，必要时可能需要注释ASSERT和去除CSV等后缀名

上述中SSH2和NGHTTP2均以Cmake编译，Cmake编译时时会生成Visual Studio相关工程文件，可以通过参数配置生成32bit和64bit及可以通过VS修改DLL\LIB方式，上述所有LIB和dll使用同版本的VS及相关命令编译，避免出现无实现体相关链接错误。

以上相关信息均在github相应的源码位置上提取的编译信息，如果需要更详细的说明可以在github上查阅，上面每个编译均附有相关源码URL