



linux下编译安装libcurl(附使用示例)

2016-06-16 _Xqiang 摘自 csdn博客 阅 1

分享： 微信 转藏到我的图书馆

TA的最新

- C语言
- C语言
- linux下
- HTTP
- 关于C
- va

推荐阅读

- C : 后
- 浅析HT
- Linux下
- gnome
- Fedora
- Fedora
- C语言
- C++标
- C++实

linux下编译安装libcurl(附使用示例)

qianghaohao(Xqiang)

说明：在编译任何开源代码前可以看看README，INSTALL等文件，会提供如何编译的信息。在此教程中通过阅读README和GIT-INFO文件可以得到很多编译安装信息。在GIT-INFO中很明显地说了执行：./buildconf产生configure配置文件。所有不要在这里因为找不到configure文件而懵逼。

- 1.下载libcurl源码：
git clone https://github.com/curl/curl.git
- 2.进入curl工程目录执行./buidconf产生configure配置文件：

```
[Crawl@localhost curl]$ ls
acinclude.m4  buildconf.bat  CMake      COPYING      docs      lib      MacOSX-Framework  maketgz
appveyor.yml  CHANGES      CMakeLists.txt  CTestConfig.cmake  GIT-INFO  libcurl.pc.in  Makefile.am      packages
buildconf     CHANGES.0    configure.ac  curl-config.in  include   m4           Makefile.dist    projects
[Crawl@localhost curl]$ vim buildconf
[Crawl@localhost curl]$ ./buildconf
buildconf: autoconf version 2.63 (ok)
buildconf: autom4te version 2.63 (ok)
buildconf: autoheader version 2.63 (ok)
buildconf: automake version 1.11.1 (ok)
buildconf: aclocal version 1.11.1 (ok)
```

- 3.执行产生的configure脚本：
./configure --enable-debug 在这里我只加了这个参数，为了后续的调试，如果还要其他参数，可以./configure --help查看其他可选参数。

注：我是默认安装openssl的，所以没有出现找不到openssl库的问题。如果遇到了就装一个，选择默认安装省事，自己指定安装目录比较麻烦，还要给configure多加个参数。具体查看工程目录下的README。

```
appveyor.yml  CHANGES      compile      configure.ac  depcomp      install-sh  m4
[Crawl@localhost curl]$ ./configure --enable-debug
checking whether to enable maintainer-specific portions of Makefiles... no
checking whether to enable debug build options... yes
checking whether to enable compiler optimizer... (assumed) no
checking whether to enable strict compiler warnings... yes
checking whether to enable compiler warnings as errors... no
checking whether to enable curl debug memory tracking (disabled by default) ... yes
checking whether to enable hiding of library internal symbols... yes
checking whether to enable c-ares for DNS lookups... no
checking whether to disable dependency on -lrt... (assumed no)
checking for path separator... :
checking for sed... /bin/sed
checking for grep... /bin/grep
```

- 4.make
- 360doc 个人图书馆 首页 阅览室 馆友 我的图书馆 搜文章 找馆友 登录 注册

```
[root@localhost curl]# ls /usr/local/lib
gcc      libcurl.a      libgomp.so      libmudfl
gcj-4.8.2-14  libcurl.la      libgomp.so.1    libmudfl
libasan.a      libcurl.so      libgomp.so.14   libmudfl
libasan.la      libcurl.so.4    libgomp.so.1.0.0 libmudfl
libasan.preinit.o  libcurl.so.4.4.0  libgomp.spec    libmudfl
libasan.so      libgcc_s.so     libgfortran.a   libmudfl
libasan.so.0    libgcc_s.so.1   libgfortran.la  libmudfl
libasan.so.0.0.0  libgcj_bc.so    libgfortran.so  libmudfl
libatomic.a      libgcj_bc.so.1  libgfortran.so.3  libitm.a
libatomic.la     libgcj_bc.so.1.0.0  libgfortran.so.3.0.0  libitm.la
libatomic.so     libgcj.la       libgij.so        libitm.so
libatomic.so.1   libgcj.so        libgij.so.14     libitm.so.1
libatomic.so.1.0.0  libgcj.so.14    libgij.so.14.0.0  libitm.spec
libbfd.a         libgcj.spec      libgomp.a         libmudflap.a
libbfd.la        libgcj.spec      libgomp.a         libmudflap.la
libopcodes.a     libopcodes.la    libopcodes.a      libopcodes.la
```

- 7.接下来写个测试代码来使用libcurl库(此测试代码下载指定URL的页面)：
测试代码如下：

[cpp] view plain copy print?

```

1. // 采用CURLOPT_WRITEFUNCTION 实现网页下载保存功能
2. #include <stdio.h>;
3. #include <stdlib.h>;
4. #include <unistd.h>;
5.
6. #include <curl/curl.h>;
7. #include <curl/types.h>;
8. #include <curl/easy.h>;
9.
10. FILE *fp; //定义FILE类型指针
11. //这个函数是为了符合CURLOPT_WRITEFUNCTION而构造的
12. //完成数据保存功能
13. size_t write_data(void *ptr, size_t size, size_t nmemb, void *stream)
14. {
15.     int written = fwrite(ptr, size, nmemb, (FILE *)fp);
16.     return written;
17. }
18.
19. int main(int argc, char *argv[])
20. {
21.     CURL *curl;
22.     if (argc != 3)
23.     {
24.         fprintf(stderr, "usage: %s url filename\n", argv[0]);
25.         exit(-1);
26.     }
27.     curl_global_init(CURL_GLOBAL_ALL);
28.     curl = curl_easy_init();
29.     curl_easy_setopt(curl, CURLOPT_URL, argv[1]);
30.
31.     if((fp = fopen(argv[2], "w")) == NULL)
32.     {
33.         curl_easy_cleanup(curl);
34.         exit(1);
35.     }
36.     //CURLOPT_WRITEFUNCTION 将后继的动作交给write_data函数处理
37.     curl_easy_setopt(curl, CURLOPT_WRITEFUNCTION, write_data);
38.     curl_easy_perform(curl);
39.     curl_easy_cleanup(curl);
40.     exit(0);
41. }

```

8.curl-config工具简介：

在安装完libcurl库后会同时安装了curl-config工具，这个工具专门用来查看已安装libcurl的一些信息。有很多参数：

`curl-config --libs` 查看我们的代码链接libcurl时需要哪些参数，这个选项我们用的比较多

`curl-config --version` 查看libcurl版本

还有更多参数，不一一列举了，`curl-config --help`查看更多参数。

9.编译链接我们的测试代码：

```

[root@localhost curl_demo]# curl-config --libs
-L/usr/local/lib -lcurl -lrt
[root@localhost curl_demo]# gcc -L/usr/local/lib -o xget get_file.c -lcurl
[root@localhost curl_demo]# ls http://blog.csdn.net/
get_file.c xget

```

此参数必须要加，要不然连接出错

10.编译完成，下载百度首页：

```

[root@localhost curl_demo]# ./xget http://www.baidu.com/ index.html
[root@localhost curl_demo]# ls
get_file.c index.html xget
[root@localhost curl_demo]# http://blog.csdn.net/
[root@localhost curl_demo]# |

```

总结：

通过源码编译安装程序，看README很重要，而不是盲目的按那老套路编译安装，很多时候按老套路是行不通的，其次，在我们遇到问题了可以直接把错误提示粘到Chrome浏览器搜索栏搜索，一般在英文论坛都有很好的解决方法，很多问题都可以在<http://stackoverflow.com>这个论坛找到，这个论坛确实不错。

上一篇：Huffman树及编码C 实现

下一篇：C语言URL解析器(代码分享)

猜你喜欢

类似文章	更多	精选文章
libcurl		年薪15万的80后小本科：只需要6分钟，告诉...
对CURL的一些研究 - ChinaUnix.net		月入百万的十三大终极秘诀
libcurl使用示例		赚钱是一件很容易的事,只要方法和观念正确
Linux php安装		中国传统色谱
libcurl API：CURLOPT的参数CURLOPT_REF...		《我怎样毁了我的一生》,字字刺人心
libcurl库介绍		怎样安排时间读书？
CURL详解		文秘助手：经常写大材料的请惠存
curl应用总结（一）		丸子汤的做法大全

发表评论：

请 [登录](#) 或者 [注册](#) 后再进行评论

社交帐号登录：