

发现生活之美

linux c++ curl https 请求并双向验证SSL证书

1、配置curl https请求需要提供 CA证书、客户端证书和客户端秘钥，这三个文件的pem格式。

分别对应 curl_easy_setopt() 函数的 下面三个参数：

CURLOPT_CAINFO - path to Certificate Authority (CA) bundle

CURLOPT_SSLKEY - specify private keyfile for TLS and SSL client cert

CURLOPT_SSLCERT - set SSL client certificate

一般创建SSL证书时会生成 ca.crt , client.crt, client.key, server.crt, server.key 等，而 curl客户端请求，只需要将 ca.crt , client.crt, client.key转成相应的 pem格式 使用。转换方法如下：

- 1) 将 CRT 转成 PEM---
- 不能直接将 .crt 转成 .pem , 需要经过 .der 中转

```
openssl x509 -in client.crt -out client.der -outform der
openssl x509 -in client.der -inform der -outform pem -out client.pem

openssl x509 -in ca.crt -out ca.der -outform der
openssl x509 -in ca.der -inform der -outform pem -out ca_info.pem
```

- 2) 将 .key 转成 .pem
- 不能直接将 .key 转成 .pem , 需要经过 .der 中转

```
openssl rsa -in client.key -out client.der -outform DER
openssl rsa -inform DER -outform PEM -in client.der -out client_key.pem
```

2、配置 curl https请求

- 1) 官方例程如下：
- curl 接口文档说明：
- https://curl.haxx.se/libcurl/c/curl_easy_setopt.html
- https://curl.haxx.se/libcurl/c/CURLOPT_CAINFO.html
- https://curl.haxx.se/libcurl/c/https.html
- CURLOPT_CAINFO - path to Certificate Authority (CA) bundle
- CURLOPT_SSLKEY - specify private keyfile for TLS and

2018年7月						
<	日	一	二	三	四	五
	24	25	26	27	28	29
	1	2	3	4	5	6
	8	9	10	11	12	13
	15	16	17	18	19	20
	22	23	24	25	26	27
	29	30	31	1	2	3

导航

[博客园](#)
[首页](#)
[新随笔](#)
[联系](#)
[订阅](#) XML
[管理](#)

统计

随笔 - 96
文章 - 0
评论 - 5
引用 - 0

公告

昵称：发现生活之美
园龄：1年4个月
粉丝：3
关注：3
+加关注

搜索

常用链接

[我的随笔](#)
[我的评论](#)
[我的参与](#)
[最新评论](#)
[我的标签](#)

随笔分类

C++ 学习(74)
curl c++ linux (3)
Qt study (1)

SSL client cert

CURLOPT_SSLCERT - set SSL client certificate

按下面代码部分进行配置，即可访问

[ROS\(2\)](#)[Ubuntu 相关\(11\)](#)[电脑问题\(2\)](#)[服务器网络相关\(3\)](#)

随笔档案

[2018年3月 \(1\)](#)[2018年2月 \(4\)](#)[2017年12月 \(21\)](#)[2017年11月 \(17\)](#)[2017年10月 \(20\)](#)[2017年9月 \(15\)](#)[2017年8月 \(16\)](#)[2017年4月 \(1\)](#)[2017年3月 \(1\)](#)

最新评论

阅读排行榜

评论排行榜

推荐排行榜



```
CURL *curl = curl_easy_init();
if(curl) {
    curl_easy_setopt(curl, CURLOPT_URL,
"https://example.com/");

    // 下面两个为验证对方和验证主机名，若为0，则跳过验证，我这个服
    务器必须验证才能得到请求数据
    curl_easy_setopt(pCurl, CURLOPT_SSL_VERIFYPEER, 1L);
    curl_easy_setopt(pCurl, CURLOPT_SSL_VERIFYHOST,
1L);

    // 配置 https 请求所需证书
    curl_easy_setopt(curl, CURLOPT_CAINFO,
"/etc/certs/cabundle.pem");
    curl_easy_setopt(curl, CURLOPT_SSLCERT,
"client.pem");
    curl_easy_setopt(curl, CURLOPT_SSLKEY, "key.pem");
    curl_easy_setopt(curl, CURLOPT_KEYPASSWD, "s3cret");

    ret = curl_easy_perform(curl);
    curl_easy_cleanup(curl);
}
```



2) 我的代码如下

这是一个完整的 curl 发送 https get 请求，并带中文参数



```
#include <iostream>
#include <sstream>
#include <jsoncpp/json/json.h>
#include <curl/curl.h>
#include <exception>
#include <string>
#include <iostream>
#include <stdlib.h>

int writer(char *data, size_t size, size_t nmemb, string
*writerData)
{
    unsigned long sizes = size * nmemb;
    if (writerData == NULL)
        return -1;

    writerData->append(data, sizes);
}
```

```
        return sizes;
    }

    string parseJsonResponse_question(string input)
    {
        Json::Value root;
        Json::Reader reader;
        bool parsingSuccessful = reader.parse(input, root);
        if(!parsingSuccessful)
        {
            std::cout<<"!!! Failed to parse the response data"<<
std::endl;
            return "";
        }
        const Json::Value text = root["obj"]["question"];
        string result = text.asString();
        return result;
    }

    string HttpsGetRequest_question(string input)
    {
        string buffer, ling_result;

        // 对请求参数中的中文和特殊字符（如空格等）进行处理，方可使用
        char * escape_control = curl_escape(input.c_str(),
input.size());
        input = escape_control;
        curl_free(escape_control);

        string str_url= "https://*.*.*.*question?question=" +
input; // alter *.*.*.* by your server address

        try
        {
            CURL *pCurl = NULL;
            CURLcode res;

            // In windows, this will init the winsock stuff
            curl_global_init(CURL_GLOBAL_ALL);
            // get a curl handle
            pCurl = curl_easy_init();
            if (NULL != pCurl)
            {
                // 设置超时时间为8秒
                curl_easy_setopt(pCurl, CURLOPT_TIMEOUT, 8);
                curl_easy_setopt(pCurl, CURLOPT_URL,
str_url.c_str());

                // 下面两个为验证对方和验证主机名，若为0，则跳过验证，我
这个服务器必须验证才能得到请求数据
                curl_easy_setopt(pCurl, CURLOPT_SSL_VERIFYPEER,
1L);
                curl_easy_setopt(pCurl, CURLOPT_SSL_VERIFYHOST,
1L);
```

```
        // 配置 https 请求所需证书

curl_easy_setopt(pCurl,CURLOPT_CAINFO,"/etc/msc/ca_info.pem"
);
        curl_easy_setopt(pCurl, CURLOPT_SSLCERT,
"/etc/msc/client.pem");
        curl_easy_setopt(pCurl, CURLOPT_SSLKEY,
"/etc/msc/client_key.pem");
        curl_easy_setopt(pCurl, CURLOPT_KEYPASSWD,
"your_key_password");

        curl_easy_setopt(pCurl, CURLOPT_WRITEFUNCTION,
writer);
        curl_easy_setopt(pCurl, CURLOPT_WRITEDATA,
&buffer);

        // Perform the request, res will get the return
code
        res = curl_easy_perform(pCurl);
        // Check for errors
        if (res != CURLE_OK)
        {
            printf("curl_easy_perform() failed:%s\n",
curl_easy_strerror(res));
        }
        curl_easy_cleanup(pCurl);
    }
    curl_global_cleanup();
}
catch (std::exception &ex)
{
    printf("curl exception %s.\n", ex.what());
}
if(buffer.empty())
{
    std::cout<< "!!! ERROR The sever response NULL" <<
std::endl;
}
else
{
    ling_result = parseJsonResponse_question(buffer);
}
return ling_result;
}
```



参考：

<http://blog.csdn.net/rztyfx/article/details/6919220>

<https://segmentfault.com/a/1190000011709784>

分类: [curl c++ linux](#)

« 上一篇: [linux curl http get 请求中带有中文参数或者特殊字符处理](#)

» 下一篇: [配置 ubuntu 14.04 shadowsocks客户端 , 使用 google chrome 翻墙](#)

posted on 2017-12-22 10:38 [发现生活之美](#) 阅读(1908) 评论(...) 编辑 收藏

抱歉! 发生了错误! 麻烦反馈至contact@cnblogs.com

[刷新评论](#) [刷新页面](#) [返回顶部](#)

【推荐】超50万VC++源码: 大型组态工控、电力仿真CAD与GIS源码库!

【推荐】如何快速搭建人工智能应用?

【大赛】2018首届“顶天立地”AI开发者大赛



最新IT新闻:

- [拼多多更新招股书：腾讯红杉拟各增持2.5亿美元](#)
 - [世界杯期间华帝股份大赚 退款后仍入袋近4亿元](#)
 - [易居中国港股IPO定价14.38港元 位于招股区间低端](#)
 - [密集爆雷潮下，P2P平台的自救之策](#)
 - [又一华尔街机构下调Netflix股票评级：海外市场竞争激烈](#)
- » [更多新闻...](#)



最新知识库文章:

- [危害程序员职业生涯的三大观念](#)
 - [断点单步跟踪是一种低效的调试方法](#)
 - [测试 | 让每一粒尘埃有的放矢](#)
 - [从Excel到微服务](#)
 - [如何提升你的能力？给年轻程序员的几条建议](#)
- » [更多知识库文章...](#)

Powered by:

[博客园](#)

Copyright © 发现生活之美

