烟雨平生

烟雨前辈

博客访问: 1828
博文数量: 2
博客积分: 0
博客等级: 民兵
技术积分: 20
用 户 组: 普通用户
注册时间: 2015-08-13 10:02


cloudhuh


章鱼小丸

OPENSSL 验证证书链

原创 分类: C/C++ 2015-08-14 11:15:13

步骤:

1)初始化环境

a.新建证书存储区X509_STORE_new()

b.新建证书校验上下文X509_STORE_CTX_new()

2)导入根证书

a.读取CA证书，从DER编码格式化为X509结构d2i_X509()

b.将CA证书导入证书存储区X509_STORE_add_cert()

3)导入要校验的证书test

a.读取证书test，从DER编码格式化为X509结构d2i_X509()

b.在证书校验上下文初始化证书test,X509_STORE_CTX_init()

c.校验X509_verify_cert

点击(此处)折叠或打开

```
#include <stdio.h>
#include <string.h>
#include <stdlib.h>

#include <openssl/evp.h>
#include <openssl/x509.h>
#include <openssl/pem.h>

#define CERT_PATH "/home/ycg/demoCA"
#define ROOTCA_CERT "rootca_cert.pem"
#define CLASS2CA_CERT "class2ca_cert.pem"
#define CLIENT_CERT "client_cert.pem"


#define GET_ROOT_CA_CERT(str) sprintf(str, "%s/%s", CERT_PATH, ROOTCA_CERT)
#define GET_CLASS2_CA_CERT(str) sprintf(str, "%s/%s", CERT_PATH, CLASS2CA_CERT)
#define GET_CLIENT_CERT(str, path, name) sprintf(str, "%s/%s", path, name)

#define MAX_LEGTH 4096

int my_load_cert(unsigned char *str, unsigned long *str_len,
                 const char *verify_cert, const unsigned int cert_len)
{
    FILE *fp;
    fp = fopen(verify_cert, "rb");
    if ( NULL == fp)
    {
        fprintf(stderr, "fopen fail\n");
        return -1;
    }

    *str_len = fread(str, 1, cert_len, fp);
    fclose(fp);
```

·【PyTorch基础教程7】多维特...
·旋转算法
·openstack注入文件到虚机源码...

相关博文

·人脸识别主板能应用哪些产品...
·整车控制单元(VCU)
·嵌入式开发与纯软件什么区别...
·启明云端分享:S系列1.54寸串...
·Rhapsody — MBSE 开发工具...
·IPD的前世今生
·香蕉派 BPI-M5单板计算机，采...
·自助售货机主板要注意哪几个...
·TAITherm—专业热管理工具...
·启明云端分享:ESP32-C3环境...

```
    return 0;
}

X509 *der_to_x509(const unsigned char *der_str, unsigned int der_str_len)
{
    X509 *x509;
    x509 = d2i_X509(NULL, &der_str, der_str_len);
    if ( NULL == x509 )
    {
        fprintf(stderr, "d2i_X509 fail\n");
        return NULL;
    }
    return x509;
}

X509 *pem_to_x509(const char *pem_file)
{
    X509 *x509;
    BIO *cert = NULL;
    if ((cert = BIO_new(BIO_s_file())) == NULL) {
        goto end;
    }

    if (BIO_read_filename(cert, pem_file) <= 0) {
        goto end;
    }

    x509 = PEM_read_bio_X509(cert, NULL,NULL, NULL);
    if ( NULL == x509 )
    {
        fprintf(stderr, "PEM_read_bio_X509_AUX fail\n");
        return NULL;
    }
    return x509;

    end:
    if (cert != NULL)
        BIO_free(cert);

    return NULL;
}

int x509_verify()
{
    int ret;
    char cert[MAX_LEGTH];

    X509 *user = NULL;
    X509 *rootca = NULL;
    X509 *class2ca = NULL;

    X509_STORE *ca_store = NULL;
    X509_STORE_CTX *ctx = NULL;
    STACK_OF(X509) *ca_stack = NULL;

    /* x509初始化 */
    ca_store = X509_STORE_new();
    ctx = X509_STORE_CTX_new();

    /* root ca*/
    GET_ROOT_CA_CERT(cert);
    rootca = pem_to_x509(cert);
    /* 加入证书存储区 */
    ret = X509_STORE_add_cert(ca_store, rootca);
    if ( ret != 1 )
    {
        fprintf(stderr, "X509_STORE_add_cert fail, ret = %d\n", ret);
        goto EXIT;
    }

    GET_CLASS2_CA_CERT(cert);
    class2ca = pem_to_x509(cert);

    /* 加入证书存储区 */
    ret = X509_STORE_add_cert(ca_store, class2ca);
    if ( ret != 1 )
    {
        fprintf(stderr, "X509_STORE_add_cert fail, ret = %d\n", ret);
        goto EXIT;
    }

    /* 需要校验的证书 */
    GET_CLIENT_CERT(cert, CERT_PATH, CLIENT_CERT);
    user = pem_to_x509(cert);

    ret = X509_STORE_CTX_init(ctx, ca_store, user, ca_stack);
    if ( ret != 1 )
    {
        fprintf(stderr, "X509_STORE_CTX_init fail, ret = %d\n", ret);
        goto EXIT;
```

```
    }

    //openssl-1.0.1c/crypto/x509/x509_vfy.h
    ret = X509_verify_cert(ctx);
    if ( ret != 1 )
    {
        fprintf(stderr, "X509_verify_cert fail, ret = %d, error id = %d, %s\n",
                ret, ctx->error, X509_verify_cert_error_string(ctx->error));
        goto EXIT;
    }
EXIT:
    X509_free(rootca);
    X509_free(class2ca);
    X509_free(user);

    X509_STORE_CTX_cleanup(ctx);
    X509_STORE_CTX_free(ctx);

    X509_STORE_free(ca_store);

    return ret == 1 ? 0 : -1;
}

int main()
{
    OpenSSL_add_all_algorithms();
    x509_verify();
    return 0;
}
```


上一篇:Linux的nm查看动态和静态库中的符号
下一篇:没有了