正确使用AFNetworking的SSL保证网络安全

Reference: https://www.cainwang.cn/afnssl/

AFNetworking, iOS开发中，以其优雅的结构设计和简便的调用方式，使其成为了最流行的网络开源库之一（另一个应该算是ASI了，但经久失修不维护的原因，已经不是首选）。

我们在大多数情况下，都能够正确使用AFNetworking的功能，但在网络安全日趋严峻的今天，加入SSL使用HTTPS已经成为了很多大中型网站的首选；这点在国外尤其流行，例如Google已经全站HTTPS。

本文便主要描述了如何正确使用AFNetworking中的SSL功能。详细步骤如下：

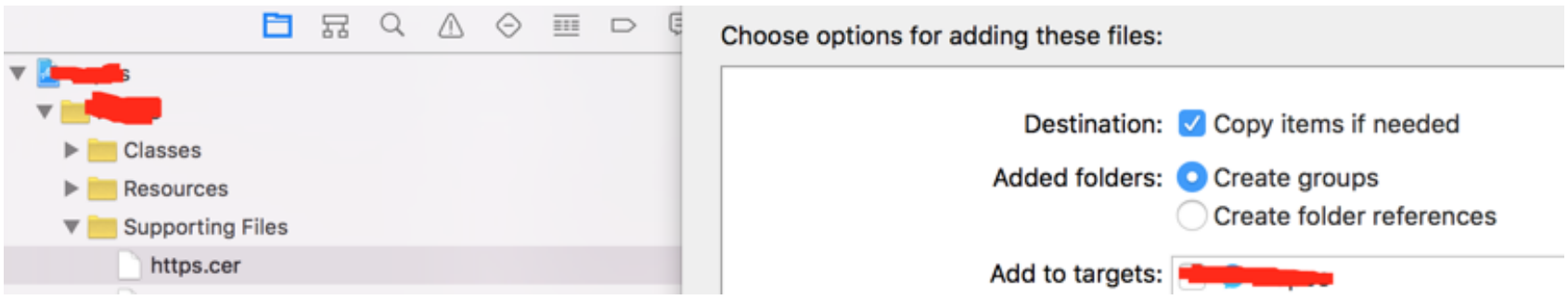
1、获取到站点的证书：

我们可以使用以下openssl命令来获取到服务器的公开二进制证书（以google为例）：

```
"openssl s_client -connect www.google.com:443 </dev/null 2>/dev/null | open
```

冒号中的为命令主要部分。该条命令将会在当前路径下，形成google.com站点的公开二进制证书，命名为https.cer。您可以将www.google.com 替换成您自己的站点以此来获取您自己站点的https.cer。

2、将证书放进我们的XCode项目工程中：



Paste_Image.png

如上图所示，将我们的https.cer拖到我们的工程Supporting Files中，把 Copy Items if needed 的勾选上。然后把您的Add to targets 选上，点击确定。就完成了证书的导入工作。

3、在我们的代码中使用我们的cer

AFNetworking中的AFSecurityPolicy是主要的类，我们可以这样来使用它(AFNetworking 2.6.0之前)：

```
AFHTTPRequestOperationManager *manager = [AFHTTPRequestOperationManager

NSString *cerPath                = [[NSBundle mainBundle] pathForResour
NSData *certData                 = [NSData dataWithContentsOfFile:cerPa
AFSecurityPolicy *securityPolicy = [[AFSecurityPolicy alloc] init];
[securityPolicy setAllowInvalidCertificates:NO];
[securityPolicy setPinnedCertificates:@[certData]];
[securityPolicy setSSLPinningMode:AFSSLPinningModeCertificate];
[securityPolicy setValidatesDomainName:YES];
[securityPolicy setValidatesCertificateChain:NO];

manager.securityPolicy = securityPolicy;
```

解析：

1）新建一个manager, 地球人都知道

2）在mainBundle中寻找我们刚才拖进项目中的https.cer, 并且将相关的数据读取出来

3）新建一个AFSecurityPolicy，并进行相应的配置

4）将这个AFSecurityPolicy 实例赋值给manager

也可以这样来使用：

```
AFHTTPRequestOperationManager *manager = [AFHTTPRequestOperationManager man

AFSecurityPolicy *securityPolicy = [[AFSecurityPolicy alloc] init];
[securityPolicy setAllowInvalidCertificates:NO];
[securityPolicy setSSLPinningMode:AFSSLPinningModeCertificate];
[securityPolicy setValidatesDomainName:YES];
[securityPolicy setValidatesCertificateChain:NO];

manager.securityPolicy = securityPolicy;
```

这种方式比前面那种方式要更加简便一些，主要原因在于AFNetworking会自动去搜索mainBundle下的所有cer结尾的文件并放进内存中；再一一对比。因此在代码中可以省略不写。

这样一个网络请求的https的安全策略就配置好了，接下来再说明一下几个AFSecurityPolicy相关的配置

1> SSLPinningMode

SSLPinningMode 定义了https连接时，如何去校验服务器端给予的证书。

```
typedef NS_ENUM(NSUInteger, AFSSLPinningMode) {
    AFSSLPinningModeNone,
    AFSSLPinningModePublicKey,
    AFSSLPinningModeCertificate,
};
```

AFSSLPinningModeNone: 代表客户端无条件地信任服务器端返回的证书。

AFSSLPinningModePublicKey: 代表客户端会将服务器端返回的证书与本地保存的证书中，PublicKey的部分进行校验；如果正确，才继续进行。

AFSSLPinningModeCertificate: 代表客户端会将服务器端返回的证书和本地保存的证书中的所有内容，包括PublicKey和证书部分，全部进行校验；如果正确，才继续进行。

2> allowInvalidCertificates

allowInvalidCertificates 定义了客户端是否信任非法证书。一般来说，每个版本的iOS设备中，都会包含一些既有的CA根证书。如果接收到的证书是iOS信任的CA根证书签名的，那么则为合法证书；否则则为“非法”证书。

allowInvalidCertificates 就是用来确认是否信任这样的证书的。当然，我们也可以给iOS加入新的信任的CA证书。iOS已有的CA根证书，可以在这里了解到：https://support.apple.com/en-us/HT204132

3> pinnedCertificates

pinnedCertificates 就是用来校验服务器返回证书的证书。通常都保存在mainBundle 下。通常默认情况下，AFNetworking会自动寻找在mainBundle的根目录下所有的.cer文件并保存在pinnedCertificates数组里，以校验服务器返回的证书。

4> validatesDomainName

validatesDomainName 是指是否校验在证书中的domain这一个字

段。每个证书都会包含一个DomainName, 它可以是一个IP地址，一个域名或者一端带有通配符的域名。如*.google.com, www.google.com 都可以成为这个证书的DomainName。设置validatesDomainName=YES将严格地保证其安全性。

5> validatesCertificateChain

validatesCertificateChain 指的是是否校验其证书链。

通常来讲，一个CA证书颁发机构有很多个子机构，用来签发不同用途的子证书，然后这些子证书又再用来签发相应的证书。只有证书链上的证书都正确，CertificateChain才算验证完成。以Google为例：

从上图可以看到，Google.com的证书的根CA证书是GeoTrust Global CA; 而CA并没有直接给google.com签证书，而是先签名了Google Internet Authority G2, 然后G2再签名了google.com。这时候就需要设备中保存有Google Internet Authority G2证书才能通过校验。

一般来讲，我推荐将validatesCertificateChain设置为NO，因为并不是太有必要做CertificateChain的校验。并且，在AFNetworking 2.6.0中，也正式将validatesCertificateChain拿掉了（https://github.com/AFNetworking/AFNetworking/blob/master/CHANGELOG.md），其原因也同样为：There was no documented security advantage to pinning against an entire certificate chain。

因此，在2.6.0之后，可以不管这个字段。而在此之前，从效率上来说，设定为NO会是个比较明智的选择。

做好以上工作后，您应该就可以正常访问您自己的https服务器了。如果还是有问题请检查：

（1）、HTTPS服务器的正确配置。一般来说，可以使用浏览器打开相同页面来查看浏览器上的小锁是否正常。

（2）、是否https.cer正确打包进了项目中。查看第2步中的内容。

（3）、其他。跟帖呗。有问题大家一起交流，共同进步：）

这几个东西别看就这么一点，理解了好久才理清楚其中代表的含义。接下来打算再写几篇关于SSL的详细博文以纪念这些深度研究的时间。

参考文档：

1、https://github.com/AFNetworking/AFNetworking/blob/master/CHANG

ELOG.md

2、http://nelson.logdown.com/posts/2015/04/29/how-to-properly-setup-afnetworking-security-connection/