

# XCA 密钥及证书管理工具的使用

## 一、ssl 证书的查看

ssl 证书的主要用途是身份认证和数据加密，在 https 里常用于网站服务端的身份认证。在浏览器地址栏左侧可看到当前网站的 ssl 证书相关信息，显示“已安全连接到此网站”则说明此网站的 ssl 证书通过了验证，浏览器认为它是可信的。



可以在浏览器里查看网站的证书:

百度一下, 你就知道 x baidu.com 的证书 x +

Firefox about:certificate?cert=MIKLjCCCRagAwIBAgIMclh4Nm6fVugdQYhIMA0GCSqGSIb3DQEBCwUAMGYxCzAJBgNVBAYTAKJMRkwF

n-Su... P 期 - Unicode codep... 在线字体转换, 字体格... IP/IPv6查询, 服务器... MSDN, 我告诉你 Windows and Linux ... Windows Server

## 证书

baidu.com	GlobalSign Organization Validation CA - SHA256 - G2	GlobalSign
-----------	---	------------

**证书使用者信息**

主题名称  
 国家/地区 CN  
 州/省 beijing  
 地市 beijing  
 组织单位 service operation department  
 组织 Beijing Baidu Netcom Science Technology Co., Ltd  
 通用名称 baidu.com

颁发者名称  
 国家/地区 BE  
 组织 GlobalSign nv-sa  
 通用名称 GlobalSign Organization Validation CA - SHA256 - G2

有效性  
 起始时间 2020/4/2 下午3:04:58 (Asia/Shanghai)  
 终止时间 2021/7/26 下午1:31:02 (Asia/Shanghai)

主题替代名称  
 DNS 名称 baidu.com  
 DNS 名称 baifubao.com  
 DNS 名称 www.baidu.cn

百度一下, 你就知道 x baidu.com 的证书 x +

Firefox about:certificate?cert=MIKLjCCCRagAwIBAgIMclh4Nm6fVugdQYhIMA0

指数 65537

模块 C1:A9:B0:AE:47:1A:D2:57:EB:1D:15:1F:6E:5C

杂项

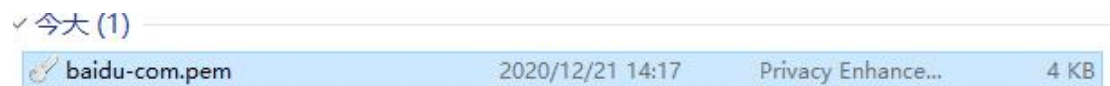
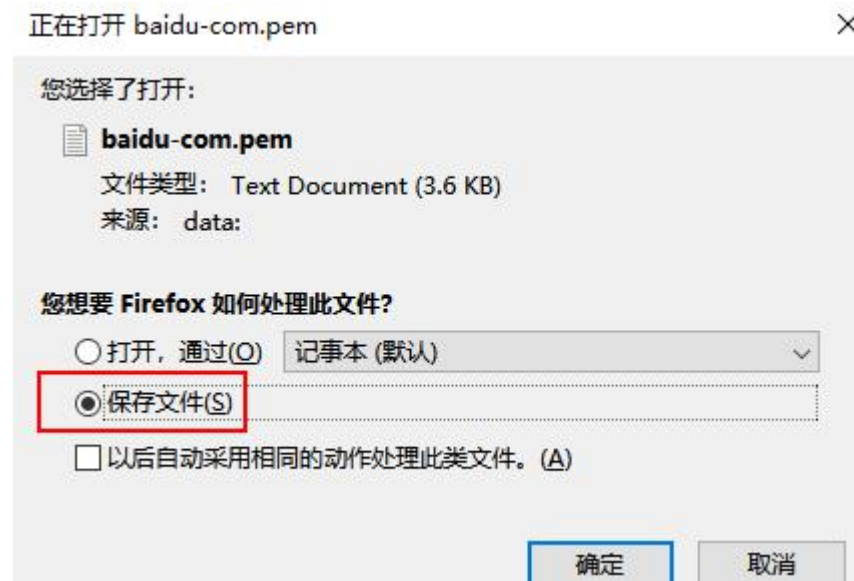
序列号 72:58:78:36:6E:9F:56:E8:1D:41:88:48

签名算法 SHA-256 with RSA Encryption

版本 3

下载 [PEM \(证书\)](#) [PEM \(证书链\)](#)

也可以下载证书到本地,



这个 ssl 证书文件本身不大，大概 2 到 6KB 左右，证书内容本身是二进制的，有时为了方便复制粘贴，也可转为 base64 编码。下图为 baidu.com 网站 ssl 证书的内容（base64 编码格式）



内容以“-----BEGIN CERTIFICATE-----”开始，以“-----END CERTIFICATE-----”结束，中间是正式内容的 base64 编码。

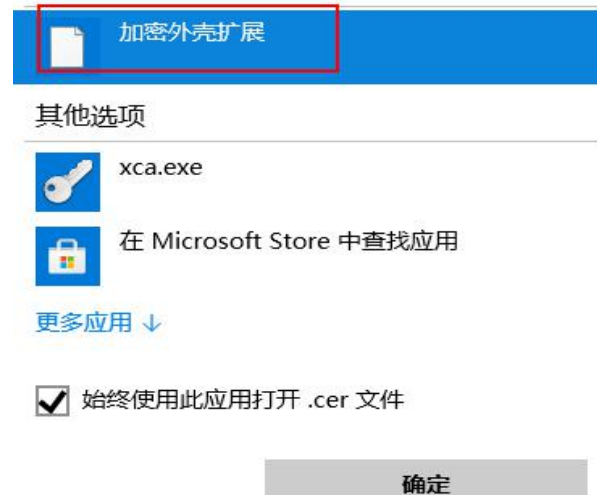
从浏览器上下载得到的证书一般为 x.509 标准的证书，默认以.pem 为后缀，可改为.cer，



然后在 windows 系统上可直接双击此证书文件 查看证书的详细信息，如果已经安装了 xca 工具，则可能无法直接双击查看了，这时使用“加密外壳扩展”打开

你要如何打开这个文件？

继续使用此应用





常规 详细信息 证书路径

显示(S): <所有>

字段	值
签名哈希算法	sha256
颁发者	GlobalSign Organization ...
有效期从	2020年4月2日 15:04:58
到	2021年7月26日 13:31:02
使用者	baidu.com, Beijing Baidu ...
公钥	RSA (2048 Bits)
公钥参数	05 00
授权信息访问	[1]Authority Info Access: ...
证书策略	[1]Certificate Policies: Policy: ...

CN = baidu.com  
 O = Beijing Baidu Netcom Science Technology Co., Ltd  
 OU = service operation department  
 L = beijing  
 S = beijing  
 C = CN

常规 详细信息 **证书路径**

证书路径(P)

- GlobalSign Root CA
  - GlobalSign Organization Validation CA - SHA256 - G2
    - baidu.com**

## 二、ssl 证书内容及验证原理

### ① ssl 证书里主要包含的内容有：

版本号（version）：v3  
证书序列号（serial number）：xxxxxx  
签发者（issuer）：某 ca 机构  
签名算法（signature algorithm）：sha256Rsa  
有效期始（valid from）：生效时间  
有效期至（valid to）：截止时间  
主题（subject）：证书拥有者/使用者的相关信息  
公钥（public key）：rsa 公钥  
.....

在“主题/主体”里一共有 7 个字段，用以表明使用者的身份信息



countryName（C）：国家或地区，只能是 2 个大写字母

stage or province（S）：省/州

locality（L）：市

organizationUnit（OU）：组织/公司的 某单位/部门，如 it-dept（IT 部）

organizatoin（O）：组织名/公司名称

commonName（CN）：通用名称，一般为网站的域名或组织名，必需字段

email address（E）：邮件地址，可选字段

（括号里的为字段的缩写）

在签发者信息里也有上面的 7 个字段，用以表明签发者的身份信息。

当浏览器收到服务器发来的证书后，会对此证书进行验证，

②ssl 证书验证流程是：

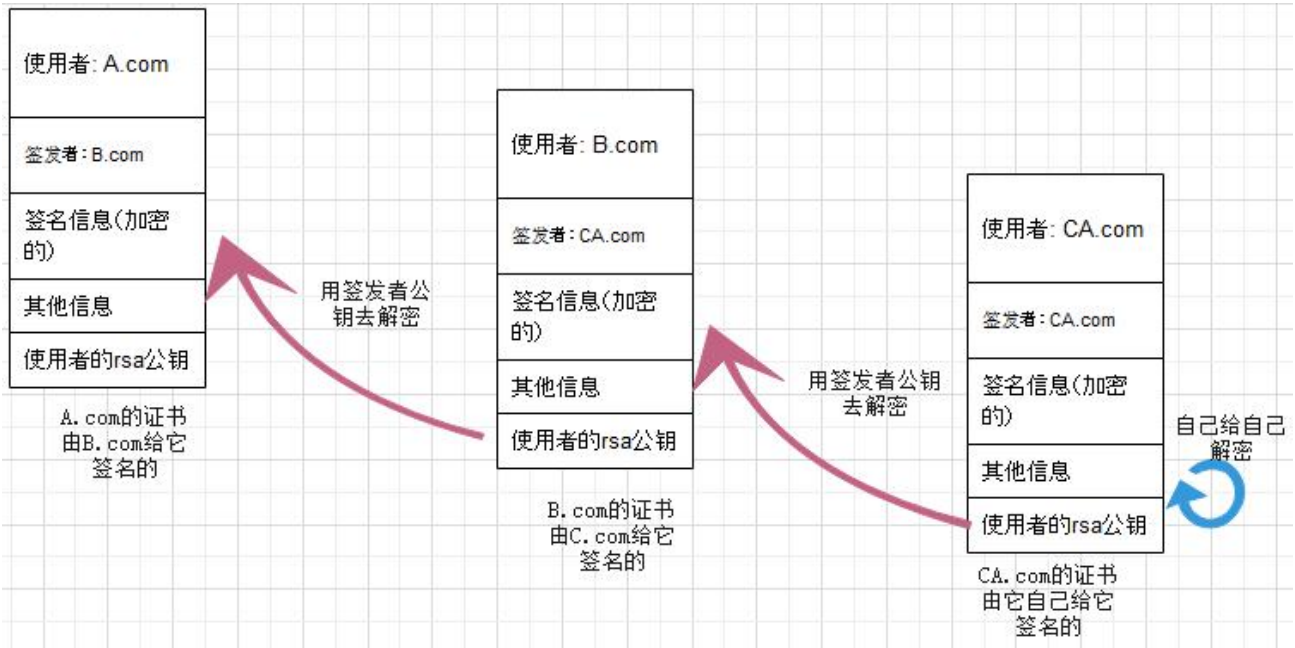
- 1.查看网站证书的 通用名称或扩展字段 SAN 里的域名是否和网站的域名相同，相同则说明此证书是给这个网站用的
- 2.查看证书里的签发者名称，再去找签发者的证书，签发者的证书可在系统预安装的证书里查找，也可由浏览器去它自己官网找，找到后，再用签发者的 rsa 公钥去解开网站的 ssl 证书里的签名信息，解开后再和网站的证书一比对，对得上就说明 此网站的 ssl 证书确实由此签发者签名的。
- 3.对签发者的证书也如上步骤进行验证，直到最终的签发者为可信的 CA，即最终的签发者证书为可信的根证书，一般预安装在操作系统里。

根证书的签名信息是由自己进行签名生成的。（是自签名证书）即签发者和使用者为同一组织。

**签名：**B 给 A 签名就是指 B 使用自己的 rsa 私钥去给 A 的相关身份信息的 hash 值进行加密，再把加密后的信息放入 A 的证书里

**验证：**要验证 A 的证书是否为 B 所签名，需要用到 B 的证书里的公钥，用 B 的公钥去给 A 的证书里的签名信息进行解密，解密后再和 A 的相关身份信息的 hash 值比对，一致则说明 A 证书确实是由 B 签名的，即 A 证书可信，要验证 B 证书的可信度，也是同样的道理。

验证时，最顶级的证书一定是自签名的，且为受信任的根证书，验证时，从底层证书一直到最高层都要验证通过才能说明最底层证书可信。



这一串证书构成了一条信任链，称为**证书链**

可把要用到的证书内容都放在一起，直接用文本处理工具把它们的 base64 编码内容都复制在一个文件里，这个新的文件就是证书链文件，后缀也可为.pem 或.crt

st.com.csr.pem baidu-com-chain.pem

-----BEGIN CERTIFICATE-----  
MIIKLjCCCRagAwIBAgIMclh4Nm6fVugdQYhIMA0GCSqGSIb3DQEBCwUAMGYxC  
-----END CERTIFICATE-----

-----BEGIN CERTIFICATE-----  
MIIEYjCCA0qgAwIBAgILBAAAAAABMYnGRMkwDQYJKoZIhvcNAQELBQAwTDEg  
-----END CERTIFICATE-----

-----BEGIN CERTIFICATE-----  
MIIDXzCCAkegAwIBAgILBAAAAAABIVhTCKIwDQYJKoZIhvcNAQELBQAwTDEgM  
-----END CERTIFICATE-----

证书1

中间证书

顶级证书



### 三、ssl 证书安全级别

ssl 证书根据受信任的程序可分为 3 种级别：域名型、机构验证型、严格身份验证型  
区别如下表：

ssl 证书级别类型	<b>DV 域名型</b> Domain Validation	<b>OV 机构验证型</b> Organization Valid	<b>EV 严格身份验证型</b> Extended Validation
商业上的称呼	超快 SSL 证书	超真 SSL 证书	超安 SSL 证书
证书用途	个人站点, 简单的 Https 加密需要	中小企业, 电子商务站点	大型金融平台, 政府机构站点
审核内容	域名所有权验证	域名所有权验证及企业身份信息	最高等级的企业身份信息验证和域名所有权验证
证书颁发需要时长	10 分钟至 1 小时	2 至 5 个工作日	2 至 5 个工作日
首次申请年限	1 年	1 至 2 年	1 至 2 年
价格参考 (2020 年)	1k 或免费	5k	12k
证书内容差异	主体信息里仅含有 CN 通用名称(一般为域名)	主体信息里含有更多的信息, 可以 7 个字段都有 (域名及公司名称等)	主体信息和 OV 证书一样多, 更多的体现在证书的扩展字段里

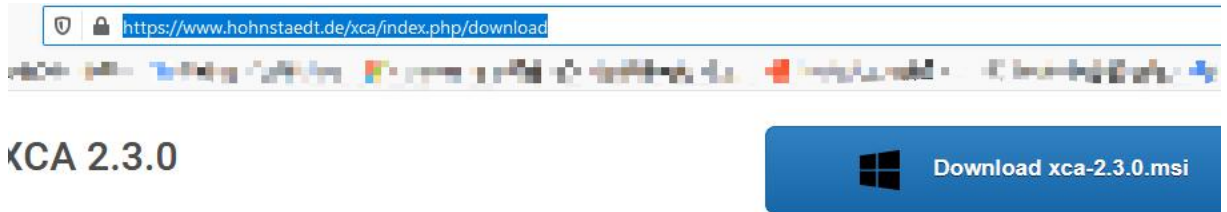
这三种证书有什么本质的区别吗?  
没有

## 四、向 ca 机构申请自己的证书

### ①安装 XCA 工具

首先 ssl 证书是基于 rsa 加密算法的，得先生成 rsa 密钥，本例使用 XCA 密钥及证书管理工具

下载地址：<https://www.hohnstaedt.de/xca/index.php/download>



Download	File size		
 <a href="#">xca-2.3.0.dmg</a>	24.03 MB	4643	798dcad616837b33ad7a92f6f62a7afba3d9eb049ad26eccc
 <a href="#">xca-2.3.0.msi</a>	17.80 MB	17684	a3d2295af4720455f20c366bd5c2c5a50625abab97b020961
 <a href="#">xca-2.3.0.tar.gz</a>	1.29 MB	3087	3d168a225efaf9b2213e9ad2ba0abbccfcad139181dd2be66

选择一个较新版本，如果是 windows 系统就下载.msi 后缀的安装包，下载后双击安装



安装完成后，双击运行

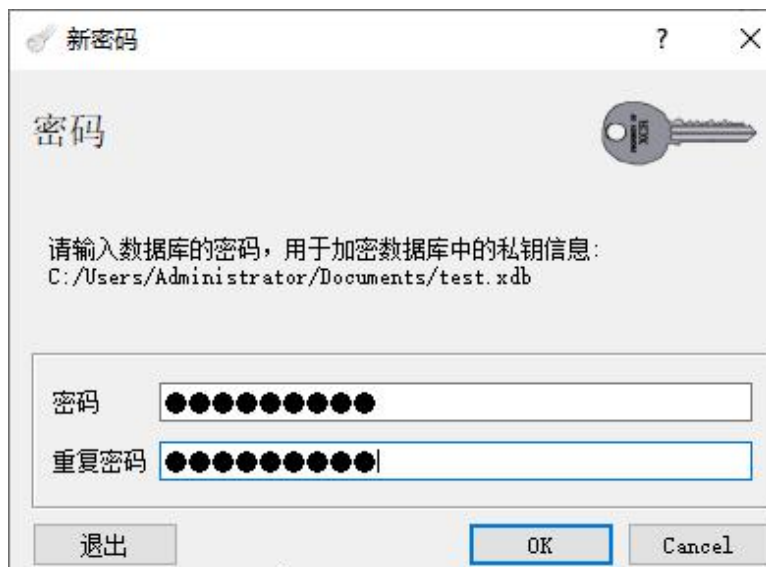


### ②新建数据库用以保存密钥及证书等文件

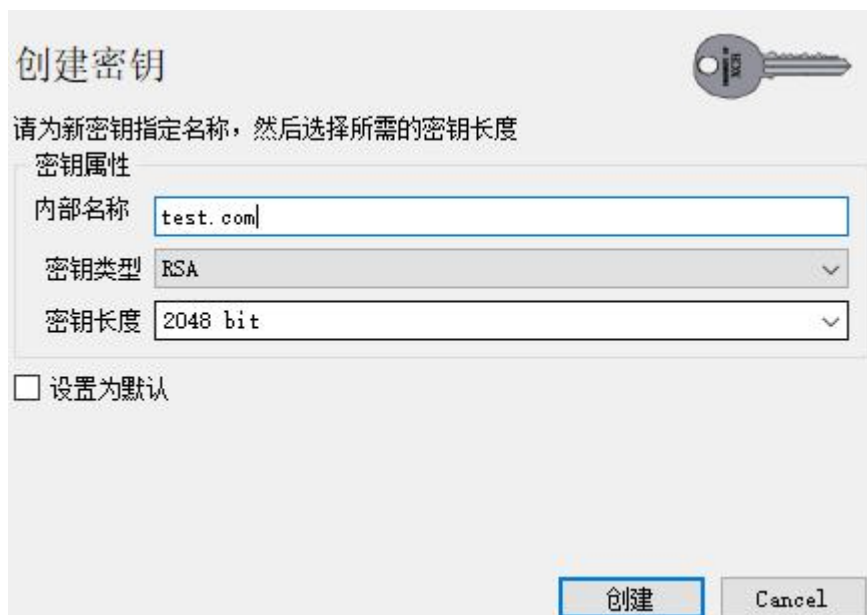
在主界面上，点击左上角菜单栏的“文件”→“新建数据库”

选择保存到某个目录下，名为 test.xdb

确定后，要求输入数据库的密码



③在此数据库里创建 **rsa** 密钥，点击主界面左上角的“私钥”，再点击右边的“创建密钥”



内部名称可随便写，比如就以自己的网站域名为名，其他参数自己设定

点击“创建”后就 ok 了

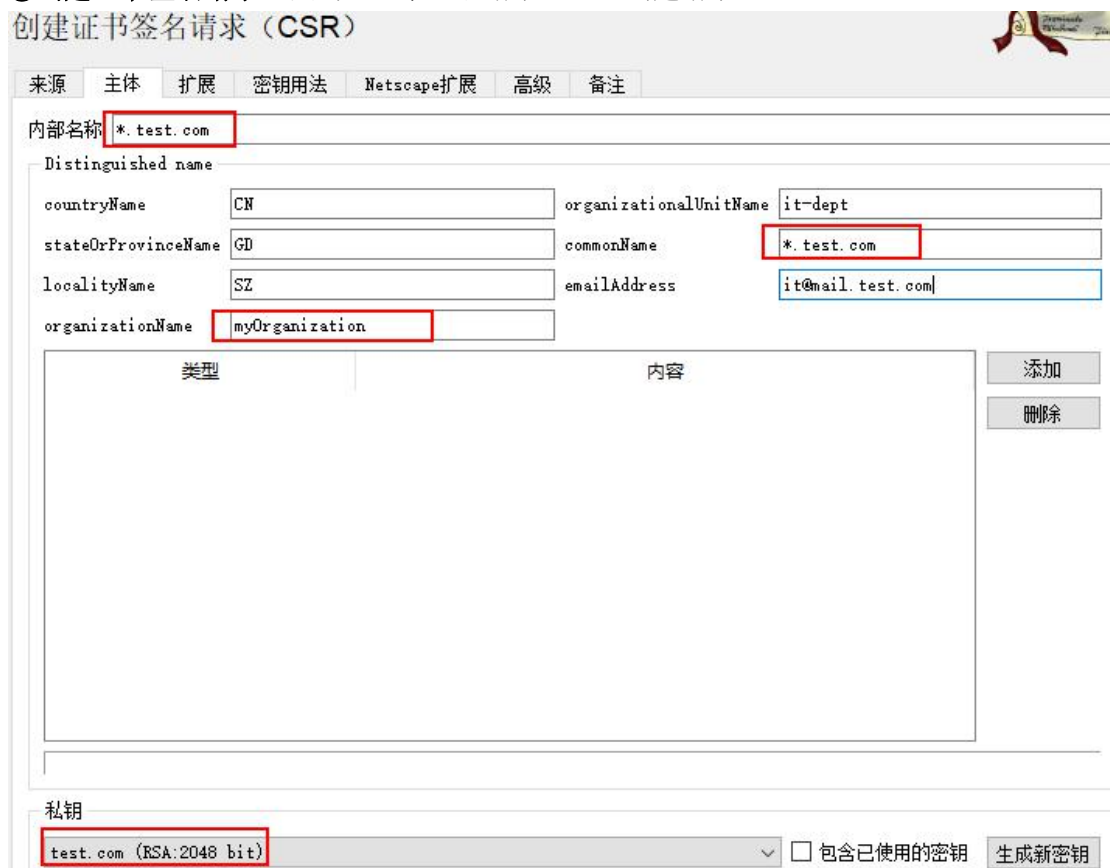


然后在主界面的“私钥”框里，就有了刚刚创建的密钥对，rsa 密钥是一对一地的，虽然在这个 xca 软件里显示的是私钥，但私钥里也是包含有公钥的。（私钥文件里一定包含有公钥）



④创建证书签名请求，点击“证书签名请求”→“创建请求”

创建证书签名请求（CSR）



在主体里，内部名称也可随便写，一般写域名，本次我们要申请一个泛域名证书，所以就写 \*.test.com，其他如国家/省/市就写大写字母简称，O 组织名称为 公司名称，OU 组织单位

一般写 IT-dept, cN 通用名称一般写域名或泛域名\*.test.com, 邮件可写/可不写  
再确认此签名请求文件使用的 rsa 密钥为要使用的那个密钥。(密钥名称和域名无必然联系)

### 创建证书签名请求 (CSR)

来源 主体 扩展 密钥用法 Netscape扩展 高级 备注

X509v3 Basic Constraints

类型 最终实体

CA路径长度

☐ Critical

Key identifier

☐ X509v3 Subject Key Identifier

☐ X509v3 Authority Key Identifier

有效期

不早于 2020-12-21 07:46 GMT

不晚于 2021-12-21 07:46 GMT

时间跨度

1 年 应用

☐ UTC午夜时间 ☐ 当地时间 ☐ 未明确定义到期日

X509v3 Subject Alternative Name

编辑

X509v3 Issuer Alternative Name

编辑

X509v3 CRL Distribution Points

编辑

然后在“扩展”框里,“类型”选择为“最终实体”,现代的证书一般都需要有 x.509v3 的扩展字段 SAN (Subject Alternative Name) 使用者可选名称:  
点击“x509v3SAN 右边的编辑”

### X Certificate and Key management

☐ Critical ☐ 复制通用名称 (CN)

	类型	内容
0	DNS	*.test.com
1	DNS	test.com

添加  
删除

应用 验证 取消

添加 2 条记录,第 0 条一定要和 cN 通用名称一致,即必须为\*.test.com  
第 1 条则写 test.com,还可有其他的,一般不加了,点击“应用”



### 创建证书签名请求 (CSR)

来源 主体 扩展 密钥用法 Netscape扩展 高级 备注

X509v3 Basic Constraints

类型 最终实体
CA路径长度

☐ Critical

Key identifier

☐ X509v3 Subject Key Identifier
☐ X509v3 Authority Key Identifier

有效期

不早于 2020-12-21 07:46 GMT
不晚于 2021-12-21 07:46 GMT

时间跨度

1 年 应用
☐ UTC午夜时间 ☐ 当地时间 ☐ 未明确定义到期日

X509v3 Subject Alternative Name ☒ DNS:\*.test.com, DNS:test.com 编辑

X509v3 Issuer Alternative Name 编辑

X509v3 CRL Distribution Points 编辑

最后添加证书的用途信息：

### 创建证书签名请求 (CSR)

来源 主体 扩展 密钥用法 Netscape扩展 高级 备注

X509v3 Key Usage

☐ Critical

Digital Signature
Non Repudiation
Key Encipherment
Data Encipherment
Key Agreement
Certificate Sign
CRL Sign

X509v3 Extended Key Usage

☐ Critical

TLS Web Server Authentication
TLS Web Client Authentication
Code Signing
E-mail Protection
Time Stamping
Microcode Signing

点击“官钥用法”，左边基础用法点击 Digital Signature 和 Key Encipherment 这 2 个就行了

### 创建证书签名请求 (CSR)

来源 主体 扩展 密钥用法 Netscape扩展 高级 备注

X509v3 Key Usage

☐ Critical

Digital Signature
Non Repudiation
Key Encipherment
Data Encipherment
Key Agreement
Certificate Sign
CRL Sign

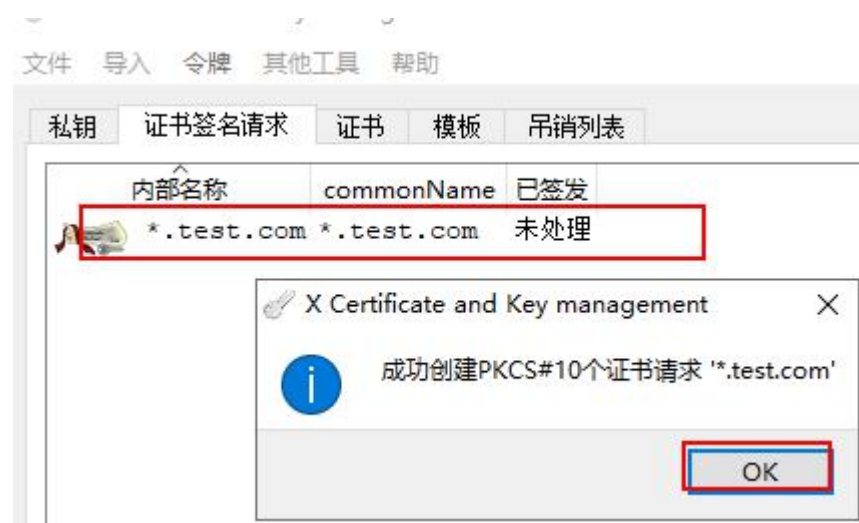
X509v3 Extended Key Usage

☐ Critical

TLS Web Server Authentication
TLS Web Client Authentication
Code Signing
E-mail Protection
Time Stamping
Microcode Signing

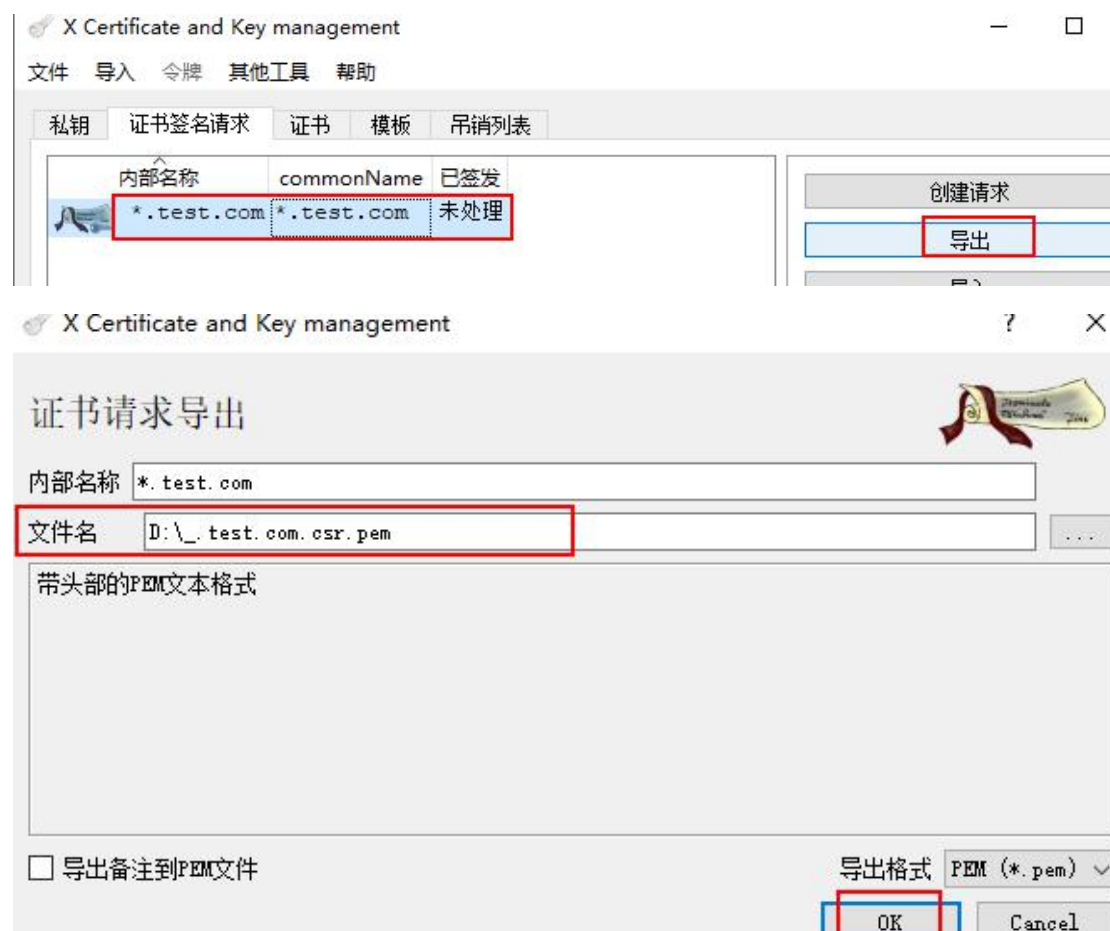
右边扩展用法点击 TLS web SERVER Authentication 和 CLIENT Authentication 就可以了

最后确认所有信息无误后，点击最右下方的 OK 就行了，



提示创建成功，在“证书签名请求”框里多了一个\*.test.com 的证书签名请求

#### ⑤接下来要导出此请求文件



导出到 D 盘，文件可随便取，一般为\_test.com.csr.pem 之类的名字，也可直接以.csr 为后缀

```

1 -----BEGIN CERTIFICATE REQUEST-----
2 MIIDNzCCA8CAQAwYgxCzAJBgNVBAYTAkNOMQswCQYDVQQLIEwJHRDELMAkGA1UE
3 BxMCU1oxFzAVBgNVBAoTDm15T3JnYW5pemF0aW9uMRAwDgYDVQQLewdpdC1kZXB0
4 MRMwEQYDVQDDAoqLnRlc3QuY29tMR8wHQYJKoZIhvcNAQkBFhBpdEBtYWlsLnRl
5 c3QuY29tMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAvmdlJtl7JSev
6 Q53CZUNB6ixC4iAmbezogiR2Af7sg0vo770/U7vORodQjXqGJE8bP543TNiaO4YI
7 cEBYSk3dagooNhEj+3pU3ljYy72QAa25N9QK5faMreTH9UTFP0a3MZ2AhvU2Udbv
8 MHHB/JIPIbd5hOUPYPizKTcyyPyS25mbJwqcKjSzAsmfQrbydKYuHNlYVOcl4Y8O
9 Lnt3epq8EoZIt8dqmGiH+nonXPgacgRRLj5CeYVIqXkiHN6od+i6tZZN6Qi2MoV1
10 98E1vKF8/9VgPE8c2byOl8vQjs2rUGW4M0jEat+Ll6RFhrhU1E1UJ+mB+mgfJd1S
11 0J9cOrIiRwIDAQABoGkwZwYJKoZIhvcNAQkOMVowWDAJBGNVHRMEAjAAMAsGA1Ud
12 DwQEAwIFoDAdBgNVHSUEFjAUBggrBgEFBQcDAQYIKwYBBQUHAWIwHwYDVROBBGw
13 FoIKKi50ZXN0LmNvbYIldGVzdC5jb20wDQYJKoZIhvcNAQELBQADggEBAES0Cg96
14 RmOMNRWrSTZ7NYUuVcf4tpBSIP9FaUT72xEealG4LUnqObES4M4FI0+H+c6iuuiS
15 BQU8i07RrDX6hdSV/mEBRAKZT9hMwR/gL6ausGZQ/TQA9bH1B1saCN9+PMP6KW4/
16 dlrGZ64cmPGinOBK2PINKD8clCeiYzX2ZIFAcP5VT5YSqvQQfjvRQNXe7ZBHX9r0
17 Rw2ntszHle02Y6nQQ/LY+LRCWDHW3jtTlt/3Lfj8GjZZr6a4xE8uHR1fiQLqfIYG
18 4juXDpyD9IvfIb8mwX2jUi8aHKmdHWB69+GddtRw++DZzlgAFKv/rumtEyxKY7Sd
19 DZvnKXJ7BAD8Boo=
20 -----END CERTIFICATE REQUEST-----
21

```

证书签名请求文件内容也是 base64 编码的

⑥之后可把此文件交给**证书签发商**的工作人员或在他们的 **ssl 证书申请网站**上复制进去就行，等待验证通过后，就会收到他们发来的已签名了的 ssl 证书。

我们收到的 ssl 证书一般有多种格式的，常见的是 x.509 编码的，后缀为.cer 或.crt 或.pem 之类的。

## 五、转换 ssl 证书格式

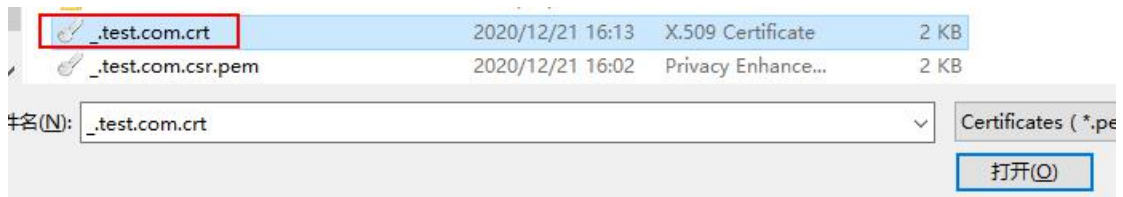
我们向 ca 机构（Certificate Authority）提交证书签名请求文件后，他们会验证我们的目标域名及公司身份等信息，验证通过后会得到有签名的 ssl 证书，不过此证书一般为 x.509 的编码格式，后缀为.cer 或.crt 或.pem，而我们的 web 服务器可能要其他的格式的证书

可用 XCA 工具进行转换，千万**不要在网上在线转换**，因为转换时可能要用到 rsa 的私钥，如果私钥给了第三方网站，则此 ssl 证书可以算是作废了。

### ①导入收到的已签名的 ssl 证书



点击“证书”→“导入”



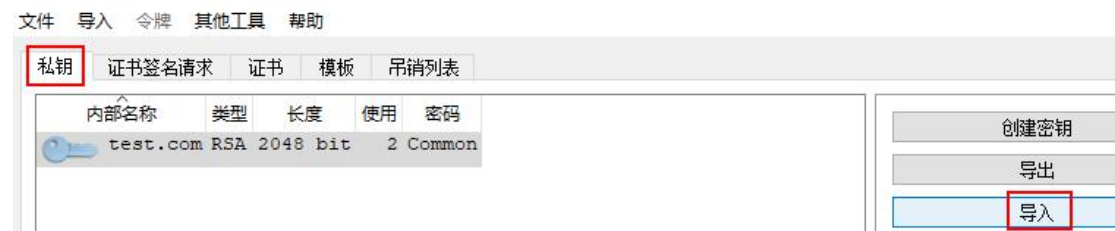
选择目标证书，点击“打开”，导入成功后，在“证书”框里有刚导入的证书信息



\* 点击“导入”默认是导入 x.509 的格式证书，要导入其他格式的证书，请点击下面的



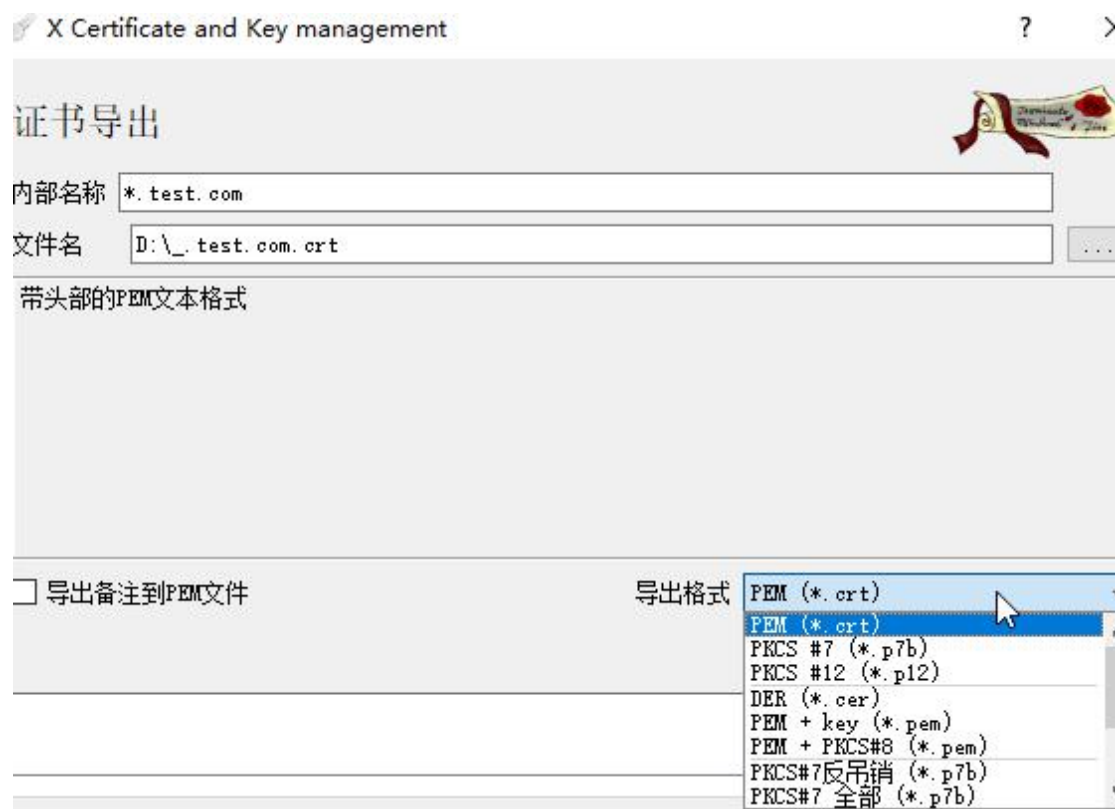
导入后就可直接再导出为其他格式的证书了，因为它的 rsa 私钥本来就在我们的这个 xca 数据库里，如果是从其他的地方发来的证书，其 rsa 密钥不在本 xca 数据库里，则要导入此证书对应的私钥：



## ②导出证书为其他格式



选中目标证书，点击“导出”



选择目标类型就行了，如导出为.p7b 的格式，或者导出为.p12 的格式  
(.p12 的格式就是.pfx，导出后把后缀改为.pfx 就能直接给 IIS web 服务器使用)



## 六、创建自签名证书

如果我们的网站是给广大客户用的，则一定要用 CA 权威机构签名的 ssl 证书，别人才信得过我们，而如果是自己做实验或一般内部使用的环境下，可用自签名 ssl 证书，就是不给 CA 机构去签名了，自己签名就行，自己信任自己即可。

①同样也是要先创建 rsa 密钥对

文件 导入 令牌 其他工具 帮助

私钥 证书签名请求 证书 模板 吊销列表

内部名称	类型	长度	使用	密码
mytest.com	RSA	2048 bit	0 Common	
test.com	RSA	2048 bit	2 Common	

创建密钥 导出

②可直接创建证书，不用先创建请求文件了

证书签名请求 证书 模板 吊销列表

内部名称	commonName	CA	序列号	过期时间	CRL过期
*.test.com	*.test.com	否	3AD4E5AE6DDF9E1E	2022/12/21	

创建证书 导出

“来源”下面的“签名”选择“创建自签名证书”

### 创建x509证书

来源 主体 扩展 密钥用法 Netscape扩展 高级 备注

签名请求

☐ 签发证书签名请求 (CSR)

☒ 从签名请求复制扩展信息

☐ 修改签名请求的主体信息

\*.test.com

显示签名请求

签名

☒ 创建自签名证书

☐ 使用此CA证书进行签名

### 创建x509证书

来源
主体
扩展
密钥用法
Netscape扩展
高级
备注

内部名称 \*.mytest.com

Distinguished name

countryName	CN	organizationalUnitName	it-dept
stateOrProvinceName	GD	commonName	*.mytest.com
localityName	SZ	emailAddress	it@mytest.com
organizationName	myorg		

类型	内容

添加
删除

私钥

mytest.com (RSA:2048 bit)

☐ 包含已使用的密钥 生成新密钥

其他的信息和之前创建 CSR 证书签名请求时一样，随便写

### 创建x509证书

来源
主体
扩展
密钥用法
Netscape扩展
高级
备注

X509v3 Basic Constraints

类型 最终实体

CA路径长度

☐ Critical

Key identifier

☐ X509v3 Subject Key Identifier

☐ X509v3 Authority Key Identifier

有效期

不早于 2020-12-21 08:38 GMT

不晚于 2021-12-21 08:38 GMT

时间跨度 20 年

☐ UTC午夜时间 ☐ 当地时间 ☐ 未明确定义到期日

X509v3 Subject Alternative Name

编辑

在“扩展”里，选择“最终实体”类型，证书有效时间随便写，1到几十年都行，点击“应用”，最后添加 SAN 主体可选名称，点击右下的“编辑”

X Certificate and Key management

☐ Critical ☐ 复制通用名称 (CN)

	类型	内容
0	DNS	*.mytest.com
1	DNS	mytest.com

DNS: 一个DNS域名或'copycn'

添加以上信息，第 0 个一定是和 cn 通用名称一致，记录类型都是 DNS

### 创建x509证书

来源 主体 扩展 **密钥用法** Netscape扩展 高级 备注

X509v3 Key Usage

☐ Critical

☒ Digital Signature

☐ Non Repudiation

☒ Key Encipherment

☐ Data Encipherment

X509v3 Extended Key Usage

☐ Critical

☒ TLS Web Server Authentication

☒ TLS Web Client Authentication

☐ Code Signing

☐ E-mail Protection

“密钥用法”一般选择以上 4 个，确认无误后，点击右下方的 OK

内部名称	commonName	CA	序列号	过期时间
*.mytest.com	*.mytest.com	否	0C138611A42CEB05	2021/12
*.test.com	*.test.com	否	3AD4E5AE6DDF9E1E	2022/12

X Certificate and Key managem...

成功创建证书 '\*.mytest.com'

创建成功

双击此证书可查看详细信息

## 证书的详细信息



状态 主体 颁发者 扩展 备注

内部名称 \*.mytest.com

签名状态 **自签名** sha256WithRSAEncryption

密钥 mytest.com

序列号 0C138611A42CEB05

### 指纹信息

MD5 0A:20:C5:AB:8E:DE:FD:BF:A4:BA:CA:34:A4:78:F8:1E

SHA1 32:A7:95:83:38:10:F7:42:99:87:20:61:C7:D6:EF:52:B8:BA:5A:1D

状态 主体 颁发者 扩展 备注

countryName CN

stateOrProvinceName GD

localityName SZ

organizationName myorg

organizationalUnitName it-dept

commonName \*.mytest.com

emailAddress it@mytest.com

状态 主体 颁发者 扩展 备注

countryName CN

stateOrProvinceName GD

localityName SZ

organizationName myorg

organizationalUnitName it-dept

commonName \*.mytest.com

emailAddress it@mytest.com

主体和颁发者都是自己，这就是自签名证书。  
之后导出证书和对应的密钥即可。

作者: cof-lee

日期: 2020-12-21