|  |
| --- |
| 哈尔滨工业大学(深圳)哈尔滨工业大学哈尔滨工业大学(深圳)(哈尔滨工业大学(深圳)深圳哈尔滨工业大学(深圳))哈尔滨工业大学(深圳)哈尔滨工业大学哈尔滨工业大学(深圳)哈尔滨工业大学哈尔滨工业大学哈尔滨工业大学哈尔滨工业大学(深圳)(哈尔滨工业大学哈尔滨工业大学(深圳)深圳哈尔滨工业大学哈尔滨工业大学(深圳))哈尔滨工业大学哈尔滨工业大学(深圳)(哈尔滨工业大学(深圳)哈尔滨工业大学(哈尔滨工业大学(深圳)((哈尔滨工业大学(深圳)深圳(哈尔滨工业大学(深圳))(哈尔滨工业大学(深圳)深圳哈尔滨工业大学(深圳)深圳哈尔滨工业大学深圳哈尔滨工业大学(深圳)(深圳哈尔滨工业大学(深圳)深圳深圳哈尔滨工业大学(深圳))深圳哈尔滨工业大学(深圳))哈尔滨工业大学(深圳)哈尔滨工业大学)哈尔滨工业大学(深圳)()哈尔滨工业大学(深圳)深圳)哈尔滨工业大学(深圳)))哈尔滨工业大学(深圳) |
| 《网络与系统安全》 实验报告《网络与系统安全》 实验报告《网络与系统安全》 实验报告《网络与系统安全》 实验报告《网络与系统安全》 实验报告《网络与系统安全》 实验报告《网络与系统安全》 实验报告《网络与系统安全》 实验报告《网络与系统安全》 实验报告《网络与系统安全》 实验报告《网络与系统安全》 实验报告《网络与系统安全》 实验报告《网络与系统安全》 实验报告《网络与系统安全》 实验报告《网络与系统安全》 实验报告《网络与系统安全》 实验报告 |
|  |
| 实验四  PKI 实验实验PKI 实验PKI 实验实验PKI 实验实验实验PKI 实验PKI 实验PKI 实验实验实验PKI 实验实验PKI 实验PKI 实验实验PKI 实验PKI 实验  学 院: 计算机科学与技术学院   |  |  | | --- | --- | | 姓 名: | 梁鑫嵘 | | 学 号: | 200110619 | | 专 业: | 计算机科学与技术 | | 日 期: | 2023年4月 | |

1. 根据如下命令查看证书信息，并回答下面两个问题。

命令为：openssl x509 -in ca.crt -text -noout。

命令的输出为：

Certificate:   
   Data:   
       Version: 3 (0x2)   
       Serial Number:   
           1c:65:5e:fd:aa:6c:36:97:e0:7b:d1:b3:44:c5:86:e0:f2:20:01:17   
       Signature Algorithm: sha256WithRSAEncryption   
       Issuer: CN = www.modelCA.com, O = Model CA LTD., C = US   
       Validity   
           Not Before: May 18 06:38:50 2023 GMT   
           Not After : May 15 06:38:50 2033 GMT   
       Subject: CN = www.modelCA.com, O = Model CA LTD., C = US   
       Subject Public Key Info:   
           Public Key Algorithm: rsaEncryption   
               Public-Key: (4096 bit)   
               Modulus:   
                   00:ae:32:13:89:02:0a:a5:ac:33:a1:49:a0:ce:37: ...  
                   c4:ec:31   
               Exponent: 65537 (0x10001)   
       X509v3 extensions:   
           X509v3 Subject Key Identifier:    
               E7:CB:0D:CA:BE:79:6E:D1:85:1F:A3:22:0B:FA:B3:F0:4D:4F:F8:78   
           X509v3 Authority Key Identifier:    
               E7:CB:0D:CA:BE:79:6E:D1:85:1F:A3:22:0B:FA:B3:F0:4D:4F:F8:78   
           X509v3 Basic Constraints: critical   
               CA:TRUE   
   Signature Algorithm: sha256WithRSAEncryption   
   Signature Value:

1. 证书的哪部分内容表明这是证书的持有方？

证书的Subject字段表明了证书的持有方信息。在生成的证书中，Subject字段为：

Subject: CN = www.modelCA.com, O = Model CA LTD., C = US

在扩展字段中，

X509v3 Basic Constraints: critical   
CA:TRUE

于是这是一个CA证书。

（2）从证书的哪部分内容可以看出这是自签名的证书？

证书中Issuer和Subject字段是完全相同的，说明颁发者和持有者相同；同时这个证书还是一个CA证书，于是这就是一个自签名证书。

1. 用如下命令查看[www.bank32.com](http://www.bank32.com/)的服务器证书，至少说出与ca.crt的证书的两点不同。

openssl x509 -in server.crt -text -noout：

命令输出：

Certificate:   
   Data:   
       Version: 3 (0x2)   
       Serial Number: 4096 (0x1000)   
       Signature Algorithm: sha256WithRSAEncryption   
       Issuer: CN = www.modelCA.com, O = Model CA LTD., C = US   
       Validity   
           Not Before: May 18 06:53:46 2023 GMT   
           Not After : May 15 06:53:46 2033 GMT   
       Subject: C = US, O = Bank32 Inc., CN = www.bank32.com   
       Subject Public Key Info:   
           Public Key Algorithm: rsaEncryption   
               Public-Key: (2048 bit)   
               Modulus:   
                   00:c4:70:58:00:12:5d:cc:87:ab:d5:04:91:03:96: ….  
                   c3:95   
               Exponent: 65537 (0x10001)   
       X509v3 extensions:   
           X509v3 Basic Constraints:    
               CA:FALSE   
           Netscape Comment:    
               OpenSSL Generated Certificate   
           X509v3 Subject Key Identifier:    
               E8:EB:C8:06:0F:10:3B:98:D9:C0:81:B8:53:19:91:50:E6:BB:75:DE   
           X509v3 Authority Key Identifier:    
               E7:CB:0D:CA:BE:79:6E:D1:85:1F:A3:22:0B:FA:B3:F0:4D:4F:F8:78   
   Signature Algorithm: sha256WithRSAEncryption   
   Signature Value:  
  
不同点：

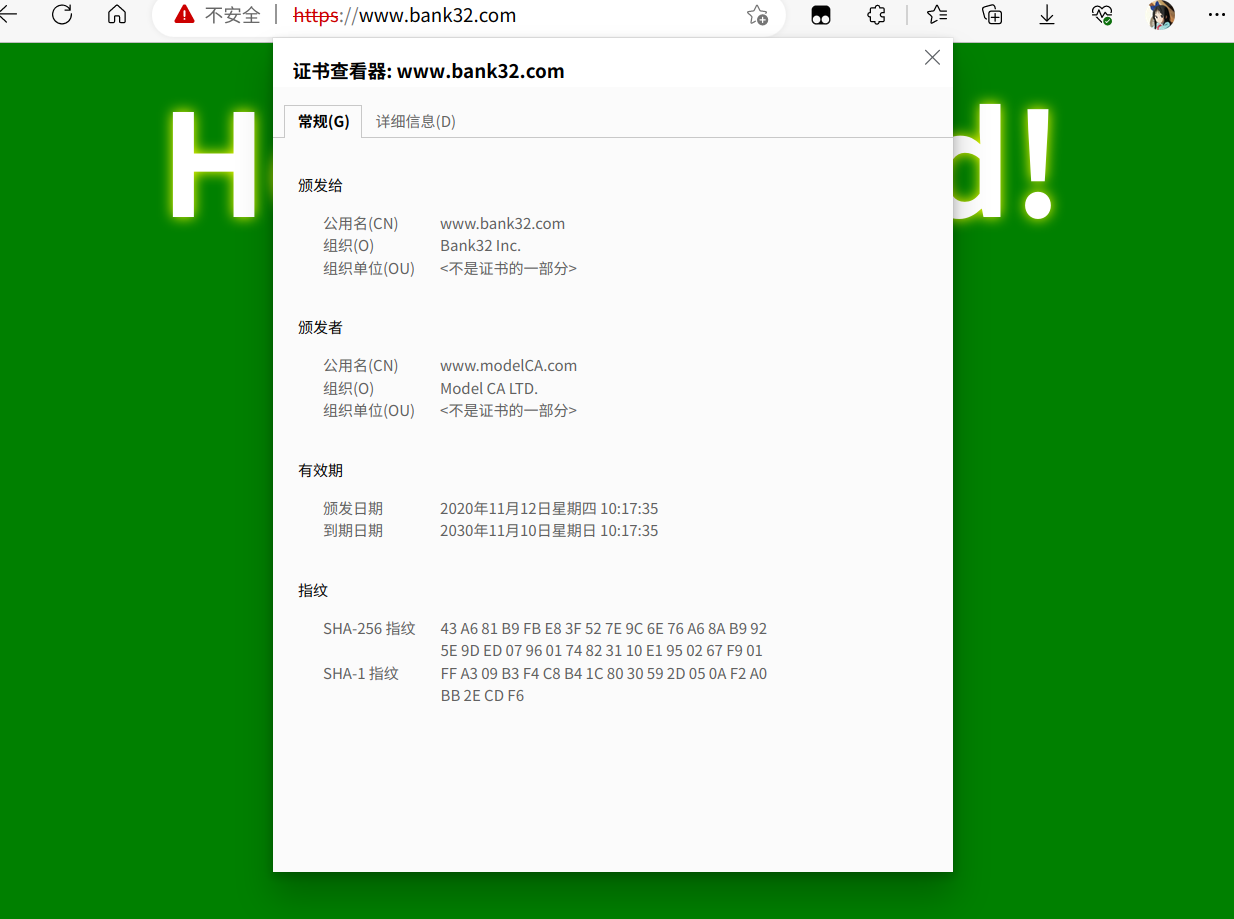
1. Subject字段不同，server.crt是由Bank Inc.持有的，而ca.crt是modelCA自己持有的。

2. X509v3 Basic Constraints 字段中server.crt是CA:FALSE，证明这不是一个CA证书，而ca.crt是一个CA证书。

3. 密钥不同。

1. 请将能够正确访问www.bank32.com的截图贴在下面。

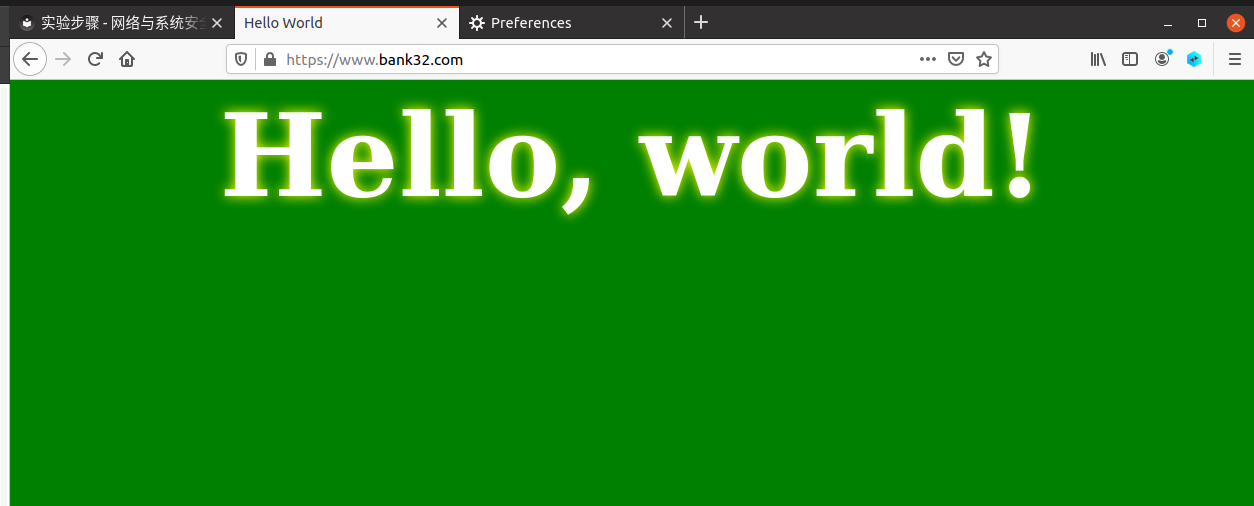
**在没有使用server.crt的时候访问：**



使用https协议访问了www.bank32.com，使用浏览器查看其证书。由于www.modelCA.com不是本机的可信CA，所以被标明为不安全访问。

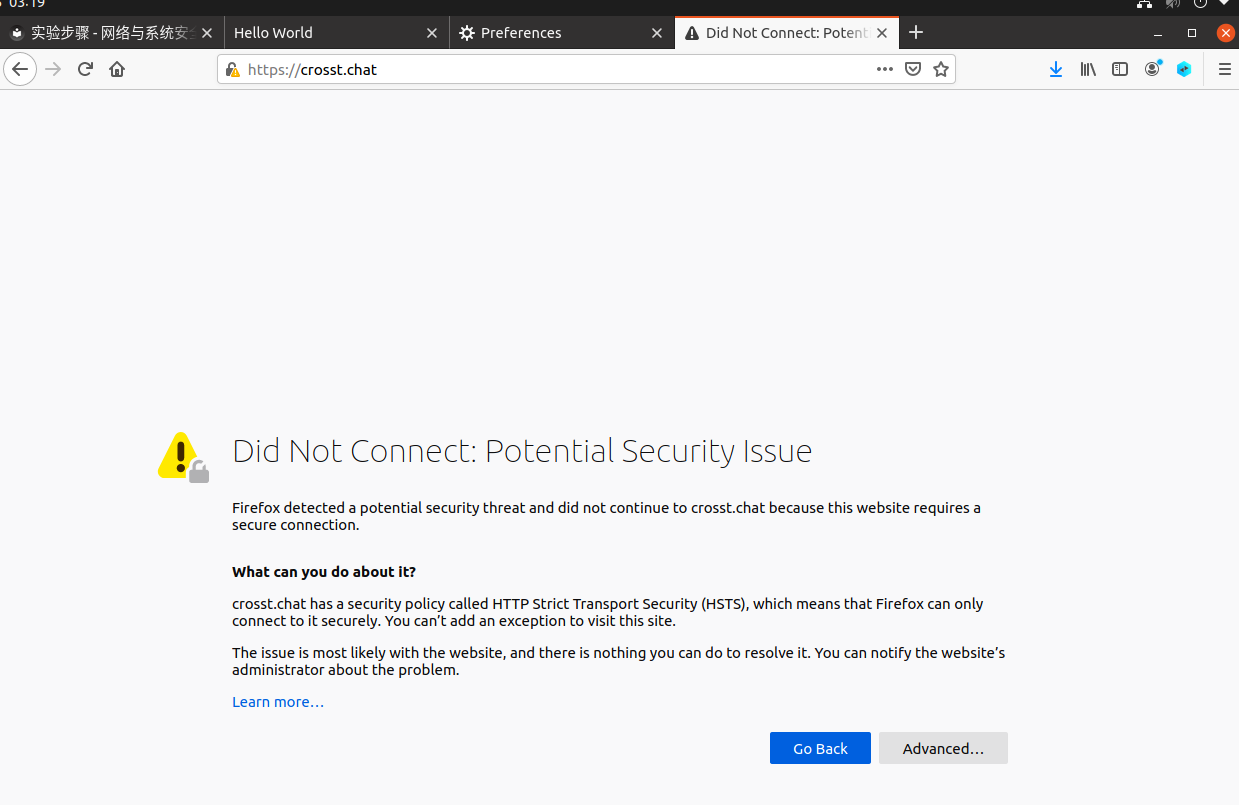
如果使用http访问，背景是红色的。

**在使用server.crt之后：**

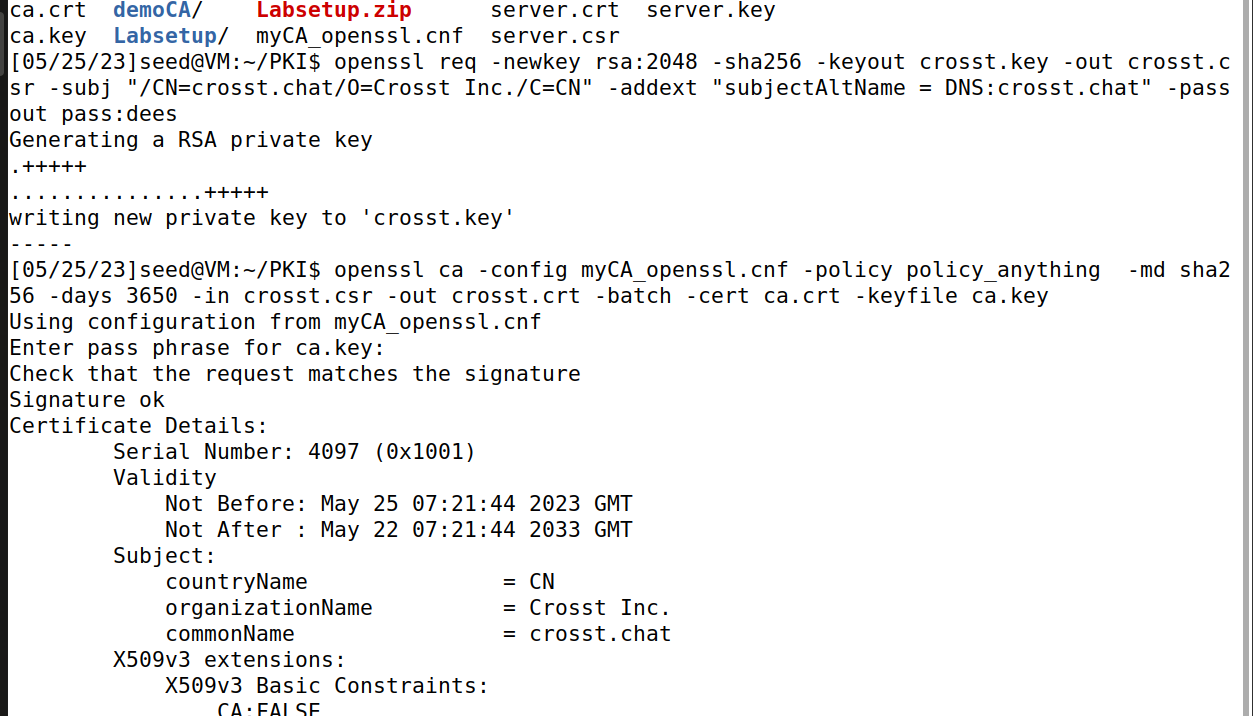


1. 将能够拦截访问一个（例如www.hitsz.edu.cn）网站的截图和CA被劫持后能够正常访问的截图贴在下面。并分析说明。（建议大家随机选取一个网站，不使用[www.hitsz.edu.cn](http://www.hitsz.edu.cn/)）

在crosst.chat使用www.bank32.com证书的时候：



为crosst.chat生成签名请求和证书：



使用生成的密钥：



1. 分析CA证书各密码算法的作用。

密码算法在 CA 证书中的作用是保证了证书的可信度和安全性,保护用户的

隐私安全和数据安全。

常用的密码算法以及特点：

1. 非对称加密算法：

- RSA（Rivest-Shamir-Adleman）：用于密钥生成、加密和解密过程。CA使用RSA算法生成公钥和私钥对，其中私钥用于签署证书请求和生成数字签名，公钥用于验证签名和加密通信。

- ECC（Elliptic Curve Cryptography）：与RSA类似，用于密钥生成、加密和解密。ECC算法基于椭圆曲线数学原理，提供与RSA相当的安全性但具有更小的密钥尺寸，适用于资源受限的环境。

2. 散列函数（哈希函数）：

- SHA-2（Secure Hash Algorithm 2）系列：包括SHA-224、SHA-256、SHA-384和SHA-512等算法。用于生成证书的数字指纹，确保证书的完整性和不可伪造性。

- SHA-3（Secure Hash Algorithm 3）系列：包括SHA3-224、SHA3-256、SHA3-384和SHA3-512等算法。作为SHA-2的后继者，提供更高的安全性和更好的性能。

3. 对称加密算法：

- AES（Advanced Encryption Standard）：用于保护私钥和敏感信息的机密性。在证书生成和传输过程中，对称加密算法用于加密和解密数据，确保数据在传输过程中的安全性。

4. 数字签名算法：

- DSA（Digital Signature Algorithm）：一种使用非对称加密算法的数字签名算法，用于生成和验证数字签名。CA使用DSA算法生成签名并将其附加到证书中，以确保证书的真实性和完整性。

- ECDSA（Elliptic Curve Digital Signature Algorithm）：与DSA类似，基于椭圆曲线的数字签名算法，提供与DSA相当的安全性但具有更小的密钥尺寸。

这些密码算法的选择取决于安全性需求、性能要求和可用性等因素。CA使用这些密码算法的组合来创建安全的数字证书，并确保证书的合法性、真实性和可信性。