X.509

Q: Give me an example of X.509 certificate and tell me how it works.

1. X.509 简介

   X.509 是被广泛使用的数字证书标准，是由国际电联电信委员会（ITU-T）为单点登录（SSO-Single Sing-on）和授权管理基础设施（PMI-Privilege Management Infrastructure）制定的 PKI 标准。X.509 定义了（但不仅限于）公钥证书、证书吊销清单、属性证书和证书路径验证算法等证书标准。

2. X.509 基本结构

X.509证书结构		
Version	版本	标识证书的版本（版本1、版本2或是版本3）
Serial Number	序列号	标识证书的唯一整数，由证书颁发者分配的本证书的唯一标识符
Issuer(CA's name)	颁发者	证书颁发者的可识别名（DN）
Validity	有效期	证书有效期的时间段。本字段由"Not Before"和"Not After"两项组成，它们分别由UTC时间或一般的时间表示（在RFC2459中有详细的时间表示规则）
Not Before	有效起始日期	
Not After	有效终止日期	
Subject	使用者	证书拥有者的可识别名，这个字段必须是非空的，除非在证书扩展中有别名。
Subject Pubilc Key Info	使用者公钥信息	主体的公钥（以及算法标识符）
Public Key Algorithm	公钥算法	
Subject Public Key	公钥	
Issuer Unique Identifier (Optional)	颁发者唯一标识	标识符—证书颁发者的唯一标识符，仅在版本2和版本3中有要求，属于可选项。
Subject Unique Identifier(Optional)	使用者唯一标识	证书拥有者的唯一标识符，仅在版本2和版本3中有要求，属于可选项。
Extensions(Optional)	拓展	可选的标准和专用的扩展（仅在版本2和版本3中使用）
Certificate Signature Algorithm	证书签名算法	用于签证书的算法标识，由对象标识符加上相关的参数组成，用于说明本证书所用的数字签名算法。例如，SHA-1和RSA的对象标识符就用来说明该数字签名是利用RSA对SHA-1杂凑加密。
Certificate Signature	证书签名	

3. X.509 实例

   以下是一个典型的 X.509 证书实例

```
// 一个典型的证书实例
Certificate: // 证书主体
   Data:
      // 证书标准版本号
       Version: 1 (0x0)
      // 序列号
       Serial Number: 7829 (0x1e95)
      // 证书签名算法
       Signature Algorithm: md5WithRSAEncryption
      // 证书发行者
       Issuer: C=ZA, ST=Western Cape, L=Cape Town, O=Thawte Consulting cc,
               OU=Certification Services Division,
               CN=Thawte Server CA/emailAddress=server-certs@thawte.com
      // 证书有效期
       Validity:
          // 起始日期
           Not Before: Jul  9 16:04:02 1998 GMT
          // 截止日期
           Not After : Jul  9 16:04:02 1999 GMT
      // 主体
       Subject: C=US, ST=Maryland, L=Pasadena, O=Brent Baccala,
                OU=FreeSoft, CN=www.freesoft.org/emailAddress=baccala@freesoft.org
```

```
      // 主体公钥信息
       Subject Public Key Info:
            // 公钥算法
           Public Key Algorithm: rsaEncryption
           // RSA算法的公钥值
           RSA Public Key: (1024 bit)
               Modulus (1024 bit):
                   00:b4:31:98:0a:c4:bc:62:c1:88:aa:dc:b0:c8:bb:
                   33:35:19:d5:0c:64:b9:3d:41:b2:96:fc:f3:31:e1:
                   66:36:d0:8e:56:12:44:ba:75:eb:e8:1c:9c:5b:66:
                   70:33:52:14:c9:ec:4f:91:51:70:39:de:53:85:17:
                   16:94:6e:ee:f4:d5:6f:d5:ca:b3:47:5e:1b:0c:7b:
                   c5:cc:2b:6b:c1:90:c3:16:31:0d:bf:7a:c7:47:77:
                   8f:a0:21:c7:4c:d0:16:65:00:c1:0f:d7:b8:80:e3:
                   d2:75:6b:c1:ea:9e:5c:5c:ea:7d:c1:a1:10:bc:b8:
                   e8:35:1c:9e:27:52:7e:41:8f
               // 证书扩展部分
               Exponent: 65537 (0x10001)
   // 证书签名算法
    Signature Algorithm: md5WithRSAEncryption
        93:5f:8f:5f:c5:af:bf:0a:ab:a5:6d:fb:24:5f:b6:59:5d:9d:
        92:2e:4a:1b:8b:ac:7d:99:17:5d:cd:19:f6:ad:ef:63:2f:92:
        ab:2f:4b:cf:0a:13:90:ee:2c:0e:43:03:be:f6:ea:8e:9c:67:
        d0:a2:40:03:f7:ef:6a:15:09:79:a9:46:ed:b7:16:1b:41:72:
        0d:19:aa:ad:dd:9a:df:ab:97:50:65:f5:5e:85:a6:ef:19:d1:
        5a:de:9d:ea:63:cd:cb:cc:6d:5d:01:85:b5:6d:c8:f3:d9:f7:
        8f:0e:fc:ba:1f:34:e9:96:6e:6c:cf:f2:ef:9b:bf:de:b5:22:
        68:9f
```

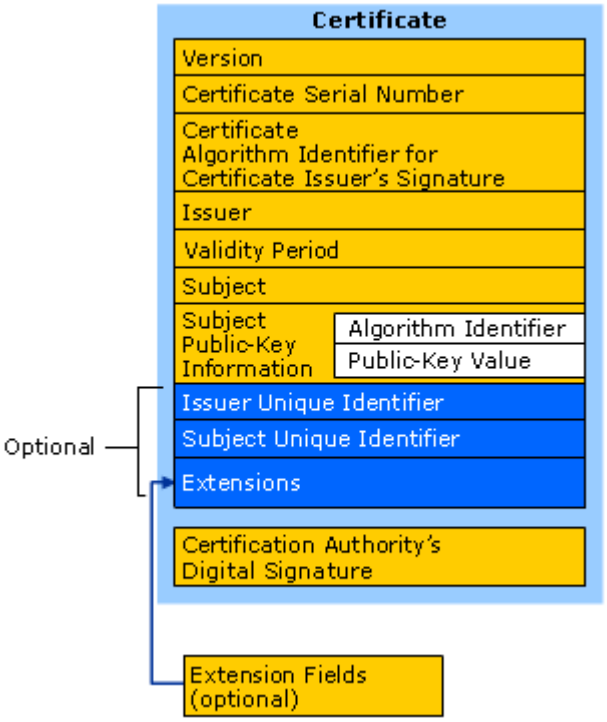
4. X.509 工作描述

(1) 基本原理

数字证书是用来确认网络上个人电脑和其他实体的在线身份的电子凭证。数字证书的作用类似于身份证，如护照和驾驶证，由认证机构(certificate authority,CA)颁发。 CA 用自己的私钥对用户的身份信息(主要是用户名和该用户的公钥)进行签名，该签名和用户的身份信息一起就形成了证书。除用户信息外，数字证书中还包括证书机构名称,证书有效期,证书的序列号,签名使用的哈希算法,公钥使用的加密算法等相关信息。

(2) 证书结构

右图为 X.509 第三版证书的内容，详细描述可见上页（2.X.509 基本结构）



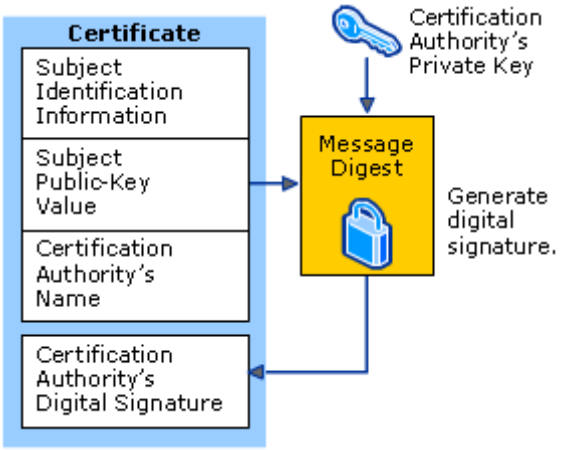
(3) 工作过程

1) 使用用户的身份信息生成数字签名

一个 CA 使用其私有密钥对每次发出的证书进行数字签名。为了创建数字签名，从该证书生成摘要，加密该摘要用其私钥，并且把数字签名作为证书的一部分。任何人都可以使用的消息摘要函数和 CA 的公钥来验证证书的完整性。如果证书已损坏或者与它人篡改，被改变的证书的信息摘要将和数字签名不匹配。下图显示了由 CA 生成数字签名的方式。

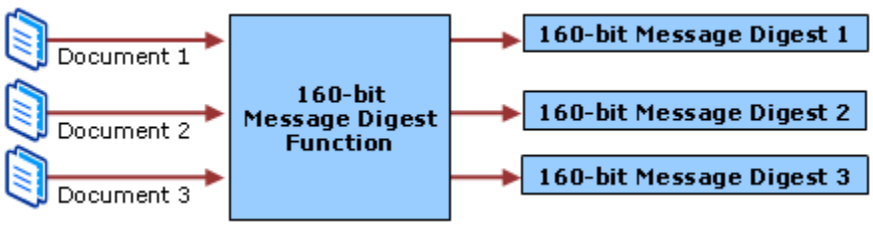
2) 签名验证

用户 A 把自己的证书发送给用户 B。用户 B 使用 CA 的公钥对证书的签名进行验证，由于只有 CA 才能生成该证书，因此只要证书验证正确，即说明证书是由 CA 发布的，证书中用户 A 的公钥是值得信赖的。用户 B 以后就可以使用该公钥验证用户 A 的签名或者和 A 进行加密通信。



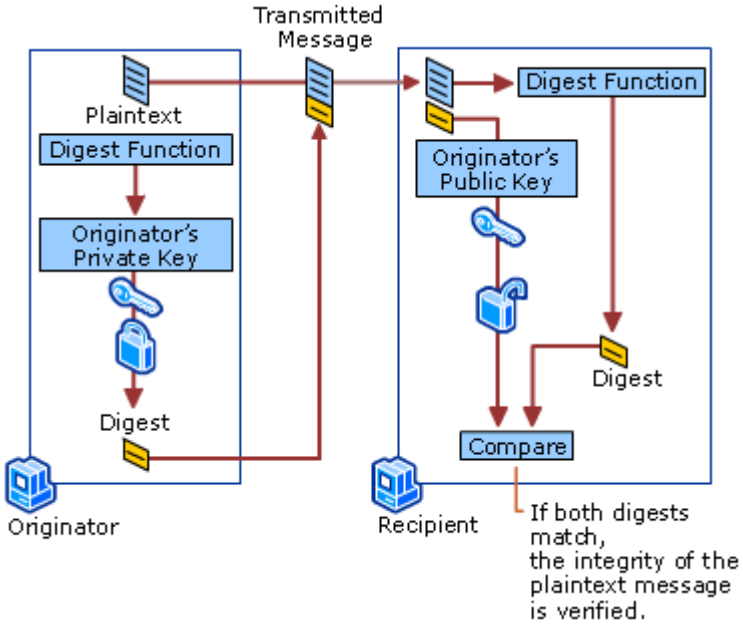
3) 消息摘要函数

消息摘要函数，也称为散列函数，常用结合非对称密钥，以进一步加强公共密钥加密。消息摘要是通常为 128 位至 160 位的长度，并为每个数字文件或文档的唯一数字标识符。文档的两个副本都会有相同的消息摘要，即使修改文件中的一个比特，都会导致消息摘要的变化。下图显示了基本的消息摘要过程。



消息摘要是结合常用的与公开密钥技术来创建数字签名或用于认证，完整性和不可抵赖性的“数字指纹”。消息摘要还与数字签名技术常用于电子文件和文档提供数据完整性。

4) 基本的完整性检查过程



如果消息摘要和证书中包含的数字签名不匹配，认为其被更改或损坏。

5) 如何验证证书机构的公钥-证书的证书

用户 B 使用证书机构的公钥来验证用户 A 的数字证书，但如何又能够知道用户 B 拿到的证书机构的公钥不是伪造的呢？解决办法是再找一个证书机构对该证书机构的公钥颁发一个证书，这样形成了一个公钥证书的嵌套循环，该循环的终点就是根证书机构。根证书机构较少，其公钥可以通过安全的方式发布，如通过 USB 拷贝、书面文件当面移交。