计算机信息安全技术基础知识点总结

1 基本概念

1.1 信息安全的要素

- **保密性**：指网络中的信息不被非授权实体获取与使用。
  保密的信息包括：
  1. 存储在计算机系统中的信息：使用访问控制机制，也可以进行加密增加安全性。
  2. 网络中传输的信息：应用加密机制。
- **完整性**：指数据未经授权不能进行改变的特性，即信息在存储或传输过程中保持不被修改、不被破坏和丢失的特性，还要求数据的来源具有正确性和可信性，数据是真实可信的。
  解决手段：数据完整性机制。
- **真实性**：保证以数字身份进行操作的操作者就是这个数字身份合法拥有者，也就是说保证操作者的物理身份与数字身份相对应。
  解决手段：身份认证机制。
- **不可否认性**：或不可抵赖性。发送信息方不能否认发送过信息，信息的接收方不能否认接收过信息。
  解决手段：数字签名机制。

1.2 信息保密技术

- 明文（Message）：指待加密的信息，用 M 或 P 表示。
- 密文（Ciphertext）：指明文经过加密处理后的形式，用 C 表示。
- 密钥（Key）：指用于加密或解密的参数，用 K 表示。
- 加密（Encryption）：指用某种方法伪装消息以隐藏它的内容的过程。
- 加密算法（EncryptionAlgorithm）：指将明文变换为密文的变换函数，用 E 表示。
- 解密（Decryption）：指把密文转换成明文的过程。
- 解密算法（DecryptionAlgorithm）：指将密文变换为明文的变换函数，用 D 表示。
- 密码分析（Cryptanalysis）：指截获密文者试图通过分析截获的密文从而推断出原来的明文或密钥的过程。
- 密码分析员（Crytanalyst）：指从事密码分析的人。
- 被动攻击（PassiveAttack）：指对一个保密系统采取截获密文并对其进行分析和攻击，这种攻击对密文没有破坏作用。
- 主动攻击（ActiveAttack）：指攻击者非法入侵一个密码系统，采用伪造、修改、删除等手段向系统注入假消息进行欺骗，这种攻击对密文具有破坏作用。
- 密码体制（密码方案）：由明文空间、密文空间、密钥空间、加密算法、解密算法构成的五元组。
  分类：
  1. 对称密码体制：单钥密码体制，加密密钥和解密密钥相同。
  2. 非对称密码体制：双钥密码体制、公开密码体制，加密密钥和解密密钥不同。
- 密码系统（Cryptosystem）：指用于加密和解密的系统，通常应当是一个包含软、硬件的系统。
- 柯克霍夫原则：密码系统的安全性取决于密钥，而不是密码算法，即密码算法要公开。

1.3 摘要算法

1.3.1 概念

摘要算法，又叫作 Hash 算法或散列算法，是一种将任意长度的输入浓缩成固定长度的字符串的算法，注意是“浓缩”而不是“压缩”，因为这个过程是不可逆的。

常见的摘要算法：MD5，SHA-1。

1.3.2 特点

1. 过程不可逆。
2. 不同内容的文件生成的散列值一定不同；相同内容的文件生成的散列值一定相同。由于这个特性，摘要算法又被形象地称为文件的“数字指纹”。
3. 不管文件多小（例如只有一个字节）或多大（例如几百 GB），生成的散列值的长度都相同，而且一般都只有几十个字符。

1.3.3 应用

这个神奇的算法被广泛应用于比较两个文件的内容是否相同——散列值相同，文件内容必然相同；散列值不同，文件内容必然不同。

2 对称密码体制

2.1 概念

单钥密码体制，加密密钥和解密密钥相同。

目前最为流行的对称加密算法是 DES（DataEncryptionStandard，数据加密标准）和 AES，此外，对称加密算法还有 IDEA、FEAL、LOKI、Lucifer、RC2、RC4、RC5、Blow fish、GOST、CAST、SAFER、SEAL 等。

WinRAR 的文件加密功能就是使用的 AES 加密算法。

2.2 举例

加密过程：

明文：good good study, day day up.

密钥：google

加密算法：将明文中的所有的字母“d”替换成密钥。

密文：将“good good study, day day up.”中的所有字母“d”替换成“google”，就得到密文“googoogle googoogle stugoogley， googleay googleay up.”。

解密过程：

密文：googoogle googoogle stugoogley， googleay googleay up.

密钥：google

解密算法：将密文中所有与密钥相同的字符串替换成“d”。

明文：将“googoogle googoogle stugoogley， googleay googleay up.”中的所有“google”替换成“d”，就得到了明文“good good study, day day up.”。

2.3 对称密码体制的不足

如果是已经保存在自己硬盘上的文件，使用对称加密技术进行加密是没有问题的；如果是两个人通过网络传输文件，使用对称加密就很危险——因为在传送密文的同时，还必须传送解密密钥。

3 非对称密码体制

3.1 概念

双钥密码体制、公开密码体制、公钥密码体制。加密密钥和解密密钥不同，一个公开，称为公钥；一个保密，称为私钥。

常见的算法：RSA 密码算法，Diffie-Hellman 密钥交换算法，ElGamal 加密算法。

3.2 非对称加密算法的特点

如果用密钥 K1 进行加密，则有且仅有密钥 K2 能进行解密；反之，如果使用密钥 K2 进行了加密，则有且仅有密钥 K1 能进行解密。

注意：“有且仅有”的意思——如果用密钥 K1 进行了加密，是不能用密钥 K1 进行解密的；同样，如果用密钥 K2 进行了加密，也无法用密钥 K2 进行解密。

3.3 基本思路

首先，生成一对满足非对称加密要求的密钥对（密钥 K1 和密钥 K2）。然后，将密钥 K1 公布在网上，任何人都可以下载它，我们称这个已经公开的密钥 K1 为公钥；密钥 K2 自己留着，不让任何人知道，我们称这个只有自己知道的密钥 K2 为私钥。当我想给 Clark 传送小电影时，我可以用 Clark 的公钥对小电影进行加密，之后这个密文就连我也无法解密了。这个世界上只有一个人能将密文解密，这个人就是拥有私钥的 Clark。

3.4 非对称密码体制的不足

非对称加密算法有一个重大缺点——加密速度慢，编码率比较低。例如在上一篇里我给 Clark 传的那个 1GB 的小电影，进行非对称加密足足用了 66 小时。那个借条小一些吧，也用了将近 2 分钟。所以在实际使用非对称加密的时候，往往不直接对文件进行加密，而是使用摘要算法与非对称算法相结合（适用于数字签名）或对称加密和非对称加密相结合（适用于加密传输文件）的办法来解决或者说绕过非对称加密算法速度慢的问题。

3.5 应用一：加密

3.5.1 基本原理

使用接收者的“公开密钥”加密，接收方用自己的“私有密钥”解密。

【举例】

流程是这样的：首先，登录当地的数字证书认证中心网站，填表->出示个人有效证件原件和复印件->缴费->等待数字证书认证中心制作数字证书->领取数字证书。如果您的公司需要申请大量的数字证书，还可以与认证中心的销售人员商量，先领取免费的试用版的数字证书供技术人员试用。

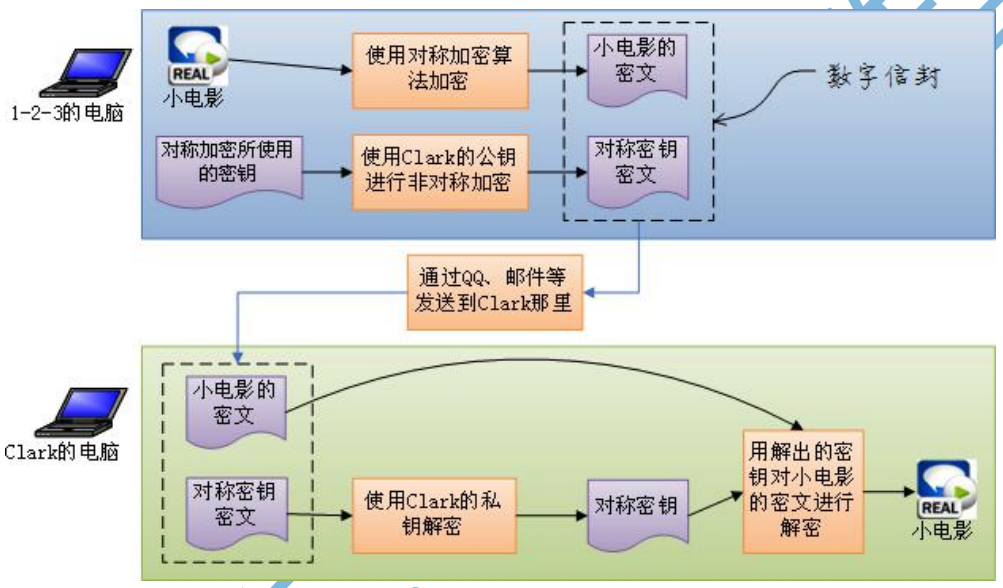
我先在数字证书认证中心下载了 Clark 的公钥证书(就是一个含有公钥信息的文件)，使用非对称加密算法对不良漫画进行加密，再将密文通过 QQ 传送给 Clark。然后，我兴冲冲地拨打 Clark 的手机：

“喂？Clark 么？好久不见，呵呵.....我给你发了个好东东呦，在 QQ 上，收到没？.....已经用你的公钥加密了。用你的私钥解密就行了”

Clark 兴冲冲地插入他的私钥（忘了说了，私钥并不是一个文件，而是一个 USB 设备，外形就跟 U 盘一样），解密，然后开始看漫画。

3.5.2　加密的优化：对称加密和非对称加密相结合

我们可以用对称加密与非对称加密相结合的方式来解决这个问题。对称加密速度快，但是必须在传送密文的同时传送解密密钥；非对称加密速度慢，但是不需要传送解密密钥。把两个技术一起使用，各取优点，就 OK 了。方法是，先把小电影用对称加密算法加密，然后把解密密钥用非对称加密算法加密。再将小电影的密文与解密密钥的密文同时传送给 Clark。Clark 收到这两样东西后，先用自己的私钥将解密密钥的密文解密，得到解密密钥，再用解密密钥将小电影的密文解密，就得到了小电影的明文。Clark 收到的这两样东西——小电影的密文和解密密钥的密文——加在一起就叫作**数字信封**。如下图：



3.6　应用二：数字签名

3.6.1　基本原理

使用发送方/签名人的“私有密钥”加密/签名，接收方/验证方收到签名时使用发送方的公开密钥验证。

【举例】

唉，这个月买了太多的书，到月底揭不开锅了。正巧在 QQ 上遇到了 Clark：

1-2-3：“Clark，我需要 200 两纹银，能否借给我？”

Clark：“没问题。我这就给你转账。请给我一张借条。”

1-2-3：“太谢谢了，我这就用 Word 写一个借条给你。”

然后，我新建一个 Word 文档，写好借条，存盘。然后，然后怎么办呢？我不能直接把借条发送给 Clark，原因有：

1. 我无法保证 Clark 不会在收到借条后将“纹银 200 两”改为“纹银 2000 两”。
2. 如果我赖账，Clark 无法证明这个借条就是我写的。
3. 普通的 Word 文档不能作为打官司的证据。

好在我早就申请了数字证书。我先用我的私钥对借条进行加密，然后将加密后的密文用 QQ 发送给 Clark。Clark 收到了借条的密文后，在数字证书认证中心的网站上下载我的公钥，然后使用我

的公钥将密文解密，发现确实写的是“借纹银 200 两”，Clark 就可以把银子放心的借给我了，我也不会担心 Clark 会篡改我的借条，原因是：

1. 由于我发给 Clark 的是密文，Clark 无法进行修改。Clark 倒是可以修改解密后的借条，但是 Clark 没有我的私钥，没法模仿我对借条进行加密。这就叫**防篡改**。

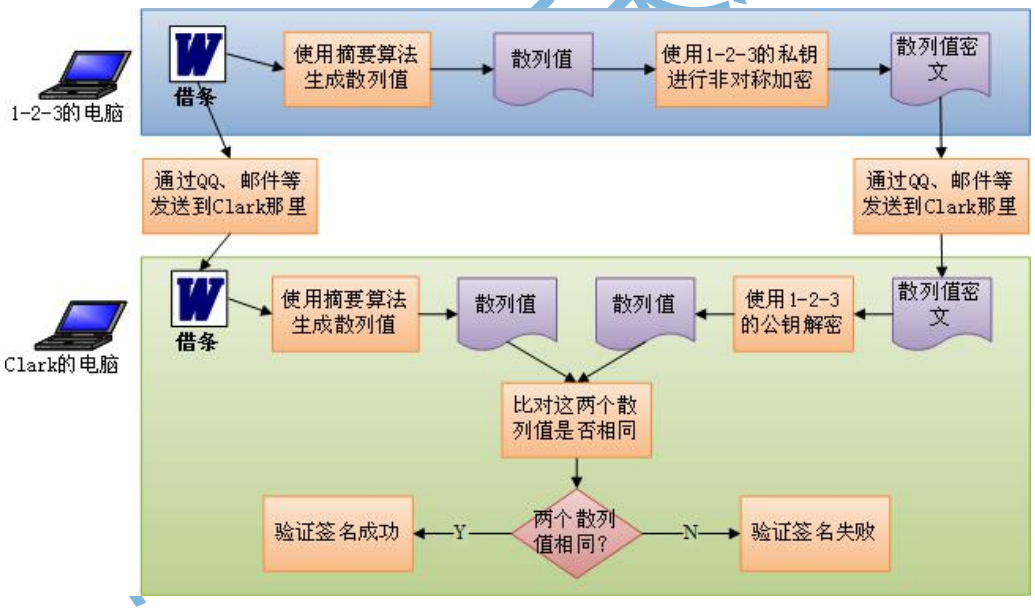
2. 由于用我的私钥进行加密的借条，有且只有我的公钥可以解密。反过来讲，能用我的公钥解密的借条，一定是使用我的私钥加密的，而只有我才拥有我的私钥，这样 Clark 就可以证明这个借条就是我写的。这就叫**防抵赖**。

3. 如果我一直赖着不还钱，Clark 把我告上了法庭，这个用我的私钥加密过的 Word 文档就可以当作程堂证供。因为我国已经出台了《中华人民共和国电子签名法》，使数字签名具有了**法律效力**。

您一定已经注意到了，这个使用我的私钥进行了加密的借条，具有了防篡改、防抵赖的特性，并且可以作为程堂证供，就跟我对这个借条进行了“签名”的效果是一样的。对了，“使用我的私钥对借条进行加密”的过程就叫做**数字签名**。

3.6.2　数字签名的优化：摘要算法与非对称算法相结合

用 Word 写给 Clark 的借条进行签名，总是先生成这个借条的散列值，然后用我的私钥对这个散列值进行非对称加密，然后把加密后的散列值（我们就叫它“散列值密文”吧）和借条一同发送到 Clark 那里。Clark 在收到借条和散列值密文后，用从网上下载的我的公钥将散列值解密，然后 Clark 自己再生成一次借条的散列值，比对这两个散列值是否相同，如果相同，就叫作验证签名成功。由于散列值只有几十个字节，所以签名的速度还可以忍受。如下图：

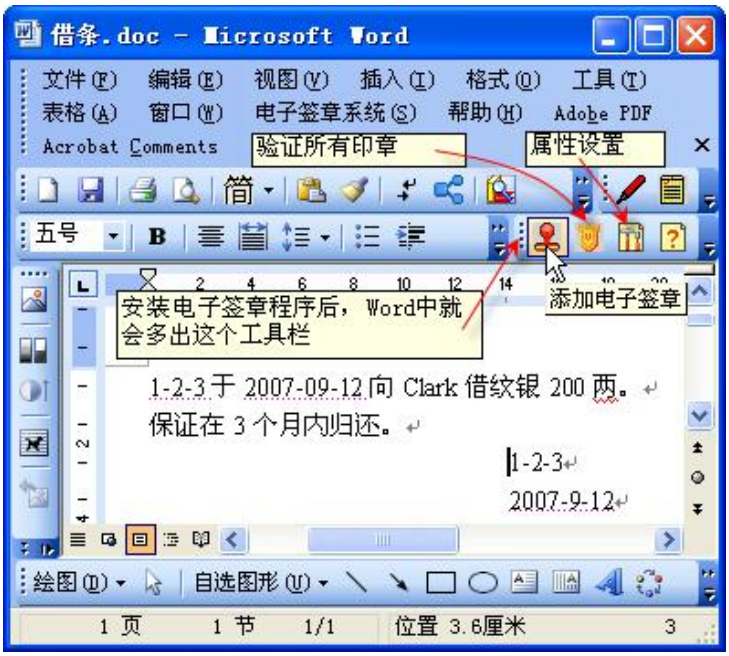


3.6.3　电子签章

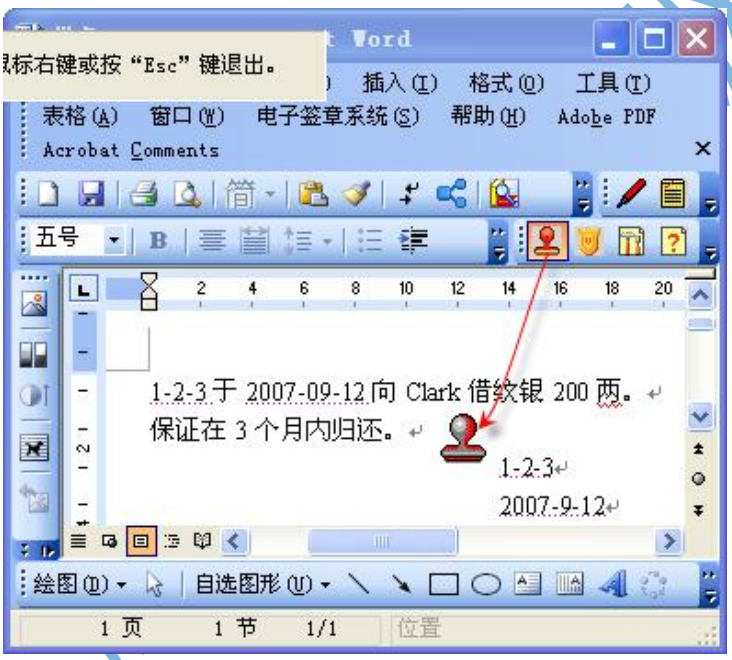
3.6.3.1　电子签章签名过程

下面就演示一下用电子签章程序对我的借条进行签名的过程。

1.　安装了电子签章程序后，Word 和 Excel 中就会多出一个签名用的工具条。如下图：

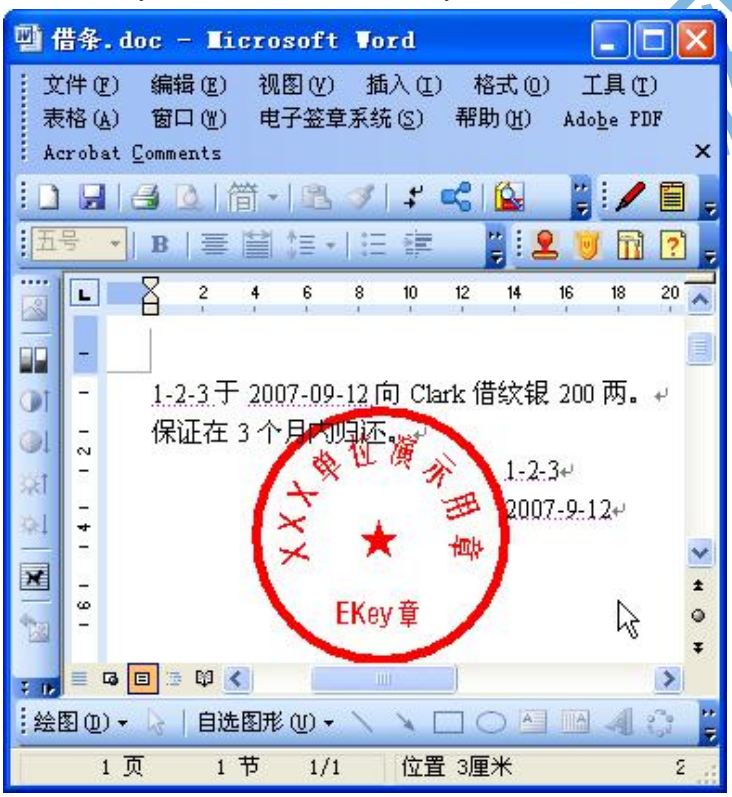


2. 写好借条，存盘。然后用鼠标点击“添加电子签章”按钮。然后在需要显示印章图片的位置上再按一次鼠标左键。



3. 电子签章程序会弹出一个对话框，注意在这步一定要勾选“签章后锁定文件”复选框，至于为什么要这样，稍后再讲。然后点击确定按钮。

4. 在上一步按确定按钮后，电子签章程序还会提示要求我输入存放私钥的 USB-Key 的使用密码，然后 Word 中就会出现一个印章了。这个印章图片是我提供给数字证书中心，在制作 USB-Key 的时候就烧录在 USB-Key 之中的。



5. 之后我把借条发送给 Clark。Clark 想验证签名的话只要按“验证所有印章”就可以了。电子签章会弹出如左图所示的对话框。

“散列值”、“公钥证书”都被电子签章程序插入到 Word 文档中的某个特定的地方了，如果你熟悉 Word 文档的结构，是不难找到它们的。

3.6.3.2　电子签章 Bug

1. 在上面的第 3 步不勾选“签章后锁定文件”复选框，这样在进行了数字签名之后，仍然可以更改 Word 文档的内容。当然如果在进行了数字签名之后又更改了 Word 文档的内容，验证签名操作就会失败。这时需要再次进行签名操作。
2. 在上面的第 3 步勾选“签章后锁定文件”复选框，这样在进行了签名操作后，Word 文档的内容就再也无法更改了。

但是，电子签章程序有一个大 Bug——在进行了签名操作后，如果只是更改了文字的颜色，验证签名操作仍然会成功。这就意味着，如果我在 Word 中写到“向公司借款 2000 元”，然后把“2000”的最后一个 0 的颜色改为白色，在领导看来就是“向公司借款 200 元”。领导欣然签章，然后我再把那最后一个 0 的颜色改为黑色，就又变成了“向公司借款 2000 元”，而且验证签名居然会成功。这也是为什么我在上面的第 3 步要强调一定要勾选“签章后锁定文件”复选框了。我猜测造成这个 Bug 的原因很可能是因为电子签章程序仅仅对文档中的纯文本生成散列值，而不是对文本+全部格式信息一同生成散列值。大家在购买电子签章程序前一定要作这方面的测试。

3.7　应用三：身份验证

3.7.1　基本原理

【举例】

正如您已经知道的，Clark 的老婆是一名超级黑客——就是传说中能用计算机作任何事的人。这不，不久前她就轻松入侵了 QQ 数据库，下载了 Clark 的所有好友的 ID 和密码以及聊天记录。然后，时不时地伪装成 Clark 的好友跟 Clark 聊天，搞得 Clark 最近总是神经兮兮、疑神疑鬼的。这不，昨天我在 QQ 上遇到了 Clark：

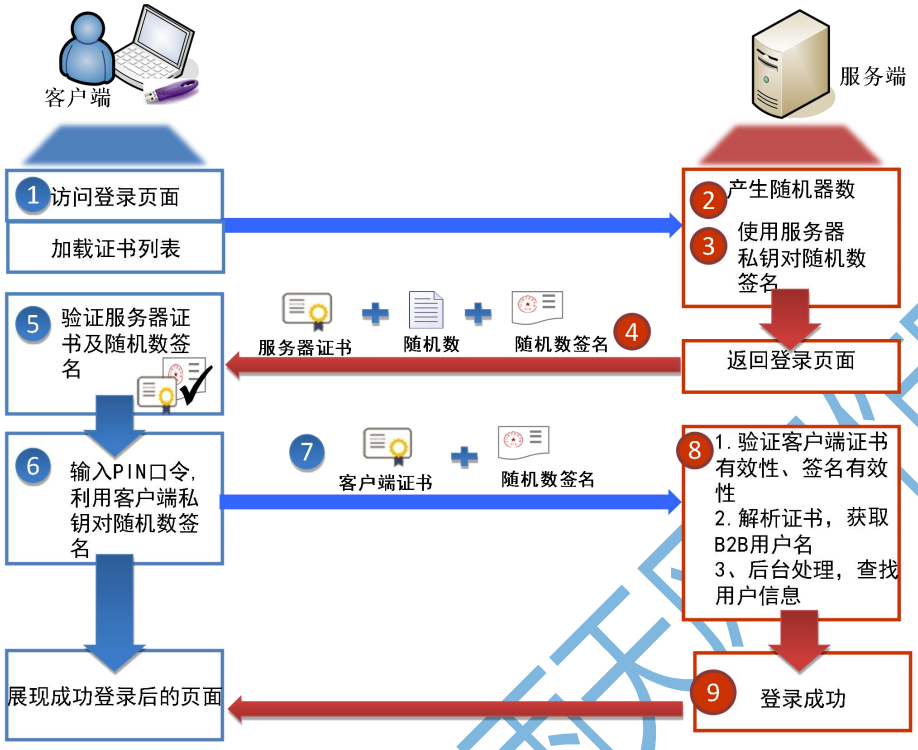
1-2-3：“Clark，最近还好吧？我又搞到一个好东东呦，要不要？”

Clark：“48475bbt556”

Clark 并不是疯掉了，那个“48475bbt556”也不是我跟 Clark 之间的什么通关暗语。这个“48475bbt556”就是 Clark 在键盘上胡乱敲上去的，不过，我却知道 Clark 是什么意思。我立刻把“48475bbt556”粘贴到 Word 里，然后用我的私钥对这个 Word 文档加密，再将这个 Word 文档发送给 Clark。Clark 在那边用我的公钥将 Word 文档解密，打开，发现里面写的就是“48475bbt556”，

就知道 QQ 这边的确就是真正的我本人了。因为拥有我的私钥的人在这个世界上就只有我一人而已，Clark 的老婆大人就是再神通广大也模仿不了，这就是数字签名的验证功能。

3.7.2　使用数字签名验证（冲击-响应应用模式）



登录认证流程：

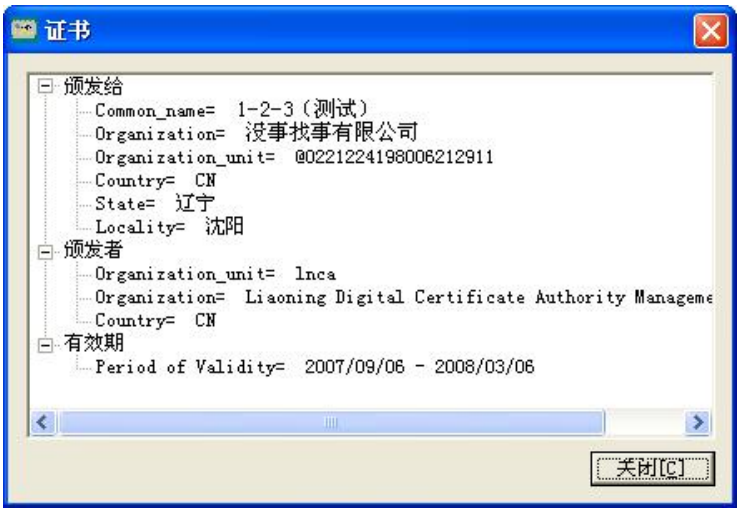
1. 用户访问登录页面，加载客户端证书列表；
2. 服务器端产生随机数，并使用服务器私钥对其进行签名，然后将随机数、签名值和服务器证书返回到登录页面，供客户端验证；
3. 用户输入证书密码，客户端私钥对随机数进行签名，并将客户端证书和签名值发送给服务器端；
4. 服务器端验证客户端签名值和证书，验证通过后解析证书，获取 B2B 平台的用户名和数据库做匹配；
5. 登录成功。

3.8　公钥/私钥存在形式

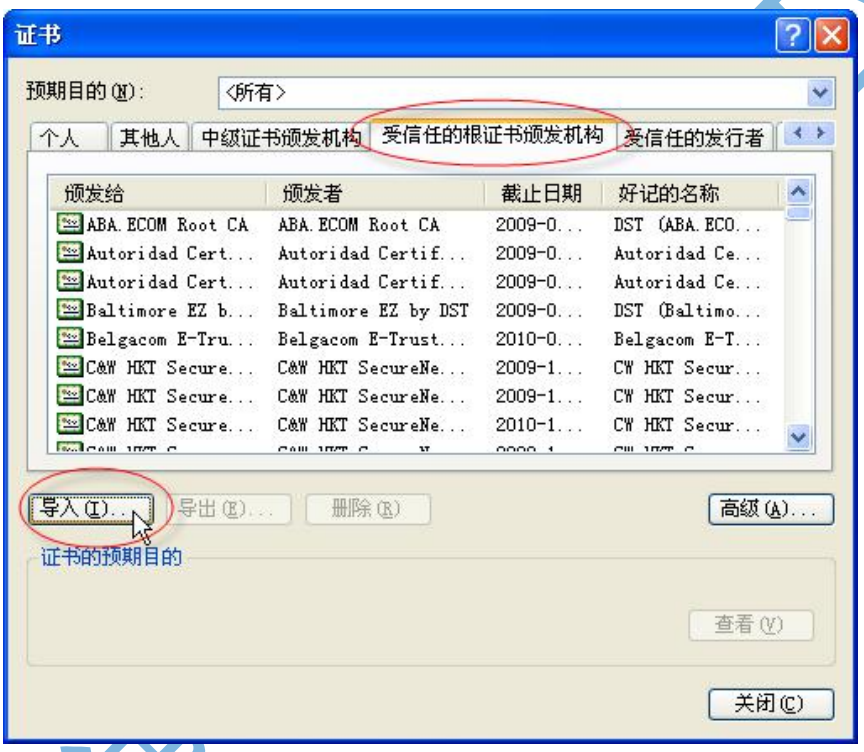
3.8.1　文件形式

数字证书里不但要包含 Clark 的公钥，还要包含 Clark 的自然信息（姓名、单位等），并且最重要的，要有证书颁发部门对这些信息的数字签名（每个证书颁发部门也都有自己的数字证书——称之为根证书——和与之配对使用的私钥）。这样我就可以验证数字证书的真伪了。

所以，让我们重新定义数字证书，数字证书是由一个权威机构发行的，至少包含一个公开密钥、证书持有人（或单位）的名称以及证书授权中心对这些信息的数字签名的文件。一般情况下证书中还包括密钥的有效时间，发证机关(证书授权中心)的名称，该证书的序列号等信息，证书的格式遵循 ITUT X.509 国际标准。如下图：



使用 IE 的菜单“工具 | Internet 选项... -> 内容 -> 证书... -> 受信任的根证书颁发机构”来查看 IE 中已经安装的根证书。点击“导入...”按钮可以导入新的根证书。



3.8.2 USB Key

3.8.2.1 概念与原理

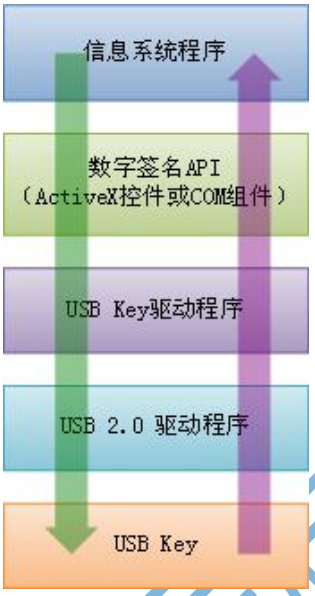
私钥保存在硬盘或者 U 盘中容易被人拷贝，我们需要的是无论如何也不可能被别人复制的私钥保存方案，USB Key 应运而生。

1. USB Key 是一种 USB 设备，外形就跟 U 盘一样，只不过**无法用它来存取文件**。
2. 证书发行单位会使用特殊的设备将你的**数字证书**、**私钥**和电子签章程序所要使用的**印章图片**烧录到 USB Key 中。
3. USB Key 本身还有一个简短的**使用密码**，每次加密前使用者必须输入正确的使用密码方能使用，这样即使 USB Key 不慎丢失，也不用担心了。

你无法使用资源管理器或木马程序取得 USB Key 中的私钥，当需要用私钥进行签名时，直接通过 USB Key 的驱动程序提供的 API 将明文传输到 USB Key 中，由 USB Key 中的加密芯片对明文进行加密，加密结果会以 API 函数的返回值的形式返回，这样就可以有效解决私钥被坏蛋复制的问题了。

3.8.2.2 缺点

USB Key 有一个不大不小的缺点——速度有点慢。例如我手里正在试用的这款 USB Key，连续签 10 个像“1234”这样的数据需要约 13 秒。这意味着如果你的信息系统只提供一次一条数据的签名方式，那么这 1 秒钟的延迟用户根本感觉不到；但是也有很多领导喜欢一次批量签名 100 条数据，那么就需要用 2 分钟来完成这项工作。经我本人测试以及向数字证书认证单位技术人员确认，速度的瓶颈主要在于数据往返于信息系统程序与 USB Key 之间所消耗的时间较长。所以很难通过优化信息系统程序或使用具有更快芯片的 USB Key 的方法来提高速度。如下图：



4 身份认证机制

4.1 定义

计算机网络世界中一切信息包括用户的身份信息都是用一组特定的数据来表示的，计算机只能识别用户的数字身份，所有对用户的授权也是针对用户数字身份的授权。

如何保证以数字身份进行操作的操作者就是这个数字身份合法拥有者，也就是说保证操作者的物理身份与数字身份相对应，身份认证就是为了解决这个问题，作为防护网络资产的第一道关口，身份认证有着举足轻重的作用。

4.2 身份认证方法的分类

在真实世界，对用户的身份认证基本方法可以分为这三种：

(1) 根据你所知道的信息来证明你的身份（what you know ，你知道什么 ）；

(2) 根据你所拥有的东西来证明你的身份（what you have ，你有什么 ）；

(3) 直接根据独一无二的身体特征来证明你的身份 (who you are ，你是谁) ，比如指纹、面貌等。

在网络世界中手段与真实世界中一致，为了达到更高的身份认证安全性，某些场景会将上面 3 种挑选 2 中混合使用，即所谓的双因素认证。

4.3 What you know

4.3.1 静态密码

用户的密码是由用户自己设定的。在网络登录时输入正确的密码，计算机就认为操作者就是合法用户。实际上，由于许多用户为了防止忘记密码，经常采用诸如生日、电话号码等容易被猜测的字符串作为密码，或者把密码抄在纸上放在一个自认为安全的地方，这样很容易造成密码泄漏。如果密码是静态的数据，在验证过程中 需要在计算机内存中和传输过程可能会被木马程序或网络中截获。因此，静态密码机制无论是使用还是部署都非常简单，但从安全性上讲，用户名/密码方式一种是不安全的身份认证方式。

4.4 What you have

4.4.1 智能卡（IC 卡）

一种内置集成电路的芯片，芯片中存有与用户身份相关的数据，智能卡由专门的厂商通过专门的设备生产，是不可复制的硬件。智能卡由合法用户随身携带，登录时必须将智能卡插入专用的读卡器读取其中的信息，以验证用户的身份。

智能卡认证是通过智能卡硬件不可复制来保证用户身份不会被仿冒。然而由于每次从智能卡中读取的数据是静态的，通过内存扫描或网络监听等技术还是很容易截取到用户的身份验证信息，因此还是存在安全隐患。

4.4.2 短信密码

短信密码以手机短信形式请求包含 6 位随机数的动态密码，身份认证系统以短信形式发送随机的 6 位密码到客户的手机上。客户在登录或者交易认证时候输入此动态密码，从而确保系统身份认证的安全性。

具有以下优点：

（1） 安全性

由于手机与客户绑定比较紧密，短信密码生成与使用场景是物理隔绝的，因此密码在通路上被截取几率降至最低。

（2） 普及性

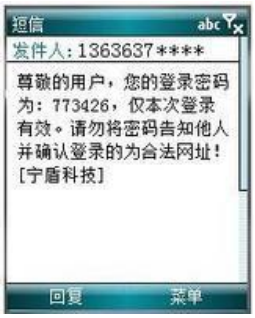
只要会接收短信即可使用，大大降低短信密码技术的使用门槛，学习成本几乎为 0，所以在市场接受度上面不会存在阻力。

（3） 易收费

由于移动互联网用户天然养成了付费的习惯，这和 PC 时代互联网截然不同的理念，而且收费通道非常的发达，如果是网银、第三方支付、电子商务可将短信密码作为一项增值业务，每月通过 SP 收费不会有阻力，因此也可增加收益。

（4） 易维护

由于短信网关技术非常成熟，大大降低短信密码系统上马的复杂度和风险，短信密码业务后期客服成本低，稳定的系统在提升安全同时也营造良好的口碑效应，这也是目前银行也大量采纳这项技术很重要的原因。



4.4.3 动态口令牌

目前最为安全的身份认证方式，也利用 what you have 方法，也是一种动态密码。

动态口令牌是客户手持用来生成动态密码的终端，主流的是基于时间同步方式的，每 60 秒变换一次动态口令，口令一次有效，它产生 6 位动态数字进行一次一密的方式认证。

由于它使用起来非常便捷，85%以上的世界 500 强企业运用它保护登录安全，广泛应用在 VPN、网上银行、电子政务、电子商务等领域。

动态口令是应用最广的一种身份识别方式，一般是长度为 5~8 的字符串，由数字、字母、特殊字符、控制字符等组成。用户名和口令的方法几十年来一直用于提供所属权和准安全的认证来对服务器提供一定程度的保护。当你每天访问自己的电子邮件服务器、服务器要采用用户名与动态口令对用户进行认证的，一般还要提供动态口令更改工具。现在系统（尤其是互联网上新兴的系统）通常还提供用户提醒工具以防忘记口令。

现行网银客户端的安全保障——动态口令牌(OTP，One time password)：

采用动态口令的认证方式就是在每次用户登录时除了输入常规的静态口令外，还要再输入一个每次都会变化的动态口令。

这个动态口令的获得方式有很多种，如刮刮卡式、二维矩阵卡式和电子令牌式，其中电子令牌就是我们所说的动态口令牌,如 VeriSign 的动态令牌 VIP 服务。刮刮卡和二维矩阵卡都是以纸质卡形式提供，但它们都存在着与生俱来的缺陷，刮刮卡有严格的使用次数限制，一般只能使用 30 次，而二维矩阵卡虽然可以无限次使用但很容易被复制，同动态口令牌相比刮刮卡和二维矩阵卡式都不具备时效性。现在很多国外银行和少数国内银行在网上银行应用中采用这种动态口令牌的方式。

采用动态口令牌方式的优点：

1. 无须安装软件，操作简单。与客户电脑无关，不需要安装其他任何程序即可直接使用网上银行服务。
2. 一次一密，解决了客户密码被盗的问题。这应该是动态口令牌在安全性方面带来的最大好处。

4.4.4 USB KEY

基于 USB Key 的身份认证方式是近几年发展起来的一种方便、安全的身份认证技术。它采用软硬件相结合、一次一密的强双因子认证模式，很好地解决了安全性与易用性之间的矛盾。USB Key 是一种 USB 接口的硬件设备，它内置单片机或智能卡芯片，可以存储用户的密钥或数

字证书，利用 USBKey 内置的密码算法实现对用户身份的认证。基于 USB Key 身份认证系统主要有两种应用模式：一是基于冲击/响应(挑战/应答)的认证模式，二是基于 PKI 体系的认证模式，目前运用在电子政务、网上银行。

4.5 Who you are

4.5.1 生物识别技术

运用 who you are 方法， 通过可测量的身体或行为等生物特征进行身份认证的一种技术。生物特征是指唯一的可以测量或可自动识别和验证的生理特征或行为方式。生物特征分为身体特征和行为特征两类。

身体特征包括：指纹、掌型、视网膜、虹膜、人体气味、脸型、手的血管和 DNA 等；

行为特征包括：签名、语音、行走步态等。

目前部分学者将视网膜识别、虹膜识别和指纹识别等归为高级生物识别技术；

将掌型识别、脸型识别、语音识别和签名识别等归为次级生物识别技术；

将血管纹理识别、人体气味识别、 DNA 识别等归为“深奥的”生物识别技术。

指纹识别技术目前应用广泛的领域有门禁系统、微型支付等。

4.6 双因素身份认证

所谓双因素就是将两种认证方法结合起来，进一步加强认证的安全性，目前使用最为广泛的双因素有：

动态口令牌 + 静态密码

USB KEY + 静态密码

二层静态密码 等等。