3. 挑战/应答

先由服务器端产生具有随机性的挑战码，通过网络传递给用户（短信、网站等），用户端基于挑战码计算出应答码，即动态口令；然后服务器端再验证该应答码是否正确。

目前关于 OTP 的国际规范主要有：RFC 1760 (The S/KEY One-Time Password System)。

OTP 机制的优点包括：

① 用户再也不能选择脆弱的静态密码。

② 用户无需记忆繁多的口令，只需保护好 OTP 产生终端（如令牌、手机等）即可。

③ 用户无需担心口令被窃取，因为口令被使用后即失效。

④ 能有效阻止网络上针对口令的各种攻击手段。

⑤ 便于管理，撤销时仅需要收回令牌或将认证服务器上的对应用户删除即可。

⑥ 能有效防止各种口令破解工具的暴力破解。

5.3.3 数字签名

数字签名又称电子签名，数字签名背后的思想是模仿传统手写签名。该思想是能够以某种方式“签署”一份数字文档，该签名具有和物理签名一样的法律效力。与物理世界中的手写签名相对应，数字签名可以理解为数字世界中的电子签名。

在《电子签名法》中，电子签名是指数据电文中以电子形式所含、所附用于识别签名人身份并表明签名人认可其中内容的数据；数据电文是指以电子、光学、磁或者类似手段生成、发送、接收或者储存的信息；民事活动中的合同或者其他文件、单证等文书，当事人可以约定使用或者不使用电子签名、数据电文。

数字签名的功能主要包括：

① 接收方能够确认发送方的签名，但不能伪造。

② 发送方发出签过名的信息后，不能再否认。

③ 接收方对接收到的签名信息也不能否认。

④ 一旦发送方和接收方出现争执，仲裁者可有充足的证据进行评判。

数字签名目前只能采用非对称密码算法实现。

数字签名的技术流程描述如下：

① 信息发送者使用摘要算法对信息生成信息摘要。

② 信息发送者使用自己的私钥对信息摘要进行签名（加密）。

③ 信息发送者把信息本身和已签名的信息摘要一起发送出去。

④ 信息接收者使用相同的摘要算法对接收的信息本身生成新的信息摘要。

⑤ 信息接收者使用信息发送者的公钥对已签名的信息摘要进行验签（解密），获得信息发送者的信息摘要。

⑥ 信息接收者比较这两个信息摘要是否相同，如果相同则确认信息发送者的身份和信息没有被修改过；否则，则表示被修改过。

数字签名的制作和验证过程如图 5-6 所示。

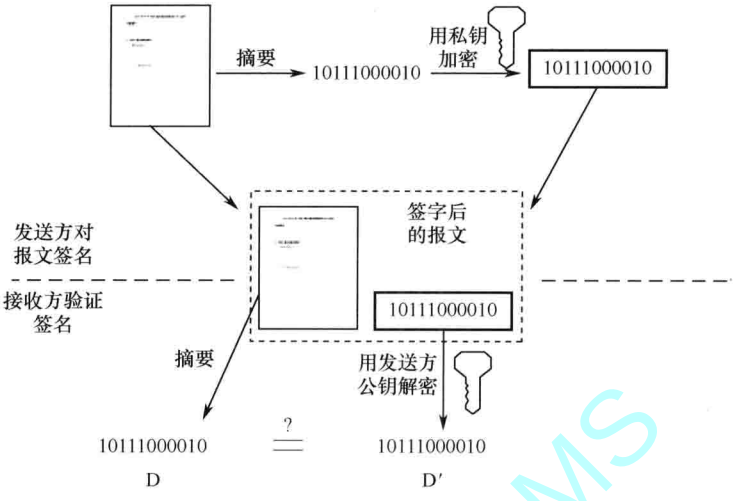

图 5-6 数字签名的制作和验证过程

PKCS #1 规范规定了使用 RSA 算法进行签名时的摘要格式。摘要格式用 ASN.1 描述如下：

```
DigestInfo ::= SEQUENCE {
    digestAlgorithm DigestAlgorithm,
    digest OCTET STRING
}
```

PKCS #7 和 RFC 2315 规范规定了数字签名消息的具体封装格式。数字签名消息封装格式用 ASN.1 描述如下：

```
SignedData ::= SEQUENCE {
    version Version,
    digestAlgorithms DigestAlgorithmIdentifiers,
    contentInfo ContentInfo,
    certificates [0] IMPLICIT ExtendedCertificatesAndCertificates OPTIONAL, crls [1] IMPLICIT
CertificateRevocationLists OPTIONAL,
     signerInfos SignerInfos }
```

5.3.4 数字信封

对称密码算法的优点是加解密运算非常快，适合处理大批量数据，但其密钥的分发与管理比较复杂。而非对称密码算法的特点是公钥与私钥分离，非常适合密钥的分发与管理；但其运行速度不快，又不适合处理大批量数据。如果将对称密码算法和非对称密码算法的优点结合起来，则既能处理大批量数据，又能简化密钥的分发与管理，于是数字信封机制应运而生。

数字信封并不需要分发和管理对称密钥，而是随机产生对称密钥，采用对称密码算法对大批量数据进行加密，并采用非对称密码算法对该对称密钥进行加密；解密时，先用非对称密码算法解密后获得对称密钥，然后使用对称密码算法解密后获得数据明文。

数字信封的功能类似于普通信封，采用对称密码算法对消息进行加密类似于信纸上的内容，采用非对称密码算法对对称密钥加密类似于信封，信封将信纸包装起来，保证了消息的安全性。

数字信封机制的具体流程如下：

① 消息发送方需要预先获得消息接收方的公钥。

② 消息发送方随机产生对称密钥，并用该密钥和对称算法对消息进行加密。

③ 消息发送方用消息接收方的公钥和非对称算法对上述对称密钥进行加密。

④ 消息发送方将消息密文和对称密钥密文一起发送给消息接收方。

⑤ 消息接收方收到消息密文和对称密钥密文。

⑥ 消息接收方使用自己的私钥和非对称算法对对称密钥密文进行解密后获得对称密钥明文。

⑦ 消息接收方使用上述对称密钥和对称算法对消息密文解密后获得消息明文。

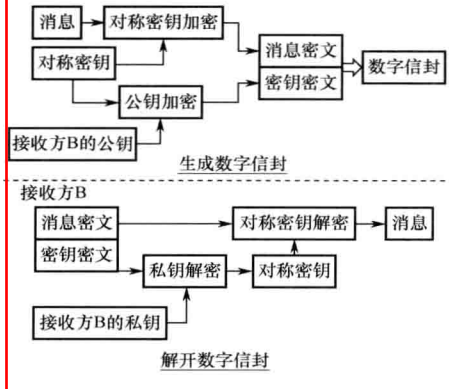

图 5-7　数字信封的生成和解开过程

数字信封的生成和解开过程如图 5-7 所示。

PKCS #7 规范规定了数字信封消息的具体封装格式。数字信封消息封装格式用 ASN.1 描述如下：

```
EnvelopedData ::= SEQUENCE {
    version Version,
    recipientInfos RecipientInfos,
    encryptedContentInfo EncryptedContentInfo }
RecipientInfos ::= SET OF RecipientInfo
EncryptedContentInfo ::= SEQUENCE {
    contentType ContentType,
    contentEncryptionAlgorithm ContentEncryptionAlgorithmIdentifier,
    encryptedContent [0] IMPLICIT EncryptedContent OPTIONAL }
```

5.4　密码应用实践

5.4.1　软件加密与硬件加密

软件加密是指加解密操作全部使用软件模块实现；硬件加密是指加解密操作全部使用专用的硬件设备实现，如加密机、加密卡或 IC 卡等。

软件加密和硬件加密的区别主要体现在以下几个方面。

1．密钥的安全性

软件加密时，密钥存储在内存或硬盘中，很容易被获取。

而硬件加密时，密钥的生成及保存均在硬件加密设备内部，而且采用了各种防护措施（如开机密钥自毁、密钥成分背对背输入等措施）防止密钥泄露；同时各种密码运算均在加密机内部进行，从而可以对密钥进行根本性的安全保护。

2．加密过程中的数据安全性

软件加密时很多重要数据或敏感信息（如用来做密码运算的密钥，或顾客的个人密码）都会在某一时刻以明文形式出现在计算机的内存或磁盘中。几乎所有应用业务系统中均采用目前流行的通用型操作系统，安全级别只达到 C2 级，其本身有很多的安全隐患和安全漏洞，而且属于国外产品，其内部留有很多后门已是众所周知，很容易被非法分子或恶意攻击者侵入或控制。所以采用软件加密方式，将使不法分子有机会从不安全的操作系统中对关键数据进行读取、利用、修改或删除，无法保证加密过程中的安全性。

采用硬件加密方式时，所有涉及的重要数据或敏感信息（如用来做密码运算的密钥，或顾客的个人密码）的加解密操作和密码运算均在加密机内部完成，不可能有任何危害客户利益的资料暴露于加密机之外，从而从根本上保证了加密过程中重要数据或敏感信息的安全性。

3．加密运算速度

软件加密通过在业务主机上运行加密软件来实现加密功能，要占用主机资源，一般来说其运算速度较硬件加密设备要慢很多。

而硬件加密方式是通过独立于主机系统之外的硬件加密设备实现的，硬件加密模块内部本身具有很强大的加解密处理模块，凡涉及加密运算的功能都由硬件加密设备来处理，几乎不占用主机资源，可以显著提高业务系统的处理能力。

4．运营维护方便性

软件加密模块一般与业务系统运行在相同主机中，模块独立性差，升级维护困难，很难保证业务系统 7×24 小时永不间断的服务宗旨，一旦出现故障，将会给客户的经济利益和社会形象带来巨大影响。

硬件加密模块与各种业务处理系统物理上独立，模块独立性好，升级维护简单，能够从根本上保证业务系统 7×24 小时永不间断的服务宗旨。

5．密码设备本身的安全性

软件加密自身安全性很难保证。

硬件加密模块能保证自身的安全性。国际上有专门的加密设备安全规范，如 FIPS PUB 140-2 加密模块的安全要求 Level 1~4，国内加密设备在上市前需要通过国家密码管理部门制定的安全测试规范。例如，FIPS 140-2 规定，密码设备必须采用双重控制和知识分离的密钥管理规范；双重控制是指两人或以上的人同时操作才能正确操作密码设备，以防止机密信息被单个人进入密码设备后掌控；知识分离是指由两人或以上的人分别安全保管不同的密钥段，只有同时操作才能重新产生密钥，以防止关键密钥被单个人掌控。FIPS 140-2 还规定，如欲非法获得或修改被输入、保存或处理的敏感信息，必须采用物理方法侵入密码设备；同时要求任何成功的物理侵入将导致密码设备在物理上的损毁，而且无法在不被发现的情况下复原。

5.4.2 网络层加密与应用层加密

按照 TCP/IP 分层模型，网络层加密是指在 IP 层对数据进行加解密处理，而应用层加密是指在应用层对数据进行加解密处理。

网络层加密主要采用 IP 加密技术，对 IP 层的所有数据进行加密，与应用系统无关。其优点是对应用系统透明、实施简单、能防止外部攻击；缺点是与应用系统脱离，无法做到选择性的加密保护，无法做到防止内部攻击，也无法保证敏感数据的全程安全性。

应用层加密的优点是能够做到选择性的加密保护，能保证敏感数据的全程安全性，能够同时防止外部攻击和内部攻击；其缺点是与应用系统结合紧密、开发工作量大、实施难度大。

网络层加密和应用层加密在保护数据全程安全性方面的比较见表 5-4。

表 5-4 网络层加密和应用层加密的比较

比较项		网络层加密	应用层加密
数据存储安全		不能	能
数据传输安全	主干网	能	能
	局域网	不能	能
数据处理安全		不能	能
数据输入安全		不能	能

5.4.3 密钥管理的基本原则

密钥管理主要遵循以下基本原则。

1. 保证密钥自身的机密性（Keys Are Secret）

除非对称公钥外，用于加解密的密钥和用于生成密钥的敏感资料必须保持机密，不允许任何人知道任何密钥。

2. 限制密钥的存在形式（Keys Have Permissible Forms）

除非对称公钥外，用于加解密的密钥只允许以下几种存在形式：

① 密钥明码（Clear Text）只能存在于硬件加密设备中。

② 在硬件加密设备外，密钥必须加密存储。

③ 在硬件加密设备外的密钥成分（用于合成密钥），必须保证双重控制、多人掌握。

3. 密钥分发（Key Deployment）

用于加解密的密钥，在进行分发时必须保证在切实可信、最少数目的地点进行。

4. 密钥隔离（Key Separation）

用于加解密的密钥，在产生和使用时必须用于其最初设计的目的。

5. 密钥同步（Key Synchronization）

必须提供机制来保证和验证已分发密钥的正确性，且该密钥的使用不会影响其他密钥的安全性。