第 1章 数字签名概述

1.1 数字签名的一般模型

随着计算机网络的发展，特别是电子商务的兴起，需要对消息进行消息完整性保护，对消息源进行鉴别，对交易进行认证，以及提供不可抵赖性保障。数字签名是手写签名的数字化形式。手写签名的基本特点是：能与被签名的信息在物理上不可分割，签名者不能否认自己的签名，签名不能伪造，签名容易被验证。数字签名是一串二进制数，应与被签名的信息“绑定”在一起。通常，数字签名应具有以下特性：

(1) 签名是可信的。任何人都可以验证签名的有效性。

(2) 签名是不可伪造的。除合法的签名者之外，任何其他人伪造签名是困难的。

(3) 签名是不可复制的。对某个消息的签名不能通过复制变为对另一个消息的签名。如果对某个消息的签名是从别处复制得到的，则任何人都可以发现签名和消息不一致，从而可以拒绝签名的消息。

(4) 签名的消息是不可改变的。经签名的消息不能被篡改。一旦已签名的消息被篡改，则任何人都可以发现消息和签名之间的不一致性。

(5) 签名是不可抵赖的。签名者事后不能否认自己的签名。

定义 1.1 一个数字签名方案是一个 5 元组$(M,S,K,\mathrm{SIGN},\mathrm{VRFY})$，满足如下的条件：

(1) M 是一个可能消息的有限集。

(2) S 是一个可能签名的有限集。

(3) 密钥空间 K 是一个可能密钥的有限集。

(4) 对每一个 $k=(k_s,k_v)\in K$，都对应一个签名函数 $\mathrm{Sign}_{k_s}\in\mathrm{SIGN}$ 和验证算法 $\mathrm{Vrfy}_{k_v}\in\mathrm{VRFY}$。每一个 $\mathrm{Sign}_{k_s}:M\to S$ 和验证函数 $\mathrm{Vrfy}_{k_v}:M\times S\to\{\mathrm{True},\mathrm{False}\}$ 是一个对任意消息 $m\in M$ 和任意签名 $s\in S$ 满足下列方程的函数：

$$\mathrm{Vrfy}(m,s)=\begin{cases}\mathrm{True}\ s=\mathrm{Sign}_{k_s}(m)\\ \mathrm{False}\ s\neq\mathrm{Sign}_{k_s}(m)\end{cases}$$

对每一个 $k\in K$，函数 Sign_{k_s} 和 Vrfy_{k_v} 都是多项式时间可计算的函数。Vrfy_{k_v} 是一个公开函数，k_v 为公钥(验证密钥)；Sign_{k_s} 是一个密码函数，k_s 为私钥(签名密钥)，要秘密保存。

第 7 章

代 理 签 名

先看一个场景，公司的经理外出度假期间，让他的秘书代理公司的业务，包括以自己的名义在一些文件上签名。通常是经理将自己的名章交给秘书，让秘书代表自己盖章，即将自己的签名权委托给代理人。

数字签名权利的委托(delegation of the digital signing power)是数字化的信息社会必然遇到的一种现象，但是，将自己的签名权利委托他人的时候，必须考虑以下几个问题：

(1) 安全性(security)。一般来说，一个人将数字签名权力委托给代理人的时候，希望代理人只能代表他在特定的时间对特定的文件生成数字签名，而不希望代理人“滥用”他的数字签名权力，且不希望非法的攻击者能因此伪造出有效的数字签名。

(2) 实用性(practicability)。委托数字签名权力的方法方便、有效，容易实现。

(3) 效率(efficiency)。委托权力的方法具有较高的速度和较小的计算复杂性、通信复杂性等。

7.1 代理签名的基本概念和分类

代理签名(proxy signature)，又称为委托签名，是 1996 年由 Mambo、Usuda 和 Okamoto 在 ACM CCS96 会议上首次提出的[①]。代理签名是指在一个签名方案中，原始签名人(original signer)把他的签名权授予代理签名人(proxy signer)，然后代理签名人代表原始签名人生成有效的签名。

一个代理签名体制可由以下几个部分组成：

(1) 参数产生。选定签名体制的参数、用户的密钥等。

(2) 数字签名权利的委托。原始签名人将自己的签名权利委托给代理签名人。

(3) 代理签名的生成。代理签名人代表原始签名人生成数字签名。

(4) 代理签名的验证过程。验证人验证代理签名的有效性。

以上是一个直观的定义，下面给出一个形式化的定义。这也是定义签名方案的第一步。

定义 7.1 设 A，B 是一个数字签名方案(M，S，SK，PK，GenKey，Sign，Vrfy)的两个用户，其中 M 是消息集合，S 是签名集合，SK 是私钥集合，PK 是公钥集合，GenKey 表示

---

① M. Mambo, K. Usuda, E. Okamoto. Proxy Signature for Delegating Signing Operation. Proc. of ACM CCS96, 48-57.

密钥生产算法集合，Sign 表示签名生成算法的集合，Vrfy 是签名验证算法集合，它们的私钥、公钥分别是(x_A, y_A)，$(x_B, y_B) \in SK \times PK$。如果以下条件成立：

(1) A 利用他的私钥 x_A 计算出一个数 δ，然后将 δ 秘密交给 B；任何人(包括 B)在试图求出 x_A 时，δ 不会对求解有任何帮助。

(2) 代理签名者 B 可以利用 δ 和 x_B 生成一个新的签名密钥 $\delta_{A\to B}$。

(3) 存在一个公开的验证算法 $Vrfy_{A\to B}: PK \times S \times M \to \{True, False\}$，使得对任何 $s \in S$ 和 $m \in M$，都有 $Vrfy_{A\to B}(y_A, s, m) = True \Leftrightarrow s = Sign(\delta_{A\to B}, m)$。

(4) 任何人在试图求解 x_A，x_B，δ 和 $\delta_{A\to B}$时，任何数字签名 $Sign(\delta_{A\to B}, m)$都不会对求解有任何帮助。

则称用户 A 将他的签名权力委托给用户 B，且称 A 为原始签名人，B 为 A 的代理签名人，δ 为委托密钥(delegating key)，$\delta_{A\to B}$为代理签名密钥(proxy signing key)，以代理签名密钥对消息 m 生成的签名 $Sign(\delta_{A\to B}, m)$为 A 的代理签名，能够生成代理签名的数字签名体制称为代理签名体制。

代理签名体制应满足如下基本性质：

(1) 不可伪造性(unforgeability)。除了原始签名人外，任何人(包括代理签名人)都不能生成原始签名人的普通数字签名。这个性质是数字签名体制的基本要求，保证了原始签名人的基本安全要求。

(2) 代理签名的不可伪造性。除了代理签名人外，任何人(包括原始签名人)都不能生成有效的代理签名。如果原始签名人委托了多个代理签名人，那么任何代理签名人都不能伪造其他代理签名人的代理签名。这一性质保证了代理签名人的基本安全需求。

(3) 代理签名的可区分性(distinguishability)。任何一个代理签名都与原始签名人的普通数字签名有明显的区别，不同的代理签名人生成的代理签名之间也有明显的区别。这个性质和性质(1)、(2)结合起来可防止签名人之间的互相抵赖。

(4) 不可抵赖性(undeniability)。任何签名人(不论是原始签名人还是代理签名人)在生成一个数字签名后，不能再对它加以否认。这个性质可由性质(1)～(3)推导出来。

(5) 身份可识别性(identifiability)。原始签名人可以根据一个有效的代理签名确定相应的代理签名人的身份。利用这个性质，原始签名人可以对代理签名人进行监督，使代理签名人不能在不被发现的情况下滥用他的代理签名权力。

Mambo 等将代理签名分为 3 类：完全代理签名(full delegation)、部分代理签名(partial delegation)、具有证书的代理签名(delegation by warrant)。

(1) 完全代理签名即原始签名人直接将自己的私钥通过安全信道发送给代理签名人，这是一种平凡的代理签名。由于代理签名人产生的签名与原始签名者产生的签名是不可区分的，故不能制止签名滥用，且不具备可识别性。

(2) 部分代理签名中，代理签名的密钥是由原始签名人的密钥计算出来的，但由代理密钥计算不出原始签名的密钥。签名时用到原始签名人的公钥。

(3) 具有证书的代理签名方案中使用了一个称为委任状的文件来实现签名权的委托。代理签名人在签名时，用自己的签名密钥进行签名。一个有效的代理签名由代理签名人生成的签名和原始委任状组成。

部分代理签名和具有证书的代理签名比完全代理签名安全，而部分代理签名比具有证书的代理签名灵活、方便，故本节主要讨论部分代理签名。

部分代理签名可分为两种类型：

(1) 不保护代理的代理签名(proxy-unprotected proxy signature)。指定的代理签名者能够代表原始签名者产生有效代理签名，没有指定为代理签名者的第三方不能产生有效代理签名，但是原始签名者可产生有效代理签名，这时代理签名密钥就是 δ。

(2) 保护代理的代理签名(proxy-protected proxy signature)。指定的代理签名者能代表原始签名者产生有效代理签名。第三方都不能产生有效代理签名，而且原始签名者也不能产生有效代理签名。这是因为代理签名密钥由 δ 和代理签名人的私钥 x_B 两部分组成。

7.2　代理签名举例

7.2.1　MUO不保护代理的代理签名

下面介绍 M. Mambo、K. Usuda 和 E. Okamoto 提出的 MUO 方案。

假设$(M,S,\mathrm{SK},\mathrm{PK},\mathrm{GenKey},\mathrm{Sign},\mathrm{Vrfy})$是一个基于离散对数问题的数字签名体制，$p$ 是一个大素数，q 为 $p-1$ 或 $p-1$ 的大素数因子；$g\in\mathbb{Z}_p^*$，且 $g^q=1 \bmod p$。用户 A、B 的私钥和公钥分别是(x_A,y_A)，(x_B,y_B)，满足 $y_A=g^{x_A} \bmod p$，$y_B=g^{x_B} \bmod p$。

(1) 委托过程。

① A 随机选取一个数 $k\in\mathbb{Z}_p^*$，计算 $K=g^k \bmod p$。

② A 计算 $\delta=x_A+kK \bmod q$，将(δ,K)秘密发送给 B。

③ B 验证等式 $g^\delta=y_AK^K \bmod p$ 是否成立，如不成立，则要求 A 重新执行步骤①或终止。

(2) 代理签名的生成。

对消息 m，B 使用 δ 生成普通的数字签名 $s=\mathrm{Sign}_\delta(m)$，然后将$(s,K)$作为代表 A 对消息 m 生成的数字签名，即代理签名。

(3) 代理签名的验证。接收方收到了消息 m 和代理签名(s,K)，按如下步骤来验证代理签名的有效性：

① 计算 $v=y_AK^K \bmod p$。

② 验证 $\mathrm{Vrfy}(y_A,(s,K),m)=\mathrm{True}\Leftrightarrow\mathrm{Vrfy}(v,s,m)=\mathrm{True}$。

MUO 方案具有以下几个性质：

(1) 基本的不可伪造性。B 难以根据他所得到的(δ,K)计算出 x_A，从而不能伪造 A 的普通数字签名。同时也说明任何其他攻击者难以伪造 A 的普通数字签名。

(2) 代理签名的不可伪造性。由于 A 和 B 都知道(δ,K)，所以 A 和 B 都能生成代理签名(也说明这是不保护代理的代理签名)。但是除了 A 和 B 以外，其他任何人都难以伪造一个有效的代理签名。

(3) 代理签名的可区分性。代理签名(s,K)由两部分组成，一部分是普通数字签名

s,另一部分是某个数 K。由于代理签名比普通签名多出一部分(即 K),容易将代理签名与普通的数字签名区分开来。同时,不同的代理签名人的代理签名也可区分,假如除 B 外还有代理签名人 C,A 在委托过程中发送给 C 的消息是(δ', K'),其中 $K'=g^{k'} \bmod p$,$k\neq k'$,于是 $K\neq K'$,C 生成的代理签名为(s', K'),于是可将 B 和 C 的代理签名区分开来。

(4) 不可抵赖性。由于任何人都不能伪造 A 的普通数字签名,所以 A 不能否认其有效的数字签名。由于除了 A 和 B 外,任何人都不能伪造 B 的代理签名,所以 A 和 B 不能否认一个有效的代理签名,即一个有效的代理签名必然是 A 和 B 两者中的一个生成的。但是,A 和 B 之间可以相互抵赖,即声称代理签名是对方而不是自己生成的。

(5) 身份可识别性。在这个代理签名方案中,如果 A 在向 B 发送(δ, K)时,将 K 和 B 的身份保存在一起,那么当 A 看到一个有效的代理签名(s, K)时,就可以通过 K 识别 B 的身份。

上面给出的方案的步骤(2)中没有给出具体的签名方案。这里给出一个完整的步骤(2),并给出对应的步骤(3)。参数设置和原步骤(1)不变。

(1) 委托过程。

步骤同前。

(2) 代理签名的生成。对消息 m,B 执行如下操作:

① 选择随机数 $r\in\mathbb{Z}_q^*$,计算 $R=g^r \bmod p$。

② 计算 $s=r^{-1}(m-\delta R) \bmod (p-1)$。

代理签名为(R, s, K)。

(3) 代理签名的验证。接收方收到了消息 m 和代理签名(R, s, K),按如下步骤来验证代理签名的有效性:

① 计算 $v=y_A K^K \bmod p$。

② 验证 $g^m=R^s v^R \bmod p$ 是否成立,如果成立,则代理签名正确。

正确性证明留作练习。

通过完整的方案可以体会到,代理签名人用委托密钥 δ 签名,验证签名时用到原始签名人的公钥,$v=y_A K^K \bmod p$ 其实可视为“委托密钥对应的公钥”。委托密钥和原始签名人的公钥通过一个随机的“承诺”K 联系起来。

7.22 MUO保护代理的代理签名

7.2.1 节介绍了不保护代理的代理签名,即 A 和 B 都能生成代理签名。保护代理的代理签名的设计目标是使得只有代理签名人才可以生成代理签名,方法是:在步骤(2)中不用委托密钥 δ 签名,而是用代理签名密钥 $\delta_{A\to B}$ 签名,对消息 m 生成的签名 $\mathrm{Sign}(\delta_{A\to B}, m)$为 A 的代理签名,而不是将 $\mathrm{Sign}(\delta, m)$作为 A 的代理签名。

注意:代理签名密钥$(\delta_{A\to B})$由委托密钥 δ 和代理签名人的私钥生成,故只有代理签名人可以生成代理签名。

于是,得到保护代理的 MUO 代理签名方案。参数生成过程同 7.2.1 节。

(1) 委托过程。

① A 随机选取一个数 $k\in\mathbb{Z}_p^*$,计算 $K=g^k \bmod p$。

② A 计算 $\delta=x_A+kK \bmod q$，将(δ,K)秘密发送给 B。

③ B 验证等式 $g^{\delta}=y_A K^K \bmod p$ 是否成立，如不成立，则要求 A 重新执行步骤①或终止。

④ B 计算 $\bar{\delta}=\delta+x_B y_B \bmod q$。

(2) 代理签名的生成。对消息 m，B 使用 $\bar{\delta}$ 生成普通的数字签名 $s=\mathrm{Sign}_{\bar{\delta}}(m)$，然后将$(s,K)$作为代表 A 对消息 m 生成的数字签名，即代理签名。

(3) 代理签名的验证。

接收方收到了消息 m 和代理签名(s,K)，按如下步骤来验证代理签名的有效性：

① 计算 $v=y_A K^K y_B^{y_B} \bmod p$。

② 验证 $\mathrm{Vrfy}(y_A,y_B,(s,K),m)=\mathrm{True}\Leftrightarrow\mathrm{Vrfy}(v,s,m)=\mathrm{True}$。