密 码 学

第十二讲　数字签名（2）

张焕国

武汉大学计算机学院

空天信息安全与可信计算教育部重点实验室

内容简介

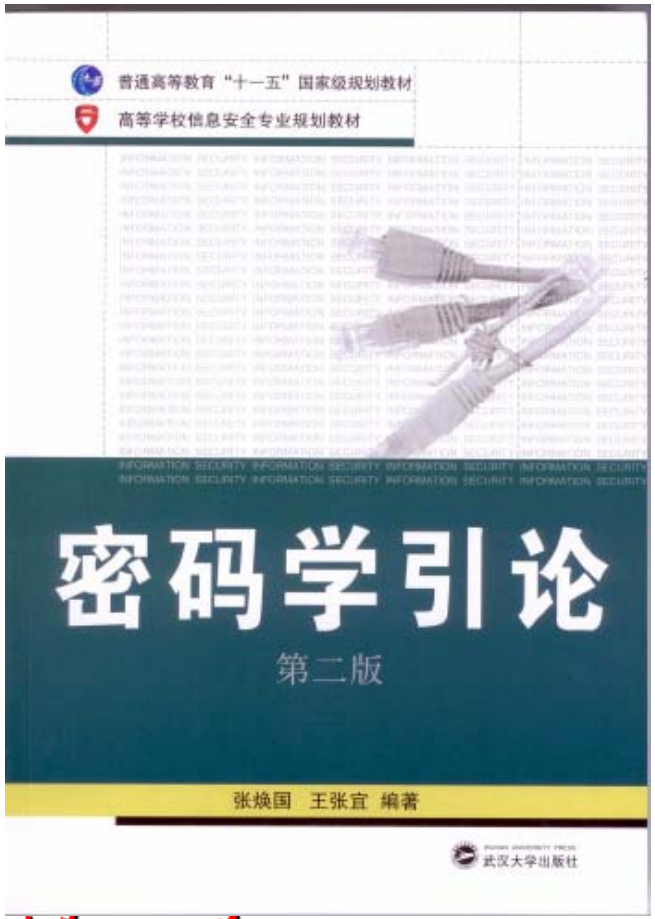
第一讲 信息安全概论
第二讲 密码学的基本概念
第三讲 数据加密标准（DES）
第四讲 高级数据加密标准（AES）
第五讲 中国商用密码（SMS4）
第六讲 分组密码的应用技术
第七讲 序列密码
第八讲 复习
第九讲 公钥密码（1）

内容简介

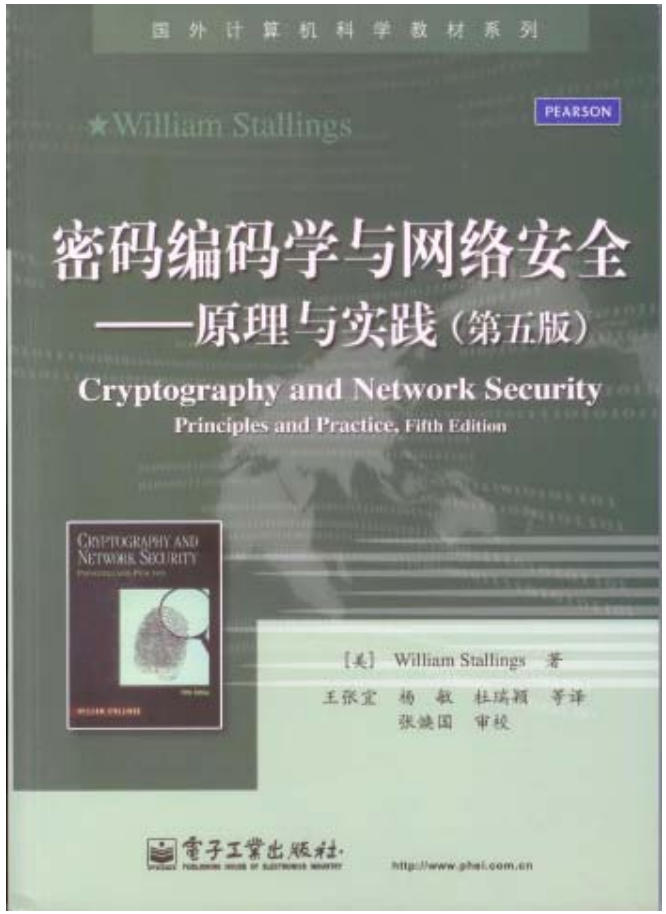
第十讲 公钥密码（2）
第十一讲 数字签名（1）
第十二讲 数字签名（2）
第十三讲 HASH函数
第十四讲 认证
第十五讲 密码协议
第十六讲 密钥管理（1）
第十七讲 密钥管理（2）
第十八讲 复习

教材与主要参考书

教材



参考书



三、利用公钥密码实现数字签名

2、利用ELGamal密码实现数字签名

⑴密钥选择

- 选P是一个大素数，p-1有大素数因子， α 是一个模p的本原元，将p 和 α 公开作为密码基础参数。
- 用户随机地选择一个整数x作为自己的秘密的解密钥，$1< x < p-1$ 。
- 计算 $y = \alpha^x \bmod p$，取y 为自己的公开的加密钥。

三、利用公钥密码实现数字签名

2、利用ELGamal密码实现数字签名

⑵ 产生签名

设明文为m，$0 \leqslant m \leqslant p-1$, 签名过程如下：

① 用户A随机地选择一个整数 k，$1<k<p-1$，且（k，$p-1$）=1；

② 计算$r = \alpha^k \bmod p$

③ 计算$s = (m - xr) k^{-1} \bmod p-1$

④ 取（r，s）作为m的签名,并以<m，r，s>的形式发给用户B。

m	r	s

三、利用公钥密码实现数字签名

2、利用ELGamal密码实现数字签名

(3) 验证签名

- 用户B接收：$<m, r, s>$
- 用户B用A的公钥y验证：$a^m = y^r r^s \mod p$是否成立，若成立则签名为真，否则签名为假。
- 签名的可验证性证明如下：

  因为$s=(m-xr)k^{-1} \mod p-1$，

  所以$m=xr+ks \mod p-1$，

  故$a^m = a^{xr+ks}=(a^x)^r(a^k)^s = y^r r^s \mod p$，故签名可验证。

三、利用公钥密码实现数字签名

2、利用ELGamal密码实现数字签名

(3) 验证签名

- 安全性
  - 从公开密钥求私钥是离散对数问题。
  - p-1要有大素数因子,否则易受攻击。
  - 为了安全，随机数k应当是一次性的。否则时间一长，k将可能泄露。因为，

$$x=(m-ks)r^{-1} \bmod p-1,$$

如果知道了m，便可求出保密的解密钥。

三、利用公钥密码实现数字签名

2、利用ELGamal密码实现数字签名

⑶ 验证签名

- 安全性
  - 如果k重复使用，如用k签名m_1和m_2。于是，

$$m_1 = xr + ks_1 \bmod p-1,$$

$$m_2 = xr + ks_2 \bmod p-1,$$

于是，$(s_1-s_2)k=(m_1-m_2) \bmod p-1$

如果知道了m_1和m_2，便可求出k，进而求出保密的解密钥。

  - 由此可知，不要随便给别人签名。
  - 不要直接对m签名，而是对HASH(m)签名。

三、利用公钥密码实现数字签名

2、利用ELGamal密码实现数字签名

⑷、ELGamal密码签名的应用

- 安全，方便。
- 缺点：由于取（*r*，*s*）作为*m*的签名，所以数字签名的长度是明文的两倍，数据扩张一倍。
- 美国数字签名标准（DSS）的签名算法DSA是它的一种变形。
- 俄罗斯数字签名标准（GOST）也是采用一种ELGamal密码签名变种。

三、利用公钥密码实现数字签名

3、利用椭圆曲线密码实现数字签名

- 一个椭圆曲线密码由下面的六元组描述：

$$T=<p,a,b,G,n,h>$$

其中，p为大于3素数，p确定了有限域GF（p）；元素$a,b\in GF(p)$,a和b确定了椭圆曲线；G为循环子群E_1的生成元，n为素数且为生成元G的阶，G和n确定了循环子群E_1。

$$y^2=x^3+ax+b \bmod p$$

三、利用公钥密码实现数字签名

3、利用椭圆曲线密码实现数字签名

(1) 密钥选择

随机数$d \in \{1,2,\cdots,n-1\}$为用户的私钥，公开钥为Q点，$Q=dG$。

(2) 产生签名

- 选择一个随机数k，$k \in \{1,2,\cdots,n-1\}$;
- 计算点R（x_R, y_R）$= kG$，并记 $r = x_R$;
- 利用保密的解密钥d计算：

$$s=(m-dr)k^{-1} \bmod n$$

- 以<r,s>作为消息m的签名，并以<m，r，s>的形式传输或存储。

m	r	s

三、利用公钥密码实现数字签名

3、利用椭圆曲线密码实现数字签名

(3) 验证签名

① 计算$s^{-1} \bmod n$;

② 利用公开的加密钥 Q 计算

$U(x_U, y_U) = s^{-1}(mG - rQ)$;

③ 如果$x_U = r$，则<r,s>是用户A对m的签名。

- 证明：因为$s = (m - dr)\, k^{-1} \bmod n$，所以

$$s^{-1} = (m - dr)^{-1}\, k \bmod n,$$

- 所以$U(x_U, y_U) = (m - dr)^{-1}\, k\ (mG - rQ)$

$= (m - dr)^{-1}\ (mkG - krdG) = (m - dr)^{-1}\ (mR - rdR)$

$= (m - dr)^{-1}\, R\ (m - dr) = R(x_R, y_R)$。

所以 $x_U = x_R = r$.

三、利用公钥密码实现数字签名

3、利用椭圆曲线密码实现数字签名

(4) 椭圆曲线密码签名的应用

- 安全，密钥短、软硬件实现节省等特点。
- **2000**年美国政府已将椭圆曲线密码引入数字签名标准**DSS**。
- 中国也采用椭圆曲线密码签名。

三、利用公钥密码实现数字签名

4、中国的椭圆曲线数字签名方案

(1) 用户A产生签名的算法

设待签名消息为M，Z_A=Hash($ENTL_A \| ID_A \| a \| b \| x_G \| y_G \| x_A \| y_A$)，$ID_A$是A的标识符，$ENTL_A$是$ID_A$的长度。$d_A$是私钥，公钥$P_A=d_AG=(x_A,y_A)$。

① 置$M^*=Z_A \| M$;

② 计算e = Hash(M^*);

③ 用随机数发生器产生随机数$k \in [1,n-1]$;

④ 计算椭圆曲线点$G_1(x_1,y_1)=kG$;

⑤ 计算$r = (e+x_1) \bmod n$，若$r=0$或$r+k=n$则返回③；

⑥ 计算$s = ((1+d_A)^{-1} \cdot (k - r \cdot d_A)) \bmod n$，若$s=0$则返回③;

⑦ 以(r,s)作为对消息M的签名。

三、利用公钥密码实现数字签名

4、中国的椭圆曲线数字签名方案

(2)验证签名的算法

设收到的消息为M'，数字签名为(r', s')，P_A为A的公钥。

① 检验$r' \in [1, n-1]$是否成立，若不成立则验证不通过；

② 检验$s' \in [1, n-1]$是否成立，若不成立则验证不通过；

③ 置$(M')^* = Z_A \| M'$；

④ 计算$e' = \mathrm{Hash}((M')^*)$；

⑤ 计算$t = (r' + s') \bmod n$，若$t = 0$，则验证不通过；

⑥ 计算椭圆曲线点$G_1'(x_1'; y_1') = s'G + tP_A$；

⑦ 计算$R = (e' + x'_1) \bmod n$，检验$R = r'$是否成立，若成立则验证通过；否则验证不通过。

三、利用公钥密码实现数字签名

4、中国的椭圆曲线数字签名方案

(3)验证的正确性

① 因为签名时确保了$r \in [1,n-1]$且$s \in [1,n-1]$，如果没有篡改和错误，则有$r' \in [1,n-1]$且$s' \in [1,n-1]$。对此进行检验，可发现其是否有错误和错改，确保其正确性。

② 因为签名时确保了$r \neq 0$ 且$s \neq 0$，如果$t=r+s=0 \bmod n$,则$r+s$是n的整数倍。而签名时确保了(r+k) 不是n的整数倍。$r+s=r+(k-rd)/(1+d)=(r+k)/(1+d)$,所以$(r+k)/(1+d)$也不是$n$的整数倍，否则因为$d$是正整数，将导致$(r+k)$是$n$的整数倍，产生矛盾。这说明，如果$r'$和$s'$是正确的，则$t = (r' + s') \bmod n = t \neq 0$。

三、利用公钥密码实现数字签名

4、中国的椭圆曲线数字签名方案

(3)验证的正确性

③ 一方面，$sG + tP_A = sG + (r+s)(dG) = (s+rd+sd)G$，另一方面 $(s+rd+sd) = s(1+d)+rd = ((k-rd)/(1+d))(1+d)+rd = k$，所以 $sG + tP_A = kG = G_1(x_1, y_1)$。如果 x_1 是和 e 正确的，则有 $e' = e$，$x'_1 = x'$，因而有 $R = r$.

四、盲签名

- 在普通数字签名中，签名者总是先知道数据的内容后才实施签名，这是通常的办公事务所需要的。但有时却需要某个人对某数据签名，而又不能让他知道数据的内容。称这种签名为盲签名（**Blind Signature**）。在无记名投票选举和数字货币系统中往往需要这种盲签名，
- 盲签名在电子商务和电子政务系统中有着广泛的应用。

四、盲签名

- 盲签名与普通签名相比有两个显著的特点：

①签名者不知道所签署的数据内容；

②在签名被接收者泄露后，签名者不能追踪签名。即：如果把签名的数据给签名者看，他确信是自己的签名，但他无法知道什么时候对什么样的盲数据施加签名而得到此签名数据。

四、盲签名

- 盲签名的技术思想
  - 接收者首先将待签数据进行盲变换，把变换后的盲数据发给签名者。
  - 经签名者签名后再发给接收者。
  - 接收者对签名再作去盲变换，得出的便是签名者对原数据的盲签名。
  - 这样便满足了条件①。要满足条件②，必须使签名者事后看到盲签名时不能与盲数据联系起来，这通常是依靠某种协议来实现的。

四、盲签名

- 盲签名原理图：

四、盲签名

1、RSA盲签名

$$A \xrightarrow{M} B$$

① A对消息M进行盲化处理：他随机选择盲化整数k,$1<k<M$，并计算

$$T=M(k)^{e} \bmod n \text{ 。}$$

② A把T发给B。

③ B对T签名：

$$T^{d}=(Mk^{e})^{d} \bmod n$$

$$=(M)^{d}k \bmod n$$

四、盲签名

1、RSA盲签名

④ B把他对T的签名发给A。

⑤ A通过计算得到B对M的签名。

$$S = T^d / k \mod n$$

$$= M^d \mod n$$

● 正确性证明：

因为$T^d = (Mk^e)^d = M^d k \mod n$，所以

$T^d / k = M^d \mod n$，而这恰好是B对消息M的签名。

四、盲签名

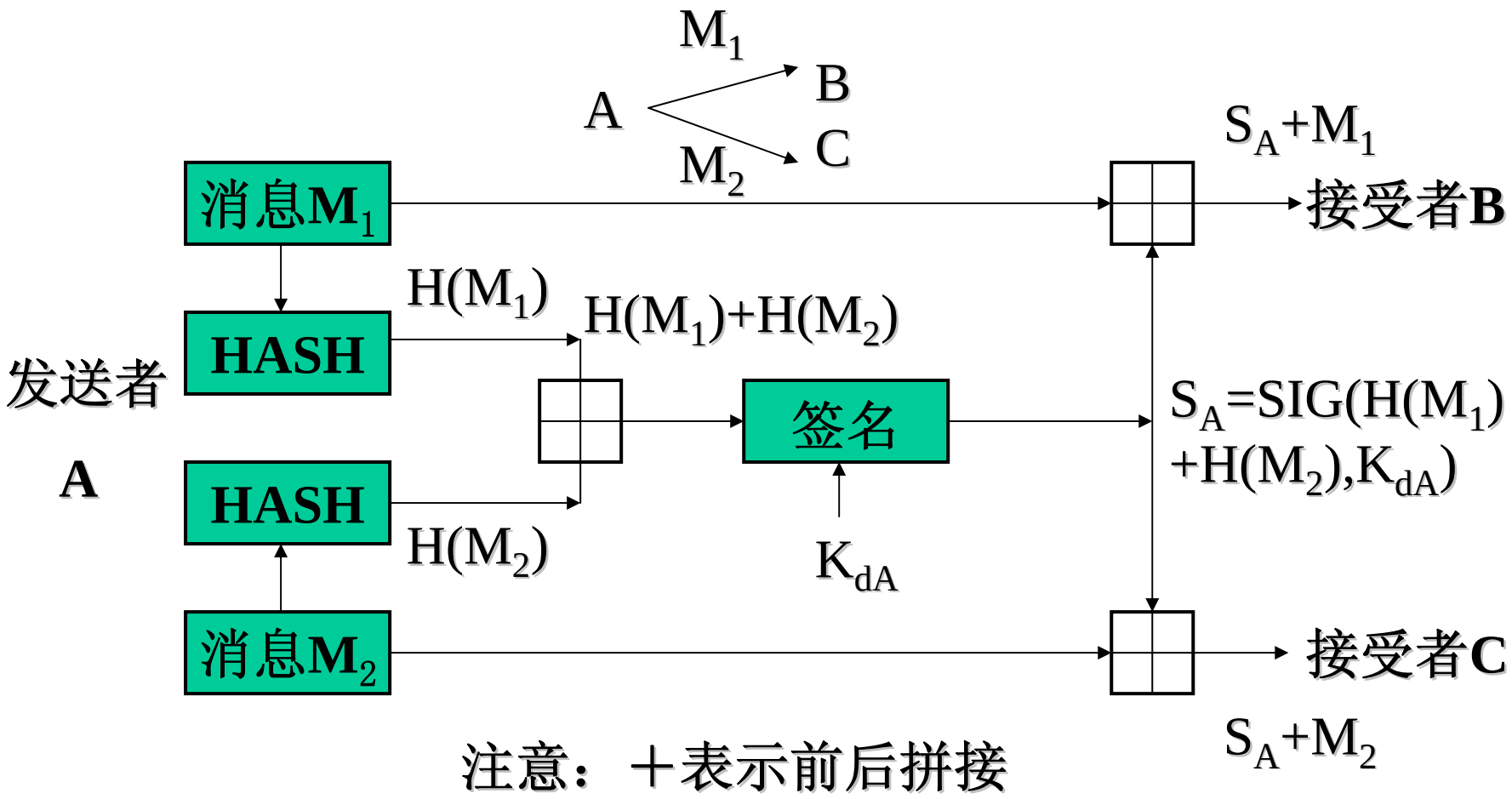
- 盲签名在某种程度上保护了参与者的利益，但不幸的是盲签名的匿名性可能被犯罪份子所滥用。为了阻止这种滥用，人们又引入了公平盲签名的概念。公平盲签名比盲签名增加了一个特性，即建立一个可信中心，通过可信中心的授权，签名者可追踪签名。

四、盲签名

2、双联签名

- 双联签名是实现盲签名的一种变通方法。它利用协议和密码将消息与人关联起来而并不需要知道消息的内容。从而实现盲签名。
- 双联签名采用单向**HASH**函数和数字签名技术相结合，实现盲签名的两个特性。

四、盲签名



注意：＋表示前后拼接

武漢大学

四、盲签名

- 接收者B和C都可用发信者A的公开钥验证双联签名S_A。，获得H（M_1）和 H（M_2）。
- B只能阅读M_1，计算H（M_1），通过H（M_1）验证M_1是否正确。而对消息M_2 却一无所知，但通过验证签名S_A可以相信消息M_2的存在。
- 同样，C也只能阅读M_2，计算H（M_2），通过H（M_2）验证M_2 是否正确。而对消息M_1 却一无所知，但通过验证签名S_A可以相信消息M_1的存在。

作 业 题

1、p189第7题。

谢　谢！