第33卷 第10期 计 算 机 工 程 2007年5月
Vol.33 No.10 Computer Engineering May 2007
·安全技术· 文章编号：1000—3428(2007)10—0136—02 文献标识码：A 中图分类号：TP301.6

基于椭圆曲线密码体制的电子公文流转方案

杨世平，李 祥
(贵州大学计算机软件与理论研究所，贵阳 550025)

摘 要：流转公文的数字签名和用于加密信息的对称密钥交换都是建立在椭圆曲线之上，利用有限域上椭圆曲线的点群中的离散对数问题难解性增强了方案的安全性。通信各方的私钥和公钥对由自己产生，公钥均由 KDC 保存并根据用户使用申请实时分发，流转的电子公文和数字签名等信息使用 IDEA 算法进行加密传输，避免了公文在传输的过程中被第 3 人窃取或篡改，确保了数据的机密性、完整性和不可否认性。

关键词：椭圆曲线；数字签名；电子公文流转；IDEA；SHA-1

Scheme of Electronic Official-documents Interchange Based on Elliptic Curve Cryptosystem

YANG Shiping, LI Xiang
(Institute of Computer Software and Theory, Guizhou University, Guiyang 550025)

【Abstract】The digital signature and symmetric key exchange in the scheme both are established over a finite field and the computational intractability of the elliptic curve discrete logarithm problem (ECDLP) over the finite field enhances security of the scheme. Each entity in a network generates a pair of keys to be used for encryption and decryption of the transmitted messages, where the private one of the two keys must be kept secret and the public key is distributed in realtime on requests by KDC. The electronic official-documents and their digital signatures are encrypted with IDEA and then transmitted in the network. The scheme provides mutual authentication between entities and ensures confidentiality, integrity and nonrepudiation of the transmitted messages.

【Key words】Elliptic curves; Digital signature; Electronic official-documents interchange; IDEA; SHA-1

公文流转是电子政务的主要应用功能之一，未经安全处理的电子文档容易复制、修改或者伪造，事后予以否认等。如果没有安全的处理手段，通过网络交换平台传输的电子文档很难保证其保密性、完整性和不可否认性。本文提出一种电子公文在网络环境下安全流转的方案，其核心是利用椭圆曲线密码体制来实现电子公文的数字签名和加密密钥交换。

1 椭圆曲线密码体制的相关概念

将椭圆曲线(Elliptic Curve)用于密码的算法，是利用有限域上椭圆曲线的点构成的群实现离散对数密码算法[1~3]。椭圆曲线公钥系统(ECC)所需要的密钥较短，与基于大整数因式分解问题的公钥系统(RSA)和有限域上离散对数问题的公钥系统(DSA)相比，在相同的安全级别上，椭圆曲线公钥系统所需要的密钥长度要小得多。在安全性相当的情况下，使用ECC比使用RSA具有计算上的优势。椭圆曲线密钥体制可以用于加/解密、密钥交换和数字签名。

满足椭圆曲线方程的一个有序对偶称为一个点，常用大写字母 P,Q 等来表示。点 P 还可以用其坐标表示为 $P=(x,y)$，x,y 属于该有限域。P 的 x,y 坐标分别表示为 $P.x$ 和 $P.y$。

椭圆曲线上的点乘(点的数乘)运算是椭圆曲线密码系统的核心运算之一。椭圆曲线上的点乘运算定义如下：给定一条椭圆曲线 E 和曲线上的一点 P，曲线 E 上的 P 点的点乘 xP 定义为点 P 与自身相加 x 次之和，即 $xP=P+P+\ldots+P$ 共 x 个 P 相加。

1.1 有限域上的椭圆曲线离散对数

对椭圆曲线关心的是将它定义在有限域 $GF(p)$上的情况。若 E 为 $GF(p)$上的椭圆曲线，P 为 E 上的一点，则 E 上关于 P 的椭圆曲线离散对数问题为：给定一点 $N \in E$，求解整数 $x \in GF(p)$，使 $xP = N$。离散对数的求解是很困难的，椭圆曲线离散对数问题(ECDLP)比有限域上的离散对数问题更难求解。在有限域 $GF(p)$上选择一个椭圆曲线 E 及一个具有比较高阶的基点 $P \in E$，计算该点的数乘 kP，相对来说是容易的，但是在已知 P 和 kP 的情况下要求解 k 是很困难的。在实际的密码系统中的 $GF(p)$，模 p 可以达到上百比特，要计算 k 非常困难。

1.2 消息摘要函数

消息摘要函数 $h=H(m)$，m 是可变长度消息，h 是固定长度的函数值。消息摘要函数 h 的作用是对消息 m 产生一个“摘要”，使得接收方能够对消息 m 的完整性进行检验。

消息摘要函数满足如下性质：

(1)函数的输入长度任意，而输出值长度固定，一般为 128bits 或者 160bits。

(2)已知 m，求 $H(m)$比较容易。

(3)已知 h，求 m，使得 $H(m)=h$ 在计算上不可行，即单向性。

(4)已知 x，寻找不等于 x 的 y，使得 $H(x)=H(y)$在计算上不可行。

(5)对任意不相等的输入对偶(x,y)，$H(x)=H(y)$在计算上不可行。

从以上性质可以看出，输入消息的每一位的改变对输出

基金项目：教育部基金资助项目(200326)
作者简介：杨世平(1955－)，男，副教授，主研方向：计算机网络与信息安全；李 祥，教授
收稿日期：2006-06-28 **E-mail：**ysp@gzu.edu.cn

摘要都有影响。目前使用最为广泛的两个标准化的 Hash 函数是 MD5 和 SHA。在数字签名中一般使用发送方的私钥来对 $h=H(m)$进行加密来实现签名。

1.3 椭圆曲线数字签名算法

设系统参数为$(GF(p), E, P, n, H)$，$GF(p)$为有限域，E是$GF(p)$上的椭圆曲线，P为基点，其阶为大素数n，$(k, P_k = kP)$为私、公密钥对，待签名的消息为m，$H(\ )$为Hash函数，ECDSA算法如下：

(1)签名算法[3]如下：

1)选取随机整数，即$t, 1 \le t \le n-1$，计算$tP = (x, y), r = x \bmod n$；2)计算$e = H(m), s = t^{-1}(e + rk) \bmod n$；3)签名消息为$(r, s)$。

(2)验证算法如下：

1) 计算 $e = H(m), u = s^{-1}e \bmod n, v = s^{-1}r \bmod n$ ； 2) 计算 $uP + vP_k = (x', y'), r' = x' \bmod n$；3)当且仅当$r'=r$，接受签名。

(3)验证算法的正确性证明如下：

$e = ts - rk_i \bmod n$

因为 $u + vk_i = es^{-1} + rs^{-1}k_i = t - rs^{-1}k_i + rs^{-1}k_i = t \bmod n$，且 $nP = O$，所以$uP + vP_{k_i} = uP + vk_iP = (u + vk_i)P = tP$。

1.4 椭圆曲线 Diffie-Hellman 密钥交换算法

椭圆曲线Diffie-Hellman密钥交换算法[5,6,8]的目的是使两个用户能够安全地交换一个密钥，以用于后继的消息加密，算法本身只用于密钥的交换。原Diffie-Hellman密钥交换算法的安全性是基于有限域上离散对数难解问题，而椭圆曲线Diffie-Hellman密钥交换算法有更好的安全性(参见 1.1 节)。

A 和 B 选择一个共同定义在有限域上的椭圆曲线 E 和一个阶为大素数 n 的基点 P。算法如下:

(1)A选择一个整数n_A，满足$n_A < n$，A计算其公开密钥 $P_A = n_A \times P$ 发送给B，n_A作为A的私有密钥被秘密保存。

(2)B选择一个整数n_B，满足$n_B < n$，保存为自己的私钥，并计算其公开密钥 $P_B = n_B \times P$ 发送给A。

(3)A 产生秘密密钥 $K = n_A \times P_B$，B 产生秘密密钥 $K = n_B \times P_A$。

产生的密钥是相同的，因为

$K = n_AP_B = n_A(n_BP) = n_B(n_AP) = n_BP_A$

2 电子公文安全流转方案

本方案要解决如下问题：参与公文流转各方的认证问题，流转的公文的机密性，完整性和不可否认问题。

设系统参数为$(GF(p), E, P, n, SHA\text{-}1)$并将其公开，$GF(p)$为有限域，$E$是$GF(p)$上的椭圆曲线，$P$为基点，基点的阶为大素数$n$，$(k_i, P_{k_i} = k_iP, k_j, P_{k_j} = k_jP, i \ne j)$为网络中用户$U_i$和$U_j$私钥和公钥对，$(k_{kdc}, P_{k_{kdc}} = k_{kdc}P)$是密钥分发中心(Key Distribution Center, KDC)的私钥和公钥对。KDC根据用户的请求，负责网络中公钥的管理和发放。网络中用户的私钥 $k_i (i = 1, 2, 3, ...)$由用户 $U_i (i = 1, 2, 3, ...)$ 持有并保密，网上所有用户的标识 ID_i 和公钥 $P_{k_i} (i = 1, 2, 3, ...)$ 则通过秘密通道在KDC上注册并保存，待发送的电子公文为m，$SHA\text{-}1(\ )$为输入为不超过2^{64}bits长的任意消息，输出为一个 160bits长的消息摘要的Hash函数。方案如下：

(1)用户U_i对电子公文进行签名

1)选取随机整数 t，$1 \le t \le n-1$，计算 $tP = (x, y), r = x \bmod n$。

2) $e = SHA\text{-}1(m), s = t^{-1}(e + rk_i) \bmod n$。

3) $M = m \| r \| s$。

(2)用户U_i和U_j从KDC获取对方的公钥

1) $U_i \to KDC : ID_i \| ID_j$。

2) $KDC \to U_i : E_{k_{kdc}}[ID_j \| P_{k_j}]$。

3) $U_i \to U_j : E_{P_{k_j}}[N_i \| ID_i]$，用$U_j$的公钥 P_{k_j} 加密临时值和标识 $N_i \| ID_i$。

4) $U_j \to KDC : ID_j \| ID_i \| E_{P_{k_{kdc}}}[N_i]$，用户$U_j$向KDC申请获取用户$U_i$的公钥 P_{k_i}。

5) $KDC \to U_j : E_{k_{kdc}}[ID_i \| P_{k_i}] \| E_{P_{k_j}}[E_{k_{kdc}}[N_i \| ID_i \| ID_j]]$；KDC用自己的私钥对用户$U_i$的标识$ID_i$和公钥 P_{k_i} 加密，同时用用户U_j的公钥 P_{k_j} 加密 $E_{k_{kdc}}[N_i \| ID_i \| ID_j]$ 并发送给用户U_j。

6) $U_j \to U_i : E_{P_{k_i}}[E_{k_{kdc}}[N_i \| ID_i \| ID_j] \| N_j]$；用户$U_j$用获取的用户$U_i$的公钥 P_{k_i} 加密 $E_{k_{kdc}}[N_i \| ID_i \| ID_j] \| N_j$ 发送给用户U_i，至此，双方完成了相互认证并获得了对方的公钥。

(3)用户U_i和用户U_j进行密钥交换获取加密密钥K_s

1)U_i选择随机整数 $1 \le r_i \le n-1$，计算 $R_i = r_iP = (x_i, y_i)$，并将R_i发送给U_j，类似地U_j选择随机整数 $1 \le r_j \le n-1$，计算 $R_j = r_jP = (x_j, y_j)$，并将R_j发送给U_i。

2)U_i计算 $S_i = (r_i + x_ik_i) \bmod n$，$U_j$计算 $S_j = (r_j + x_jk_j) \bmod n$。

3)U_i从U_j处收到R_j，并利用U_j的公钥 P_{k_j}，计算

$K_s = S_i(R_j + x_jP_{k_j}) = S_i(r_jP + x_jk_jP) = S_i(r_j + x_jk_j)P = S_iS_jP = (x_s, y_s)$

类似地，U_j从U_i得到R_i，并利用U_i的公钥 P_{k_i}，计算

$K_s = S_j(R_i + x_iP_{k_i}) = S_j(r_iP + x_ik_iP) = S_j(r_i + x_ik_i)P = S_iS_jP = (x_s, y_s)$

至此，通信双方得到会话密钥，此处，本文选取点K_s的坐标x_s作为加密密钥。

4) $U_i \to U_j : EN_{IDEA}(M \| N_j)_{K_s.x_s}$，用会话密钥$K_s.x_s$加密 $M \| N_j$，加密算法为IDEA。

(4)用户U_j解密收到的电子公文，并分离出N_j, m, r, s，检查临时值N_j以保证消息是当次的。

1)U_j：$M = DE_{IDEA}(EN_{IDEA}(M \| N_j)_{K_s.x_s})_{K_s.x_s}$；用户$U_j$用会话密钥$K_s.x_s$解密出 $M \| N_j$。

2)分离出N_j, m, r, s。

(5)用户U_j对收到的电子公文进行验证

1) $e = SHA\text{-}1(m), u = s^{-1}e \bmod n, v = s^{-1}r \bmod n$。

2)计算 $uP + vP_{k_i} = (x', y'), r' = x' \bmod n$。

3)当且仅当$r'=r$，接受签名，参见 1.3 节。

到此为止，一次完整的公文流转完成。

在方案运行开始之前，用户生成公钥对、私钥对，并通过秘密通道定期传送或更新至KDC。当用户U_i准备向U_j发送一个电子公文m时，首先进行如下处理：公文m由U_i使用SHA-1 算法压缩成为 160bits的消息摘要e，再通过椭圆曲线密码(ECC)签名算法，利用U_i的私钥k_i对消息摘要e进行签名运算得到s，(r, s)即为U_i对公文m的签名。电子公文m、r与s拼接在一起生成报文M。U_i向KDC申请获得U_j的公开密钥 P_{k_j}，U_j申请获得U_i的公开密钥 P_{k_i}。在相互认证中生成的临时值N_i和N_j用来保证本次通信不是一个重放攻击。此后U_i和U_j进行基于椭圆曲线的密钥交换获取会话密钥K_s用于报文M的加密传输，加密的算法选用IDEA。U_j收到报文后进行解密，然后进行检查和签名验证，检查电子公文的合法性和完整性。

3 公文流转方案的安全性分析

在上述公文流转方案中采用椭圆曲线密码体制来完成数字签名和密钥交换，其安全性依赖于椭圆曲线 E 上的离散对数的难解性。 (下转第 140 页)