软件学报 ISSN 1000-9825, CODEN RUXUEW
Journal of Software,2012,23(8):2163−2172 [doi: 10.3724/SP.J.1001.2012.04127]

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel/Fax: +86-10-62562563

带标签的基于证书密钥封装机制*

李继国+, 杨海珊, 张亦辰

(河海大学 计算机与信息学院,江苏 南京 210098)

Certificate-Based Key Encapsulation Mechanism with Tags

LI Ji-Guo+, YANG Hai-Shan, ZHANG Yi-Chen

(College of Computer and Information Engineering, Hohai University, Nanjing 210098, China)

+ Corresponding author: E-mail: ljg1688@163.com

Li JG, Yang HS, Zhang YC. Certificate-Based key encapsulation mechanism with tags. *Journal of Software*, 2012,23(8):2163−2172 (in Chinese). http://www.jos.org.cn/1000-9825/4127.htm

Abstract: The certificate-based encryption schemes often limit the message space to a particular group and are not adaptive to encrypt large messages. In order to solve this problem, the study extends the concept of key encapsulation mechanism with tags to the certificate-based encryption, and then proposes the notion and security model of the certificate-based key encapsulation mechanism with tags (CB-TKEM). In addition, the paper presents a construction of CB-TKEM that is provably secure against IND-CCA2 in the random oracle model.

Key words: key encapsulation mechanism; certificate-based encryption; random oracle model

摘 要: 基于证书加密方案通常将消息空间限制于某个特殊的群并且不适合大块消息加密.为了解决这一问题,将带标签的密钥封装机制引入到基于证书系统中,提出了带标签的基于证书密钥封装机制的形式化定义及安全模型.在此基础上构造了一个带标签的基于证书密钥封装方案,并证明了该方案在随机预言模型下是自适应选择密文不可区分的.

关键词: 密钥封装机制;基于证书加密;随机预言模型

中图法分类号: TP309 **文献标识码**: A

密码学的主要任务之一是保证在公开信道上发送消息的安全.目前主要有两种方法可以达到这个目的,分别是使用公钥密码系统对消息加密或者是使用对称密码系统对消息加密.使用公钥密码系统加密,不仅加、解密速度比较慢(与对称加、解密速度相比),而且对明文空间有限制或要求明文属于某个群,这在实际应用中是不实用的;而使用对称密码体制加、解密速度快,对明文长度没有限制,但存在密钥管理困难的问题.基于速度和安全性的考虑,有些学者结合两种密码体制的优点提出了混合加密的思想.即,用对称加密算法对需要通信的数据进行加、解密,用公钥加密算法对对称加密的密钥进行加、解密.直到 2003 年,Cramer 和 Shoup[1]才第一次形式化定义了混合加密的安全模型,即采用密钥封装机制(key encapsulation mechanism,简称 KEM)与数据封装机制

* 基金项目: 国家自然科学基金(60842002, 60673070, 61103183, 61103184); 国家高技术研究发展计划(863)(2007AA01Z409); 中国博士后基金(20100471373); 中央高校基本科研业务费专项资金(2009B21114, 2010B07114); 江苏省"六大人才高峰"项目(2009182); 河海大学新世纪优秀人才计划

收稿时间: 2011-05-15; 定稿时间: 2011-10-08

(data encapsulation mechanism,简称 DEM)进行组合,其模型简记为 KEM/DEM.KEM 与公钥加密相似,只是加密的任务变为生成一个随机密钥及对该随机密钥的封装.也就是说,加密算法除了随机值和接收者的公钥以外没有其他输入,生成一个对称密钥以及对该对称密钥的加密.DEM 是一个一次密钥对称加密方案,即每个密钥只用于 1 条消息的加密.

在 Cramer 和 Shoup[1]定义的 KEM/DEM 结构中,如果 KEM 和 DEM 都是自适应选择密文不可区分的,则由该 KEM 和 DEM 构造的混合加密方案是自适应选择密文不可区分的.以这样的方式构造混合加密看起来是合理的,也是必要的.在 CRYPTO 2004 上,Kurosawa 和 Desmedt[2]提出一个混合加密方案,其中,KEM 没有达到自适应选择密文安全,但是混合加密方案却达到了自适应选择密文安全.该方案是在 Cramer 和 Shoup 方案的基础上进行一次变形,它不再进行密文有效性验证,这样节约了一个哈希函数的计算和一个指数运算.除了效率上的优势以外,该方案在理论上也非常有意义.它说明了 IND-CCA2 安全的 KEM 虽然是 KEM/DEM 混合加密 IND-CCA2 安全的充分条件,但不是必要条件.另外,还有许多在随机预言模型下有效的混合加密方案[3−6],也不能满足 Cramer 和 Shoup[1]定义的 KEM/DEM 结构.

为了设计一个更普遍、更有效的混合结构,2005 年,Abe 等人[7]提出了 Tag-KEM/DEM 混合范例,Kurosawa 和 Desmedt 的混合加密方案就可以用该结构来解释.在该混合范例中,使用 DEM 的输出作为 Tag-KEM 中的标签,如果 Tag-KEM 是 IND-CCA2 安全的并且 DEM 对被动攻击者是安全的,则混合加密可以达到 IND-CCA2 安全.Abe 等人指出,可以由比 CCA2 安全强度弱的 KEM 构造 CCA2 安全的 Tag-KEM,并给出 CCA2 安全的 Tag-KEM 的几种构造.2006 年,Bjørstad 和 Dent 等人[8]首次将 Tag-KEM/DEM 引入到混合签密方案中,并提出了几个安全的适用于签密的 Tag-KEM 的例子.Yoshida 和 Fujiwara[9]在 Bjørstad 和 Dent 等人[8]研究的基础上,重新对签密 Tag-KEM 的安全性进行定义,并对 Bjørstad 和 Dent 等人[8]提出的混合签密的一般构造在新的定义下给出了安全性证明.2007 年,Huang 和 Wong[10]首次将 Tag-KEM 引入到无证书加密方案中,提出了带标签的无证书密钥封装机制(CL-TKEM),并构造了一个有效的 CL-TKEM.2008 年,Tan[11]构造了在标准模型下内部安全的用于签密的 Tag-KEM 方案.2009 年,Matsuda 等人[12]分析了 Tag-KEM/DEM 混合加密方案几种安全概念之间的关系,以及混合加密方案的每种安全概念成立的充分条件和必要条件.2009 年,Li 等人[13]将 Tag-KEM 引入到无证书签密方案中,并给出一个无证书签密 Tag-KEM 方案,以及用该无证书签密 Tag-KEM 构造安全的无证书混合签密方案.2010 年,Selvi 等人[14]指出,Li 等人的无证书混合签密方案可以伪造出签密,是不安全的,并提出一个改进的带标签的无证书混合签密方案,给出了安全性证明.

基于证书密码系统(certificate-based cryptography,简称 CBC)是由 Gentry[15]在 2003 年欧密会上提出的一个新型公钥密码系统.该密码系统具有基于身份和传统公钥密码系统的优点,消除了传统公钥密码系统对证书的第三方询问问题,简化了传统 PKI 系统中的证书撤销问题,克服了基于身份密码体制的密钥托管和密钥分发问题.结合 TKEM 和基于证书加密方案的优点,本文将 TKEM 引入到基于证书加密方案中,提出带标签的基于证书密钥封装机制(certificate-based key encapsulation mechanism with tags,简称 CB-TKEM)的形式化定义和安全模型;在此基础上,构造出一个带标签的基于证书密钥封装方案,并证明了该方案在随机预言模型下是自适应选择密文不可区分的.

1 预备知识

在可证明安全理论中,对相关方案的安全性证明是通过把方案的安全性规约为某个公开的数学难题来实现的.本节简要介绍与本文相关的几个困难问题及困难假定,具体可参阅文献[16].

定义 1(双线性映射). G_1 是 q 阶加法循环群,G_2 为 q 阶乘法循环群,P 为群 G_1 的生成元.一个可计算的双线性映射(admissible bilinear map)$e:G_1\times G_1\to G_2$ 具有以下性质:

(1) 双线性(bilinearity):对任意的 $P,Q\in G_1$ 以及 $a,b\in Z_q^*$,有 $e(aP,bQ)=e(P,Q)^{ab}$.

(2) 非退化性(non-degeneracy):存在 $P,Q\in G_1$,使得 $e(P,Q)\neq 1$.

(3) 可计算性(computability):存在有效的算法来计算 $e(P,Q)\in G_2$.

定义 2(Diffie-Hellman 元组). 给定群元素(P,aP,bP,cP),其中$a,b,c\in Z_q^*$, 判断 $cP=abP$ 是否成立.如果成立,则称(P,aP,bP,cP)是一个有效的 Diffie-Hellman 元组.

定义 3(Bilinear Diffie-Hellman 问题). 给定群元素(P,aP,bP,cP),其中$a,b,c\in Z_q^*$,计算 $e(P,P)^{abc}$.

概率多项式时间算法 A 解决$\langle G_1,G_2\rangle$上的 BDH 问题的优势定义为

$$Succ_{A,G_1,G_2}^{BDH}=\Pr[A(P,aP,bP,cP)=e(P,P)^{abc}:a,b,c\in Z_q^*].$$

如果任意的概率多项式时间算法 A 解决$\langle G_1,G_2\rangle$上的 BDH 问题的优势都是可忽略的,则称$\langle G_1,G_2\rangle$的 BDH 问题是困难的.

定义 4(Decision Bilinear Diffie-Hellman 问题). 对于$a,b,c\in Z_q^*$,给定群元素(P,aP,bP,cP)以及 $T\in G_2$,判断 $T=e(P,P)^{abc}$ 是否成立.如果成立,则输出 1;否则,输出 0.

概率多项式时间算法 A 解决$\langle G_1,G_2\rangle$上的 DBDH 问题的优势定义为

$$Succ_{A,G_1,G_2}^{DBDH}=|\Pr[A(P,aP,bP,cP,e(P,P)^{abc})=1]-\Pr[A(P,aP,bP,cP,T)=1]|,$$

其中,$a,b,c\in Z_q^*$.

如果任意的概率多项式时间算法 A 解决$\langle G_1,G_2\rangle$上的 DBDH 问题的优势都是可忽略的,则称$\langle G_1,G_2\rangle$的 DBDH 问题是困难的.

定义 5(Decision Generalized Bilinear Diffie-Hellman 问题). 对于$a,b,c\in Z_q^*$,给定群元素(P,aP,bP,cP)以及 $T\in G_2$,算法 A 选择 $Y\in G_1^*$ 并判断 $T=e(P,Y)^{abc}$ 是否成立.如果成立,则输出 1;否则,输出 0.

概率多项式时间算法 A 解决$\langle G_1,G_2\rangle$上的 DGBDH 问题的优势定义为

$$Succ_{A,G_1,G_2}^{DGBDH}=|\Pr[A(P,aP,bP,cP,e(P,Y)^{abc})=1]-\Pr[A(P,aP,bP,cP,T)=1]|,$$

其中,$a,b,c\in Z_q^*$.

如果任意的概率多项式时间算法 A 解决$\langle G_1,G_2\rangle$上的 DGBDH 问题的优势都是可忽略的,则称$\langle G_1,G_2\rangle$的 DGBDH 问题是困难的.

2 带标签的基于证书密钥封装机制的形式化定义及安全模型

本节将带标签的密钥封装机制与基于证书密码体制结合,给出了带标签的基于证书密钥封装机制的形式化定义和安全模型.

2.1 带标签的基于证书密钥封装机制的形式化定义

参考 Abe 等人[7]提出的 Tag-KEM 的形式化定义以及 Gentry[15]提出的基于证书加密的形式化定义,我们给出带标签的基于证书密钥封装机制的形式化定义.CB-TKEM 由 6 个算法组成(Setup,UserKeyGen,Certify,Key,Encap,Decap),其中,系统参数设置和证书产生算法由证书认证中心(CA)运行,用户密钥生成算法、密钥产生算法、密钥封装算法和密钥解封装算法由用户执行.

- 系统参数设置算法(Setup):以安全参数 k 为输入的概率多项式时间算法,CA 运行此算法产生主密钥 MSK 和系统参数 $params$(包含对串空间的描述以及主公钥 P_{pub}).CA 将系统参数公开.
- 用户密钥生成算法(UserKeyGen):用户获得系统参数 $params$ 后,运行此算法,生成用户私钥 SK 和用户公钥 PK.
- 证书产生算法(Certify):给定用户的身份信息 ID 和用户公钥 PK,输入(MSK,$params$,ID,PK),输出用户证书 $Cert_{ID}$,并将 $Cert_{ID}$ 通过公开信道发送给用户.
- 密钥产生算法(KeyGen):该算法以($params$,ID,PK)输入,生成密钥 K 及内部状态信息ω,其中,$K\in\kappa_D$ 为数据封装 DEM 的加密密钥,κ_D 为数据封装的密钥空间.K 与ω均不公开.
- 密钥封装算法(Encap):这是一种概率多项式时间的封装算法.该算法以ω及标签$\tau\in\{0,1\}^*$为输入,输出密钥 K 对应的封装ψ.将封装ψ及标签τ发送给接收者.

- 密钥解封装算法(Decap):这是一种确定性算法.输入$(ID,Cert_{ID},SK,\psi,\tau)$,进行解封装并输出数据封装 DEM 的密钥 K,或者无效标志$\perp$.

2.2 带标签的基于证书密钥封装机制的安全模型

参考 Abe 等人[7]提出的 Tag-KEM 的安全模型以及 Li 等人[17−19]提出的基于证书的安全模型,我们给出带标签的基于证书密钥封装机制的安全模型.

带标签的基于证书密钥封装机制中包含两类敌手 A_{I} 和 A_{II}.A_{I} 模拟了一个不诚实的用户,它不知道系统主密钥,但可以替换公钥以及询问任何用户的私钥,可以获得除目标用户外任意用户的证书,可以进行除了目标封装以外的其他解封装询问.A_{II} 模拟了一个恶意的 CA,它拥有主密钥,可以生成任何用户的证书,但不可以进行替换公钥. A_{II} 还可以询问除目标用户外任何用户的私钥,可以对除目标封装外的其他密钥封装进行解封装询问.

带标签的基于证书密钥封装机制的安全模型是通过挑战者与敌手之间的游戏来定义的.根据敌手的分类,带标签的基于证书密钥封装机制的安全模型定义如下:

游戏 1:

系统参数设置:挑战者运行算法 Setup,返回主密钥 MSK 和系统参数 $params$,保密主密钥 MSK,把系统参数 $params$ 发送给 A_{I}.

第 1 阶段询问:在这一阶段中,挑战者 C 维护一个记录用户 ID_i 私钥和公钥的表 $L_0=\{ID_i,PK_i,SK_i,f_i\}$,该表初始为空,$f_i$=0 表示用户公钥没有被替换,$f_i$=1 表示公钥被替换.$A_{\mathrm{I}}$ 向挑战者 C 进行如下询问:

- 公钥询问:敌手 A_{I} 对 ID_i 进行公钥询问.如果表 L_0 中不存在 ID_i 项,则 C 将运行 UserKeyGen 算法生成 ID_i 的公私钥对(PK_i,SK_i),并将$\{ID_i,PK_i,SK_i,0\}$加入表 L_0,然后返回 PK_i;如果表 L_0 中已存在$\{ID_i,PK_i,SK_i,f_i\}$项,则直接返回 PK_i.
- 私钥询问:敌手 A_{I} 询问身份 ID_i 的私钥.如果表 L_0 中不存在 ID_i 项,则 C 运行 UserKeyGen 算法生成用户 ID_i 的公私钥对(PK_i,SK_i),并将$\{ID_i,PK_i,SK_i,0\}$加入表 L_0,然后返回 SK_i;若表 L_0 中已存在$\{ID_i,PK_i,SK_i,f_i\}$项,则直接返回 SK_i.
- 公钥替换询问:敌手 A_{I} 随机选择公钥 PK_i' 对 ID_i 进行公钥替换询问.如果表 L_0 中不存在 ID_i 项,则 C 将 $(ID_i,PK_i',\perp,1)$ 加入;否则,C 将表 L_0 中 ID_i 项更新为 $(ID_i,PK_i',\perp,1)$.
- 证书询问:敌手 A_{I} 询问 ID_i 的证书.如果表 L_0 中不存在 ID_i 项,则 C 运行算法 UserKeyGen 生成 ID_i 的公私钥对(PK_i,SK_i),并将$\{ID_i,PK_i,SK_i,0\}$加入表 L_0;否则,C 获取 PK_i.两种情况下,C 都以$(params,MSK,ID_i,PK_i)$为输入运行 Certify 算法,返回 $Cert_{ID_i}$.
- 解封装询问:敌手 A_{I} 对(ID_i,ψ_i,τ_i)进行解封装询问.C 查找表 $L_0=\{ID_i,PK_i,SK_i,f_i\}$,如果 f_i=1,则要求敌手提供 PK_i 对应的私钥;否则,直接读取 SK_i.利用解封装算法解封装 ψ_i,返回密钥 K_i.

挑战阶段:敌手 A_{I} 选择目标身份 ID^*并将目标身份发送给挑战者.挑战者对目标身份运行算法 $KeyGen(params,ID^*,PK^*)\rightarrow(K_1,\omega^*)$,同时选取 $K_0\in\kappa_D$.然后随机选取 $b\in\{0,1\}$,并将 K_b 返回给敌手.敌手 A_{I} 继续像第 1 阶段那样进行询问,但是不可以对目标身份 ID^*进行证书询问.A_{I} 决定询问结束时,提交一个目标标签 τ^*,挑战者计算封装 $Encap(\omega^*,\tau^*)\rightarrow\psi^*$,并将目标封装 ψ^*返回给敌手 A_{I}.

第 2 阶段询问:A_{I} 继续进行询问,但不能询问目标用户 ID^*的证书以及对(ID^*,ψ^*,τ^*)的解封装询问.

猜测:A_{I} 输出 b'.如果 $b'=b$,则称 A_{I} 赢得游戏.

A_{I} 赢得游戏的优势定义为

$$Adv_{CB\text{-}TKEM}^{A_{\mathrm{I}}}=|2\Pr[b'=b]-1|.$$

定义 6(对第 1 类攻击者在自适应选择密文攻击下的不可区分性). 如果不存在概率多项式时间的敌手 A_{I} 能够以不可忽略的优势赢得游戏 1,则称带标签的基于证书密钥封装机制对第 1 类敌手在自适应选择密文攻击下是不可区分的.

游戏 2:

系统参数设置:挑战者运行算法 Setup,返回主密钥 MSK 和系统参数 $params$,并把主密钥 MSK 和系统参数 $params$ 都发送给 A_{II}.

第 1 阶段询问:在这一阶段中,挑战者 C 维护一个记录用户 ID_i 私钥和公钥的表 L_0={ID_i,PK_i,SK_i},该表初始为空.A_{II} 向挑战者 C 进行如下询问:

- 公钥询问:敌手 A_{II} 对 ID_i 进行公钥询问.如果表 L_0 中不存在 ID_i 项,则 C 将运行 UserKeyGen 算法生成 ID_i 的公私钥对(PK_i,SK_i),并将{ID_i,PK_i,SK_i}加入表 L_0,然后返回 PK_i;如果表 L_0 中已存在{ID_i,PK_i,SK_i}项,则直接返回 PK_i.
- 私钥询问:敌手 A_{II} 询问身份 ID_i 的私钥.如果表 L_0 中不存在 ID_i 项,则 C 运行 UserKeyGen 算法生成用户 ID_i 的公私钥对(PK_i,SK_i),并将{ID_i,PK_i,SK_i}加入表 L_0,然后返回 SK_i;若表 L_0 中已存在{ID_i,PK_i,SK_i}项,则直接返回 SK_i.
- 解封装询问:敌手 A_{II} 对(ID_i,ψ_i,τ_i)进行解封装询问.C 利用解封装算法解封装 ψ_i,返回密钥 K_i 或者⊥.

挑战阶段:敌手 A_{II} 选择目标身份 ID^*,并将目标身份发送给挑战者.挑战者对目标身份运行算法 $KeyGen(params,ID^*,PK^*)\rightarrow(K_1,\omega^*)$,同时选取 $K_0\in\kappa_D$.然后随机选取 $b\in\{0,1\}$,并将 K_b 返回给敌手.敌手 A_{II} 继续像第 1 阶段那样进行询问,但是不可以对目标身份 ID^* 进行私钥询问.A_{II} 决定询问结束时,提交一个目标标签 τ^*,挑战者计算封装 $Encap(\omega^*,\tau^*)\rightarrow\psi^*$,并将目标封装 ψ^* 返回给敌手 A_{II}.

第 2 阶段询问:A_{II} 继续像第 1 阶段那样询问,但不能对目标身份 ID^* 进行私钥询问,也不能对(ID^*, ψ^*,τ^*)进行解封装询问.

猜测:A_{II} 输出 b'.如果 $b'=b$,则称 A_{II} 赢得游戏.

A_{II} 赢得游戏的优势定义为

$$Adv_{CB\text{-}TKEM}^{A_{II}}=|2\Pr[b'=b]-1|.$$

定义 7(对第 2 类攻击者在自适应选择密文攻击下的不可区分性). 如果不存在概率多项式时间攻击者 A_{II} 能够以不可忽略的优势赢得游戏 2,则称带标签的基于证书密钥封装机制对第 2 类攻击者在自适应选择密文攻击下是不可区分的.

3 带标签的基于证书密钥封装机制

在这部分中,我们首次提出了带标签的基于证书密钥封装方案,并对它进行正确性分析.该方案由以下 6 个算法组成:

- 系统参数设置算法(Setup):该算法由 CA 执行如下:
  (1) G_1 为素数 q 阶加法循环群,P 是群 G_1 的生成元;G_2 为素数 q 阶乘法循环群,存在可计算的双线性映射 e:$G_1\times G_1\rightarrow G_2$;
  (2) 选择两个 Hash 函数 $H_1:\{0,1\}^*\times G_1\times G_1\rightarrow G_1^*$ 和 $H_2:G_1\rightarrow G_1^*$,G_1^* 为群 G_1 中的非零元素;
  (3) 随机选取 $s\in Z_q^*$,计算系统主公钥 $P_{pub}=sP$,则系统主私钥 $MSK=s$ 且由 CA 保存,并将系统参数 $params$={G_1,G_2,q,e,P,P_{pub},H_1,H_2}公开.
- 用户密钥生成算法(UserKeyGen):用户在 Z_q^* 中随机选取 x,利用系统参数生成自己的公钥 $PK=(PK_1,PK_2)=(xP,xP_{pub})$,用户私钥 $SK=x$.
- 证书产生算法(Certify):用户给出(ID,PK),认证中心 CA 为当前用户计算证书.输入($params$,MSK,ID,PK),计算 $Q_{ID}=H_1(ID,PK)$,则用户(ID,PK)的证书 $Cert_{ID}=sH_1(ID,PK)=sQ_{ID}$.
- 密钥产生算法(KeyGen):发送者使用系统参数 $params$、接收者的公钥 PK 和身份 ID,生成密钥 K 及内部状态信息 ω.首先,发送者验证 $e(PK_1,P_{pub})=e(PK_2,P)$ 是否成立:若不成立,则输出⊥并终止;否则,发送者计算 $Q_{ID}=H_1(ID,PK)$,随机选取 $r\in Z_q^*$, 计算:

(1) $K=e(Q_{ID},PK_2)^r$;

(2) $C_1=rP$;

(3) $\omega=(r,C_1)$.

- 密钥封装算法(Encap):以$\omega=(r,C_1)$和随机标签τ为输入,计算:

  (1) $W=H_2(C_1,\tau)$;

  (2) $C_2=rW$.

  返回对密钥K的封装$\psi=(C_1,C_2)$及标签τ.
- 密钥解封装算法(Decap):接收者收到封装$\psi=(C_1,C_2)$及标签τ后,用证书及私钥进行解封装.解封装过程如下:

  (1) 计算 $W=H_2(C_1,\tau)$.当且仅当(P,C_1,W,C_2)是 Diffie-Hellman 元组时,ψ是正确的封装,并计算$K=e(C_1,SK\times Cert_{ID})$.

  (2) 否则,ψ是无效的.

正确性分析:

$$K=e(C_1,SK\times Cert_{ID})=e(rP,xsQ_{ID})=e(xsP,Q_{ID})^r=e(PK_2,Q_{ID})^r.$$

4 带标签的基于证书密钥封装方案的安全性证明

在这一部分中,证明本文提出的带标签的基于证书密钥封装方案在随机预言模型下是安全的.根据第 2 节提出的带标签的基于证书密钥封装机制的安全模型和定义,给出以下证明.

定理 1. 在随机预言模型下,如果存在概率多项式时间敌手 A_I,能够在多项式时间内,经过最多 q_C 次证书询问、q_D 次解封装询问后,以概率ε在游戏 1 中区分出 K_b:

- 若目标身份 ID^*的公钥没有被替换,那么存在一种算法 B,在多项式时间内以 $\varepsilon'\geqslant\frac{\varepsilon}{e(1+q_C)}-\frac{q_D}{q}$ 的概率解决 DBDH 困难问题;
- 若目标身份 ID^*的公钥被替换,那么存在一种算法 B,在多项式时间内以概率 $\varepsilon'\geqslant\frac{\varepsilon}{e(1+q_C)}-\frac{q_D}{q}$ 解决判定 DGBDH 困难问题.

证明:算法 B 以$(G_1,G_2,e,q,P,aP,bP,cP,T)$为输入,其目的是在目标身份公钥未被替换的情况下判定 $T=e(P,P)^{abc}$ 是否成立;在目标身份公钥被替换的情况下判定 $T=e(P,Y)^{abc}$ 是否成立.B 将扮演挑战者与 A_I 进行交互,并利用 A_I 实现目标.为了对 A_I 询问回答前后一致,B 需维持 3 个初始为空的表 L_0,L_1,L_2,记录 B 对敌手 A_I 询问的回答.

系统参数设置:B 令 $P_{pub}=aP$,并将 $params=\{G_1,G_2,q,e,P,P_{pub},H_1,H_2\}$发送给 A_I.其中,H_1,H_2 视为随机预言器.

第 1 阶段询问:在该阶段中,B 与敌手 A_I 进行交互,过程如下:

- 公钥询问:B 维持一个记录用户公钥和私钥的表 $L_0=\{ID_i,PK_i=(PK_{i1},PK_{i2}),SK_i,f_i\}$,$f_i=0$ 表示用户公钥没有被替换,$f_i=1$ 表示公钥被替换,该表初始为空.A_I 询问用户 ID_i 的公钥,若表 L_0 中有 ID_i 项,则 B 直接返回 ID_i 的公钥 PK_i 给 A_I;若表 L_0 中没有 ID_i 项,则 B 随机选取 $x_i\in Z_q^*$,计算 $PK_i=(PK_{i1},PK_{i2})=(x_iP,x_i(aP))$,返回 PK_i 给 A_I,并将元组$(ID_i,PK_i,x_i,0)$添加到表 L_0.
- 私钥询问:A_I 询问用户 ID_i 的私钥,B 检查$\{ID_i,PK_i,SK_i,f_i\}$是否在表 L_0 中.若 ID_i 对应的元组在表 L_0 中且 $f_i=1$,则 B 拒绝回答 A_I 的询问;若 ID_i 对应的元组在表 L_0 中且 $f_i=0$,则 B 返回私钥 SK_i;若 ID_i 对应的元组不在表 L_0 中,则 B 随机选取 $x_i\in Z_q^*$,计算 $PK_i=(PK_{i1},PK_{i2})=(x_iP,x_i(aP))$,返回 $SK_i=x_i$ 给 A_I,并将元组$(ID_i,PK_i,x_i,0)$添加到表 L_0 中.
- 公钥替换询问:敌手 A_I 提交公钥替换询问 (ID_i,PK_i').B 检查 $e(P_{pub},PK_{i1}')=e(P,PK_{i2}')$ 是否成立.若不成立,则返回无效的公钥;否则,B 检查表 L_0 中是否存在 ID_i 项.若不存在,则 B 将 $(ID_i,PK_i',\perp,1)$ 加入;否则,B 将表 L_0 中的 ID_i 项更新为 $(ID_i,PK_i',\perp,1)$.

- H_1 询问:敌手 A_1 对(ID_i,PK_i)进行 H_1 询问,其中,(ID_i,PK_i)是第 i 个身份与其对应的公钥.B 投掷硬币 $coin_{ID_i}(coin_{ID_i}\in\{0,1\},\Pr[coin_{ID_i}=0]=\delta=1-1/(q_C+1))$, 并维护列表 $L_1=\{coin_{ID_i},ID_i,PK_i,b_i,Q_{ID_i},Cert_{ID_i},f_i\}$:若 $coin_{ID_i}=0$, 则 B 随机选取 $b_i\in Z_q^*$, 计算 $Q_{ID_i}=b_iP=H_1(ID_i,PK_i)$ 并返回,将 $\{0,ID_i,PK_i,b_i,Q_{ID_i},Cert_{ID_i},f_i\}$ 填加到表 L_1 中;若 $coin_{ID_i}=1$, 则 B 随机选取 $b_i\in Z_q^*$, 计算 $Q_{ID_i}=b_i(bP)=H_1(ID_i,PK_i)$ 并返回,将 $\{1,ID_i,PK_i,b_i,Q_{ID_i},Cert_{ID_i},f_i\}$ 填加到表 L_1 中.
- H_2 询问:B 维护列表 L_2={T_j,C_{j1},τ_j,W_j},初始为空.敌手 A_1 对 C_{j1},τ_j 进行 H_2 询问,B 随机选 $T_j\in Z_q^*$, 计算 W_j=$H_2(C_{j1},\tau_j)$=$T_j(aP)$,并将{T_j,C_{j1},τ_j,W_j}填加到表 L_2 中.
- 证书询问:敌手 A_1对(ID_i,PK_i)询问证书(假设证书询问前对 ID_i 进行过公钥询问及 H_1 询问).若表 L_1 中身份 ID_i 项对应的证书 $Cert_{ID_i}$ 存在,则 B 将该证书返回给敌手 A_1.否则,当 $coin_{ID_i}=0$ 时,B 返回证书 $Cert_{ID_i}=sQ_{ID_i}=b_i(aP)=a(b_iP)$. 当 $coin_{ID_i}=1$ 时,B 终止游戏并输出失败.
- 解封装询问:敌手 A_1对(ID_i,PK_i)的封装 ψ_j=(C_{j1},C_{j2})进行解封装询问,标签为 τ_j.B 进行解封装如下:

  (1) 如果 f_i=0,则假设 C_{j1},τ_j 在表 L_2 中.B 通过 C_{j1},τ_j 查询表 L_2{T_j,C_{j1},τ_j,W_j},检查$(P,C_{j1},T_j(aP),C_{j2})$是否为 Diffie-Hellman 元组,若不是,则 B 返回无效;若是,则 B 按如下方式计算 K_j:

  - 若 $coin_{ID_i}=0$,则 B 计算如下:K_j=$e(C_{j1},x_ib_i(aP))$,并返回 K_j 给 A_1;
  - 若 $coin_{ID_i}=1$,则虽然 B 无法从 C_{j1}=r_jP 中得知 r_j,但是 B 知道 C_{j2}=$r_jH_2(C_{j1})$,可知

  $$e(C_{j1},SK_iCert_{ID_i})=e(r_jP,x_ib_i(abP))=e(bP,x_ib_i(r_jT_jaP))^{1/T_j}.$$

  所以可知,$K_j=e\left(bP,\frac{x_ib_i}{T_j}C_{j2}\right)$.

  (2) 如果 f_i=1,若敌手 A_1没有提供替换后公钥对应的私钥,则 B 拒绝返回;若 A_1提供替换后公钥对应的私钥,则 B 按照方式(1)进行解封装.

挑战阶段:当敌手 A_1 决定第 1 阶段询问结束时,选取目标用户 ID^*发送给 B,B 运行如下:如果 $coin_{ID^*}=0$, 则 B 终止游戏并输出失败.否则,若 f^*=0,则 $C_1^*=(x^*)^{-1}(b^*)^{-1}(cP)$; 若 f^*=1,则 $C_1^*=(b^*)^{-1}(cP)$. 挑战者将 $K_b^*=T\in G_2$ 返回给敌手.A_1收到 K_b^* 后,继续像第 1 阶段那样进行询问,但是不可以对目标身份 ID^*进行证书询问.A_1 决定询问结束时,提交一个目标标签 τ^*.B 随机选取 $t^*\in Z_q^*$, 计算 $H_2(C_1^*,\tau^*)=t^*P$, 则有 $C_2^*=t^*C_1^*$. B 将封装 $\psi^*=(C_1^*,C_2^*)$ 返回给 A_1.

第 2 阶段询问:敌手 A_1继续进行一系列询问,但不能询问目标用户 ID^*的证书以及对(ψ^*,ID^*,PK^*)的解封装.B 的回答方式与第 1 阶段询问相同.

猜测:敌手 A_1 输出对 b 的猜测 $b'\in\{0,1\}$,猜测 b'=1 意味着 K_b^* 是正确密钥.算法 B 推断如下:

(1) 敌手 A_1 没有对目标用户进行公钥替换,B 将 b'作为它的输出 b'',b''=1 意味着 B 猜测出 T=$e(P,P)^{abc}$.

(2) 敌手 A_1用 (PK_1',PK_2') 对目标用户进行公钥替换.若 b'=1,则 B 将 $(b''=1,PK_1')$ 作为它的输出,也即 B 能找到 PK_1' 并能猜测出 $T=e(P,PK_1')^{abc}$; 若 b'=0,B 输出 b''=0,也即 B 找不到能够解决 DGBDH 困难问题的值.

分析:如果敌手 A_1 没有对目标用户进行公钥替换,则对 $\psi^*=(C_1^*,C_2^*)$ 及目标标签 τ^*解封装得

$$K_1^*=e(C_1^*,SK^*Cert_{ID^*}^*)=e((x^*)^{-1}(b^*)^{-1}(cP),x^*b^*abP)=e(P,P)^{abc}.$$

而 $K_0^*\in G_2$ 是挑战者随机选取的.B 将 $K_b^*=T\in G_2(b\in\{0,1\})$ 返回给敌手.敌手 A_1 输出对 b 的猜测 $b'\in\{0,1\}$:b'=1 意味着 T=$e(P,P)^{abc}$,b'=0 意味着 T 为随机值.此时,B 将 b'作为它的输出 b'':b''=1,也即 B 猜测出 T=$e(P,P)^{abc}$ 成立,否则不成立.如果 A_1在游戏中能区分出 K_b^*,则 B 可以根据敌手解决 DBDH 困难问题.

若敌手 A_1 用 $PK'=(PK_1',PK_2')=(x'P,x'P_{pub})$ 对目标用户进行公钥替换,则由解封装 $\psi^*=(C_1^*,C_2^*)$ 及目标标签 τ^*可得密钥 $K_1^*=e((b^*)^{-1}(cP),x'b^*abP)=e(x'P,P)^{abc}=e(PK_1',P)^{abc}$, 而 $K_0^*\in G_2$ 是挑战者随机选取的.

B 将 $K_b^*=T\in G_2(b\in\{0,1\})$ 返回给敌手.敌手 A_1 输出对 b 的猜测 $b'\in\{0,1\}$:b'=1 意味着 $T=e(PK_1',P)^{abc}$,b'=0 意味着 T 为随机值.此时,B 将 b'作为它的输出 b'':b''=1 意味着 B 猜测出 $T=e(PK_1',P)^{abc}$ 成立,否则不成立.如果 A_1 在游戏中能区分出 K_b^*,则 B 可以根据敌手解决 DGBDH 困难问题.

概率计算:在解封装询问中,若$(P,C_{j1},T_j(aP),C_{j2})$不是 Diffie-Hellman 元组,则封装无效.因此在解封装询问中接受一个无效封装的概率为 $1/q$,在 q_D 次解封装询问中无效封装的概率为 q_D/q.记 E 为事件 B 没有终止游戏,E_1 为事件 $coin_{ID_i}=1$ 时敌手 A_1 对(ID_i,PK_i)询问证书,E_2 为事件 $coin_{ID^*}=0$ 时敌手 A_1 选取的目标身份,则

$$\Pr[E]=\Pr[(\neg E_1)\wedge(\neg E_2)]=\delta^{q_C}(1-\delta)\geqslant 1/e(1+q_C),$$

于是,B 解决困难问题的概率 $\varepsilon'\geqslant\dfrac{\varepsilon}{e(1+q_C)}-\dfrac{q_D}{q}$. □

定理 2. 在随机预言模型下,如果存在概率多项式的时间敌手 A_{II},能够在多项式时间内,经过最多 q_{SK} 次 H_2 询问、q_{SK} 次私钥询问、q_D 次解封装询问后,以概率ε在游戏 2 中区分出 K_b,那么存在算法 B,在多项式时间内以概率 $\varepsilon'\geqslant\dfrac{\varepsilon}{e(1+q_{SK})}-\dfrac{q_D}{q}$ 解决 DBDH 困难问题.

证明:算法 B 以$(G_1,G_2,e,q,P,aP,bP,cP,T)$为输入,其目标是输出 T 与 $e(P,P)^{abc}$ 是否相等.B 将扮演挑战者与 A_{II} 进行交互,并利用 A_{II} 实现目标.为了使对 A_{II} 询问回答前后一致,B 需要维持两个初始为空的表:表 L_0,L_1.表 L_0,L_1 用于记录 B 对敌手 A_{II} 询问的回答.

系统参数设置:B 选取 $s\in Z_q^*$,计算 $P_{pub}=sP$,并将系统主私钥 $MSK=s$ 和系统参数 $params=\{G_1,G_2,e,q,P,P_{pub},H_2\}$发送给 A_{II},其中,H_1,H_2 是随机预言器.

第 1 阶段询问:在该阶段中,B 与敌手 A_{II} 进行交互,过程如下:

- 公钥询问:B 维持一个初始为空的记录用户私钥和公钥的表 $L_0=\{coin_{ID_i},ID_i,PK_i=(PK_{i1},PK_{i2}),SK_i,x_i,b_i,Q_{ID_i}\}$.$A_{\mathrm{II}}$ 询问用户 ID_i 的公钥,B 进行如下计算:
  (1) 若表 L_0 中有 ID_i 项,则 B 直接返回 ID_i 的公钥 PK_i 给 A_{II}.
  (2) 否则,B 投掷硬币 $coin_{ID_i}$ ($coin_{ID_i}\in\{0,1\}$, $\Pr[coin_{ID_i}=0]=\delta=1-1/(q_{SK}+1)$). 若 $coin_{ID_i}=0$,则 B 随机选取 $x_i\in Z_q^*$,计算 PK_i=(PK_{i1},PK_{i2})=$(x_iP,x_i(sP))$,返回 PK_i 给 A_{II},并将元组$(0,ID_i,PK_i,x_i,x_i,\perp,\perp)$添加到表 L_0;若 $coin_{ID_i}=1$,则 B 随机选取 $x_i\in Z_q^*$,计算 PK_i=(PK_{i1},PK_{i2})=$(x_i(bP),x_i(sbP))$,返回 PK_i 给 A_{II},并将元组$(1,ID_i,PK_i,\perp,x_i,\perp,\perp)$添加到表 L_0.
- 私钥询问:A_{II} 询问用户 ID_i 的私钥,B 检查$\{coin_{ID_i},ID_i,PK_i,SK_i,x_i,b_i,Q_{ID_i}\}$是否在表 L_0 中.若 ID_i 的私钥存在表 L_0 中且 $coin_{ID_i}=1$,则 B 结束游戏并失败;若 ID_i 的私钥存在表 L_0 中但 $coin_{ID_i}=0$,则 B 返回私钥 SK_i.若 ID_i 的私钥不在表 L_0 中,则 B 投掷硬币 $coin_{ID_i}$,当 $coin_{ID_i}=0$ 时,B 随机选取 $x_i\in Z_q^*$,计算 PK_i=(PK_{i1},PK_{i2})=$(x_iP,x_i(sP))$,将元组$(0,ID_i,PK_i,x_i,x_i,\perp,\perp)$添加到表 L_0,并将 SK_i=x_i 返回给 A_{II};若 $coin_{ID_i}=1$,则 B 随机选取 $x_i\in Z_q^*$,计算 PK_i=(PK_{i1},PK_{i2})=$(x_i(bP),x_i(sbP))$,并将元组$(1,ID_i,PK_i,\perp,x_i,\perp,\perp)$添加到表 L_0,B 结束游戏并失败.
- H_1 询问:敌手 A_{II} 对(ID_i,PK_i)进行 H_1 询问,其中,(ID_i,PK_i)是第 i 个身份和其对应的公钥.B 回答如下:若 $coin_{ID_i}=0$,则 B 随机选取 $b_i\in Z_q^*$,计算 $Q_{ID_i}=b_iP$,将元组 $(0,ID_i,PK_i,SK_i,x_i,b_i,Q_{ID_i})$ 添加到表 L_0 中,并将 $H_1(ID_i,PK_i)=Q_{ID_i}$ 返回给 A_{II};若 $coin_{ID_i}=1$,则 B 随机选取 $b_i\in Z_q^*$,计算 $Q_{ID_i}=b_i(aP)$,将元组$(1,ID_i,PK_i,SK_i,x_i,b_i,Q_{ID_i})$ 添加到表 L_0 中并将 $H_1(ID_i,PK_i)=Q_{ID_i}$ 返回给 A_{II}.
- H_2 询问:B 维护列表 L_1=$\{T_j,C_{j1},\tau_j,W_j\}$,初始为空.敌手 A_{II} 对 C_{j1},τ_j 进行 H_2 询问,B 随机选取 $T_j\in Z_q^*$,计算 W_j=$H_2(C_{j1},\tau_j)$=$T_j(aP)$,并将$\{T_j,C_{j1},\tau_j,W_j\}$填加到表 L_1 中.
- 解封装询问:敌手 A_{II} 对(ID_i,PK_i)的封装 ψ_j=(C_{j1},C_{j2})进行解封装询问,标签为 τ_j.假设 C_{j1},τ_j 在 L_2 表中.B 通过 C_{j1},τ_j 查询 L_1 表$\{T_j,C_{j1},\tau_j,W_j\}$,检查$(P,C_{j1},T_j(aP),C_{j2})$是否为 Diffie-Hellman 元组:若不是,则 B 返回

无效;若是,则 B 按如下方式计算 K_j:

(1) 若 $coin_{ID_i}=0$,则 B 如下计算:$K_j=(C_{j1},x_ib_isP)$,并返回 K_j 给 A_{II};

(2) 若 $coin_{ID_i}=1$,则虽然 B 无法从 $C_{j1}=r_jP$ 中得知 r_j,但是 B 知道 $C_{j2}=r_jH_2(C_{j1})$,可获得解封装 $e(C_{j1},SK_iCert_{ID_i})=e(r_jP,x_ib_i(sabP))=e(bP,x_ib_is(r_jT_jaP))^{1/T_j}$,输出 $K_j=e\left(bP,\frac{x_ib_is}{T_j}C_{j2}\right)$.

挑战阶段:当敌手 A_{II} 决定第 1 阶段询问结束时,选取目标用户 ID^*发送给 B.B 运行如下:如果 $coin_{ID^*}=0$,则 B 终止游戏并输出失败;否则,计算 $C_1^*=(x^*)^{-1}(b^*)^{-1}(s)^{-1}(cP)$,挑战者将 $K_b^*=T\in G_2$ 返回给敌手.A_1 收到 K_b^* 后,继续像第1阶段那样进行询问,但是不可以对目标身份 ID^*进行证书询问.A_1决定询问结束时,提交一个目标标签 τ^*.B 随机选取 $t^*\in Z_q^*$,$C_2^*=t^*C_1^*$,并定义 $H_2(C_1^*,\tau^*)=t^*P$ .B 返回封装 $\psi^*=(C_1^*,C_2^*)$.

第2阶段询问:敌手 A_{II} 继续进行一系列询问,但不能对(ψ^*,ID^*,PK^*)进行解封装询问.B 的回答方式与第1阶段询问相同.

猜测:敌手 A_{II} 输出对 b 的猜测 $b'\in\{0,1\}$.B 将 b'作为它的输出 b'':$b''=1$ 意味着 B 猜测出 $T=e(P,P)^{abc}$.

分析:$\psi^*=(C_1^*,C_2^*)$ 为在目标标签τ^*下挑战阶段形成的封装,解封装可得:

$$K_1^*=e(C_1^*,SK^*Cert_{ID^*}^*)=e((x^*)^{-1}(b^*)^{-1}(s)^{-1}(cP),x^*b^*sabP)=e(P,P)^{abc}.$$

而 $K_0^*\in G_2$ 是挑战者随机选取的.B 将 $K_b^*=T\in G_2(b\in\{0,1\})$ 返回给敌手.敌手 A_{II} 输出对 b 的猜测 $b'\in\{0,1\}$:$b'=1$ 意味着 $T=e(P,P)^{abc}$,$b'=0$ 意味着 T 为随机值.此时,B 将 b'作为它的输出 b'':$b''=1$,也即 B 猜测出 $T=e(P,P)^{abc}$ 成立,否则不成立.如果 A_{II} 在游戏中能够区分出 K_b^*,则 B 可以根据敌手解决 DBDH 困难问题.

概率计算:在解封装询问中,若$(P,C_{j1},T_j(aP),C_{j2})$不是 Diffie-Hellman 元组,则封装无效.因此在解封装询问中接受一个无效封装的概率为 $1/q$,在 q_D 次解封装询问中无效封装的概率为 q_D/q.记 E 为事件 B 没有终止游戏,E_1 为事件 $coin_{ID_i}=1$ 时敌手 A_{II} 对(ID_i,PK_i)询问证书,E_2 为事件 $coin_{ID^*}=0$ 时敌手 A_{II} 选取的目标身份,则

$$\Pr[E]=\Pr[(\neg E_1)\wedge(\neg E_2)]=\delta^{q_{SK}}(1-\delta)\geqslant 1/e(1+q_{SK}),$$

于是,B 解决困难问题的概率 $\varepsilon'\geqslant\frac{\varepsilon}{e(1+q_{SK})}-\frac{q_D}{q}$. □

5 结束语

本文将带标签的密钥封装机制引入到基于证书的密码系统中,提出了带标签的基于证书密钥封装机制的形式化定义及安全模型,并构造了一个带标签的基于证书密钥封装方案.该方案在随机预言模型下证明是自适应选择密文不可区分的.本文提出的方案只是在随机预言模型下证明了安全性,而随机预言模型下的安全性证明只是一种启发式证明,这并不是严格意义上的可证明安全性.因此,构造标准模型下可证安全的带标签的基于证书密钥封装方案是我们进一步的研究方向.

References:

[1] Cramer R, Shoup V. Design and analysis of practical public-key encryption schemes secure against adaptive chosen ciphertext attack. SIAM Journal on Computing, 2003,33(1):167−226. [doi: 10.1137/S0097539702403773]

[2] Kurosawa K, Desmedt Y. A new paradigm of hybrid encryption scheme. In: Franklin M, ed. Proc. of the CRYPTO 2004. LNCS 3152, Berlin, Heidelberg: Springer-Verlag, 2004. 426−442.

[3] Bellare M, Rogaway P. Random oracles are practical: A paradigm for designing efficient protocols. In: Denning D, Pyle R, Ganesan R, eds. Proc. of the 1st ACM Conf. on Computer and Communication Security. New York: ACM Press, 1993. 62−73. [doi: 10.1145/168588.168596]

[4] Okamoto T, Pointcheval D. REACT: Rapid enhanced-security asymmetric cryptosystem transform. In: Naccache D, ed. Proc. of the CT-RSA 2001. LNCS 2020, Berlin, Heidelberg: Springer-Verlag, 2001. 159−174.

[5] Shoup V. Using hash functions as a hedge against chosen ciphertext attack. In: Preneel B, ed. Proc. of the EUROCRYPT 2000. LNCS 1807, Berlin, Heidelberg: Springer-Verlag, 2000. 275−288.

[6] Fujisaki E, Okamoto T. Secure integration of asymmetric and symmetric encryption schemes. In: Wiener M, ed. Proc. of the CRYPTO'99. LNCS 1666, Berlin, Heidelberg: Springer-Verlag, 1999. 537−554.

[7] Abe M, Gennaro R, Kurosawa K, Shoup V. Tag-KEM/DEM: A new framework for hybrid encryption and a new analysis of Kurosawa-Desmedt KEM. In: Cramer R, ed. Proc. of the EUROCRYPT 2005. LNCS 3494, Berlin, Heidelberg: Springer-Verlag, 2005. 128−146. [doi: 10.1007/s00145-007-9010-x]

[8] Bjørstad TE, Dent AW. Building better signcryption schemes with tag-KEMs. In: Yung M, *et al*., eds. Proc. of the PKC 2006. LNCS 3958, Berlin, Heidelberg: Springer-Verlag, 2006. 491−507. [doi: 10.1007/11745853_32]

[9] Yoshida M, Fujiwara T. On the security of Tag-KEM for signcryption. Electronic Notes in Theoretical Computer Science, 2007, 171(1):83−91. [doi: 10.1016/j.entcs.2006.11.011]

[10] Huang Q, Wong DS. Generic certificateless key encapsulation mechanism. In: Pieprzyk J, Ghodosi H, Dawsom E, eds. Proc. of the ACISP 2007. LNCS 4586, Berlin, Heidelberg: Springer-Verlag, 2007. 215−229.

[11] Tan CH. Insider-Secure signcryption KEM/Tag-KEM schemes without random oracles. In: Proc. of the 2008 3rd Int'l Conf. on Availability, Reliability and Security. Barcelona: IEEE, 2008. 1275−1281. [doi: 10.1109/ARES.2008.112]

[12] Matsuda T, Nishimaki R, Numayama A, Tanaka K. Security on hybrid encryption with the Tag-KEM/DEM framework. In: Boyd C, González J, eds. Proc. of the ACISP 2009. LNCS 5594, Berlin, Heidelberg: Springer-Verlag, 2009. 343−359. [doi: 10.1007/978-3-642-02620-1_24]

[13] Li F, Shirase M, Takagi T. Certificateless hybrid signcryption. In: Bao F, Li H, Wang G, eds. Proc. of the ISPEC 2009. LNCS 5451, Berlin, Heidelberg: Springer-Verlag, 2009. 112−123. [doi: 10.1007/978-3-642-00843-6_11]

[14] Selvi S, Vivek S, Rangan C. Certificateless KEM and hybrid signcryption schemes revisited. In: Kwak J, *et al*., eds. Proc. of the Information Security, Practice and Experience 2010. LNCS 6047, Berlin, Heidelberg: Springer-Verlag, 2010. 294−307. [doi: 10.1007/978-3-642-12827-1_22]

[15] Gentry C. Certificate-Based encryption and the certificate revocation problem. In: Biham E, ed. Proc. of the EUROCRYPT 2003. LNCS 2656, Berlin, Heidelberg: Springer-Verlag, 2003. 272−293.

[16] Long Y, Chen KF. Efficient chosen-ciphertext secure certificateless threshold key encapsulation mechanism. Information Sciences, 2010,180(7):1167−1181. [doi: 10.1016/j.ins.2009.12.008]

[17] Li JG, Huang XY, Mu Y, Susilo W, Wu QH. Constructions of certificate-based signature secure against key replacement attacks. Journal of Computer Security, 2010,18(3):421−449.

[18] Li JG, Huang XY, Mu Y, Susilo W, Wu QH. Certificate-Based signature: security model and efficient construction. In: Lopez J, Samarati P, Ferrer JL, eds. Proc. of the EuroPKI 2007. LNCS 4582, Berlin, Heidelberg: Springer-Verlag, 2007. 110−125. [doi: 10.1007/978-3-540-73408-6_8]

[19] Li JG, Huang XY, Zhang YC, Xu LZ. An efficient short certificate-based signature scheme. Journal of Systems and Software, 2012, 85(2):314−322. [doi: http://dx.doi.org/10.1016/j.jss.2011.08.014]

李继国(1970－),男,黑龙江富裕人,博士,教授,博士生导师,CCF 会员,主要研究领域为信息安全,密码学理论与技术.

张亦辰(1971－),女,博士生,讲师,主要研究领域为密码学理论与技术.

杨海珊(1985－),女,工程师,主要研究领域为密码学理论与技术.