抗泄露的(分层)身份基密钥封装机制

周彦伟[1),2),3)] 杨 波[1),2)] 胡冰洁[1)] 夏 喆[4)] 张明武[2),3)]

[1)](陕西师范大学计算机科学学院 西安 710062)

[2)](密码科学技术国家重点实验室 北京 100878)

[3)](桂林电子科技大学广西可信软件重点实验室 广西 桂林 541004)

[4)](武汉理工大学计算机科学与技术学院 武汉 430070)

摘 要 在真实应用环境中,冷启动、边信道等物理攻击方式的出现,使得敌手能够获得参与者内部私有状态的泄露信息,从而导致传统可证明安全的密码机制在有泄露的环境下已无法继续保持其原有的安全性,因此更多的密码学研究者开始致力于抗泄露密码机制的研究.混合加密技术同时具备了对称加密和非对称加密的优势,由于身份基密钥封装机制(Identity-Based Key-Encapsulation Mechanism,IB-KEM)是身份基混合加密机制的重要组成部分,近年来得到了广泛关注.为满足真实环境的抗泄露性需求,抗泄露 IB-KEM 被提出;然而现有的构造在计算、传输和存储等方面均存在不足.针对上述不足,本文提出了选择密文攻击(Chosen-Ciphertext Attacks,CCA)安全的抗泄露 IB-KEM 的通用构造,并基于底层 IB-KEM 的选择明文攻击(Chosen-Plaintext Attacks,CPA)安全性对通用构造的 CCA 安全性进行了形式化证明.此外,为展示本文通用构造的实用性及普遍性,分别设计了 IB-KEM 和分层身份的身份基密钥封装机制(Hierarchical Identity-Based Key-Encapsulation Mechanism,HIB-KEM)的具体实例,并在选择身份的安全模型下,基于判定的双线性 Diffie-Hellman 假设和双线性 Diffie-Hellman 指数假设对本文实例的 CPA 安全性分别进行了证明.最后,为了实现抵抗连续泄露攻击的目标,本文研究了各实例的密钥更新算法.相较于已有 CCA 安全的抗泄露 IB-KEM,本文构造在计算、传输和存储等方面具有一定的优势.

关键词 身份基密码学;身份基密钥封装机制;身份分层的身份基密钥封装机制;有界泄露模型;连续泄露模型

中图法分类号 TP393 **DOI 号** 10.11897/SP.J.1016.2021.00820

(Hierarchical) Identity-Based Key-Encapsulation Mechanism with Leakage-Resilience

ZHOU Yan-Wei[1),2),3)] YANG Bo[1),2)] HU Bing-Jie[1)] XIA Zhe[4)] ZHANG Ming-Wu[2),3)]

[1)](*School of Computer Science, Shaanxi Normal University, Xi'an* 710062)

[2)](*State Key Laboratory of Cryptology, Beijing* 100878)

[3)](*Guangxi Key Laboratory of Trusted Software, Guilin University of Electronic Technology, Guilin, Guangxi* 541004)

[4)](*School of Computer Science and Technology, Wuhan University of Technology, Wuhan* 430070)

Abstract In the actual application, any adversary can obtain a certain amount of additional information on the internal secret states by performing various leakage attacks, such as cold boot attacks, side-channel attacks, etc. Thus, the security of the traditional cryptography system cannot keep their claimed security in the leakage setting, because we always assume that an adversary cannot capture the leakage on the internal secret states of participator in the traditional ideal security model, and the traditional security was proved in the above ideal security model.

收稿日期:2020-02-19;在线发布日期:2020-05-02.本课题得到国家重点研发计划(2017YFB0802000)、国家自然科学基金(U2001205,61802242,61772326,61802241)、"十三五"国家密码发展基金(MMJJ20180217)、中央高校基本科研业务费(GK202003079,GK202007033)资助.**周彦伟**,博士,高级工程师,硕士生导师,主要研究兴趣为密码学、匿名通信技术等.E-mail:zyw_snnu@foxmail.com.**杨 波**(通信作者),博士,教授,博士生导师,陕西省"百人计划"特聘教授,主要研究领域为信息安全、密码学等.E-mail:byang@snnu.edu.cn.**胡冰洁**,硕士研究生,主要研究方向为信息安全、密码学等.**夏 喆**,博士,副教授,硕士生导师,主要研究方向为信息安全、密码学等.**张明武**,博士,教授,博士生导师,主要研究领域为信息安全、密码学等.

Since the hybrid encryption technology has the advantages of both symmetric and asymmetric encryption, the research on identity-based key-encapsulation mechanism (IB-KEM) has received extensive attention in recent years, which is the important underlying technology of identity-based hybrid encryption. To obtain the leakage resilience, a leakage-resilient IB-KEM with chosen ciphertext attacks (CCA) security was proposed. However, the previous scheme has shortcomings in computing, transmission and storage. To address the above shortcomings, a generic construction of CCA secure IB-KEM with leakage resilience is proposed in this paper, and the formal proof can be given from the underlying chosen-plaintext attacks (CPA) secure IB-KEM. In additional, to further show the practicability of the above generic construction, two instances of IB-KEM and hierarchical identity-based key-encapsulation mechanism (HIB-KEM) are proposed, the corresponding CPA security is proved based on the decisional bilinear Diffie-Hellman (DBDH) and the bilinear Diffie-Hellman exponent (BDHE) assumptions, respectively. In additional, based on our generic construction of IB-KEM, a leakage-resilient HIB-KEM with CCA security can be also proposed. Finally, in order to achieve the goal of resisting continuous leakage attacks, key update algorithms of each instance are also researched in this paper, because the previous conclusions shown that the continuous leakage-resilient problem can be obtained from the corresponding bounded leakage resilience by performing an additional key update algorithm. Analysis and comparison show that our construction of leakage-resilient IB-KEM with CCA security has certain advantages in computing, transmission and storage.

Keywords　identity-based cryptography; identity-based key-encapsulation mechanism; hierarchical identity-based key-encapsulation mechanism; bounded leakage model; continuous leakage model

《计算机学报》

1　引　言

为了改善传统公钥密码机制中复杂的证书管理问题，1984 年，Shamir 提出了身份基密码机制(Identity-Based Cryptography，IBC)的概念[1]. 在 IBC 中，用户的电话、邮箱、证件号码等唯一的身份信息将直接作为用户的公钥，其对应的私钥由可信第三方—私钥生成中心(Private Key Generation，PKG)为其生成，由于身份信息与用户间具有自然的绑定关系，因此无需额外的证书来完成两者间的联系，从而简化了传统公钥机制的证书管理问题. Boneh 和 Franklin[2] 基于双线性映射提出了第一个实用的身份基加密(Identity-Based Encryption，IBE)机制，并在随机谕言机模型下证明了其方案的安全性. 为进一步提升 IBE 机制的实用性，多个标准模型下的 IBE 机制相继被提出[3-6]. 为改进传统 IBE 机制中单 PKG 模式所存在的 PKG 负载大，易遭受攻击等不足，研究者提出了身份分层的身份基加密(Hierarchical Identity-Based Encryption，HIBE)机制[7-9]. HIBE 实际上是一个多 PKG 的 IBE 机制，其中每个节点既可以是用户，也可是下一层用户的 PKG，也就是说，每个节点能够负责部分用户的私钥生成，那么节点所对应的私钥可以由根 PKG 生成，也可以由上一层父节点生成，同时它也是下一层子节点的 PKG.

传统安全模型均假设敌手只能观察到密码机制特定的输入和输出，密码机制具体执行过程的相关内部私有状态(如用户私钥等)都无法接触；然而，冷启动、边信道等物理攻击方式的出现，使得敌手能够通过上述泄露攻击方式获得参与者内部私有状态的部分泄露信息，因此在传统安全模型下已证明安全的密码机制在真实应用中由于泄露攻击的存在导致其已无法满足相应的安全性. 为进一步增强密码机制的实用性，近年来抗泄露密码学原语的研究得到了众多密码学研究者的关注[10-22]. 此外，在实际应用中由于敌手能够连续多次的执行泄露攻击，那么限制敌手获得泄露信息的总量不能超过相应泄露参数的假设还是无法满足实际应用的需求，因此需继续研究密码原语的连续泄露容忍性.

1.1　研究现状

2010 年，Alwen 等人[11] 基于提出的新密码原

语—身份基哈希证明系统(Identity-Based Hash Proof System,IB-HPS)和强随机性提取器设计了构造选择明文攻击(Chosen Plaintext Attacks,CPA)安全的抗泄露 IBE 机制的通用方法;基于上述方法,Chow 等人[23]利用已有的 IBE 机制[4-5,9]设计了三个 IB-HPS 的具体实例.为了达到选择密文攻击(Chosen-Ciphertext Attacks,CCA)安全性,基于 Gentry 的 IBE 构造方法[6],文献[24]设计了 CCA 安全的抗有界泄露攻击的 IBE 机制.为了实现接收者的匿名性保护需求,文献[25]设计了一个新颖的抗泄露 HIBE 机制.由于部分抗泄露 IBE 机制的泄露参数会随着明文消息的增加而减少,文献[26]受文献[12]研究思路的启发,设计了泄露参数不受待加密消息长度限制的抗泄露 IBE 机制.文献[27]基于对偶系统加密(Dual System Encryption,DSE)技术设计了一个完全安全的抗泄露 IBE 机制,然而,合数阶双线性群的使用,导致该构造的效率受到了一定影响.

现实环境中的敌手能够进行持续的泄露攻击以获得更多秘密信息的泄露,因此为增强密码机制的实用性,应进一步研究其抵抗连续泄露攻击的能力.Dodis 等人在文献[17]中指出具有有界泄露容忍性的密码机制若满足条件(1)保持密码机制公开参数不变的前提下,对用户私钥能够进行定期更新,(2)用户私钥更新前后相应机制的功能及安全性保持不变和(3)任意敌手均无法区分更新后的用户私钥与原始私钥,那么该机制能够抵抗连续的泄露攻击.基于上述结论,文献[28-29]在连续泄露模型中提出了两种抵抗连续泄露攻击的 IBE 机制,增强了 IBE 机制的抗泄露攻击的能力.由于不同应用环境的泄露需求各不相同,用一个不变的泄露界很难满足现实中不同环境的实际应用需求,为了实现根据应用环境的需求动态设置 IBE 机制泄露界的目标,文献[30]设计了泄露界可灵活变化的抗泄露 IBE 机制,其中可根据实际环境的泄露需求通过改变相应的初始化参数来控制 IBE 机制的泄露上界,切实做到了泄露界的按需设计目标;该机制在保持公开参数不变的前提下,通过增加用户私钥的长度达到提升机制抗泄露攻击的能力.基于现有 IB-HPS 的定义,文献[31]提出了可更新的身份基哈希证明系统(Updatable Identity-Based Hash Proof System,U-IB-HPS)的新密码学原语,并基于该技术设计了 IBE 机制、基于身份的混合加密机制和基于身份的密钥协商协议等密码机制抗泄露版本的通用构造.

相较于公钥加密运算而言,对称加密的运算效率较高,而公钥加密机制的安全性能更优,因此联合两种运算的混合加密形式被提出.由于混合加密兼顾了两种密码机制的优势,在实际应用中得到广泛的使用.身份基密钥封装机制(Identity-Based Key-Encapsulation Mechanism,IB-KEM)作为身份基混合加密机制的重要组成部分,近年来得到了广泛关注,文献[32]设计了一个新的 IB-KEM,并在云计算环境对该技术的应用进行了介绍.文献[33]针对电子邮件系统的应用需求设计了一个具有通配符功能的 IB-KEM.2019 年,为满足 IB-KEM 抵抗泄露攻击的实际应用需求,文献[34]设计了第一个 CCA 安全的抗泄露 IB-KEM;然而,由于该机制基于矩阵运算构造导致公开参数、主密钥和用户私钥的长度较长,一定程度上增加了用户的存储、计算和传输的负载,不具备在实际环境中的应用潜力.

针对上述不足,本文以 IB-KEM 的抗泄露性为研究目标,设计安全高效的抗泄露 IB-KEM;鉴于身份分层结构在实际应用中的优势,本文在 IB-KEM 的基础上研究身份分层的身份基密钥封装机制(Hierarchical Identity-Based Key-Encapsulation Mechanism,HIB-KEM)的泄露容忍性;为进一步提升实用性,通过对 IB-KEM 和 HIB-KEM 实例设计相应的密钥更新算法,本文对上述实例抵抗连续泄露攻击的能力进行了讨论.

1.2 我们的思路

为实现高效构造 CCA 安全的公钥加密机制(Public-Key Encryption,PKE)的目标,Canetti、Halevi 和 Katz 在文献[35]中提出了由选择身份安全的 IBE 机制来构造 CCA 安全 PKE 机制的通用构造方法(该方法简称为 CHK 转换),底层 IBE 机制只需满足标准模型下较弱的选择身份安全性,CHK 转换即可构造出标准模型下 CCA 安全的 PKE 机制.受 CHK 转换思路的启发,本文首先基于 CPA 安全的 IB-KEM 设计 CCA 安全的 IB-KEM 的通用构造(由该方法很容易推广得到 CCA 安全的 HIB-KEM 的通用构造);然后分别设计标准模型下 CPA 安全的 IB-KEM 和 HIB-KEM 的具体实例.

如图 1 所示为 CCA 安全的 IB-KEM 通用构造封装算法的设计思路.该算法通过不同的随机数 r_1 和 r_2 两次调用底层 CPA 安全的 IB-KEM 的封装算法 Encap′,分别产生两组封装密文密钥对 (c_1,k_1) 和

(c_2,k_2)，其中 k_1 作为强随机性提取器 Ext 的输入，在随机提取种子 S 的作用下产生具有抗泄露攻击能力的封装密钥 k；k_2 作为消息验证码标签生成算法 Tag 的对称密钥，协助该算法输出关于消息 $\mathcal{H}(c_1,c_2,S)$ 的认证标签 Tag，其中消息验证码的强不可伪造性保证了封装密文 $C=(c_1,c_2,S,Tag)$ 的不可延展性. 也就是说，基于强随机性提取器将底层 IB-KEM 封装密钥的随机性转换为对任意敌手而言的均匀随机性，用消息验证码防止密文被扩张.

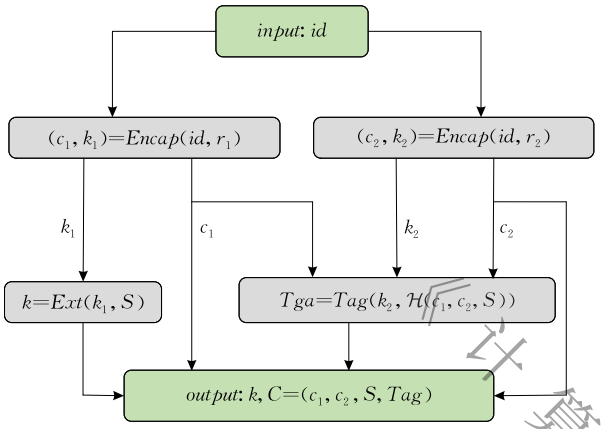

图 1　封装算法的设计思路

1.3　我们的工作

为了进一步提升 IB-KEM 和 HIB-KEM 的实用性，我们设计了抗泄露的 IB-KEM 和 HIB-KEM，主要贡献有：

(1) 联合 CPA 安全的 IB-KEM、消息验证码和强随机性提取器，设计了 CCA 安全的抗泄露 IB-KEM 的通用构造，并基于底层技术的安全性对通用构造的安全性进行了形式化证明；并且该方法可推广到 HIB-KEM 中，即由 CPA 安全的 HIB-KEM 能够设计 CCA 安全的 HIB-KEM 的通用构造.

(2) 为表明通用构造的实用性，本文构造了相应的 CPA 安全的 IB-KEM 和 HIB-KEM 的实例，并分别基于判定的双线性 Diffie-Hellman 假设和双线性 Diffie-Hellman 指数假设对相应构造的 CPA 安全性在选择身份的安全模型下进行了形式化证明.

(3) 已有研究[17]表明抗有界泄露攻击的密码机制在一定条件下能够基于密钥更新操作实现抵抗连续泄露攻击的目标，基于该理论，为了构造 CCA 安全的抗连续泄露的(H)IB-KEM 的实例，我们分别讨论了 IB-KEM 和 HIB-KEM 实例密钥更新算法的设计.

2　基础知识

本节将介绍最小熵、平均最小熵、随机性提取器和消息验证码等基础知识.

2.1　相关符号

用 κ 表示安全参数；$a\leftarrow_R A$ 表示从集合 A 中均匀随机的选取元素 a；$\mathrm{negl}(\kappa)$ 表示在安全参数 κ 上是计算可忽略的；$x\leftarrow\mathcal{A}(y)$ 表示算法 $\mathcal{A}$ 在输入 y 的作用下输出相应的计算结果 x.

2.2　统计距离和最小熵

我们首先介绍统计距离的概念. 令 X 与 Y 是有限域 Ω 上的任意两个随机变量，那么上述变量间的统计距离可表示为

$$\mathrm{SD}(X,Y)=\frac{1}{2}\sum_{w\in\Omega}|\Pr[X=w]-\Pr[Y=w]|.$$

定义 1.　设 X 是确定的随机变量，则变量 X 的最小熵可表示为

$$H_\infty(X)=-\log(Max_x\Pr[X=x]).$$

特别地，$H_\infty(X)$ 表示无任何信息协助的前提下任意敌手猜中变量 X 的最大概率，即变量 X 的最小熵 $H_\infty(X)$ 体现了 X 的不可预测性.

定义 2.　当变量 B 已知时，变量 A 的平均最小熵可表示为

$$\widetilde{H}_\infty(A|B)=-\log(E_{b\leftarrow B}[2^{-H_\infty(A|B=b)}]),$$

其中，E 表示数学期望运算. 平均最小熵 $\widetilde{H}_\infty(A|B)$ 表示在变量 B 已知的前提下，变量 A 的不可预测性；也就是说，任意敌手在变量 B 的协助下猜中变量 A 的概率.

定理 1.　对于任意的随机变量 A,B 和 C，若 B 的取值最多有 2^l 个，则有

$$\widetilde{H}_\infty(A|(B,C))\geq\widetilde{H}_\infty(A|C)-l.$$

2.3　随机性提取

定义 3.　令随机变量 X 和 Y 满足条件 $X\in\{0,1\}^{l_n}$ 和 $\widetilde{H}_\infty(X|Y)\geq k$，对于任意的 $R\in\{0,1\}^{l_t}$ 和 $Z\in\{0,1\}^{l_m}$，若 $\mathrm{SD}((\mathrm{Ext}(X,R),R,Y),(Z,R,Y))\leq\varepsilon$ 成立，则称函数 $\mathrm{Ext}:\{0,1\}^{l_n}\times\{0,1\}^{l_t}\rightarrow\{0,1\}^{l_m}$ 是平均情况的 (k,ε)-强随机性提取器，其中 $R\in\{0,1\}^{l_t}$ 是可公开的随机性种子.

2.4　区别引理

引理 1(区别引理)[12].　令 $\mathcal{F}_1,\mathcal{F}_2$ 和 $\mathcal{E}$ 分别表示三个事件，当事件 $\mathcal{E}$ 不发生时，事件 $\mathcal{F}_1$ 和 $\mathcal{F}_2$ 发生的概率是相同的，即 $\Pr[\mathcal{F}_1|\bar{\mathcal{E}}]=\Pr[\mathcal{F}_2|\bar{\mathcal{E}}]$，那么有关系 $|\Pr[\mathcal{F}_1]-\Pr[\mathcal{F}_2]|\leq\Pr[\mathcal{E}]$ 成立.

计算机学报

2.5 双线性映射

群生成算法 $\mathcal{G}(1^\kappa)$ 的输入为安全参数 κ，输出是元组 $(q,g,G,G_T,e(\cdot))$，其中 G 和 G_T 是阶为大素数 q 的乘法循环群，g 为群 G 的生成元，$e:G\times G\rightarrow G_T$ 是满足下述性质的双线性映射：

双线性：对于任意的 $a,b\in Z_q^*$，有 $e(g^a,g^b)=e(g,g)^{ab}$ 成立；

非退化性：有 $e(g,g)=1_{G_T}$ 成立，其中 1_{G_T} 是群 G_T 的单位元；

可计算性：对于任意的 $P,Q\in G$，$e(P,Q)$ 可在多项式时间内完成计算.

2.6 安全性假设

判定的双线性 Diffie-Hellman(Decisional Bilinear Diffie-Hellman，DBDH)问题. 对于公开参数 $(q,g,G,G_T,e(\cdot))\leftarrow\mathcal{G}(1^\kappa)$ 和任意未知的指数 $a,b,c,d\in Z_q^*$，给定两个元组 $(g,g^a,g^b,g^c,e(g,g)^{abc})$ 和 $(g,g^a,g^b,g^c,e(g,g)^d)$. DBDH 问题的目标是判断 $e(g,g)^{abc}=e(g,g)^d$ 是否成立.

定义 4(DBDH 假设). 任意的概率多项式时间(Probabilistic Polynomial Time，PPT)算法 $\mathcal{A}$ 成功解决 DBDH 问题的优势

$$\mathrm{Adv}^{\mathrm{DBDH}}(\kappa)=\Pr[\mathcal{A}(g,g^a,g^b,g^c,e(g,g)^{abc})=1]-\Pr[\mathcal{A}(g,g^a,g^b,g^c,e(g,g)^d)=1]$$

是可忽略的，其中概率来源于 a,b,c,d 在 Z_q^* 上的随机选取和算法 $\mathcal{A}$ 的随机选择.

双线性 Diffie-Hellman 指数(Bilinear Diffie-Hellman Exponent，BDHE)问题. 对于已知的公开参数 $(q,g,G,G_T,e(\cdot))\leftarrow\mathcal{G}(1^\kappa)$ 和任意未知的随机指数 $\alpha,c\in Z_q^*$，给定两个元组 $\mathcal{T}_1=(g,g^c,g^\alpha,g^{\alpha^2},\cdots,g^{\alpha^{\mu-1}},g^{\alpha^{\mu+1}},\cdots,g^{\alpha^{2\mu}},T_1)$ 和 $\mathcal{T}_0=(g,g^c,g^\alpha,g^{\alpha^2},\cdots,g^{\alpha^{\mu-1}},g^{\alpha^{\mu+1}},\cdots,g^{\alpha^{2\mu}},T_0)$，其中 $T_1=e(g^c,g)^{\alpha^\mu}$ 和 $T_0\leftarrow_R G_T$. BDHE 问题的目标是区分上述两个元组 $\mathcal{T}_1$ 和 $\mathcal{T}_0$. 为了表述的方便，令 $x=g^c$ 和 $y_i=g^{(\alpha^i)}$.

定义 5(BDHE 假设). 任意的 PPT 算法 $\mathcal{A}$ 成功解决 BDHE 问题的优势

$$\mathrm{Adv}^{\mu\text{-BDHE}}(\kappa)=\Pr[\mathcal{A}(g,x,y_1,\cdots y_{\mu-1},y_{\mu+1},\cdots,y_{2\mu},T_1)=1]-\Pr[\mathcal{A}(g,x,y_1,\cdots,y_{\mu-1},y_{\mu+1},\cdots,y_{2\mu},T_0)=1]$$

是可忽略的，其中概率来源于随机值 a,c 在 Z_q^* 上的选取和算法 $\mathcal{A}$ 的随机选择.

2.7 消息验证码

密钥空间 $\mathcal{K}$ 和消息空间 $\mathcal{M}$ 上的消息验证码 MAC=(Tag，Ver)包含以下两个算法：

(1) $\mathrm{Tag}(k,m)$. 输入密钥空间中的对称密钥 $k\in\mathcal{K}$ 和消息空间中的消息 $m\in\mathcal{M}$，标签算法 Tag 输出一个认证标签 Tag.

(2) $\mathrm{Ver}(k,m,Tag)$. 输入密钥空间中的对称密钥 $k\in\mathcal{K}$、消息空间中的消息 $m\in\mathcal{M}$ 和认证标签 Tag，验证算法 Ver 输出相应的验证结果 0 或 1，其中 1 表示 Tag 是关于消息 m 的认证标签，否则输出 0.

消息验证码 MAC=(Tag，Ver)的正确性要求，对于密钥空间 $\mathcal{K}$ 上任意的对称密钥 $k\in\mathcal{K}$，有

$$\mathrm{Ver}(k,m,\mathrm{Tag}(k,m))=1$$

成立.

消息验证码 MAC=(Tag，Ver)的安全性通过下述交互式实验 $\mathrm{Exp}_{\mathrm{MAC}}^{\text{suf-cmva}}(\kappa)$ 描述：

(1) 从密钥空间中随机选取对称密钥 $k\in\mathcal{K}$.

(2) 运行 $\mathcal{A}^{\mathrm{Tag}(k,\cdot),\mathrm{Ver}(k,\cdot,\cdot)}(\kappa)$，其中 $\mathrm{Tag}(k,\cdot)$ 是标签谕言机，敌手 $\mathcal{A}$ 能够从它获得相应消息 m 的认证标签 Tag；$\mathrm{Ver}(k,\cdot,\cdot)$ 是验证谕言机，敌手 $\mathcal{A}$ 能从它获得消息 m 和相应标签 Tag 的验证结果.

(3) 敌手 $\mathcal{A}$ 输出一个挑战消息标签对 (m^*,Tag^*)，并且 (m^*,Tag^*) 与之前标签谕言机 $\mathrm{Tag}(k,\cdot)$ 返回的所有值 (m_i,Tag_i) 均不相同. 若有 $\mathrm{Ver}(k,m^*,Tag^*)=1$ 成立，则输出 1，否则输出 0. 特别地，若元组 (m^*,Tag') 满足条件 $(m^*,Tag^*)\neq(m^*,Tag')$，则 (m^*,Tag') 可以出现在标签谕言机 $\mathrm{Tag}(k,\cdot)$ 的返回值列表中，即敌手 $\mathcal{A}$ 可以对挑战消息 m^* 进行标签生成询问，但挑战标签 Tag^* 不能由标签谕言机生成.

敌手 $\mathcal{A}$ 在上述实验中获胜的优势定义为

$$\mathrm{Adv}_{\mathrm{MAC}}^{\text{suf-cmva}}(\kappa)=\Pr[\mathrm{Exp}_{\mathrm{MAC}}^{\text{suf-cmva}}(\kappa)=1].$$

定义 6(消息验证码的强不可伪造性). 对于任意的 PPT 敌手 $\mathcal{A}$，若有 $\mathrm{Adv}_{\mathrm{MAC}}^{\text{suf-cmva}}(\kappa)\leq\mathrm{negl}(\kappa)$ 成立，那么该消息验证码 MAC=(Tag，Ver)在选择消息和选择验证询问攻击下是强不可伪造的.

3 (分层)身份基密钥封装机制

本节我们将回顾 IB-KEM 和 HIB-KEM 的形式化定义及安全属性.

3.1 形式化定义

一个 IB-KEM 包含 4 个 PPT 算法 Setup、KeyGen、Encap 和 Decap. 算法的具体描述如下所述：

(1) $(mpk,msk)\leftarrow\mathrm{Setup}(1^\kappa)$. 初始化算法 Setup 以系统安全参数 κ 为输入，输出相应的系统公开参

《计算机学报》

数 mpk 和主密钥 msk，其中 mpk 定义了系统的用户身份空间 $\mathcal{ID}$，封装密钥空间 $\mathcal{K}$，封装密文空间 $\mathcal{C}$ 和用户私钥空间 $\mathcal{SK}$. 此外，mpk 是其它算法 KeyGen、Encap 和 Decap 的隐含输入；为了方便起见，下述算法的输入列表中并未将其列出.

(2) $sk_{id}\leftarrow$KeyGen(msk,id). 对于输入的任意身份 $id\in\mathcal{ID}$，密钥生成算法 KeyGen 以主密钥 msk 作为输入，输出身份 id 所对应的私钥 sk_{id}. 特别地，每次运行该算法，概率性的密钥生成算法基于不同的随机数为用户生成不同的私钥.

(3) $(C,k)\leftarrow$Encap(id). 对于输入的任意身份 $id\in\mathcal{ID}$，封装算法 Encap 输出封装密文 $C\in\mathcal{C}$ 及相应的封装密钥 $k\in\mathcal{K}$.

(4) $k\leftarrow$Decap(sk_{id},C). 对于确定性的解封装算法，输入身份 id 所对应的私钥 d_{id} 和封装密文 C，输出相应的解封装密钥 k.

在身份分层的身份基密码机制中，身份是一个向量，将深度为 k 的身份表示为一个长度为 k 的向量 $\boldsymbol{id}_k=(I_1,I_2,\cdots,I_k)$，其中第 i 个分量 I_i 表示第 i 层的身份信息. 一个 HIB-KEM 除包含上述四个 PPT 算法之外，还额外包含一个密钥派生算法 Delegate，该算法的具体定义如下所述：

$sk_{\boldsymbol{id}_k}\leftarrow$Delegate$(sk_{id_{k-1}},\boldsymbol{id}_k)$. 输入第 k 层的身份 $\boldsymbol{id}_k$ 和第 $k-1$ 层身份 id_{k-1} 所对应的私钥 $sk_{id_{k-1}}$，密钥派生算法 Delegate 将输出第 k 层身份 $\boldsymbol{id}_k$ 所对应的私钥 sk_{id}.

特别地，在 HIB-KEM 中，初始化算法 Setup 输出的系统公开参数 mpk 中将额外定义一个分层身份结构的最大深度 l，限制了系统所能支持的最大身份深度.

3.2　正确性

对于 IB-KEM 而言，身份空间 $\mathcal{ID}$ 中的任意身份 $id\in\mathcal{ID}$，有

$$\begin{aligned}\Pr[k\neq k'\mid & sk_{id}\leftarrow\text{KeyGen}(msk,id),\\ & (C,k)\leftarrow\text{Encap}(id),\\ & k'\leftarrow\text{Decap}(sk_{id},C)]\leq\text{negl}(\kappa)\end{aligned}$$

成立，其中 $(mpk,msk)\leftarrow$Setup(1^κ).

对于 HIB-KEM 而言，身份空间 $\mathcal{ID}$ 中的任意身份 $\boldsymbol{id}_k\in\mathcal{ID}$，有

$$\begin{aligned}\Pr[k\neq k'\mid & sk_{\boldsymbol{id}_k}\leftarrow\text{KeyGen}(msk,\boldsymbol{id}_k),\\ & (C,k)\leftarrow\text{Encap}(\boldsymbol{id}_k),\\ & k'\leftarrow\text{Decap}(sk_{\boldsymbol{id}_k},C)]\leq\text{negl}(\kappa)\end{aligned}$$

成立，其中 $(mpk,msk)\leftarrow$Setup(1^κ). 特别地，对于派生算法生成的私钥 $sk_{\boldsymbol{id}_k}\leftarrow$Delegate$(sk_{id_{k-1}},\boldsymbol{id}_k)$，上述概率依然是可忽略的，即有下述不等式成立.

$$\begin{aligned}\Pr[k\neq k'\mid & sk_{\boldsymbol{id}_k}\leftarrow\text{Delegate}(sk_{id_{k-1}},\boldsymbol{id}_k),\\ & (C,k)\leftarrow\text{Encap}(\boldsymbol{id}_k),\\ & k'\leftarrow\text{Decap}(sk_{\boldsymbol{id}_k},C)]\leq\text{negl}(\kappa).\end{aligned}$$

3.3　安全性

在泄露环境下，通过赋予敌手访问泄露谕言机的能力实现对泄露攻击的模拟.

定义 7(泄露谕言机).　泄露谕言机 $\mathcal{O}_{sk_{id}}^{\lambda,\kappa}(\cdot)$ 的输入是安全参数 κ，泄露参数 λ 和用户私钥 sk_{id}，收到敌手提交的任意高效可计算的泄露函数 $f_i:\{0,1\}^*\rightarrow\{0,1\}^{\lambda_i}$ 后，返回 sk_{id} 的泄露信息 $f_i(sk_{id})$，但是对于同一私钥 sk_{id} 的泄露总量不能超过 λ，即 $\sum_{t=1}^{i}f_t(sk_{id})\leq\lambda$；否则将返回终止符 $\perp$.

对于 IB-KEM 而言，泄露容忍的选择身份 CCA(Selective-Identity CCA，SID-CCA)安全性游戏由模拟器 $\mathcal{S}$ 和敌手 $\mathcal{A}$ 执行，其中 κ 是安全参数，λ 是泄露参数，其中敌手 $\mathcal{A}$ 的目标是判断挑战阶段来自模拟器 $\mathcal{S}$ 的 k_ν^* 是与挑战封装密文 C^* 相对应的封装密钥，还是封装密钥空间 $\mathcal{K}$ 中的随机值. 模拟器 $\mathcal{S}$ 和敌手 $\mathcal{A}$ 间具体的消息交互过程如下所述：

选择身份的安全模型要求，在系统初始化之前敌手 $\mathcal{A}$ 向模拟器 $\mathcal{S}$ 提交其选定的挑战身份 id^*，并且限制 id^* 不能在任何密钥生成询问中出现，此外 id^* 对应私钥 sk_{id^*} 的泄露信息总量不能超过系统设定的泄露界 λ.

(1) 初始化. 模拟器 $\mathcal{S}$ 输入安全参数 κ，运行初始化算法 Setup(1^κ)，产生公开的系统参数 mpk 和保密的主密钥 msk，发送 mpk 给敌手 $\mathcal{A}$.

(2) 阶段 1(训练). 在该阶段敌手 $\mathcal{A}$ 适应性地进行多项式有界次的下述询问：

① 密钥生成询问. 对于身份 $id(id\neq id^*)$ 的密钥生成询问，模拟器 $\mathcal{S}$ 运行密钥生成算法 KeyGen，返回相应的密钥 sk_{id} 给敌手 $\mathcal{A}$.

② 解封装询问. 对于身份 id 和封装密文 C 的解封装询问，模拟器 $\mathcal{S}$ 首先运行密钥生成算法 KeyGen，产生与身份 id 相对应的密钥 sk_{id}；然后以 sk_{id} 作为输入运行解封装算法 Decap，并返回相应的解封装结果给敌手 $\mathcal{A}$.

③ 泄露询问. 对于身份 id 对应私钥 sk_{id} 的泄露询问，模拟器 $\mathcal{S}$ 运行密钥生成算法 KeyGen，产生身份 id 对应的私钥 sk_{id}，再运行泄露谕言机 $\mathcal{O}_{sk_{id}}^{\lambda,\kappa}(\cdot)$，产生密钥 sk_{id} 的泄露信息 $f_i(sk_{id})$，并把 $f_i(sk_{id})$ 发送给敌手 $\mathcal{A}$，其中 $f_i:\{0,1\}^*\rightarrow\{0,1\}^{\lambda_i}$ 是高效可计

计算机学报

算的泄露函数;但是在整个泄露询问过程中关于同一密钥 sk_{id} 泄露信息的总量不能超过系统设定的泄露界 λ,即有 $\sum_{t=1}^{i} f_t(sk_{id}) \leq \lambda$ 成立;否则 $\mathcal{S}$ 将输出终止符号 $\perp$ 给 $\mathcal{A}$.

(3)挑战. $\mathcal{S}$ 计算 $(C^*, k_1^*) \leftarrow \text{Encap}(id^*)$,然后随机选取 $k_0^* \leftarrow \mathcal{K}$ 和 $\nu \leftarrow_R \{0,1\}$,并将 (C^*, k_ν^*) 发送给 $\mathcal{A}$.

(4)阶段 2(训练). 该阶段敌手可进行多项式有界次的密钥生成询问和解封装询问. 特别地,敌手在该阶段不能提交任何泄露询问. 能够对除挑战身份 id^* 之外的任何身份 $id(id \neq id^*)$ 进行密钥生成询问;但不能对挑战身份 id^* 和挑战封装密文 C^* 进行解封装询问. 模拟器 $\mathcal{S}$ 以阶段 1 中的方式对相应的询问进行应答.

(5)猜测. 敌手 $\mathcal{A}$ 输出对随机数 ν 的猜测 ν'. 若 $\nu = \nu'$,则敌手 $\mathcal{A}$ 在该游戏中获胜. 敌手 $\mathcal{A}$ 输出 $\nu' = 1$,意味着收到了与挑战密文相对应的封装密钥;否则,输出 $\nu' = 0$,意味着收到了封装密钥空间中的随机值.

敌手 $\mathcal{A}$ 在上述游戏中获胜的优势定义为

$$\text{Adv}_{\text{IB-KEM},\mathcal{A}}^{\text{LR-SID-CCA}}(\kappa,\lambda) = \left| \Pr[\mathcal{A}\ wins] - \frac{1}{2} \right|,$$

其中概率来自于模拟器 $\mathcal{S}$ 和敌手 $\mathcal{A}$ 对随机数的使用.

定义 8(泄露容忍的 SID-CCA 安全性). 若对任意的 PPT 敌手 $\mathcal{A}$,其在上述交互式游戏中获胜的优势 $\text{Adv}_{\text{IB-KEM},\mathcal{A}}^{\text{LR-SID-CCA}}(\kappa,\lambda)$ 是可忽略的,那么相应的 IB-KEM 具有泄露容忍的选择身份 CCA 安全性.

此外,IB-KEM 泄露容忍的 CPA 安全性游戏中,敌手不具备进行解封装询问的能力. 类似地,我们能够得到 HIB-KEM 机制抗泄露 CPA 和 CCA 安全性游戏的描述及定义,交互过程与 IB-KEM 的相关游戏相类似,区别是使用分层结构的身份信息.

特别地,在适应性安全模型中,敌手在挑战阶段根据前期阶段 1 的询问结果适应性的提交挑战身份给模拟器.

4 CCA 安全的抗泄露 IB-KEM 的通用构造

本节将联合 CPA 安全的 IB-KEM、消息验证码和强随机性提取器设计 CCA 安全的抗泄露 IB-KEM 的通用构造. 为方便描述,对 CPA 安全的 IB-KEM 的封装算法进行简单的修改,即算法所使用的随机数来自于算法输入,则封装算法可表示为

$$(C,k) \leftarrow \text{Encap}(id,r),$$

其中,r 表示封装算法运算过程中所使用的随机数.

4.1 通用构造

令 $\Pi' = (\text{Setup}', \text{KeyGen}', \text{Encap}', \text{Decap}')$ 是封装密钥空间为 $\mathcal{K} = \{0,1\}^{l_1}$ 和封装密文空间为 $\mathcal{C}$ 的 CPA 安全的 IB-KEM,MAC = (Tag, Ver) 是对称密钥空间为 $\mathcal{K} = \{0,1\}^{l_1}$、消息空间为 $\mathcal{M}$ 的消息验证码;$\text{Ext}: \{0,1\}^{l_1} \times \{0,1\}^{l_t} \rightarrow \{0,1\}^{l_2}$ 是平均情况的 $(l_1 - \lambda, \varepsilon)$-强随机性提取器,其中 λ 是泄露参数,ε 是可忽略的值,$l_2 < l_1$;$\mathcal{H}: \mathcal{C} \times \mathcal{C} \times \{0,1\}^{l_t} \rightarrow \mathcal{M}$ 是安全的抗碰撞哈希函数.

本文 CCA 安全的抗泄露 IB-KEM Π = (Setup, KeyGen, Enc, Dec) 的通用构造由下述算法组成:

(1) $(mpk, msk) \leftarrow \text{Setup}(1^\kappa)$

输出 $mpk = (mpk', \text{MAC})$ 和 $msk = msk'$,其中 $(mpk', msk') \leftarrow \text{Setup}'(1^\kappa)$.

(2) $sk_{id} \leftarrow \text{KeyGen}(msk, id)$

输出 $sk_{id} = sk'_{id}$,其中 $sk'_{id} \leftarrow \text{KeyGen}'(msk, id)$.

(3) $C \leftarrow \text{Encap}(id, M)$

① 随机选取 $r_1, r_2 \leftarrow_R Z_q^*$,并计算 $(c_1, k_1) \leftarrow \text{Encap}'(id, r_1)$ 和 $(c_2, k_2) \leftarrow \text{Encap}'(id, r_2)$.

② 随机选取 $S \leftarrow_R \{0,1\}^{l_t}$,并计算 $k = \text{Ext}(k_1, S)$ 和 $Tag \leftarrow \text{Tag}(k_2, \mathcal{H}(c_1, c_2, S))$.

③ 输出封装密文 $C = (c_1, c_2, S, Tag)$ 及相应的封装密钥 k.

其中,$\Pi' = (\text{Setup}', \text{KeyGen}', \text{Encap}', \text{Decap}')$ 的 CPA 安全性保证了其输出的封装密钥与封装密钥空间上的任意随机值是不可区分的;也就是说,输出的封装密钥具有足够的随机性,满足强随机性提取器 $\text{Ext}: \{0,1\}^{l_1} \times \{0,1\}^{l_t} \rightarrow \{0,1\}^{l_2}$ 的提取要求.

(4) $M \leftarrow \text{Decap}(sk_{id}, C)$

① 计算 $k_2 \leftarrow \text{Decap}'(sk_{id}, c_2)$.

② 若有 $\text{Ver}(k_2, Tag, \mathcal{H}(c_1, c_2, S)) = 1$ 成立,则计算 $k_1 \leftarrow \text{Decap}'(sk_{id}, c_1)$,并输出相应的封装密钥 $k = \text{Ext}(k_1, S)$;否则输出终止符 $\perp$.

4.2 正确性和安全性

由底层 IB-KEM、消息验证码和强随机性提取器的正确性可知本文通用构造的正确性.

定理 2. 若底层的基础机制 IB-KEM $\Pi' = (\text{Setup}', \text{KeyGen}', \text{Encap}', \text{Decap}')$ 是 CPA 安全的,消息验证码 MAC = (Tag, Ver) 是强不可伪造的,

那么对于泄露参数 $\lambda \leq l_1 - l_2 - \omega(\log\kappa)$，上述实例 Π=(Setup,KeyGen,Encap,Decap)是 CCA 安全的抗泄露 IB-KEM 的通用构造.

定理 2 的证明过程详见附录 A.

特别地，由 Dodis 等人[17]的结论可知，当私钥具有定期更新的能力，并且更新前后密码机制的功能及公开参数未发生改变，那么抗有界泄露攻击的密码机制即可达到抵抗连续泄露攻击的目的. 因此本文的上述通用构造中，一旦底层 CPA 安全的 IB-KEM 具有密钥更新功能，那么上述通用构造就具有连续泄露容忍性.

5　CPA 安全的 IB-KEM 实例

上文的通用构造表明，任意的 CPA 安全的 IB-KEM 结合消息验证码和强随机性提取器即可得到 CCA 安全的 IB-KEM，因此，本节将给出 CPA 安全的 IB-KEM 的具体实例.

5.1　具体构造

本文实例 Π=(Setup,KeyGen,Encap,Decap)具体包含下述 4 个 PPT 算法：

(1) $(mpk,msk) \leftarrow \text{Setup}(1^\kappa)$

① 运行群生成算法生成相应的元组 $(q,G,g,G_T,e(\cdot))$，其中 G 是阶为大素数 q 的乘法循环群，g 是群 G 的生成元，$e: G\times G \rightarrow G_T$ 是高效可计算的双线性映射.

② 随机选取 $\alpha \leftarrow_R Z_q^*$ 和 $u,h \leftarrow_R G$，计算主密钥 $msk=g^\alpha$，并公开系统参数

$$mpk=\{q,G,g,G_T,e(\cdot),u,h,e(g,g)^\alpha\}.$$

(2) $sk_{id} \leftarrow \text{KeyGen}(msk,id)$

① 随机选取 $r \leftarrow_R Z_q^*$，并计算

$$d_1=g^\alpha(u^{id}h)^r \text{ 和 } d_2=g^{-r}.$$

② 输出身份 id 所对应的私钥 $sk_{id}=(d_1,d_2)$.

(3) $(C,k) \leftarrow \text{Encap}(id)$

① 随机选取 $z \leftarrow_R Z_q^*$，并计算

$$c_1=g^z \text{ 和 } c_2=(u^{id}h)^z.$$

② 输出封装密文 $C=(c_1,c_2)$ 及相对应的封装密钥 $k=e(g,g)^{\alpha z}$.

(4) $k \leftarrow \text{Decap}(d_{id},C)$

输出封装密文 $C=(c_1,c_2)$ 所对应的封装密钥

$$k=e(c_1,d_1)e(c_2,d_2).$$

5.2　正确性

由下述等式即可获得本文 IB-KEM 实例 Π=(Setup,KeyGen,Encap,Decap)的正确性.

$$\begin{aligned} &e(c_1,d_1)e(c_2,d_2)\\ &=e(g^z,g^\alpha(u^{id}h)^r)e((u^{id}h)^z,g^{-r})\\ &=e(g,g)^{\alpha z}. \end{aligned}$$

5.3　安全性

下面将基于经典的 DBDH 困难性假设给出上述 IB-KEM 实例的安全性形式化证明. 特别地，本文仅考虑用户私钥的泄露，对主私钥的泄露未考虑.

定理 3.　在选择身份的安全模型下，若存在一个 PPT 敌手 $\mathcal{A}$ 在多项式时间内能以不可忽略的优势 $\text{Adv}^{\text{SID-CPA}}_{\text{IB-KEM},\mathcal{A}}(\kappa)$ 攻破上述 IB-KEM 实例 Π=(Setup,KeyGen,Encap,Decap)的 CPA 安全性，那么我们就能够构造一个敌手 $\mathcal{B}$ 在多项式时间内能以优势 $\text{Adv}^{\text{DBDH}}_{\mathcal{B}}(\kappa)$ 攻破经典的 DBDH 困难性假设，其中

$$\text{Adv}^{\text{DBDH}}_{\mathcal{B}}(\kappa) \geq \text{Adv}^{\text{SID-CPA}}_{\text{IB-KEM},\mathcal{A}}(\kappa).$$

定理 3 的证明详见附录 B.

5.4　用户私钥的更新

基于定期的密钥更新操作即可将有界泄露容忍的密码机制转换为抵抗连续泄露攻击的密码机制. 基于该结论，本节将为 IB-KEM 实例 Π 设计相应的密钥更新算法，定期完成对用户私钥的更新任务.

对于输入的原始用户私钥 $sk_{id}=(d_1,d_2)$，密钥更新算法 $sk'_{id} \leftarrow \text{Update}(sk_{id},id)$ 输出更新后的用户私钥 $sk'_{id}=(d'_1,d'_2)$，具体操作主要包括：

① 随机选取 $r_i \leftarrow_R Z_q^*$，并计算

$$d'_1=d_1(u^{id}h)^{r_i} \text{ 和 } d'_2=d_2g^{-r_i}.$$

② 输出身份 id 更新后的私钥 $sk'_{id}=(d'_1,d'_2)$.

对于任意的更新索引 j，那么第 j 次执行密钥更新算法后的私钥为

$$sk^j_{id}=(d^j_1,d^j_2)=\left(g_2^\alpha(u^{id}h)^{r+\sum_{i=1}^{j}r_i},g^{-\left(r+\sum_{i=1}^{j}r_i\right)}\right).$$

由于 $r_i(i=1,\cdots,j)$ 是从 Z_q^* 中均匀随机选取的，因此对于任意的敌手而言，密钥更新算法执行前后，公开参数和 IB-KEM 的功能是保持不变的，并且 $sk^j_{id}=(d^j_1,d^j_2)$ 与密钥生成算法 KeyGen 输出的原始私钥 $sk_{id}=(d_1,d_2)$ 是不可区分的，也就是说有 $\text{SD}(sk^j_{id},sk_{id}) \leq \text{negl}(\kappa)$ 成立. 特别地，第 j 次执行密钥更新算法输出的更新私钥 sk^j_{id} 相当于密钥生成算法输出的随机数为 $r+\sum_{i=i}^{j}r_i$ 的用户私钥，因此密钥更新算法并未改变 IB-KEM 的性能和安全性.

5.5　分析对比

下面将 Π=(Setup,KeyGen,Encap,Decap)对应的 CCA 安全的抗泄露 IB-KEM 的构造与现有的

《计算机学报》

相关研究工作[34]进行对比.

表 1 中的相关符号的含义分别为:Z 表示 Z_q^* 中元素的长度,G 表示群中元素的长度,l_t 表示强随机性提取器的种子长度,l_n 和 l_κ 分别表示相应的封装密钥长度,l_{tag} 表示消息验证码生成相应标签的长度,A 表示矩阵的加法运算,M 表示矩阵的乘法运算,N 表示群上的指数运算,E 表示强随机性提取运算,T 表示消息验证的码的标签生成运算;此外,m 和 n 是文献[34]中分别表示元素个数和矩阵大小的参数.特别地,相关符号前的系数表示相应运算的次数.

表 1 与现有研究工作的对比

分组	对比项目	文献[34]	本文构造
存储效率	msk 的长度	$(3n+mn^2)Z$	$1G$
	私钥的长度	$(4n+2)G$	$2G$
	封装密钥长度	l_n	l_2
计算效率	密钥生成算法	$1A+1M$	$2N$
	密钥封装算法	$3M+E$	$2N+E+T$
传输效率	mpk 的长度	$(2n+mn^2)G$	$4G$
	封装密文长度	$(2n+1)G+l_t$	$2G+l_t+l_{tag}$

由表 1 可知,相较于文献[34]中的构造,本文 CCA 安全的抗泄露 IB-KEM 实例在存储效率、传输效率和计算效率方面都具有优势.

6 CPA 安全的 HIB-KEM 实例

将本文通用构造的相关结论推广到身份分层的身份基密码结构中,则有 CPA 安全的 HIB-KEM 结合消息验证码和强随机性提取器即可得到 CCA 安全的 HIB-KEM,因此,本节将给出 CPA 安全的 HIB-KEM 的具体实例.

6.1 具体构造

本文 CPA 安全的 HIB-KEM 实例 $\Pi'=(\text{Setup}', \text{KeyGen}', \text{Delegate}', \text{Encap}', \text{Decap}')$ 各算法的具体运算过程如下:

(1) $(mpk, msk) \leftarrow \text{Setup}'(1^\kappa)$

① 以安全参数 κ 作为输入运行群生成算法 $\mathcal{G}(1^\kappa)$,输出相应的公开元组 $(q, g, G, G_T, e(\cdot))$;并且设定 HIB-KEM 的身份最大深度值是 l;

② 随机选取 $a\in Z_q^*$,并计算 $g_1=g^a$;

③ 随机选取 $g_2, h, u_1, u_2, \cdots, u_l \in G$,并公开相应的系统公开参数 mpk,同时秘密保存系统主密钥 $msk=g_2^a$,其中

$$mpk=(q, g, G, G_T, e(\cdot), g_1, g_2, h, u_1, u_2, \cdots, u_l).$$

(2) $sk_{\boldsymbol{id}_k} \leftarrow \text{KeyGen}'(msk, \boldsymbol{id}_k)$

随机选取 $r\in Z_q^*$,输出身份 $\boldsymbol{id}_k=(I_1, I_2, \cdots, I_k)_{k\le l}$ 所对应的私钥

$$sk_{\boldsymbol{id}_k}=(d_1, d_2, \omega_{k+1}, \omega_{k+2}, \cdots, \omega_l)$$
$$=(g_2^a(u_1^{I_1}u_2^{I_2}\cdots u_k^{I_k}h)^r, g^r, u_{k+1}^r, u_{k+2}^r, \cdots, u_l^r).$$

(3) $sk_{\boldsymbol{id}_k} \leftarrow \text{Delegate}'(sk_{id_{k-1}}, \boldsymbol{id}_k)$

对于私钥 $sk_{id_{k-1}}=(d_1', d_2', \omega_{k-1}', \omega_k', \cdots, \omega_l')$ 和身份 $\boldsymbol{id}_k=(I_1, I_2, \cdots, I_k)_{k\le l}$,密钥派生算法具体包含下述过程:

随机选取 $t\in Z_q^*$,输出身份 $\boldsymbol{id}_k=(I_1, I_2, \cdots, I_k)_{k\le l}$ 所对应的私钥

$$sk_{\boldsymbol{id}_k}=(d_1, d_2, \omega_{k+1}, \cdots, \omega_l)$$
$$=(d_1'(u_1^{I_1}u_2^{I_2}\cdots u_k^{I_k}h)^t(\omega_k')^{I_k}, d_2'g^t, \omega_{k+1}'u_{k+1}^t, \cdots, \omega_l'u_l^r),$$

已知

$$sk_{id_{k-1}}=(d_1', d_2', \omega_k', \omega_{k+1}', \cdots, \omega_l')$$
$$=(g_2^a(u_1^{I_1}u_2^{I_2}\cdots u_{k-1}^{I_{k-1}}h)^r, g^r, u_k^r, u_{k+1}^r, \cdots, u_l^r),$$

那么有

$$d_1'(u_1^{I_1}u_2^{I_2}\cdots u_k^{I_k}h)^t(\omega_k')^{I_k}$$
$$=g_2^a(u_1^{I_1}u_2^{I_2}\cdots u_{k-1}^{I_{k-1}}h)^r(u_1^{I_1}u_2^{I_2}\cdots u_k^{I_k}h)^t(u_k^{I_k})^r$$
$$=g_2^a(u_1^{I_1}u_2^{I_2}\cdots u_k^{I_k}h)^{r+t};$$
$$d_2'g^t=g^{r+t};$$
$$\omega_{k+1}'u_{k+1}^t=u_{k+1}^{r+t};$$
$$\vdots$$
$$\omega_l'u_l^t=u_l^{r+t}.$$

因此,密钥派生算法基于第 $k-1$ 层身份 id_{k-1} 的私钥 $sk_{id_{k-1}}$,为第 k 层身份 $\boldsymbol{id}_k$ 为生成了随机数为 $r+t\in Z_q^*$ 的合法私钥 $sk_{\boldsymbol{id}_k}$.

(4) $(C, k) \leftarrow \text{Encap}'(\boldsymbol{id}_k, M)$,其中 $\boldsymbol{id}_k=(I_1, I_2, \cdots, I_k)_{k\le l}$

① 随机选取 $s\in Z_q^*$,并计算

$$c_1=g^s \text{ 和 } c_2=(u_1^{I_1}u_2^{I_2}\cdots u_k^{I_k}h)^{-s}.$$

② 输出封装密文 $C=(c_1, c_2)$ 及相对应的封装密钥 $k=e(g_1, g_2)^s$.

(5) $k \leftarrow \text{Decap}'(sk_{id}, C)$

输出封装密文 $C=(c_1, c_2)$ 所对应的封装密钥

$$k=e(c_1, d_1)e(c_2, d_2).$$

6.2 正确性

上述 HIB-KEM 实例的正确性可由下述等式获得.

$$e(c_1, d_1)e(c_2, d_2)$$
$$=e(g^s, g_2^a(u_1^{I_1}u_2^{I_2}\cdots u_k^{I_k}h)^r)e((u_1^{I_1}u_2^{I_2}\cdots u_k^{I_k}h)^{-s}, g^r)$$
$$=e(g^s, g_2^a)e((u_1^{I_1}u_2^{I_2}\cdots u_k^{I_k}h), g)^{sr}e(g, (u_1^{I_1}u_2^{I_2}\cdots u_k^{I_k}h))^{-sr}$$
$$=e(g_1, g_2)^s.$$

计算机学报

6.3 安全性

本文将基于判定性 BDHE 假设，在选择身份安全模型下对上述 HIB-KEM 构造的 CPA 安全性进行证明.

定理 4. 在选择身份的安全模型下，若存在一个 PPT 敌手 $\mathcal{A}$ 在多项式时间内能以不可忽略的优势 $\mathrm{Adv}^{\text{SID-CPA}}_{\text{HIB-KEM},\mathcal{A}}(\kappa)$ 攻破本文 HIB-KEM 实例 Π' 的 CPA 安全性，那么我们就能够构造一个敌手 $\mathcal{B}$ 在多项式时间内能以优势 $\mathrm{Adv}^{\text{BDHE}}_{\mathcal{B}}(\kappa)$ 攻破经典的 BDHE 困难性假设，其中

$$\mathrm{Adv}^{\text{BDHE}}_{\mathcal{B}}(\kappa)\geq \mathrm{Adv}^{\text{SID-CPA}}_{\text{HIB-KEM},\mathcal{A}}(\kappa).$$

定理 4 的形式化证明详见附录 C.

6.4 用户私钥的更新

与 5.4 节相类似，本节将为 HIB-KEM 实例 Π' 设计相应的密钥更新算法.

对于输入的原始私钥 $sk_{\boldsymbol{id}_k}=(d_1,d_2,\omega_{k+1},\cdots,\omega_l)$，密钥更新算法 $sk'_{\boldsymbol{id}_k}\leftarrow \mathrm{Update}'(sk_{\boldsymbol{id}_k},\boldsymbol{id}_k)$ 输出更新后的用户私钥 $sk'_{\boldsymbol{id}_k}=(d'_1,d'_2,\omega'_{k+1},\cdots,\omega'_l)$，具体操作主要包括：

① 随机选取 $r_i\leftarrow_R Z_q^*$，并计算

$$\begin{aligned}sk'_{\boldsymbol{id}_k}&=(d'_1,d'_2,\omega'_{k+1},\cdots,\omega'_l)\\&=(d_1(u_1^{I_1}u_2^{I_2}\cdots u_k^{I_k}h)^{r_i},d_2g^{r_i},\omega_{k+1}u_{k+1}^{r_i},\cdots,\omega_l u_l^{r_i}).\end{aligned}$$

② 输出身份 $\boldsymbol{id}_k$ 更新后的私钥 $sk'_{\boldsymbol{id}_k}=(d'_1,d'_2,\omega'_{k+1},\cdots,\omega'_l)$.

对于任意的更新索引 j，第 j 次执行密钥更新算法后的私钥为

$$\begin{aligned}sk^j_{\boldsymbol{id}_k}&=(d^j_1,d^j_2,\omega^j_{k+1},\cdots,\omega^j_l)\\&=\left(g_2^{\alpha}(u_1^{I_1}u_2^{I_2}\cdots u_k^{I_k}h)^{r+\sum_{i=1}^{j}r_i},g^{r+\sum_{i=1}^{j}r_i},u_{k+1}^{r+\sum_{i=1}^{j}r_i},\cdots,u_l^{r+\sum_{i=1}^{j}r_i}\right).\end{aligned}$$

由于 $r_i(i=1,\cdots,j)$ 是从 Z_q^* 中均匀随机选取的，因此对于任意敌手而言，密钥更新算法执行前后，公开参数和 HIB-KEM 的功能是保持不变的，并且 $sk^j_{\boldsymbol{id}_k}=(d^j_1,d^j_2,\omega^j_{k+1},\cdots,\omega^j_l)$ 与密钥生成算法输出的原始私钥 $sk_{\boldsymbol{id}_k}=(d_1,d_2,\omega_{k+1},\cdots,\omega_l)$ 是不可区分的，即有 $\mathrm{SD}(sk^j_{\boldsymbol{id}_k},sk_{\boldsymbol{id}_k})\leq \mathrm{negl}(\kappa)$. 特别地，第 j 次执行密钥更新算法输出的更新私钥 $sk^j_{\boldsymbol{id}_k}$ 相当于密钥生成算法输出的随机数为 $r+\sum_{i=i}^{j}r_i$ 的用户私钥，因此密钥更新算法并未改变 HIB-KEM 的性能和安全性.

6.5 性能分析

下面对上述 HIB-KEM 实例对应的 CCA 安全的抗泄露 HIB-KEM 构造的性能进行分析，其中(1)存储效率方面：主私钥的长度是 $1G$，私钥的长度是 $(l-k+2)G$，封装密钥的长度是 l_2；(2)计算效率方面：密钥生成算法和密钥封装算法的计算量分别是 $(l-k+2)N$ 和 $2N+E+T$；(3)传输效率方面：主私钥和封装密文的长度分别为 $(l+3)G$ 和 $2G+l_t+l_{tag}$；其中 l 表示分层身份结构中的最大层数，k 表示用户身份所处的层数，其它符号的含义与表 1 相同. 特别地，本文 IB-KEM 实例的私钥长度随用户身份层数的增加而缩短. 此外，基于上述 CPA 安全的 HIB-KEM 实例和 CCA 安全的 IB-KEM 实例的通用构造，我们能够得出第一个 CCA 安全的 HIB-KEM 的实例.

7 结束语

为满足身份基密钥封装机制的抗泄露性需求，本文提出了联合 CPA 安全的 IB-KEM、消息验证码和强随机性提取器构造了 CCA 安全的抗泄露 IB-KEM 的通用构造，并基于底层基础机制的安全性对通用构造的 CCA 安全性进行了形式化证明；此外，为了进一步展示本文通用构造的实用性，本文设计了 IB-KEM 和 HIB-KEM 的具体实例，分别基于 DBDH 和 BDHE 困难性假设在选择身份安全模型下对上述实例的 CPA 安全性进行了形式化证明. 由于 IB-KEM 是身份基混合加密机制的重要组成部分，那么本文的相关构造能够在云计算等应用环境下结合安全的数据封装机制实现云数据的抗泄露授权和共享；此外，混合加密机制兼顾了公钥加密和对称加密的优势，在现实中具有广泛的应用前景，本文结论为身份基混合加密机制提供了抗泄露的能力，增强了其实用性.

IB-KEM 的构造与 IBE 之间有一定的联系，相应的安全性游戏中敌手都能进行用户私钥的生成询问和解封装(解密)询问；IBE 机制中部分构造技术由于无法生成挑战身份的对应私钥导致无法在 CCA 安全的 IB-KEM 实例的构造中使用. 特别地，本文相应实例的安全性是在选择身份安全模型下证明的，下一阶段，我们将在现有适应性安全 IBE 机制的研究基础上，研究适应性安全的 IB-KEM 和 HIB-KEM 的设计.

参 考 文 献

[1] Shamir A. Identity-based cryptosystems and signature schemes//Proceedings of the Advances in Cryptology-CRYPTO 1984, 4th Annual International Cryptology Conference. Santa Barbara, USA, 1984: 47-53

《计算机学报》

[2] Boneh D, Franklin M K. Identity-based encryption from the Weil pairing//Proceedings of the Advances in Cryptology-CRYPTO 2001, 21st Annual International Cryptology Conference. Santa Barbara, USA, 1984: 213-229

[3] Boneh D, Boyen X. Secure identity based encryption without random oracles//Proceedings of the Advances in Cryptology-CRYPTO 2004, 24th Annual International Cryptology Conference. Santa Barbara, USA, 2004: 443-459

[4] Boneh D, Boyen X. Efficient selective-ID secure identity-based encryption without random oracles//Proceedings of the Advances in Cryptology-EUROCRYPT 2004, International Conference on the Theory and Applications of Cryptographic Techniques. Interlaken, Switzerland, 2004: 223-238

[5] Waters B. Efficient identity-based encryption without random oracles//Proceedings of the Advances in Cryptology-EUROCRYPT 2005, 24th Annual International Conference on the Theory and Applications of Cryptographic Techniques. Aarhus, Denmark, 2005: 114-127

[6] Gentry C. Practical identity-based encryption without random oracles//Proceedings of the Advances in Cryptology-EUROCRYPT 2006, 25th Annual International Conference on the Theory and Applications of Cryptographic Techniques. St. Petersburg, Russia, 2006: 445-464

[7] Boyen X, Waters B. Anonymous hierarchical identity-based encryption (without random oracles)//Proceedings of the Advances in Cryptology-CRYPTO 2006, 26th Annual International Cryptology Conference. Santa Barbara, USA, 2006: 290-307

[8] Abdalla M, Kiltz E, Neven G. Generalized key delegation for hierarchical identity-based encryption//Proceedings of the Computer Security-ESORICS 2007, 12th European Symposium on Research in Computer Security. Dresden, Germany, 2007: 139-154

[9] Lewko A B, Waters B. New techniques for dual system encryption and fully secure HIBE with short ciphertexts//Proceedings of the Theory of Cryptography, 7th Theory of Cryptography Conference (TCC 2010). Zurich, Switzerland, 2010: 455-479

[10] Naor M, Segev G. Public-key cryptosystems resilient to key leakage//Proceedings of the Advances in Cryptology-CRYPTO 2009, 29th Annual International Cryptology Conference. Santa Barbara, USA, 2009: 18-35

[11] Alwen J, Dodis Y, Naor M, et al. Public-key encryption in the bounded-retrieval model//Proceedings of the Advances in Cryptology-EUROCRYPT 2010, 29th Annual International Conference on the Theory and Applications of Cryptographic Techniques. Monaco, 2010: 113-134

[12] Liu Shengli, Weng Jian, Zhao Yunlei. Efficient public key cryptosystem resilient to key leakage chosen ciphertext attacks//Proceedings of the Topics in Cryptology-CT-RSA 2013, the Cryptographers' Track at the RSA Conference 2013. San Francisco, USA, 2013: 84-100

[13] Qin Baodong, Liu Shengli. Leakage-resilient chosen-ciphertext secure public-key encryption from hash proof system and one-time lossy filter//Proceedings of the Advances in Cryptology-ASIACRYPT 2013, 19th International Conference on the Theory and Application of Cryptology and Information Security. Bengaluru, India, 2013: 381-400

[14] Qin Baodong, Liu Shengli. Leakage-flexible CCA-secure public-key encryption: Simple construction and free of pairing //Proceedings of the Public-Key Cryptography-PKC 2014, 17th International Conference on Practice and Theory in Public-Key Cryptography. Buenos Aires, Argentina, 2014: 19-36

[15] Li Sujuan, Zhang Futai, Sun Yinxia, Shen Limin. Efficient leakage-resilient public key encryption from DDH assumption. Cluster Computing, 2013, 16(4): 797-806

[16] Wang Zhi-Wei, Li Dao-Feng, Zhang Wei, et al. CCA secure PKE with auxiliary input. Chinese Journal of Computers, 2016, 39(3): 562-570(in Chinese)
(王志伟, 李道丰, 张伟等. 抗辅助输入 CCA 安全的 PKE 构造. 计算机学报, 2016, 39(3): 562-570)

[17] Dodis Y, Haralambiev K, López-Alt A, Wichs D. Cryptography against continuous memory attacks//Proceedings of the 51st Annual IEEE Symposium on Foundations of Computer Science (FOCS 2010). Las Vegas, USA, 2010: 511-520

[18] Zhang Jie, Chen Jie, Gong Junqing, et al. Leakage-resilient attribute based encryption in prime-order groups via predicate encodings. Designs, Codes and Cryptography, 2018, 86(6): 1339-1366

[19] Zhang Leyou, Zhang Jingxia, Mu Yi. Novel leakage-resilient attribute-based encryption from hash proof system. The Computer Journal, 2017, 60(4): 541-554

[20] Li Jiguo, Guo Yuyan, Yu Qihong, et al. Continuous leakage-resilient certificate-based encryption. Information Sciences, 2016, 355-356: 1-14

[21] Yu Qihong, Li Jiguo, Zhang Yichen, et al. Certificate-based encryption resilient to key leakage. Journal of Systems and Software, 2016, 116: 101-112

[22] Huang Jianye, Huang Qiong, Susilo Willy. Leakage-resilient group signature: Definitions and constructions. Information Sciences, 2020, 509: 119-132

[23] Chow S S M, Dodis Y, Rouselakis Y, Waters B. Practical leakage-resilient identity-based encryption from simple assumptions//Proceedings of the 17th ACM Conference on Computer and Communications Security (CCS 2010). Chicago, USA, 2010: 152-161

[24] Li Jiguo, Teng Meilin, Zhang Yichen, Yu Qihong. A leakage-resilient CCA-secure identity-based encryption scheme. The Computer Journal, 2016, 59(7): 1066-1075

[25] Zhang Yinghui, Yang Menglei, Zheng Dong, et al. Leakage-resilient hierarchical identity-based encryption with recipient anonymity. International Journal of Foundations of Computer Science, 2019, 30(4): 665-681

《计算机学报》

[26] Sun Shifeng, Gu Dawu, Liu Shengli. Efficient chosen ciphertext secure identity-based encryption against key leakage attacks. Security and Communication Networks, 2016, 9(11): 1417-1434

[27] Sun Shifeng, Gu Dawu, Huang Zhengan. Fully secure wicked identity-based encryption against key leakage attacks. The Computer Journal, 2015, 58(10): 2520-2536

[28] Zhou Yanwei, Yang Bo, Hou Hongxia, et al. Continuous leakage-resilient identity-based encryption with tight security. The Computer Journal, 2019, 62(8): 1092-1105

[29] Zhou Yanwei, Yang Bo, Mu Yi. Continuous leakage-resilient identity-based encryption without random oracles. The Computer Journal, 2018, 61(4): 586-600

[30] Zhou Yanwei, Yang Bo, Mu Yi. Continuous leakage-resilient identity-based encryption with leakage amplification. Designs, Codes and Cryptography, 2019, 87(9): 2061-2090

[31] Zhou Yanwei, Yang Bo, Mu Yi. The generic construction of continuous leakage-resilient identity-based cryptosystems. Theoretical Computer Science, 2019, 772: 1-45

[32] Wang Hao, Zheng Zhihua, Yang Bo. New identity-based key-encapsulation mechanism and its applications in cloud computing. International Journal of High Performance Computing and Networking, 2015, 8(2): 124-134

[33] Yang Yang. Efficient identity-based key encapsulation scheme with wildcards for email systems. International Journal of Communication Systems, 2014, 27(1): 171-183

[34] Tomita T, Ogata W, Kurosawa K. CCA-secure leakage-resilient identity-based key-encapsulation from simple (Not q-type) assumptions//Proceedings of the Advances in Information and Computer Security-14th International Workshop on Security (IWSEC 2019). Tokyo, Japan, 2019: 3-22

[35] Canetti R, Halevi S, Katz J. Chosen-ciphertext security from identity-based encryption//Proceedings of the Advances in Cryptology-EUROCRYPT 2004, International Conference on the Theory and Applications of Cryptographic Techniques. Interlaken, Switzerland, 2004: 207-222

附录 A. 定理 2 的证明.

证明. 将通过游戏论证的方式对 IB-KEM 通用构造 $\Pi=(\text{Setup},\text{KeyGen},\text{Encap},\text{Decap})$ 的泄露容忍的 CCA 安全性进行证明，每个游戏由模拟器 $\mathcal{S}$ 和敌手 $\mathcal{A}$ 执行. 令事件 $\mathcal{F}_i$ 表示敌手 $\mathcal{A}$ 在游戏 Game_i 中获胜，即有

$$\Pr[\mathcal{F}_i]=\Pr[\mathcal{A}\ wins\ \text{in}\ \text{Game}_i].$$

换句话讲，事件 $\mathcal{F}_i$ 发生指敌手 $\mathcal{A}$ 在游戏 Game_i 中输出了对挑战封装密钥的正确判断.

特别地，证明过程中与挑战封装密文相关的变量均标记为“$*$”，即挑战身份和挑战封装密文分别是 id^* 和 $C_v^*=(c_1^*,c_2^*,S^*,Tag^*)$. 令事件 $\mathcal{E}_1$ 表示敌手 $\mathcal{A}$ 在解封装询问中提交的解封装密文 $C=(c_1,c_2,S,Tag)$ 满足条件 $C\neq C^*$ 和 $\mathcal{H}(c_1,c_2,S)=\mathcal{H}(c_1^*,c_2^*,S^*)$；令事件 $\mathcal{E}_2$ 表示敌手 $\mathcal{A}$ 在获得挑战封装密文 $C_v^*=(c_1^*,c_2^*,S^*,Tag^*)$ 之后提交了关于二元组 $(id^*,C'=(c_1^*,c_2^*,S^*,Tag'))$ 的解封装询问，其中 Tag' 是关于消息 $\mathcal{H}(c_1^*,c_2^*,S^*)$ 的合法标签，并且 $Tag'\neq Tag^*$.

Game_0. 该游戏是 IB-KEM 原始的泄露容忍 CCA 安全性游戏，其中挑战封装密文 $C_v^*=(c_1^*,c_2^*,S^*,Tag^*)$ 的生成过程如下所述：

① 随机选取 $r_1^*,r_2^*\leftarrow_R Z_q^*$，并计算

$(c_1^*,k_1^*)\leftarrow\text{Encap}'(id^*,r_1^*)$ 和 $(c_2^*,k_2^*)\leftarrow\text{Encap}'(id^*,r_2^*)$.

② 随机选取 $S^*\leftarrow_R\{0,1\}^{l_t}$，并计算

$\hat{k}_1^*=\text{Ext}(k_1^*,S^*)$ 和 $Tag^*\leftarrow\text{Tag}(k_2^*,\mathcal{H}(c_1^*,c_2^*,S^*))$.

③ 输出挑战封装密文 $C_v^*=(c_1^*,c_2^*,S^*,Tag^*)$ 及相应的封装密钥 $k^*=\hat{k}_1^*$.

特别地，该游戏中敌手输出的 k^* 是与挑战封装密文 $C_v^*=(c_1^*,c_2^*,S^*,Tag^*)$ 相对应的封装密钥.

Game_1. 该游戏与 Game_0 相类似，但该游戏在解封装询问阶段增加了新的拒绝规则，即当事件 $\mathcal{E}_1$ 发生，模拟器 $\mathcal{S}$ 拒绝敌手 $\mathcal{A}$ 提出的解封装询问.

在 Game_0 中即使事件 $\mathcal{E}_1$ 发生，模拟器 $\mathcal{S}$ 依然响应敌手 $\mathcal{A}$ 提出的解封装询问；而在 Game_1 中，当事件 $\mathcal{E}_1$ 发生时，模拟器 $\mathcal{S}$ 将拒绝敌手 $\mathcal{A}$ 提出的解封装询问. 因此，当事件 $\mathcal{E}_1$ 不发生时，Game_1 和 Game_0 是不可区分的，则有 $\Pr[\mathcal{F}_1|\bar{\mathcal{E}}_1]=\Pr[\mathcal{F}_0|\bar{\mathcal{E}}_1]$. 根据引理 1 可知

$$|\Pr[\mathcal{F}_1]-\Pr[\mathcal{F}_0]|\leq\Pr[\mathcal{E}_1].$$

事件 $\mathcal{E}_1$ 发生意味着函数 $\mathcal{H}$ 产生了碰撞，由于函数 $\mathcal{H}$ 是安全的抗碰撞哈希函数，那么事件 $\mathcal{E}_1$ 发生的概率是可忽略的. 因此有

$$|\Pr[\mathcal{E}_1]-\Pr[\mathcal{E}_0]|\leq\text{negl}(\kappa).$$

Game_2. 该游戏与 Game_1 相类似，但该游戏在解封装询问阶段增加了新的拒绝规则，即当事件 $\mathcal{E}_2$ 发生时，模拟器 $\mathcal{S}$ 拒绝敌手 $\mathcal{A}$ 提出的解封装询问. 类似地，当事件 $\mathcal{E}_2$ 不发生时，游戏 Game_2 和 Game_1 是不可区分的，则有 $\Pr[\mathcal{F}_2|\bar{\mathcal{E}}_2]=\Pr[\mathcal{F}_1|\bar{\mathcal{E}}_2]$. 根据引理 1 可知

$$|\Pr[\mathcal{F}_2]-\Pr[\mathcal{F}_1]|\leq\Pr[\mathcal{E}_2].$$

断言：$\Pr[\mathcal{E}_2]\leq\text{negl}(\kappa)$.

证明. 假设事件 $\mathcal{E}_2$ 以压倒性的概率发生；也就是说，敌手 $\mathcal{A}$ 在获得挑战封装密文 $C_v^*=(c_1^*,c_2^*,S^*,Tag^*)$ 之后提交了关于二元组 $(id^*,C'=(c_1^*,c_2^*,S^*,Tag'))$ 的解封装询问，其中 Tag' 是关于消息 $\mathcal{H}(c_1^*,c_2^*,S^*)$ 的合法标签，并且 $Tag'\neq Tag^*$.

敌手 $\mathcal{B}$ 与敌手 $\mathcal{A}$ 之间执行 IB-KEM 的泄露容忍的 CCA 安全性游戏，并且作为攻击者对底层消息验证码 $\text{MAC}=(\text{Tag},\text{Ver})$ 的强不可伪造性进行攻击，敌手 $\mathcal{B}$ 能够适应性的询问标签谕言机 $\text{Tag}(k,\cdot)$ 和验证谕言机 $\text{Vrfy}(k,\cdot,\cdot)$.

挑战阶段敌手 $\mathcal{B}$ 收到来自敌手 $\mathcal{A}$ 的挑战消息 M_0,M_1

及挑战身份 id^*，敌手 $\mathcal{B}$ 通过下述运算生成相应的挑战密文 $C_v^*=(c_1^*,c_2^*,Tag^*)$

① 随机选取 $r_1^*,r_2^*\leftarrow_R Z_q^*$，并计算

$(c_1^*,k_1^*)\leftarrow \text{Encap}'(id^*,r_1^*)$ 和 $(c_2^*,k_2^*)\leftarrow \text{Encap}'(id^*,r_2^*)$.

② 发送消息 $\mathcal{H}(c_1^*,c_2^*,S^*)$ 给标签谕言机 Tag$(k,\cdot)$，获得相应的应答 Tag^*. 特别地，消息验证码的挑战者对标签谕言机 Tag$(k,\cdot)$ 和验证谕言机 Ver$(k,\cdot,\cdot)$ 进行了初始化，并且 Tag$(k,\cdot)$ 能以不可忽略的概率 $\frac{1}{|\mathcal{K}|}$ 输出消息 $\mathcal{H}(c_1^*,c_2^*,S^*)$ 的有效标签 Tag^*，其中 $|\mathcal{K}|$ 表示消息验证码对称密钥空间的大小.

③ 输出挑战密文 $C_v^*=(c_1^*,c_2^*,S^*,Tag^*)$ 给敌手 $\mathcal{A}$.

敌手 $\mathcal{A}$ 获得挑战密文之后，提交了关于二元组 $(id^*,C'=(c_1^*,c_2^*,S^*,Tag'))$ 的解封装询问给敌手 $\mathcal{B}$. 然后，敌手 $\mathcal{B}$ 输出 (c_1^*,c_2^*,S^*,Tag') 作为伪造的消息标签发送给挑战者. 由于 Tag' 是关于消息 $\mathcal{H}(c_1^*,c_2^*,S^*)$ 的合法标签，并且 $Tag'\neq Tag^*$，所以敌手 $\mathcal{B}$ 以不可忽略的概率输出一个合法的消息标签对 $(\mathcal{H}(c_1^*,c_2^*,S^*),Tag')$，攻破了底层消息验证码 MAC=(Tag,Ver) 的强不可伪造性，然而上述结论与底层消息验证码 MAC=(Tag,Ver) 的安全性事实相矛盾，因此我们的假设不成立，则有 $\Pr[\mathcal{E}_2]\leq \text{negl}(\kappa)$.

由于 $\Pr[\mathcal{E}_2]\leq \text{negl}(\kappa)$，那么有

$$|\Pr[\mathcal{F}_2]-\Pr[\mathcal{F}_1]|\leq \text{negl}(\kappa).$$

Game$_3$. 该游戏与 Game$_2$ 相类似，除了挑战封装密文的生成阶段，即该游戏使用挑战身份 id^* 所对应的私钥 sk_{id^*} 计算挑战封装密文 $C_v^*=(c_1^*,c_2^*,S^*,Tag^*)$，具体过程描述如下：

① 计算 $sk_{id}^*\leftarrow \text{KeyGen}'(msk,id^*)$.

② 随机选取 $r_1^*,r_2^*\leftarrow_R Z_q^*$，并计算

$(c_1^*,k_1^*)\leftarrow \text{Encap}'(id^*,r_1^*)$ 和 $(c_2^*,k_2^*)\leftarrow \text{Encap}'(id^*,r_2^*)$.

③ 计算

$\bar{k}_1^*\leftarrow \text{Decap}'(sk_{id^*},c_1^*)$ 和 $\bar{k}_2^*\leftarrow \text{Decap}'(sk_{id^*},c_2^*)$.

④ 随机选取 $S^*\leftarrow_R\{0,1\}^{l_t}$，并计算

$\hat{k}_1^*=\text{Ext}(\bar{k}_1^*,S^*)$ 和 $Tag^*\leftarrow \text{Tag}(\bar{k}_2^*,\mathcal{H}(c_1^*,c_2^*,S^*))$.

⑤ 输出挑战封装密文 $C_v^*=(c_1^*,c_2^*,S^*,Tag^*)$ 及相应的封装密钥 $k^*=\hat{k}_1^*$.

由 $\Pi'=(\text{Setup}',\text{KeyGen}',\text{Encap}',\text{Decap}')$ 解封装算法的正确性可知，Game$_3$ 和 Game$_2$ 是不可区分的，因此有

$$|\Pr[\mathcal{F}_3]-\Pr[\mathcal{F}_2]|\leq \text{negl}(\kappa).$$

Game$_4$. 该游戏与 Game$_3$ 相类似，除了挑战封装密文的生成阶段，即该游戏使用从封装密钥空间 $\mathcal{K}$ 中随机选取的封装密钥 $\bar{k}_2^*$ 来计算 $C_v^*=(c_1^*,c_2^*,S^*,Tag^*)$ 中的标签元素 Tag^*，具体过程描述如下：

① 计算 $sk_{id}^*\leftarrow \text{KeyGen}'(msk,id^*)$.

② 随机选取 $r_1^*,r_2^*\leftarrow_R Z_q^*$，并计算

$(c_1^*,k_1^*)\leftarrow \text{Encap}'(id^*,r_1^*)$ 和 $(c_2^*,k_2^*)\leftarrow \text{Encap}'(id^*,r_2^*)$.

③ 计算

$\bar{k}_1^*\leftarrow \text{Decap}'(sk_{id^*},c_1^*)$.

④ 随机选取 $S^*\leftarrow_R\{0,1\}^{l_t}$ 和 $\bar{k}_2^*\leftarrow_R\mathcal{K}$，并计算

$\hat{k}_1^*=\text{Ext}(\bar{k}_1^*,S^*)$ 和 $Tag^*\leftarrow \text{Tag}(\bar{k}_2^*,\mathcal{H}(c_1^*,c_2^*,S^*))$.

⑤ 输出挑战封装密文 $C_v^*=(c_1^*,c_2^*,S^*,Tag^*)$ 及相应的封装密钥 $k^*=\hat{k}_1^*$.

由 $\Pi'=(\text{Setup}',\text{KeyGen}',\text{Encap}',\text{Decap}')$ 的安全性可知，封装密钥与封装密钥空间的任意随机值是不可区分的（Game$_4$ 和 Game$_3$ 中的 $\bar{k}_2^*$ 是不可区分的），则 Game$_4$ 和 Game$_3$ 是不可区分的，因此有

$$|\Pr[\mathcal{F}_4]-\Pr[\mathcal{F}_3]|\leq \text{negl}(\kappa).$$

Game$_5$. 该游戏与 Game$_4$ 相类似，除了挑战封装密文的生成阶段，即该游戏使用从封装密钥空间 $\mathcal{K}$ 中随机选取的封装密钥 $\bar{k}_1^*$ 来代替 $C_v^*=(c_1^*,c_2^*,S^*,Tag^*)$ 中与元素 c_1^* 相对应的密钥，具体过程描述如下：

① 计算 $sk_{id}^*\leftarrow \text{KeyGen}'(msk,id^*)$.

② 随机选取 $r_1^*,r_2^*\leftarrow_R Z_q^*$，并计算

$(c_1^*,k_1^*)\leftarrow \text{Encap}'(id^*,r_1^*)$ 和 $(c_2^*,k_2^*)\leftarrow \text{Encap}'(id^*,r_2^*)$.

③ 随机选取 $S^*\leftarrow_R\{0,1\}^{l_t},\bar{k}_1^*\leftarrow_R\mathcal{K}$ 和 $\bar{k}_2^*\leftarrow_R\mathcal{K}$，并计算

$\hat{k}_1^*=\text{Ext}(\bar{k}_1^*,S^*)$ 和 $Tag^*\leftarrow \text{Tag}(\bar{k}_2^*,\mathcal{H}(c_1^*,c_2^*))$.

④ 输出挑战封装密文 $C_v^*=(c_1^*,c_2^*,S^*,Tag^*)$ 及所对应的封装密钥 $k^*=\hat{k}_1^*$.

特别地，在游戏 Game$_5$ 中，挑战封装密钥 k^* 完全由随机信息生成，即 k^* 是封装密钥空间 $\{0,1\}^{l_\kappa}$ 中的任意随机值.

由 $\Pi'=(\text{Setup}',\text{KeyGen}',\text{Encap}',\text{Decap}')$ 的安全性可知，封装密钥与封装密钥空间的任意随机值是不可区分的（Game$_5$ 和 Game$_4$ 中的 $\bar{k}_1^*$ 是不可区分的），即 Game$_5$ 和 Game$_4$ 是不可区分的，因此有

$$|\Pr[\mathcal{F}_5]-\Pr[\mathcal{F}_4]|\leq \text{negl}(\kappa).$$

由于在 Game$_0$ 中，k^* 是与封装密文相对应的封密钥；而在 Game$_5$ 中，k^* 是封装密钥空间上的随机值，因此有

$$\text{Adv}_{\text{IB-KEM},\mathcal{A}}^{\text{LR-CCA}}(\kappa,\lambda)=|\Pr[\mathcal{F}_5]-\Pr[\mathcal{F}_0]|.$$

由于 Game$_5$ 和 Game$_0$ 是不可区分的，则有 $|\Pr[\mathcal{F}_5]-\Pr[\mathcal{F}_0]|\leq \text{negl}(\kappa)$ 成立，那么

$$\text{Adv}_{\text{IB-KEM},\mathcal{A}}^{\text{LR-CCA}}(\kappa,\lambda)\leq \text{negl}(\kappa).$$

由于 $\text{Ext}:\{0,1\}^{l_1}\times\{0,1\}^{l_t}\rightarrow\{0,1\}^{l_2}$ 是平均情况的 $(l_1-\lambda,\varepsilon)$-强随机性提取器，由其安全性可知

$$\lambda\leq l_1-l_2-\omega(\log\kappa).$$

综上所述，若底层的 Π' 是 CPA 安全的 IB-KEM，MAC 是强不可伪造的消息验证码，且 Ext 是平均情况的 $(l_1-\lambda,\varepsilon)$-强随机性提取器，那么对于任意的泄露参数 $\lambda\leq l_1-l_2-\omega(\log\kappa)$，上述 IB-KEM 的通用构造 Π 具有抗泄露的 CCA 安全性. 证毕

《计算机学报》

附录 B.　定理 3 的证明.

对于任意未知的随机数 $a,b,c\in Z_q^*$，若 $T=e(g,g)^{abc}$，则称元组(g,g^a,g^b,g^c,T)是 DBDH 元组，否则称其为非 DBDH 元组.

证明.　敌手$\mathcal{B}$与敌手$\mathcal{A}$开始执行选择身份的 CPA 安全性游戏之前，敌手$\mathcal{B}$从 DBDH 挑战者处获得一个 DBDH 挑战元组(g,g^a,g^b,g^c,T)及相应的公开元组$(q,G,g,G_T,e(\cdot))$，其中 $a,b,c\in Z_q^*$，$T=e(g,g)^{abc}$或 $T\leftarrow_R G_T$. 敌手$\mathcal{B}$的目标是当 $T=e(g,g)^{abc}$时输出 1；否则输出 0. 根据选择身份安全模型的要求，在游戏开始之前，敌手$\mathcal{A}$将选定的挑战身份 id^* 发送给敌手$\mathcal{B}$. 敌手$\mathcal{A}$与敌手$\mathcal{B}$间的消息交互过程如下所述：

(1)初始化. 初始化阶段敌手$\mathcal{B}$执行下述操作：

① 令 $u=g^a$，随机选取 $\tilde{h}\leftarrow_R Z_q^*$，计算 $h=(g^a)^{-id^*}g^{\tilde{h}}$.

② 计算 $e(g,g)^{\alpha}=e(g^a,g^b)$. 特别地，通过上述运算敌手$\mathcal{B}$隐含的设置了 $\alpha=ab$.

③ 发送公开参数

$$mpk=\{q,G,g,G_T,e(\cdot),u,h,e(g,g)^{\alpha}\}$$

给敌手$\mathcal{A}$.

我们注意到，a 和 b 由 DBDH 挑战者从 Z_q^* 中均匀随机选取. 因此，对于敌手$\mathcal{A}$而言，mpk 中的所有公开参数都是均匀随机的，即模拟游戏与真实环境中的游戏是不可区分的.

(2)阶段 1. 该阶段敌手$\mathcal{A}$适应性地进行多项式时间次的密钥生成询问.

敌手$\mathcal{A}$能够适应性的对身份空间$\mathcal{ID}$的任意身份 $id\in\mathcal{ID}$ $(id\neq id^*)$进行密钥生成询问，敌手$\mathcal{B}$随机选取 $\tilde{r}\leftarrow_R Z_q^*$，输出身份 id 相对应的私钥

$$sk_{id}=(d_1,d_2)=\left((g^b)^{\frac{-\tilde{h}}{id-id^*}}(u^{id}h)^{\tilde{r}},g^{-\tilde{r}}(g^b)^{\frac{1}{id-id^*}}\right).$$

对于任意的随机数 $\tilde{r}\leftarrow_R Z_q^*$，存在随机值 $r\in Z_q^*$，满足 $r=\tilde{r}-\frac{b}{id-id^*}$($r$ 的随机性由 $\tilde{r}$ 保证)，因此有

$$\begin{aligned}(g^b)^{\frac{-\tilde{h}}{id-id^*}}(u^{id}h)^{\tilde{r}}&=(g^b)^{\frac{-\tilde{h}}{id-id^*}}(u^{id}h)^{r+\frac{b}{id-id^*}}\\&=(g^b)^{\frac{-\tilde{h}}{id-id^*}}(u^{id}h)^{\frac{b}{id-id^*}}(u^{id}h)^{r}\\&=(g^b)^{\frac{-\tilde{h}}{id-id^*}}(g^{a(id-id^*)}g^{\tilde{h}})^{\frac{b}{id-id^*}}(u^{id}h)^{r}\\&=g^{ab}(u^{id}h)^{r};\end{aligned}$$

$$g^{-\tilde{r}}(g^b)^{\frac{1}{id-id^*}}=g^{-\left(\tilde{r}-\frac{b}{id-id^*}\right)}=g^{-r}.$$

因此，敌手$\mathcal{B}$为身份 id 生成了随机数为 r 的对应私钥 $sk_{id}=(d_1,d_2)$.

(3)挑战. 敌手$\mathcal{B}$计算

$$c_1=g^c\text{ 和 }c_2=(g^c)^{\tilde{h}},$$

并输出挑战封装密文 $C^*=(c_1,c_2)$及相应的封装密钥 $k^*=T$ 给敌手$\mathcal{A}$，其中

$$(u^{id^*}h)^c=(g^{aid^*}(g^a)^{-id^*}g^{\tilde{h}})^c=(g^c)^{\tilde{h}}.$$

(4)阶段 2. 与阶段 1 相类似，敌手$\mathcal{A}$能够适应性的对任意身份 $id\in\mathcal{ID}$ 进行密钥生成询问(除了挑战身份 id^*)，敌手$\mathcal{B}$按与阶段 1 相同的方式返回相应的应答 sk_{id}.

(5)输出. 敌手$\mathcal{A}$输出对封装密钥 k^* 的判断 ω. 若 $\omega=1$，则敌手$\mathcal{B}$输出 1，意味着挑战元组是 DBDH 元组；否则输出 0，即挑战元组是非 DBDH 元组.

当 $T=e(g,g)^{abc}$时，由于 $\alpha=ab$，则有

$$k^*=T=e(g,g)^{abc}=e(g,g)^{\alpha c},$$

则 k^* 是挑战封装密文 $C^*=(c_1,c_2)$所对应的有效封装密钥；否则，$T\leftarrow_R G_T$，可以表示为 $T=e(g,g)^{abc'}$，其中 $c'\leftarrow_R Z_q^*$ 且 $c'\neq c$，那么有 $k^*=T=e(g,g)^{\alpha c'}$，则 k^* 是封装密钥空间上的任意随机值.

综上所述，在选择身份的安全模型中，如果敌手$\mathcal{A}$能以不可忽略的优势 $\mathrm{Adv}_{\text{IB-KEM},\mathcal{A}}^{\text{SID-CPA}}(\kappa)$攻破本文 IB-KEM 实例的 CPA 安全性，且敌手$\mathcal{B}$将敌手$\mathcal{A}$以子程序的形式运行，那么敌手$\mathcal{B}$在能以显而易见优势

$$\mathrm{Adv}_{\mathcal{B}}^{\text{DBDH}}(\kappa)\geq\mathrm{Adv}_{\text{IB-KEM},\mathcal{A}}^{\text{SID-CPA}}(\kappa)$$

攻破经典的 DBDH 困难性假设.　　证毕.

附录 C.　定理 4 的证明.

对于任意未知的随机数$\alpha,c\in Z_q^*$，令 $x=g^c$ 和 $y_i=g^{\alpha^i}$ $(i=1,\cdots,\mu-1,\mu+1,\cdots,2\mu)$，那么当 $T=e(x,y_{\mu})$，则称元组

$$(g,x,y_1,\cdots y_{\mu-1},y_{\mu+1},\cdots,y_{2\mu},T)$$

是 BDHE 元组，否则 $T\leftarrow_R G_T$，称其为非 BDHE 元组.

证明.　敌手$\mathcal{B}$在与敌手$\mathcal{A}$进行 HIB-KEM 的选择身份的 CPA 安全性游戏之前，首先收到来自判定性 BDHE 假设挑战者的挑战元组

$$(g,x,y_1,y_2,\cdots,y_l,y_{l+2},\cdots,y_{2l+2},T),$$

其中 $x=g^c$，$y_i=g^{(\alpha^i)}$，$T=e(g,x)^{(\alpha^{l+1})}$或 $T\leftarrow_R G_T$. 敌手$\mathcal{B}$的目标是当 $T=e(g,x)^{(\alpha^{l+1})}$时输出 1，否则输出 0. 此外，需要说明的是挑战元组缺少的是第 $l+1$ 项，即 y_{l+1} 是未知的. 特别地，由于 $y_i=g^{(\alpha^i)}$，那么

$$y_i^{\alpha^j}=(g^{(\alpha^i)})^{\alpha^j}=g^{\alpha^i\alpha^j}=g^{\alpha^{i+j}}=y_{i+j}.$$

敌手$\mathcal{A}$在敌手$\mathcal{B}$进行系统初始化之前，发送选定的挑战身份 $\boldsymbol{id}_k^*=(I_1^*,I_2^*,\cdots,I_k^*)_{k\leq l}$给算法$\mathcal{S}$. 若 $k<l$，则敌手$\mathcal{B}$将对身份 $\boldsymbol{id}_k^*$ 进行扩充，对其补充 $l-k$ 个 0，使得挑战身份 $\boldsymbol{id}_k^*=(I_1^*,I_2^*,\cdots,I_k^*,\underbrace{0,\cdots,0}_{l-k})$是一个长度为 l 的向量. 敌手$\mathcal{A}$与$\mathcal{B}$间的消息交互过程具体叙述如下：

(1)初始化. 该阶段敌手$\mathcal{B}$主要进行下述操作

① 随机选取 $r\in Z_q^*$，并计算

$$g_1=y_1=g^{\alpha}\text{ 和 }g_2=y_l\cdot g^r=g^{r+\alpha^l}.$$

② 随机选取 $r_1,r_2,\cdots,r_l\in Z_q^*$，对于 $i=1,2,\cdots,l$，计算

$u_i=\frac{g^{r_i}}{y_{l-i+1}}$.

③ 随机选取 $\eta\in Z_q^*$,并计算 $h=g^{\eta}\prod_{i=1}^{l}y_{l-i+1}^{I_i^*}$.

④ 发送公开参数 $mpk=\{g,g_1,g_2,u_1,u_2,\cdots,u_l,h\}$ 给敌手 $\mathcal{A}$.

特别地,通过上述计算敌手 $\mathcal{B}$ 隐含的设置系统主私钥为 $g_2^{\alpha}=g^{\alpha(\alpha^{l}+r)}=y_{l+1}y_1^{r}$,由于 y_{l+1} 是未知的,因此敌手 $\mathcal{B}$ 并不掌握主私钥.此外,由于 α 是由 BDHE 挑战者从 Z_q^* 中随机选取的,那么对于敌手 $\mathcal{A}$ 而言,mpk 中的所有公开参数都是均匀随机的,即模拟游戏与真实环境中的游戏是不可区分的.

(2)阶段 1.该阶段敌手 $\mathcal{A}$ 适应性地进行多项式时间次的密钥生成询问.

敌手 $\mathcal{A}$ 能够适应性的对身份空间 $\mathcal{ID}$ 的任意身份 $id=(I_1,I_2,\cdots,I_u)\in(Z_q^*)^{u}$(其中 $u\leq l$)进行密钥生成询问,且要求询问身份 id 不能跟挑战身份 id^* 相同,并且不能是挑战身份 id^* 的前缀.也就是说,存在 $t\in\{1,2,\cdots,u\}(t\leq l)$ 满足 $I_t\neq I_t^*$.为了应答 $id_u=(I_1,I_2,\cdots,I_t,\cdots,I_u)$ 的私钥 sk_{id_u},首先生成身份 $id_t=(I_1,I_2,\cdots,I_t)$ 所对应的私钥 sk_{id_t},然后通过多次调用密钥派生算法生成身份 id_u 对应的私钥 sk_{id_u}.

敌手 $\mathcal{B}$ 随机选取 $\tilde{\gamma}\leftarrow_R Z_q^*$,并输出身份 $id_t=(I_1,I_2,\cdots,I_t)$ 相对应的私钥

$$sk_{id_t}=\left(y_1^{r}\left(y_t^{\frac{\eta-\sum_{i=1}^{t}I_ir_i}{I_t-I_t^*}}\prod_{i=1}^{t-1}y_{l-i+t+1}^{\frac{I_i^*-I_i}{I_t-I_t^*}}\prod_{i=t+1}^{l}y_{l-i+t+1}^{\frac{I_i^*}{I_t-I_t^*}}\right)(u_1^{I_1}u_2^{I_2}\cdots u_t^{I_t}h)^{\tilde{\gamma}},\right.$$

$$\left.y_t^{\frac{1}{I_t-I_t^*}}g^{\tilde{\gamma}},\frac{y_t^{\frac{r_{t+1}}{I_t-I_t^*}}}{y_l^{\frac{1}{I_t-I_t^*}}}u_{t+1}^{\tilde{\gamma}},\cdots,\frac{y_t^{\frac{r_l}{I_t-I_t^*}}}{y_{t+1}^{\frac{1}{I_t-I_t^*}}}u_l^{\tilde{\gamma}}\right).$$

已知

$$u_1^{I_1}u_2^{I_2}\cdots u_t^{I_t}h=\prod_{i=1}^{t}g^{I_ir_i}y_{l-i+1}^{-I_i}g^{\eta}\prod_{i=1}^{l}y_{l-i+1}^{I_i^*}$$

$$=g^{\eta+\sum_{i=1}^{t}I_ir_i}\cdot\prod_{i=1}^{t-1}y_{l-i+1}^{I_i^*-I_i}\cdot y_{l-t+1}^{I_t^*-I_t}\cdot\prod_{i=t+1}^{l}y_{l-i+1}^{I_i^*}.$$

对于任意的随机数 $\tilde{\gamma}\leftarrow_R Z_q^*$,存在随机数 $\gamma\in Z_q^*$,满足 $\gamma=\frac{\alpha^{t}}{I_t-I_t^*}+\tilde{\gamma}\in Z_q^*$($\gamma$ 的随机性由 $\tilde{\gamma}$ 保证),则有

$$y_1^{r}\left(y_t^{\frac{\eta+\sum_{i=1}^{t}I_ir_i}{I_t-I_t^*}}\prod_{i=1}^{t-1}y_{l-i+t+1}^{\frac{I_i^*-I_i}{I_t-I_t^*}}\prod_{i=t+1}^{l}y_{l-i+t+1}^{\frac{I_i^*}{I_t-I_t^*}}\right)(u_1^{I_1}u_2^{I_2}\cdots u_t^{I_t}h)^{\tilde{\gamma}}$$

$$=y_{l+1}y_1^{r}\left(y_t^{\frac{\eta+\sum_{i=1}^{t}I_ir_i}{I_t-I_t^*}}\prod_{i=1}^{t-1}y_{l-i+t+1}^{\frac{I_i^*-I_i}{I_t-I_t^*}}y_{l+1}^{-1}\prod_{i=t+1}^{l}y_{l-i+t+1}^{\frac{I_i^*}{I_t-I_t^*}}\right)(u_1^{I_1}u_2^{I_2}\cdots u_t^{I_t}h)^{\tilde{\gamma}}$$

$$=g_2^{\alpha}\left(g^{\eta+\sum_{i=1}^{t}I_ir_i}\prod_{i=1}^{t-1}y_{l-i+1}^{I_i^*-I_i}y_{l-t+1}^{I_t^*-I_t}\prod_{i=t+1}^{l}y_{l-i+1}^{I_i^*}\right)^{\frac{\alpha^{t}}{I_t-I_t^*}}(u_1^{I_1}u_2^{I_2}\cdots u_t^{I_t}h)^{\tilde{\gamma}}$$

$$=g_2^{\alpha}(u_1^{I_1}u_2^{I_2}\cdots u_t^{I_t}h)^{\frac{\alpha^{t}}{I_t-I_t^*}}(u_1^{I_1}u_2^{I_2}\cdots u_t^{I_t}h)^{\tilde{\gamma}}$$

$$=g_2^{\alpha}(u_1^{I_1}u_2^{I_2}\cdots u_t^{I_t}h)^{\tilde{\gamma}+\frac{\alpha^{t}}{I_t-I_t^*}}=g_2^{\alpha}(u_1^{I_1}u_2^{I_2}\cdots u_t^{I_t}h)^{\gamma};$$

$$y_t^{\frac{1}{I_t-I_t^*}}g^{\tilde{\gamma}}=g^{\frac{\alpha^{t}}{I_t-I_t^*}}g^{\tilde{\gamma}}=g^{\tilde{\gamma}+\frac{\alpha^{t}}{I_t-I_t^*}}=g^{\gamma};$$

$$\frac{y_t^{\frac{r_{t+1}}{I_t-I_t^*}}}{y_l^{\frac{1}{I_t-I_t^*}}}u_{t+1}^{\tilde{\gamma}}=\left(\frac{g^{r_{t+1}}}{y_{l-t}}\right)^{\frac{\alpha^{t}}{I_t-I_t^*}}u_{t+1}^{\tilde{\gamma}}=u_{t+1}^{\tilde{\gamma}+\frac{\alpha^{t}}{I_t-I_t^*}}=u_{t+1}^{\gamma};$$

$$\vdots$$

$$\frac{y_t^{\frac{r_l}{I_t-I_t^*}}}{y_{t+1}^{\frac{1}{I_t-I_t^*}}}u_l^{\tilde{\gamma}}=\left(\frac{g^{r_l}}{y_1}\right)^{\frac{\alpha^{t}}{I_t-I_t^*}}u_{t+1}^{\tilde{\gamma}}=u_l^{\tilde{\gamma}+\frac{\alpha^{t}}{I_t-I_t^*}}=u_l^{\gamma}.$$

由于 γ 是 Z_q^* 上的随机数,则敌手 $\mathcal{B}$ 输出了身份 $id_t=(I_1,I_2,\cdots,I_t)$ 对应的有效私钥

$$sk_{id_t}=(g_2^{\alpha}(u_1^{I_1}u_2^{I_2}\cdots u_t^{I_t}h)^{\gamma},g^{\gamma},u_{t+1}^{\gamma},u_{t+2}^{\gamma},\cdots,u_l^{\gamma}).$$

(3)挑战.敌手 $\mathcal{B}$ 计算

$$c_1=x\text{ 和 }c_2=(x)^{-\left(\eta+\sum_{i=1}^{t}I_i^*r_i\right)},$$

并输出挑战封装密文 $C^*=(c_1,c_2)$ 及相应的封装密钥 $k^*=T\cdot e(y_1,x^{r})$ 给敌手 $\mathcal{A}$,其中

$$c_2=(x)^{-\left(\eta+\sum_{i=1}^{t}I_i^*r_i\right)}=(g^{c})^{-\left(\eta+\sum_{i=1}^{t}I_i^*r_i\right)}$$

$$=\left(\prod_{i=1}^{l}\left(\frac{g^{r_i}}{y_{l-i+1}}\right)^{I_i^*}g^{\eta}\prod_{i=1}^{l}y_{l-i+1}^{I_i^*}\right)^{-c}$$

$$=(u_1^{I_1^*}u_2^{I_2^*}\cdots u_l^{I_l^*}h)^{-c}.$$

(4)阶段 2.与阶段 1 相类似,敌手 $\mathcal{A}$ 能够适应性的对任意身份 id_k 进行密钥生成询问(除了挑战身份 $\boldsymbol{id}_k^*$),敌手 $\mathcal{B}$ 按与阶段 1 相类似的方法返回相应的 sk_{id_u}.

(5)输出.敌手 $\mathcal{A}$ 输出对封装密钥 k^* 的判断 ω.若 $\omega=1$,则敌手 $\mathcal{B}$ 输出 1,意味着挑战元组是 BDHE 元组;否则输出 0,即挑战元组是非 BDHE 元组.

若 $T=e(g,x)^{\alpha^{l+1}}$,则有

$$k=e(g,x)^{\alpha^{l+1}}\cdot e(y_1,x^{r})$$

$$=(e(y_1,y_l)\cdot e(y_1,g^{r}))^{c}$$

$$=e(y_1,y_lg^{r})^{c}=e(g_1,g_2)^{c}.$$

因此,当 $T=e(g,x)^{\alpha^{l+1}}$ 时,k^* 是挑战封装密文 $C^*=(c_1,c_2)$ 所对应的有效封装密钥;否则 $T\leftarrow_R G_T$ 时,k^* 是封装密钥空间上的一个随机值.

综上所述,在选择身份的安全模型下,若敌手 $\mathcal{A}$ 能以不可忽略的优势 $\mathrm{Adv}_{\text{HIB-KEM},\mathcal{A}}^{\text{SID-CPA}}(\kappa)$ 攻破本文 HIB-KEM 实例的 CPA 安全性,那么能够构造一个敌手 $\mathcal{B}$ 将敌手 $\mathcal{A}$ 以子程序的形式运行,且能以显而易见优势

$$\mathrm{Adv}_{\mathcal{B}}^{\text{BDHE}}(\kappa)\geq Adv_{\text{HIB-KEM},\mathcal{A}}^{\text{SID-CPA}}(\kappa)$$

攻破 BDHE 困难性假设. 证毕.

计算机学报

ZHOU Yan-Wei, Ph. D., senior engineer, M. S. supervisor. His research interests include cryptography and anonymous communication, etc.

YANG Bo, Ph. D., professor, Ph. D. supervisor. His research interests include information security and cryptography, etc.

HU Bing-Jie, M. S. candidate. Her research interests include information security and cryptography, etc.

XIA Zhe, Ph. D., associate professor, M. S. supervisor. His research interests include information security and cryptography, etc.

ZHANG Ming-Wu, Ph. D., professor, Ph. D. supervisor. His current research interests include information security and cryptography, etc.

Background

In the traditional security model, it is assumed that only legitimate participants possess the internal secret states (e. g., the user's private key), and these states are completely inaccessible to the adversary. However, in many real-world applications, these states could be leaked through various leakage attacks, such as side-channel attacks, cold boot attacks, etc. Therefore, if an adversary obtains some information of the internal secret states, the cryptographic schemes may fail to achieve their claimed security. In other words, the traditional cryptographic assumptions are insufficient in the leakage setting. To solve this problem, leakage-resilient cryptography has been advocated to maintain the security properties for real world applications, and several concrete constructions were proposed to capture the leakage-resilience requirement, such as leakage-resilient public-key encryption, leakage resilient identity-based encryption, leakage-resilient authenticated key exchange, leakage resilient certificate-based encryption, etc.

Since the hybrid encryption technology has the advantages of both symmetric and asymmetric encryption, the research on identity-based key-encapsulation mechanism (IB-KEM) has received extensive attention in recent years. To obtain the leakage resilience, a leakage-resilient IB-KEM with chosen ciphertext attacks security was created. However, the previous scheme has shortcomings in computing, transmission and storage. To address the above shortcomings, a generic construction of CCA secure IB-KEM with leakage resilience is proposed in this paper, and the formal proof can be obtained from the underlying chosen-plaintext attacks (CPA) secure IB-KEM. In additional, to further show the practicability of the above generic construction, two instances of IB-KEM and hierarchical identity-based key-encapsulation mechanism are proposed, the corresponding CPA security is proved based on the decisional bilinear Diffie-Hellman (DBDH) and bilinear Diffie-Hellman exponent (BDHE) assumptions, respectively. Finally, in order to achieve the goal of resisting continuous leakage attacks, key update algorithms of each instance are also researched in this paper. Analysis and comparison show that our construction of leakage-resilient IB-KEM with CCA security has certain advantages in computing, transmission and storage.

This work was supported in part by the National Key R&D Program of China (2017YFB0802000), in part by the National Natural Science Foundation of China (U2001205, 61802242, 61772326, 61802241), and in part by the National Crypto-graphy Development Foundation During the 13th Five-Year Plan Period (MMJJ20180217).

《计算机学报》