计算机研究与发展
Journal of Computer Research and Development
DOI:10.7544/issn1000-1239.2021.20210596
58(10): 2222-2237, 2021

标准模型下的灵活细粒度授权密文一致性检测方案

邓翔天　　钱海峰
(华东师范大学软件工程学院　上海　200062)
(51194506009@stu.ecnu.edu.cn)

Flexible Fine-Grained Authorization Public Key Encryption with Equality Test Under Standard Model

Deng Xiangtian and Qian Haifeng
(*Software Engineering Institute*, *East China Normal University*, *Shanghai* 200062)

Abstract　Public key encryption with equality test(PKEET) is a public key encryption scheme which allows testers to perform equality tests on ciphertexts without holding corresponding private keys, that is, detecting whether the plaintexts decrypted from given cyphertext are equal. The fine-grained authorization PKEET (FG-PKEET) and PKEET with flexible authorization (PKEET-FA) schemes mentioned in previous works improve the functionality of PKEET in aspect of authorization granularity: FG-PKEET scheme allows one user to interact with another user to generate a token specifically for detecting the equality of all these two users' ciphertexts, while PKEET-FA scheme extends the type of authorization object of the token from user level to ciphertext level, permitting one user to authorize on a specific ciphertext. Both solutions have their own application scenarios and do not include each other in terms of functionality. Thus we propose flexible fine-grained authorization PKEET scheme. Our scheme obtains security properties related to adaptive ciphertext attack and fine-grained authorization. Our solution combines both fine-grained authorization and flexible fine-grained features, allowing two users to authorize respectively on one specified ciphertext or all his ciphertexts. Meanwhile, comparing to existing PKEET schemes with fine-grained authorization or flexible authorization features which rely on random oracle model, our scheme's security properties are proved under standard model.

Key words　public key encryption; equality test; standard model; fine-grained authorization; adaptive chosen ciphertext attack

摘　要　密文一致性检测公钥加密方案是一种检测者能够在无需解密密文的情况下检测一对密文的一致性,即该对密文解密所得明文是否一致的公钥加密方案.已有工作中提及的细粒度授权方案和灵活授权方案在授权粒度方面对密文一致性检测公钥加密方案的功能性进行了改进:细粒度授权方案允许2名用户生成专用于检测这2名用户的所有密文的一致性的令牌.灵活授权方案则将令牌的授权客体从

收稿日期:2021-06-10;**修回日期**:2021-07-29
基金项目:国家自然科学基金委员会与以色列科学基金会合作研究项目(61961146004);上海市教育委员会科研创新项目(2021-01-07-00-08-E00101)
This work was supported by the NSFC-ISF Joint Scientific Research Program (61961146004) and Innovation Program of Shanghai Municipal Education Commission (2021-01-07-00-08-E00101).
通信作者:钱海峰(hfqian@cs.ecnu.edu.cn)

用户级别拓展至指定密文级别.2 种方案拥有各自的应用场景且在功能性方面互不包含.因此提出灵活细粒度授权密文一致性检测公钥加密方案.与已有方案相比,该方案在计算效率和参数大小方面相当,并具备适应性选择密文攻击安全性、细粒度授权安全性.该方案兼具细粒度授权特性与灵活细粒度特性.同时,对比依赖预言机模型的已有相关方案,其安全性证明基于标准模型之上.

关键词 公钥加密;密文一致性检测;标准模型;细粒度授权;适应性选择密文攻击

中图法分类号 TP309

密文一致性检测公钥加密方案(public key encryption with equality test, PKEET)最初由 Yang 等人提出[1],其允许用户在不持有对应私钥的情况下检测一组密文解密所得明文的一致性,受检测的密文对加密时所用的公钥不必相同.

然而,Yang 等人提出的 PKEET 方案允许任意用户对密文进行一致性检测.为了避免这一特性带来的安全风险,Tang[2]提出了全有或全无密文一致性检测方案(all-or-nothing PKEET, AoN-PKEET,在该方案中,私钥持有者通过选择性地发放由私钥生成的令牌,自主授权"代理"执行密文一致性检测操作.令牌机制在隐私保护方面起到关键作用,因而被大量相关工作借鉴沿用[2-7]①.

后续工作中,细粒度授权密文一致性检测公钥加密方案(fine-grained PKEET, FG-PKEET)[3]允许一对用户通过交互生成令牌并将其发送至代理.该令牌只可被用于检测 2 个用户公钥加密所得任意密文的一致性.

由 Ma 等人[4]提出的灵活授权密文一致性检测公钥加密方案(PKEET with flexible authorization, PKEET-FA)则将授权客体从用户层次扩展至密文层次:其在 PKEET 方案的基础上添加多类令牌,分别用于检测 2 个指定密文的一致性,即密文-密文级别令牌,或是检测一指定密文和另一指定用户的任意密文的一致性,即密文-用户级别令牌.

到目前为止,相关工作中提及的 FG-PKEET, PKEET-FA 方案在功能性方面互不包含,并且拥有各自适用的应用场景.FG-PKEET 方案的授权客体类型局限于用户类型.同时,以具体的 PKEET-FA 方案[4]为例,尽管该方案中,密文-密文级别、密文-用户级别的令牌的授权客体可以为指定某一密文,但是该方案未能实现 FG-PKEET 方案中的令牌功能,即能够检测指定 2 位特定用户的任意密文一致性的用户-用户级别令牌.此外,目前提出的 FG-PKEET,PKEET-FA 方案的安全性证明均建立于随机预言机模型之上,需要在安全证明中设立随机预言机扮演方案中的 Hash 函数.

针对现状,我们的 FG-PKEET 方案具备 2 个创新点:

1) 我们所提出的 FG-PKEET 方案在功能性上兼顾了 FG-PKEET 方案与 PKEET-FA 方案的特性,拥有更细的令牌授权粒度,并且具备对应的安全性质.

2) 在实际应用场景中,使用具体的 Hash 函数来代替随机预言机存在安全隐患[8].我们所提出方案的安全性则建立于标准模型之上,在安全性方面更加可靠.

1 预备知识

本节我们将对文中所用符号与密码学原语的定义进行简要解释;汇总并介绍相关工作成果;对本方案安全性质所依赖的密码学假设进行介绍.

1.1 符号定义与密码学原语介绍

1) 符号定义

对于一个有限集合 S,我们使用公式 $s\leftarrow_r S$ 来指代从有限集合 S 中均匀随机采样得一个随机元素 S 这一过程.

我们声称函数 f 关于 λ 是可忽略的,当且仅当不等式 $f(\lambda)\leqslant 1/p(\lambda)$ 对于所有的多项式 $p(\cdot)$ 和足够大的 λ 成立.

2) Hash 函数

在我们所提出的方案中使用的 Hash 函数,应具备 2 个性质:

① 单向性.函数 $H:X\rightarrow Y$ 满足单向性,当且仅当其能够在多项式时间内执行完毕,且多项式时间敌手 $\mathcal{A}$(PPT Adversary)成功还原函数原像的几率是可忽略的,即:

$$y\leftarrow_r Y;Pr[H(x)=y\,|\,x\leftarrow\mathcal{A}(\lambda,H,y)]$$

关于 λ 是可忽略的.

① 多项已有工作将 AoN-PKEET 方案视作一种特殊的 PKEET 方案,并直接称呼其为 PKEET 方案,本文将沿用这一做法.

② 抗碰撞性.函数 $H:X\to Y$ 满足抗碰撞性,当且仅当 H 可在多项式时间内计算完毕.并且多项式敌手 $\mathcal{A}$ 在函数中发现一个碰撞的概率是可忽略的,即:

$Pr[(x'\neq x)\wedge(H(x)=H(x'))|x,x'\leftarrow\mathcal{A}(\lambda,H)]$

关于 λ 是可忽略的.

3) 双线性映射

令 q 是一个素数,G_1 和 G_2 是 2 个阶为 q 的群,G_1 到 G_2 的双线性映射 $e:G_1\times G_2\to G_T$ 应满足 3 个性质:

① 双线性.对任意 $P,Q\in G_1,R,S\in G_2$ 和 $a,b\in\mathbb{Z}_q$,满足 $e(aP,bR)=e(P,R)^{ab}$,$e(PQ,R)=e(P,R)\times e(Q,R)$和 $e(P,RS)=e(P,R)\times e(P,S)$.

② 非退化性.映射不会把 $G_1\times G_2$ 中的任何元素映射到 G_T 中的单位元.

③ 可计算性.对于任意的 $P\in G_1,Q\in G_2$,存在一个有效算法计算 $e(P,Q)$.

同时,双线性映射可根据 G_1,G_2 所具备的性质进一步分类[9],在一类双线性映射中 $G_1=G_2$;在 3 类双线性映射中,$G_1\neq G_2$ 且 G_1,G_2 间不存在可高效计算的同构关系.

1.2 相关工作介绍

FG-PKEET 方案由 Tang 首次提出[3].在 FG-PKEET 方案中,一对用户需进行交互以生成令牌.该令牌只可被用于检测该 2 名用户的公钥加密所得密文的一致性.与 PKEET 方案相比较,FG-PKEET 方案之中用户需消耗更多的时间消耗令牌,但是能够获得更细粒度的授权机制.

PKEET-FA 方案由 Ma 等人首次提出[4].在该方案中,授权客体类别包括用户类和密文类.

以 Ma 的具体方案为例,它总共包含 4 类令牌:1)一类令牌,用户级别令牌,其定义与 PKEET 方案中令牌的定义一致;2)二类令牌,密文级别令牌,如果一名代理接收到 2 个密文对应的二类令牌,那么他能够检测该 2 个密文的一致性;3)三类令牌,密文-密文级别令牌,其授权客体为 2 个指定密文,即该令牌只能够被用于检测这 2 个指定密文的一致性;4)四类令牌,密文-用户级别令牌,该令牌的授权客体是一个密文和一名用户.接收到该令牌的代理能够使用其检测指定密文和指定用户公钥加密所得的密文的一致性.

在安全性方面,文献[10]指出在标准模型下设计可证明安全的密码学方案是一项重要的研究课题.近来,已有多种具备标准模型安全性的 PKEET 方案被提出.其中 Zhang 等人[5]提出的方案基于特定的数论假设,其在具体设计思路上借鉴了由 Lai 等人提出的公钥加密方案[11];其余 2 种 PKEET 方案分别由 Lee 等人提出[6]和 Zeng 等人[7]提出,前者在设计方案中使用了密码学原语基于身份加密方案(identity-based encryption, IBE)和一次性签名方案(one-time signature scheme),而后者使用 Hash 证明系统(Hash proof system, HPS),因此这 2 种方案均为可灵活选择所依赖的具体数论假设的通用方案.

1.3 数论假设介绍

1) 判定性双线性 Diffie-Hellman(decisional bilinear Diffie-Hellman, DBDH)假设

考虑在挑战者 $\mathcal{C}$ 与敌手 $\mathcal{A}$ 间的游戏:令 *Gen* 为一个接收安全参数 λ 作为输入的函数,其输出元组 (p,G,G_T,e,g),其中 G,G_T 是阶为 p 的乘法循环群.e 是一个映射 $G\times G$ 至 G_T 的双线性对,g 是 G 的生成元.

首先 $\mathcal{C}$ 运行 *Gen* 生成公共参数元组(p,G,G_T,e,g),随后其从$\mathbb{Z}_p$中随机选择 a,b,c 三元素,并从 0,1 中随机选取元素 β.如果 $\beta=1$,$\mathcal{C}$ 随机选取 G_T 中的一个元素 $e(g,g)^z$ 并令其为 T,反之令 $T=e(g,g)^{abc}$.

$\mathcal{C}$ 将挑战元组(g^a,g^b,g^c,T)发送至 $\mathcal{A}$,$\mathcal{A}$ 输出其对于 β 的猜测结果 β'.我们定义敌手在游戏中的优势为

$$\left|Pr[\beta=\beta']-\frac{1}{2}\right|.$$

G,G_T 上的 DBDH 假设声称对于任意 PPT 敌手,其在上述游戏中的优势对于安全参数 λ 是可忽略的.

2) 外部判定性 Diffie-Hellman(external decisional bilinear Diffie-Hellman, Ex-DDH)假设

该假设首次在文献[12]中被提出.考虑在挑战者 $\mathcal{C}$ 与敌手 $\mathcal{A}$ 间的游戏:令 *Gen* 为一个接收安全参数 λ 作为输入的函数,其输出元组(p,G_1,G_2,G_T,e,g_1),其中 G_1,G_2,G_T 是阶为 p 的乘法循环群.g_1,g_2 分别是 G_1,G_2 的生成元.e 是将 $G_1\times G_2$ 映射至 G_T 的第 3 类双线性对映射.

首先 $\mathcal{C}$ 运行 *Gen* 生成公共参数元组(p,G_1,G_2,G_T,e,g),随后其从$\mathbb{Z}_p$中随机选择 a,b 两个元素,并从 0,1 中随机选取元素 β.如果 $\beta=1$,$\mathcal{C}$ 随机选取 G_1 中的一个元素 g_1^z,并令其为 T,反之令 $T=(g_1)^{ab}$.

$\mathcal{C}$ 将挑战元组 (g_1, g_1^a, g_1^b, T) 发送至 $\mathcal{A}$,$\mathcal{A}$ 输出其对于 β 的猜测结果 β'.我们定义敌手在上述游戏中的优势为

$$\left| Pr[\beta=\beta'] - \frac{1}{2} \right|.$$

G_1, G_2, G_T 上的 Ex-DDH 假设声称对于任意 PPT 敌手,其在游戏中的优势对于安全参数 λ 是可忽略的.

3) 外部判定性双线性 Diffie-Hellman 假设(external decisional bilinear Diffie-Hellman assumption, Ex-DBDH)

该假设在 FG-PKEET 方案[3]中首次被提出,并被用于方案安全性的归约证明.

考虑在挑战者 $\mathcal{C}$ 与敌手 $\mathcal{A}$ 间的游戏:令 Gen 为一个接收安全参数 λ 作为输入的函数,其输出元组 $(p, G_1, G_2, G_T, e, g_1)$,其中 G_1, G_2, G_T 是阶为 p 的乘法循环群.g_1, g_2 分别是 G_1, G_2 的生成元.e 是将 $G_1 \times G_2$ 映射至 G_T 的第 3 类双线性对映射.首先 $\mathcal{C}$ 运行 Gen 生成公共参数元组 (p, G_1, G_2, G_T, e, g).

其次,$\mathcal{C}$ 生成 $g_a, g_d, g_e \leftarrow_r G_1, g_b, g_c \leftarrow_r G_2$ 作为公共参数,以及 $x_1, y_1 \leftarrow_r \mathbb{Z}_p, \alpha, \beta \leftarrow_r G_1, bit \leftarrow_r \{0, 1\}$ 作为秘密参数.$\mathcal{C}$ 向敌手发送元组 X_{bit},其定义为

$$X_0 = (g_a^{x_1}, g_b^{x_1}, g_d^{x_1}\alpha, g_a^{y_1}, g_c^{y_1}, g_e^{y_1}\alpha);$$

$$X_1 = (g_a^{x_1}, g_b^{x_1}, g_d^{x_1}\alpha, g_a^{y_1}, g_c^{y_1}, g_e^{y_1}\beta).$$

$\mathcal{A}$ 输出其对于 bit 的猜测结果 bit'.我们定义敌手在游戏中的优势为

$$\left| Pr[bit=bit'] - \frac{1}{2} \right|.$$

G_1, G_2, G_T 上的 Ex-DBDH 假设声称对于任意 PPT 敌手,其在上述游戏中的优势对于安全参数 λ 是可忽略的.

2 细粒度密文一致性检测方案定义

本节我们将对所设计的细粒度密文一致性检测方案进行介绍,包括方案模型定义、合理性定义、方案安全模型定义以及方案安全性质定义.

本节会提及多个特定用户的密文元组/密钥对,本文将使用下标区分它们.

2.1 方案模型定义及合理性定义

1) 方案各函数定义

$KeyGen(\lambda)$:该非确定性算法接收一个安全参数作为输入,输出公私钥对 (PK, SK).

$Enc(m, PK)$:该非确定性算法接收一个取自消息空间 M 的明文消息 m 以及一个公钥作为输入,输出密文 C.

$Dec(C, SK)$:该确定性算法接收一个密文、一个私钥作为输入,其输出解密所得的消息 m,或输出 $\perp$ 表示解密失败.

$Aut(SK_i, SK_j)$:该非确定性算法需要用户 U_i, U_j 以及一名代理交互执行.2 名用户使用他们各自的私钥作为输入、输出令牌 $T_{i,j}$.

$Com(C_i, C_j, T_{i,j})$:该确定性算法接收由用户公钥 PK_i, PK_j 分别加密所得的密文 C_i, C_j 和对应的令牌 $T_{i,j}$ 作为输入.该算法输出 $\perp$ 表示拒绝验证,如果 C_i, C_j 解密后对应的明文 M_i, M_j 相同则输出 1,否则输出 0.

$Aut_1(SK_i, C_i, SK_j)$:该非确定性算法需要 U_i, U_j 以及一名代理交互执行,使用 U_i, U_j 各自的私钥作为输入,此外,其中一名用户将其公钥加密得到的密文作为输入,在此不妨令该密文为使用 U_i 的公钥 PK_i 加密得到的 C_i,该算法输出令牌 T_{i,j,C_i} 作为结果,或者输出 $\perp$ 表示拒绝生成.

$Com_1(C_i, T_{i,j,C_i}, C_j)$:该确定性算法接收一类令牌 T_{i,j,C_i} 和另一用户 U_j 公钥加密所得密文 C_j 作为输入.该算法输出 $\perp$ 表示拒绝验证,如果 C_i, C_j 解密后对应的明文 M_i, M_j 相同则输出 1,否则输出 0.

$Aut_2(SK_i, C_i, SK_j, C_j)$:该非确定性算法需要 U_i, U_j 以及一名代理交互执行,使用 2 名用户各自的私钥和各自公钥加密所得的密文作为输入,输出二类令牌 T_{i,j,C_i,C_j},或 $\perp$ 表示拒绝生成.由于代理可直接通过接收到的令牌检测 C_i, C_j 的一致性,因此不给出 Com_2 的定义.

2) 方案合理性(Soundness)定义

细粒度密文一致性检测方案具备合理性,当且仅当加解密正确性、一致性检测正确性这 2 个性质对任意 2 名用户,和任意一对明文 $m, m' \in M$ 成立.令用户为 U_t, U_w,对应的公私钥对为 (pk_t, sk_t),(pk_w, sk_w).

加解密正确性表达式定义为

$$Dec(Enc(m, pk_t), sk_t) = m,$$

$$Dec(Enc(m', pk_w), sk_w) = m'.$$

一致性检测正确性的具体表述为:

当 $m = m'$ 时,Com 算法的输出必定为 1.反之,Com 输出 1 的概率是可忽略的.

$$Com(Enc(m, pk_t), Enc(m', pk_w), Aut(sk_t, sk_w)).$$

当 $m = m'$ 时,Com_1 算法的输出必定为 1.反之,Com_1 输出 1 的概率是可忽略的.

$$Com_1(Enc(m',pk_w),Aut_1(sk_t,sk_w,Enc(m,pk_t))).$$

接收 Aut_2 算法执行生成的令牌后，当 $m=m'$ 时，代理必定判定 2 组密文代理必定判定 2 组密文经对应私钥解密所得的明文是相同的，即这对密文具备一致性.反之，代理误判一致性的概率是可忽略的.

$$Aut_2(SK_t,Enc(m,pk_t),SK_w,Enc(m',pk_w)).$$

2.2 方案安全模型定义

首先，为了简化模型，我们需对模型中各类用户的行为进行假设：

1) 所有用户诚实地生成他们的公私钥对，执行 Aut，Aut_1，Aut_2 算法，并且 Aut，Aut_1，Aut_2 算法的交互过程是在可信信道中完成的.

2) 代理是半诚信的，并且能够被授权检测多对用户的密文的一致性.

3) 普通用户集合与代理集合不存在交集.

PKEET 的功能性使得基于不可区分的安全性质能够被拥有特定令牌的敌手轻易攻破.为此我们参考 Tang 提出的建模方法[3]，将敌手分为 2 类：

1) 一类敌手.该类敌手为半诚信的，经 U_t，U_w 授权的代理.

2) 二类敌手.该类敌手指代的潜在敌手包括除 U_t，U_w 以外的普通用户，以及未被 U_t 授权的代理.

对于不同种类的敌手，灵活细粒度密文一致性检测方案能够保证相应级别的安全性质，在此不妨令受攻击的用户为 U_t，U_w：

针对一类敌手，灵活细粒度密文一致性检测方案应满足 2 个性质：

1) 适应性选择密文攻击下的密文单向性(one-wayness adaptive chosen ciphertext attack, OW-CCA2).OW-CCA2 安全性保证一名敌手无法从挑战密文中恢复对应明文，即使他能够在特定限制下访问解密预言机.

2) 细粒度授权性(fine-grained authorization, FG-Auth).如果 2 名用户没有授权一名代理对他们的密文进行一致性检测，FG-Auth 性质保证该代理无法越权检测 2 名用户密文的一致性.

针对二类敌手，细粒度一致性检测方案应满足性质为：

适应性选择密文攻击下的密文不可区分性(indistinguishability adaptive chosen ciphertext attack, IND-CCA2).IND-CCA2 性质保证敌手无法从挑战密文中恢复明文，即使他能够指定一对明文并要求挑战者只能选择其中一个生成挑战密文.CCA2 性质允许敌手带有限制地访问解密预言机，同时，根据对于二类敌手的定义，其能够以普通用户的身份参与令牌的生成过程.

2.3 安全性质定义

定义 1. 细粒度密文一致性检测方案具备针对一类敌手的 OW-CCA2 安全性，则多项式时间敌手在下述的 OW-CCA2 游戏中的优势是可忽略的，敌手的优势由式 $Pr[m_t'=m_t]$ 表示.

为便于后文描述，我们将查询阶段中序号为 i 的 Dec 请求称为关于用户 U_i 的解密请求.

OW-CCA2 游戏的具体定义为：

1) 初始化.敌手声明一个序号 $t\in[1,N]$ 作为其攻击对象.挑战者执行 $KeyGen$ 函数，生成若干生成公私钥对 (PK_i,SK_i)，其中 $1\leqslant i\leqslant N$.

2) 查询阶段-1.在该阶段中敌手被允许向挑战者提交 3 个请求：

① Dec.提交密文 C 和序号 i 作为输入，挑战者返回 $Dec(C,SK_i)$

② Aut.提交 2 个序号 i，j 作为输入，挑战者执行 Aut 算法，敌手在算法中扮演“代理”角色.

③ Aut_1，Aut_2.提交序号 i，j 和密文 c_i(密文 c_i，c_j)作为输入，挑战者执行 Aut_1，Aut_2 算法，敌手在算法中扮演“代理”角色.

3) 挑战阶段.挑战者随机选取消息 $m\in M$，执行函数 $Enc(m_t,PK_t)$ 并将结果 C_t^* 发送给敌手.

4) 查询阶段-2.敌手可继续发起查询阶段-1 中同类的请求，但包含一项额外限制：挑战密文 C_t^* 不能和序号 t 一同被作为 Dec 预言机的请求参数.

5) 敌手输出他对于挑战消息的猜测，游戏结束.

定义 2. 细粒度密文一致性检测方案具备 FG-Auth 安全性，则多项式时间的敌手在 FG-Auth 游戏的优势是可忽略的，敌手的优势由式 $\left|Pr[b=b']-\frac{1}{2}\right|$ 表示.

FG-Auth 游戏的具体定义为：

1) 初始化.敌手声明 2 个不同的序号 t，$w\in[1,N]$ 作为其攻击对象.挑战者执行 $KeyGen$ 函数，为 $1\leqslant i\leqslant N$ 生成公私钥对 (PK_i,SK_i).

2) 查询阶段-1.在该阶段中敌手被允许向挑战者提交 3 个请求：

① Dec.提交密文 C 和序号 i 作为输入，挑战者返回 $Dec(C,SK_i)$.

② Aut.提交 2 个序号 i,j 作为输入,挑战者执行 Aut 算法,敌手在算法中扮演"代理"角色.限制是 i,j 不能同时为 t,w.

③ Aut_1,Aut_2.提交序号 i,j 和密文 C_i(密文 C_i,密文 C_j)作为输入,挑战者执行 Aut_1,Aut_2 算法,敌手在算法中扮演"代理"角色.

3) 挑战阶段.挑战者随机选取消息 $m_0,m_1\in M$,随机生成一个比特 b.如果 $b=0$,发送挑战密文 $C_t^*=Enc(m_0,pk_t)$ 及 $C_w^*=Enc(m_0,pk_w)$ 至敌手,否则发送 $C_t^*=Enc(m_0,pk_t)$ 及 $C_w^*=Enc(m_1,pk_w)$.

4) 查询阶段-2.敌手可以发起与查询阶段-1 中同类的请求,但存在额外限制:挑战密文 C_t^*(C_w^*)不能和序号 $t(w)$一同作为 Dec 预言机的请求参数;(U_t,U_w,C_t^*),(U_w,U_t,C_w^*)无法作为提交至 Aut_1 预言机的参数;(U_t,U_w,C_t^*,C_w^*)无法作为提交至 Aut_2 预言机的参数.

5) 敌手输出他对于挑战消息的猜测 b',游戏结束.

定义 3. 细粒度密文一致性检测方案具备针对二类敌手的 IND-CCA2 安全性,则多项式时间的敌手在 IND-CCA2 游戏中的优势是可忽略的,敌手的优势由式 $\left|Pr[b=b']-\frac{1}{2}\right|$ 表示.

IND-CCA2 游戏的具体定义为:

1) 初始化.敌手声明特定序号 $t\in[1,n]$作为其攻击对象.挑战者执行 $KeyGen$ 函数,为 $1\leqslant i\leqslant N$ 生成公私钥对(PK_i,SK_i).

2) 查询阶段-1.在该阶段中敌手被允许向挑战者提交 4 个请求:

① $KeyRetrieve$.提交序号 i 作为输入,挑战者返回 SK_i,敌手不可使用 t 作为参数.

② Dec.提交密文 C 和序号 i 作为输入,挑战者返回 $Dec(C,SK_i)$.

③ Aut.提交 2 个序号 t,j 作为输入,其中 $j\neq t$,挑战者执行 Aut 算法,敌手在算法中扮演 U_j 角色.

④ $Aut_1(Aut_2)$.提交序号 t,j 和密文 c_1(密文 c_1,c_2)作为输入,其中 $j\neq t$.挑战者执行 $Aut_1(Aut_2)$ 算法,敌手在算法中扮演 U_j 角色.

敌手选择 2 个消息 m_0,m_1 并发送至挑战者,进入下一阶段.

3) 挑战阶段.挑战者随机生成一个比特 b.如果 $b=0$,发送挑战密文 $C_t^*=Enc(m_0,pk_t)$至敌手,反之发送 $C_t^*=Enc(m_1,pk_t)$.

4) 查询阶段-2.敌手可以发起与查询阶段-1 中同类的请求,但存在额外限制:挑战密文 C_t^* 不能和序号 t 一同作为 Dec 预言机的请求参数;(U_t,U_j,C_t^*)无法作为 Aut_1 预言机的请求参数;(U_t,U_j,C_t^*,C_j)无法作为 Aut_2 预言机的请求参数,其中 $j\in[1,n]$,C_j 为任意密文.

5) 敌手输出他对于挑战消息的猜测 b',游戏结束.

3 细粒度密文一致性检测方案构造

3.1 细粒度密文一致性检测方案描述

1) 公共参数定义

我们所提出的细粒度密文一致性检测方案包含 $\lambda,G,g,p,H_1,\bar{e},e,G_1,G_2,g_3,g_4,G_T,G_T^1,q,H_2$ 作为公共参数,它们的定义为:

λ 为安全参数,G 为阶为 p 的乘法群,g 是 G 的生成元.$e:G\times G\to G_T$ 是一类双线性对.G_1,G_2 是阶为 q 的乘法群,以 g_3,g_4 作为它们各自的生成元.$\bar{e}:G_1\times G_2\to G_T^1$ 是第 3 类双线性对.$H_1:G_T\times G\times G_1\times G_1\to\mathbb{Z}_p$ 是一个安全 Hash 函数.$H_2:\{0,1\}^*\to G_1$ 是另一个安全 Hash 函数.

2) 函数定义

$KeyGen(\lambda)$:该算法随机选取 $\alpha,x,y,z\in\mathbb{Z}_p$,$g_2\in G,\theta\in\mathbb{Z}_q$,并赋值 $g_1=g^\alpha,u=g^x,v=g^y,d=g^z$.最终生成的公钥元组为$(Z=e(g_1,g_2),u,v,d,g_3^\theta)$,私钥元组为$(g_2^\alpha,x,y,z,\theta)$.

$Enc(m,pk)\to(C_0,C_1,C_2,C_3,C_4,C_5)$:该算法需执行运算:

$$s,r\leftarrow_r\mathbb{Z}_p^*;\omega\leftarrow_r\mathbb{Z}_q^*;C_0=(m\parallel\omega)_{\text{encode}}Z^s;$$
$$C_3=g_3^\omega;C_4=g_3^{\omega\theta}\times H_2(m);C_5=r;$$
$$C_1=g^s;C_2=(u^tv^rd)^s;$$
$$t=H_1(C_0\parallel C_1\parallel C_3\parallel C_4).$$

在加密算法和解密算法中,我们需要一个公共的编码/解码函数,将二元组 m,ω 与 G_T 间映射.为简洁起见,我们在后面的公式中省略"encode","decode"下标.该公共函数使得$|\mathbb{Z}_p|$,$|\mathbb{Z}_q|$间存在隐含关系:令消息明文空间为$\mathbb{Z}_q$,那么不等式$|\mathbb{Z}_q|^2\leqslant|\mathbb{Z}_p|$必须成立.

$Dec(C,sk)\to m$:出于校验目的,Dec 函数首先计算:$t'=H_1(C_0\parallel C_1\parallel C_3\parallel C_4)$;$hash=\frac{C_4}{C_3^\theta}$ 并判断 $C_1^{t'x+C_5y+z}=C_2$ 是否成立,如果不成立,则输出$\perp$并中止.反之,则继续执行:

$$(m',\omega')=\frac{C_0}{e(C_1,g_2^a)};$$

$$Dec(ct,sk,pk)=\begin{cases}\perp, \text{若 } hash\neq H_2(m')\vee\\ \quad g_3^{\omega'}\neq C_3,\\ m', \text{其他.}\end{cases}$$

$Aut(sk_t,sk_w)\rightarrow token(t,w)$为用户 U_t 和 U_w 交互并输出令牌,其结构为 g_4^{γ},$g_4^{\gamma\theta_t}$,$g_4^{\gamma\theta_w}$,其中 $\gamma\leftarrow_r\mathbb{Z}_q^*$.具体过程为:

① 用户 U_t,U_w 各自随机选取 $\gamma_t,\gamma_w\leftarrow_r\mathbb{Z}_q^*$,并将 $g_4^{\gamma_t}$,$g_4^{\gamma_w}$ 发送给彼此.

② 用户 U_t,U_w 各自秘密地生成 $g_4^{\gamma_t+\gamma_w}$,$g_4^{(\gamma_t+\gamma_w)\theta_t}$ 和 $g_4^{\gamma_t+\gamma_w}$,$g_4^{(\gamma_t+\gamma_w)\theta_w}$ 并将其发送给代理.

③ 代理将 $g_4^{\gamma_t+\gamma_w}$,$g_4^{(\gamma_t+\gamma_w)\theta_t}$,$g_4^{(\gamma_t+\gamma_w)\theta_w}$ 设作算法 $Aut(sk_t,sk_w)$ 产生的令牌 (tk_r,tk_t,tk_w).如果 2 位用户表现诚实,那么 g_4^{γ},$g_4^{\gamma\theta_t}$,$g_4^{\gamma\theta_w}$ 与上述公式不可区分.上述过程步骤①中用户间传递的值被称作 Aut 中产生的中间值.

$Aut_1(C_t,sk_t,sk_w)\rightarrow token(t,w,C_t)$:$U_t$ 和 U_w 交互并生成一类令牌,其结构为 g_4^{γ},$\tilde{e}(g_4,H_2(m_t))^{\gamma}$,$g_4^{\gamma\theta_w}$,其中 $\gamma\leftarrow_r\mathbb{Z}_q^*$,$m_t$ 表示密文 C_t 解密后所对应的明文.我们声称 $token(t,w,C_t)$是密文 C_t 关于用户 U_t 的一类令牌.

具体过程为:

① 用户 U_t,U_w 各自随机选取 $\gamma_t,\gamma_w\leftarrow_r\mathbb{Z}_q^*$,分别将 $g_4^{\gamma_t}$,$\tilde{e}(C_{3,t},g_4^{\gamma_t})^{\theta_t}$ 和 $g_4^{\gamma_w}$,$g_4^{\gamma_w\theta_w}$ 发送至对方.

② 用户 U_t 使用私钥解密 C_t,如果解密结果为 $\perp$,那么整个过程中止并输出 $\perp$.这一解密行为可视作对密文 C_t 的校验操作,输出 $\perp$ 表示校验失败,反之则校验成功.

③ 用户 U_t 秘密地生成 $g_4^{\gamma_t\gamma_w}$,$g_4^{\gamma_w\gamma_t\theta_w}$,并将其发送给代理;用户 U_w 进行计算并将计算结果和 $g_4^{\gamma_t\gamma_w}$ 一同发送给代理:

$$\frac{\tilde{e}(C_{4,t},g_4^{\gamma_t})^{\gamma_w}}{\tilde{e}(C_{3,t},g_4^{\gamma_t})^{\gamma_w\theta_t}}=\frac{\tilde{e}(H_2(m_t),g_4^{\gamma_t\gamma_w})\tilde{e}(g_3^{\omega_t\theta_t},g_4^{\gamma_t\gamma_w})}{\tilde{e}(g_3^{\omega_t},g_4^{\gamma_t\gamma_w\theta_t})}=\tilde{e}(H_2(m_t),g_4^{\gamma_t\gamma_w}).$$

④ 代理令算法 $Aut_1(C_t,sk_t,sk_w)$ 产生的令牌 (tk_r,tk_t,tk_w)为

$$g_4^{\gamma_t\gamma_w},\tilde{e}(g_4,H_2(m_t))^{\gamma_t\gamma_w},g_4^{(\gamma_t\gamma_w)\theta_w}.$$

如果 2 位用户表现诚实,g_4^{γ},$\tilde{e}(g_4,H_2(m_t))^{\gamma}$,$g_4^{\gamma\theta_w}$ 与④中公式不可区分.过程步骤①中用户间传递的值被称作 Aut_1 中产生的中间值.

$Aut_2(C_t,C_w,sk_t,sk_w)\rightarrow token(t,w,C_t,C_w)$:$U_t$ 和 U_w 交互并生成二类令牌,其结构为 $\tilde{e}(H_2(m_t),g_4^{\gamma})$,$\tilde{e}(H_2(m_w),g_4^{\gamma})$,其中 $\gamma\leftarrow_r\mathbb{Z}_q^*$,$m_t$,$m_w$ 表示密文 C_t,C_w 解密后所得明文.我们声称 $token(t,w,C_t,C_w)$是密文 C_t 关于用户 U_t、密文 C_w 关于用户 U_w 的二类令牌.

具体过程为:

① 用户 U_t 使用私钥解密 C_t,如果解密结果为 $\perp$,那么整个过程中止并输出 $\perp$,用户 U_w 执行类似的步骤.解密行为可对应视作为对密文 C_t,C_w 的校验操作,输出 $\perp$ 表示校验失败,反之则校验成功.

② U_w 随机选择 $\gamma_w\leftarrow_r\mathbb{Z}_q^*$ 并发送 $g_4^{\gamma_w}$,$\tilde{e}(C_{3,w},g_4^{\gamma_w})^{\theta_w}$ 至 U_t;U_t 随机选取 $\gamma_t\leftarrow\mathbb{Z}_q^*$ 并发送 $g_4^{\gamma_t}$,$\tilde{e}(C_{3,t},g_4^{\gamma_t})^{\theta_t}$ 至 U_w.

③ U_t 进行计算,并将计算结果和 $g_4^{\gamma_t\gamma_w}$ 一同发送至代理,U_w 执行类似的步骤:

$$\frac{\tilde{e}(C_{4,w},g_4^{\gamma_w})^{\gamma_t}}{\tilde{e}(C_{3,w},g_4^{\gamma_w})^{\gamma_t\theta_w}}=\frac{\tilde{e}(H_2(m_w),g_4^{\gamma_t\gamma_w})\tilde{e}(g_3^{\omega_w\theta_w},g_4^{\gamma_t\gamma_w})}{\tilde{e}(g_3^{\omega_w},g_4^{\gamma_t\gamma_w\theta_t})}=\tilde{e}(H_2(m_w),g_4^{\gamma_t\gamma_w}).$$

④ 代理令 $H_2(m_t)^{\gamma_t\gamma_w}$,$H_2(m_w)^{\gamma_t\gamma_w}$ 为所接收到的二类令牌 (tk_t,tk_w).如果 2 位用户表现诚实,$\tilde{e}(H_2(m_t),g_4^{\gamma})$,$\tilde{e}(H_2(m_w),g_4^{\gamma})$ 与③中公式不可区分.

显然,代理可通过判断 tk_t 与 tk_w 是否相等以检测对应密文的一致性.过程步骤②中用户间传递的值被称为 Aut_2 中产生的中间值.

$Com(C_t,C_w,token_{t,w})$作为参数所提交的密文,代理计算 $t=H_1(C_0,C_1,C_3,C_4)$,并判断等式 $e(C_1,g)=e(u^tv^{C_5}w,C_2)$是否成立,如果不成立,则输出 $\perp$ 并中止.该算法的输出结果:

$$Com(C_t,C_w,token(t,w))=\begin{cases}1, \text{若 } \dfrac{\tilde{e}(C_{3,t},tk_r)}{\tilde{e}(C_{1,t},tk_t)}=\dfrac{\tilde{e}(C_{3,w},tk_r)}{\tilde{e}(C_{1,w},tk_w)},\\ 0, \text{其他.}\end{cases}$$

$Com_1(C_w,token_{t,w,C_t})$作为参数所提交的密文,代理计算使用 Com 中述及的方法对密文进行校验,如果不成立,则输出 $\perp$ 并中止.该算法的输出结果:

$$Com_1(C_w,token(t,w,C_t))=\begin{cases}1, \text{若 } tk_t=\dfrac{\tilde{e}(C_{3,w},tk_r)}{\tilde{e}(C_{1,w},tk_w)},\\ 0, \text{其他.}\end{cases}$$

3.2 细粒度密文一致性检测方案合理性证明

1) 加解密正确性

以用户 U_t 的私钥 sk_t 和使用 pk_t 诚实生成的密文 C 为例,正确性证明的推导为

$$C_1^{tx+ry+z}=g^{s\times(tx+ry+d)}=(u^t v^r w)^s=C_2.$$

等式对应解密步骤中检查 $C_1^{t'x+C_5 y+z}=C_2$ 是否成立一步.

$$\frac{C_0}{e(C_1,g_2^a)}=\frac{(m\parallel\omega)\times e(g^a,g_2)^s}{e(g^s,g_2^a)}=(m\parallel\omega).$$

等式对应提取明文信息 m 与组件 ω 一步.

$$\frac{C_4}{C_3^\theta}=\frac{g_3^{\omega\theta}H_2(m)}{g_3^{\omega\theta}}=H_2(m).$$

该等式对应从 C_4 中提取 Hash 值一步.

2) 一致性检测正确性

在算法 Com 中,给定 2 个诚实生成的密文 C_t, C_w 和相应的令牌,C_t,tk_t,tk_r 对应的计算过程为

$$\frac{\tilde{e}(C_{3,t},tk_r)}{\tilde{e}(C_{1,t},tk_t)}=\frac{\tilde{e}(g_3^{\omega\theta_t}\times H_2(m_t),g_4^\gamma)}{\tilde{e}(C_3^\omega,g_4^{\gamma\theta_t})}=$$

$$\tilde{e}(H_2(m_t),g_4^\gamma).$$

将 C_w,tk_w,tk_r 带入式中,在 $m_t=m_w$ 的情况下,将得到相同的结果,因此算法必定能够判定 2 组密文保持一致性.反之,当且仅当发生 Hash 碰撞时,该函数会输出错误判断.根据前文中 Hash 函数的定义,该事件的发生概率是可忽略的.

对于 Com_1 算法以及 Aut_2 算法的一致性检测正确性证明与该证明过程类似,故此省略.

4 细粒度密文一致性检测方案安全性证明

4.1 IND-CCA2 安全性

定理 1. 在 G_1 上的 XDH 假设、G 上的 DBDH 假设成立的前提下,3.1 节构造的 FG-PKEET 方案在标准模型下具备针对二类敌手的 IND-CCA2 安全性.

证明. 为证明该方案具备相关安全性质,可以第 2 节中定义的 IND-CCA2 游戏为基础,构建一系列游戏,最终使敌手在最后一个游戏中的优势是可忽略的,且敌手无法将其与原游戏区分.

游戏 0. 与定义 3 中的 IND-CCA2 游戏描述一致.

游戏 1. 挑战者对挑战密文进行改动:

$$Invalid_{m\omega}\leftarrow_r G_T;C_0^*=(Invalid_{m\omega})Z^s.$$

我们将通过引理 1,证明当 DBDH 假设成立时,PPT 敌手区分游戏 0 和游戏 1 的概率是可忽略的.

引理 1. 如果 G 上的 DBDH 假设成立,则以不可忽略概率区分 $Game_0$,$Game_1$ 的 PPT 敌手不存在.

证明. $\mathcal{B}$ 接收一个 DBDH 挑战元组(g,g^a,g^b,g^c,T),其中 T 为$e(g,g)^{abc}$或随机采样自G_T. $\mathcal{B}$ 的目标是区分 DBDH 挑战元组,并且能够调用尝试区分 $Game_0$,$Game_1$ 的敌手 $\mathcal{A}$ 作为子过程.$\mathcal{B}$ 将以如下方法扮演 $\mathcal{A}$ 的挑战者:

1) 初始化.$\mathcal{B}$ 随机选取 $x_v,x_d,y_u,y_v,y_d\in\mathbb{Z}_p$, $\theta\in\mathbb{Z}_q$,设置 $g_1=g^a$, $g_2=g^b$, $u=g^b g^{y_u}$, $v=g^{bx_v}g^{y_v}$, $d=g^{bx_d}g^{y_d}$.用户 U_t 公钥元组 pk_t 被发送给敌手 $\mathcal{A}$:

$$pk_t=(Z=e(g_1,g_2),u,v,d,g_3^\theta).$$

对应的私钥元组 sk_t 为$(g_2^a=g^{ab},x=b+y_u$, $y=bx_v+y_v,z=bx_d+y_d,\theta)$.

2) 查询阶段-1.

① 解密预言机.当 $\mathcal{A}$ 发起密文$c=(C_0,C_1,C_2$, $C_3,C_4,r)$关于用户 U_t 的解密请求时,B 首先计算 $t=H(C_0,C_1,C_3,C_4)$并判断等式是否成立:

$$e(C_1,u^t v^r d)=e(g,C_2).$$

若不成立,则中止解密过程并返回⊥.其次 $\mathcal{B}$ 检查 $t+rx_v+x_d=0$ 是否成立,如果成立,$\mathcal{B}$ 中止整个游戏并输出一个随机值.

由于敌手 $\mathcal{A}$ 不知道 x_v,x_d 的值,该类中止事件发生的概率为 $query/p$,其中 $query$ 为 $\mathcal{A}$ 查询解密预言机的次数.

$\mathcal{B}$ 随机选取 $\delta\leftarrow_r\mathbb{Z}_p$ 并进行计算:

$$d_{C,1}=g_1^{-\frac{ty_u+ry_v+y_d}{(t+rx_v+x_d)}}(u^t v^r d)^\delta;d_{C,2}=g_1^{\frac{-1}{(t+rx_v+x_d)}}g^\delta.$$

令 $\tilde{\delta}=\delta-\dfrac{a}{t+rx_v+x_d}$,得

$$d_{C,1}=g_2^a(u^t v^r d)^{\tilde{\delta}},d_{C,2}=g^{\tilde{\delta}}.$$

$\mathcal{B}$ 可从待解密密文中提取明文的值:

$$C_0\times\frac{e(C_2,d_{C,2})}{e(C_1,d_{C,1})}=(m\parallel\omega)Z^s\frac{e((u^t v^r d)^s,g^{\tilde{\delta}})}{e(g^s,g_2^a(u^t v^r d)^{\tilde{\delta}})}=$$

$$(m\parallel\omega)Z^s\frac{1}{e(g^s,g_2^a)}=(m\parallel\omega).$$

$\mathcal{B}$ 需要检查密文元组 C_0,C_4 所包含的明文和 Hash 是否对应,因此,$\mathcal{B}$ 的输出结果为

$$Dec(ct,sk_t)=\begin{cases}\perp, & \text{若 } C_3^\theta H_2(m)\neq C_4\vee g_3^\omega\neq C_3,\\ m, & \text{其他.}\end{cases}$$

② 令牌预言机. $\mathcal{B}$ 已知 θ 的值,因此其可以扮演用户 U_t 生成一类、二类令牌步骤中对应的中间值.由于在 IND-CCA2 游戏中,敌手扮演的是另一用户,$\mathcal{B}$ 无需完整生成令牌并发送给代理.

3) 挑战阶段. 敌手 $\mathcal{A}$ 输出 2 个明文 m_0, m_1. $\mathcal{B}$ 生成 $\beta \leftarrow_r 0,1$ 并构造挑战密文:

$$\omega \leftarrow_r \mathbb{Z}_p, C_0^* = (m_\beta \| \omega) T, C_1^* = g^c, C_3^* = g_3^\omega,$$
$$C_4^* = H_2(m_\beta) g_3^{\omega\theta}, t^* = H_1(C_0^*, C_1^*, C_3^*, C_4^*),$$
$$C_5^* = r^* = -(t^* + x_d)/x_v,$$
$$C_2^* = (g^c)^{(t^* y_u + r^* y_v + y_d)}.$$

$\mathcal{B}$ 输出 $C^* = (C_0^*, C_1^*, C_2^*, C_3^*, C_4^*, C_5^*)$, 作为挑战密文元组.

4) 查询阶段-2. 在该阶段中,敌手 $\mathcal{A}$ 对解密预言机的访问将增加如下限制:

① 如果 $\mathcal{A}$ 提交的解密请求中密文 $C = C^*$, 且用户序号为 t, 输出 $\perp$.

② 如果 $C = (C_0, C_1^*, C_2^*, C_3, C_4, C_5^*)$ 且 $H_1(C_0, C_1^*, C_3, C_4) = t^*$, $\mathcal{B}$ 中止整个游戏并输出一个随机比特,该类中止的发生概率 ADV^{CR} 为 H_1 抗碰撞性被攻破的概率.

当敌手 $\mathcal{A}$ 提交挑战密文关于用户 U_t 的一类、二类令牌请求时, $\mathcal{B}$ 将忽略对挑战密文的校验操作并直接生成中间值发送至敌手 $\mathcal{A}$.

5) 输出阶段. 敌手 $\mathcal{A}$ 输出 0 或者 1, 表示其对于游戏类别的猜测, $\mathcal{B}$ 对应输出对于 DBDH 挑战元组的猜测 $T = e(g,g)^{abc}$ 或者 $T = e(g,g)^z$.

类似于文献[5]中的结论, $\mathcal{B}$ 中止的概率上界为 $ADV^{CR} + query/p$, 对应查询阶段-1,2 中的描述.

当 $T = e(g,g)^{abc}$ 且 $\mathcal{B}$ 没有中止时,敌手 $\mathcal{A}$ 的视角与其在游戏 0 中的视角一致,反之则与游戏 1 中的视角一致:游戏 1 中均匀随机生成 $C_{0,Game_1}^* = (Invalid_{m\omega}) Z^s$ 与 $T = e(g,g)^z$ 时挑战密文组件 C_0^* 不可区分.因为令 $z \in \mathbb{Z}_p$ s.t. $T = e(g_1, g_2)^z$, 有:

$$(m^* \| \omega^*) T = (m^* \| \omega^*) \times e(g_1, g_2)^s \times e(g_1, g_2)^{z-s} =$$
$$((m^* \| \omega^*) \times e(g_1, g_2)^{z-s}) \times e(g_1, g_2)^s.$$

由于 $(m^* \| \omega^*) \times e(g_1, g_2)^{z-s}$ 与 $(Invalid_{m\omega})$ 属于同一分布,该结论成立.

综上所述,敌手 $\mathcal{A}$ 区分 $Game_0, Game_1$ 的优势为 $Adv_{\text{distinguish}_{0,1}}$ 为

$$\varepsilon_{\text{dbdh}} + Pr_{\text{Abort}} \leqslant \varepsilon_{\text{dbdh}} + ADV^{CR} + query/p,$$

其中 $\varepsilon_{\text{dbdh}}$ 为敌手在 DBDH 问题中的优势.易见,在引理 1 中述及的相关假设成立的前提下,敌手 $\mathcal{A}$ 的优势是可忽略的. 引理 1 证毕.

游戏 2. 在游戏 1 的基础上,挑战者对挑战密文进行改动为

$$z \leftarrow_r \mathbb{Z}_q; C_4^* = g_3^z H_2(m_b).$$

引理 2 将证明敌手视角下,游戏 1 与游戏 2 具备不可区分性.

引理 2. 如果 G_1 上的 XDH 假设成立,则以不可忽略概率区分 $Game_1, Game_2$ 的 PPT 敌手不存在.

证明. 接收到一个 XDH 挑战元组后, $\mathcal{B}$ 扮演敌手 $\mathcal{A}$ 的挑战者:

1) 初始化. $\mathcal{B}$ 使用 3.1 节中所描述的方法产生密钥对,并使用 XDH 挑战元组 g_3^a, g_3^b, g_3^T 代替用户 U_t 的密钥、挑战密文中的对应部分 $g_3^{\omega^*}, g_3^\theta, g_3^{\omega^*\theta}$.

2) 查询阶段 1.

① 解密预言机.在扮演解密预言机对用户 U_t 的密文进行解密时, $\mathcal{B}$ 需通过如下方式判断解密密文所包含的明文值、明文 Hash 值是否对应并输出相应结果为

$$Dec(ct, sk^*, pk) = \begin{cases} \perp, \text{若} (g_3^\theta)^\omega \times H_2(m) \neq C_4 \vee \\ \quad g_3^\omega \neq C_3, \\ m, \text{其他}. \end{cases}$$

由于在该游戏中, $\mathcal{B}$ 已知除 θ 外的所有私钥元组元素,因此他能够提取 C_0 中的 ω 并将其用于上述过程.

② 令牌预言机.由于 $\mathcal{B}$ 已知 ω, g_3^θ 的值,因此可以正常地生成一类、二类令牌.在 IND-CCA2 游戏中,敌手扮演的是另一用户,所以 $\mathcal{B}$ 无需生成令牌并发送给代理.

3) 挑战阶段. $\mathcal{B}$ 使用所给的 XDH 元组生成挑战密文中的 C_3^*, C_4^*.挑战密文元组具体生成方式为

$$Invalid_{m\omega} \leftarrow_r G_T; C_0^* = Invalid_{m\omega} Z^s;$$
$$C_3^* = g_3^{\omega^*}; C_4^* = H_3(m_\beta) \times g_3^T;$$
$$C_5^* = r \leftarrow_r \mathbb{Z}_q; C_2^* = (u^t v^r d)^s.$$

4) 查询阶段 2.解密预言机添加额外限制:当 C^* 和用户序号 t 作为参数被一起提交时,解密预言机输出 $\perp$.

当敌手 $\mathcal{A}$ 提交挑战密文关于用户 U_t 的一类、二类令牌请求时, $\mathcal{B}$ 忽略对挑战密文的校验操作,将 $Ex\text{-}DDH$ 挑战元组中的 g_3^T 视作 $g_3^{\omega^*\theta_t}$, 并用以生成令牌交互过程中所需的中间值.由于在 IND-CCA2 游戏中,敌手扮演的是另一用户, $\mathcal{B}$ 不需要生成令牌组件并发送给代理.

5) 输出阶段. 敌手 $\mathcal{A}$ 输出 1 或者 2, 表示其对于游戏类别的猜测, $\mathcal{B}$ 对应输出对于 XDH 挑战元组的猜测 $g_3^T = g_3^{ab}$ 或者 $g_3^T = g_3^z$.

如果 $g_3^T = g_3^{ab}$, 敌手 $\mathcal{A}$ 的视角与游戏 1 中的视角一致,否则与游戏 2 中的视角一致.因此敌手 $\mathcal{A}$

区分游戏 1 和游戏 2 的优势小于等于攻破 XDH 假设的优势 ε_{xdh}. 引理 2 证毕.

在游戏 2 中,敌手 $\mathcal{A}$ 的优势显然是可忽略的,因为挑战密文元组中的信息与 m_β 无关,由此,敌手在 IND-CCA2 游戏中的优势为

$$\varepsilon_{\text{IND-CCA2}}=Adv_{\text{distinguish}_{0,1}}+Adv_{\text{distinguish}_{1,2}}+\varepsilon_{Game_2}\leqslant \varepsilon_{dbdh}+ADV^{CR}+query/p+\varepsilon_{xdh}+0,$$

其中 ε_{xdh} 为敌手在 XDH 问题中的优势.易见在定理 1 中述及的相关假设成立的前提下,$\varepsilon_{\text{IND-CCA2}}$ 是可忽略的. 定理 1 证毕.

4.2 OW-CCA2 安全性

定理 2. 在 G 上的 DBDH 假设及 H_2 函数的抗碰撞性成立的前提下,3.1 节中提出的 FG-PKEET 方案对于 1 类敌手具备 OW-CCA2 安全性.

证明. 为了简化证明过程,首先需模拟令牌生成的整个过程.OW-CCA2 游戏中设立的令牌预言机生成 $token(t,i)$:

$$\gamma\leftarrow_r\mathbb{Z}_q;token(t,i)\leftarrow g_4^{\gamma\theta_t},g_4^{\gamma},g_4^{\gamma\theta_i}.$$

类似地,一类令牌预言机生成 $token(t,i,C_t)$ 的过程为

$$\gamma\leftarrow\mathbb{Z}_q;token(t,i,C_t)\leftarrow\tilde{e}(H_2(m),g_4)^{\gamma},g_4^{\gamma},g_4^{\gamma\theta_i},$$

其中,m 是由 PK_t 加密所得密文 C_t 解密所得的明文.

二类令牌预言机以如下结构生成二类令牌 $token(t,i,C_t,C_i)$:

$$\tilde{e}(H_2(m_t),g_4)^{\gamma},\tilde{e}(H_2(m_i),g_4)^{\gamma},$$

其中,$\gamma\leftarrow_r\mathbb{Z}_q^*$,$m_t,m_i$ 分别为 C_t,C_i 解密所得的明文.

3.1 节中生成一类、二类令牌时对密文实施的校验步骤同样需要执行.显然,对于身为 1 类敌手的 $\mathcal{A}$ 而言,定理 2 证明中生成令牌的方式与原方案中的方式不可区分.

游戏 0.在该游戏中,挑战者 $\mathcal{B}$ 的行为与定义 1 的 OW-CCA2 游戏中挑战者的行为一致,但是按上述方法生成各类令牌.

游戏 1.在该游戏中,挑战者 $\mathcal{B}$ 生成挑战密文:

$$Invalid_{m\omega}\leftarrow_rG_T;C_0^*=Invalid_{m\omega}Z^s;$$

$$C_4^*=H_2(m^*)g_3^{\omega\theta}.$$

挑战密文其他组件的生成方式与原方案描述一致.

当被要求生成关于挑战密文的一类令牌、二类令牌时,挑战者 $\mathcal{B}$ 直接使用 $H_2(m^*)$ 生成,并跳过校验步骤.

我们将在引理 3 中证明游戏 0 和游戏 1 的不可区分性.

引理 3. 如果 G 上的 DBDH 假设成立,则以不可忽略概率区分 $Game_0$,$Game_1$ 的 PPT 敌手不存在.

证明. 我们令 $\mathcal{B}$ 为一个尝试攻破 DBDH 假设的敌手,其试图通过调用尝试区分 OW-CCA2 游戏 0 与 OW-CCA2 游戏 1 的敌手 $\mathcal{A}$ 以攻破该假设.

1) 初始化.$\mathcal{B}$ 采用与引理 1 中相同的方法生成用户 U_t 的公钥元组并发送给敌手 $\mathcal{A}$.

2) 查询阶段-1.

① 解密预言机.与引理 1 中的描述相同;

② 令牌预言机.由于敌手 $\mathcal{A}$ 在 OW-CCA2 游戏中扮演代理角色,因此 $\mathcal{B}$ 需要完整地生成令牌并发送给 $\mathcal{A}$,令牌的生成方法与游戏 0 中的描述一致.

3) 挑战阶段.$\mathcal{B}$ 生成挑战明文 $m^*\leftarrow_rM$ 并使用与引理 1 相同的方法生成挑战密文.

4) 查询阶段-2.在该阶段中 $\mathcal{B}$ 使用与引理 1 查询阶段-2 相同的方法扮演解密预言机、使用与查询阶段-1 中相同的方法扮演令牌预言机.但当 $\mathcal{B}$ 被要求生成挑战密文与用户 U_t 相关的一类、二类令牌时,$\mathcal{B}$ 使用 $H(m^*)$ 生成令牌并忽略对挑战密文的检验步骤.

5) 输出阶段.敌手 $\mathcal{A}$ 输出 0 或者 1,表示其对于游戏类别的猜测,$\mathcal{B}$ 对应输出猜测 $T=e(g,g)^{abc}$ 或者 $T=e(g,g)^z$.

当 DBDH 挑战元组中 $T=e(g,g)^{abc}$ 时,敌手 $\mathcal{A}$ 的视角与其在游戏 0 中的视角一致,否则与其在游戏 1 中的视角一致,该结论成立的理由与在引理 1 中所述理由一致.

和引理 1 中的情况相仿,挑战者 $\mathcal{B}$ 可能会中止运行,继而无法利用敌手 $\mathcal{A}$ 输出对于 DBDH 元组的猜解.$\mathcal{B}$ 中止的概率同为 $Pr_{\text{abortion}}\leqslant Adv^{CR}+query/p$.

最后我们使用 $Adv_{\text{distinguish}_{0,1}}$ 表示敌手 $\mathcal{A}$ 区分 2 个游戏的概率,其上界为

$$Adv_{\text{distinguish}_{0,1}}\leqslant\varepsilon_{dbdh}+Pr_{\text{abortion}}\leqslant \varepsilon_{dbdh}+Adv^{CR}+query/p.$$

显而易见,在引理 3 中述及的相关假设成立的前提下,$Adv_{\text{distinguish}_{0,1}}$ 是可忽略的. 引理 3 证毕.

如果一名敌手能够在游戏 1 中取得不可忽略的优势,那么他可以被用来攻破 Hash 函数的抗碰撞性,这是因为游戏 1 中的挑战密文中,除了 C_4^* 以外其余元组均与挑战明文 m^* 无关,且 $\mathcal{B}$ 能够利用 Hash 函数抗碰撞的挑战值 $H_2(m^*)$ 扮演游戏 1 中的挑战者.

综上所述,一类敌手在 OW-CCA2 游戏中的优势为

$$Adv_{\text{OW-CCA2}}=Adv_{\text{distinguish}_{0,1}}+Adv_{Game_1}\leqslant$$
$$\varepsilon_{\text{dbdh}}+Adv^{CR}+query/p+Adv^{PR},$$

其中,Adv^{PR}指代 Hash 函数 H_2 的原像性被攻破的概率.易见敌手在 OW-CCA2 游戏中的优势是可忽略的. 定理 2 证毕.

4.3 FG-AUTH 安全性

定理 3. 在 G_1,G_2 上的 Ex-DBDH 假设、G 上的 DBDH 假设成立的前提下,3.1 节构造的 FG-PKEET 方案在标准模型下具备针对一类敌手的 FG-Auth 安全性质.

证明. 为证明 3.1 节所提及方案具备 FG-Auth 性质,我们需要构建一系列游戏:

游戏 0.挑战者的行为与定义 2 的 FG-Auth 游戏中所描述一致.

游戏 1.在该游戏中,生成关于用户 U_t 或 U_w 令牌的方法被改变,挑战者秘密地选取随机值 h_t,$h_w\leftarrow_r\mathbb{Z}_q$.

当接收到参数为(i,t)的 Aut 令牌生成请求时,挑战者返回元组 $g_4^{h_t\gamma}$,$g_4^{h_t\gamma\theta_i}$,$g_4^{h_t\gamma\theta_t}$,其中 $\gamma\leftarrow_r\mathbb{Z}_q$.当 Aut 的参数为(j,w)时,执行类似的生成步骤.

当接收到参数为(i,t,C_t)的 Aut_1 一类令牌生成请求时,挑战者解密获取 m_t,及对应的 Hash 值并返回令牌三元组 $\bar{e}(H_2(m_t),g_4)^{h_t\gamma}$,$g_4^{\gamma}$,$g_4^{h_t\gamma\theta_i}$,其中 $\gamma\leftarrow_r\mathbb{Z}_q$.当参数为$(j,w,ct_w)$时,执行类似步骤.

当接收到参数为(t,i,C_t,C_i)的 Aut_2 二类令牌请求时,挑战者解密获取 m_t,m_i,并生成对应的 Hash 值 $H_2(m_t)$,$H_2(m_i)$,和对应令牌:$\bar{e}(H_2(m_t),g_4)^{\gamma}$,$\bar{e}(H_2(m_i),g_4)^{\gamma}$,其中 $\gamma\leftarrow_r\mathbb{Z}_q$.当参数为$(w,i,C_w,C_i)$时,执行类似步骤.

在该过程中,均需按照原方案所述,对密文进行校验.

游戏 2.在该游戏中,我们对挑战密文对中的 C_w^* 中的 $C_{0,w}^*$ 组件进行修改,在挑战阶段输出的密文对为

$$C_{0,t}^*=(m_0\parallel\omega_t^*)Z_t^{s_t^*},C_{1,t}^*=g^{s_t^*},C_{3,t}^*=g_3^{\omega_t^*}$$
$$C_{4,t}^*=g_3^{\omega_t^*\theta_t}H_2(m_0),C_{5,t}^*=r_t^*,$$
$$C_{2,t}^*=(u_t^{t_t^*}v_t^{r_t^*}d_t)^{s_t^*},C_{0,w}^*=(\overline{value}_w^*)Z_w^{s_w^*},$$
$$C_{1,w}^*=g^{s_w^*},C_{3,w}^*=g_3^{\omega_w^*},C_{4,w}^*=g_3^{\omega_w^*\theta_w}H_2(m_b),$$
$$C_{5,w}^*=r_w^*,C_{2,w}^*=(u_w^{t_w^*}v_w^{r_w^*}d_w)^{s_w^*},$$

其中,$\overline{value}_w^*$ 的定义为

$$\overline{value}_w^*=\begin{cases}(m_1\parallel\omega_w^*),\text{ 若 }b=1,\\ Invalid_{m\omega}\leftarrow_r G_T,\text{ 其他.}\end{cases}$$

在游戏 2 中,当 $b=0$ 时,挑战者使用 $H_3(m_b)$ 生成 C_w^* 关于 U_w 的一类令牌、二类令牌,并在生成令牌过程中忽略对 C_w^* 的校验步骤.

为证明游戏 1 和游戏 2 的不可区分性,我们沿用引理 1 的方法证明引理 4.

引理 4. 如果 G 上的 DBDH 假设成立,以不可忽略概率区分游戏 1 和游戏 2 的 PPT 敌手不存在.

当 $b=1$ 时,显然游戏 1 与游戏 2 对于敌手而言是不可区分的,下文中我们将证明如果当 $b=0$ 时,敌手无法区分游戏 1 和游戏 2.

证明. 令 $\mathcal{B}$ 为尝试攻破 DBDH 问题的敌手,其能够调用尝试区分游戏 1 和游戏 2 的敌手 $\mathcal{A}$ 作为子过程.

1) 初始化. $\mathcal{B}$ 接收 DBDH 挑战五元组(g,g^a,g^b,g^c,T).同时依照 3.1 节方案为用户 U_t 生成一个正常的密钥对.$\mathcal{B}$ 随机选取 $bit\leftarrow_r\{0,1\}$.如果 $bit=0$,其使用 chg_{dbdh} 和随机生成的 $\theta_w\leftarrow_r\mathbb{Z}_q$ 扮演用户 U_w,否则,其生成另一个正常密钥对来扮演 U_w.

2) 查询阶段-1.

① 解密预言机:当 $bit=0$ 时,$\mathcal{B}$ 使用引理 1 中的方法处理关于 U_w 的解密请求,否则使用 3.1 节中述及的方法处理解密请求;

② 令牌预言机:与游戏 1 中所述行为一致.

3) 挑战阶段. 如果 $bit=0$,$\mathcal{B}$ 使用引理 1 中所述及的方式生成挑战密文 C_w^* 为

$$C_0^*=(m_0\parallel\omega^*)T,C_4^*=H_2(m_0)g_3^{\omega^*\theta},C_1^*=g^{s^*},$$
$$C_3^*=g_3^{\omega^*},C_5^*=r^*,C_2^*=(u^tv^{r^*}d)^{s^*}.$$

其他情况下 $\mathcal{B}$ 使用正常密钥生成挑战密文.

4) 查询阶段-2.

① 解密预言机.$\mathcal{B}$ 会根据 FG-Auth 游戏中的描述拒绝特定的挑战密文解密请求.当 $bit=0$ 时,$\mathcal{B}$ 使用引理 1 中查询阶段-2 的方法关于 U_w 的解密请求.在其他场合下,$\mathcal{B}$ 使用 3.1 节中的解密方法处理解密请求.

② 令牌预言机.与查询阶段-1 描述一致.但是当 $bit=0$ 时,挑战者 $\mathcal{B}$ 直接使用 $H_2(m_0)$ 以生成挑战密文 C_w^* 关于用户 U_w 一类令牌、二类令牌,并忽略校验步骤.

5) 输出阶段.如果 $bit=1$,$\mathcal{B}$ 保持 chg_{dbdh} 不变并重新调用敌手 $\mathcal{A}$,否则,若 $\mathcal{A}$ 将游戏识别为游戏 1 则输出对 DBDH 元组的猜测 $T=e(g,g)^{abc}$,若不然,输出 $T=e(g,g)^z$.

与引理 1 中理由类似,当 $T=e(g,g)^z$ 时,敌手 $\mathcal{A}$ 的视角与其在游戏 2 中的视角一致,反之与其在游戏 1 中的视角一致.

类似引理 1 中得出的结论,游戏 1 与游戏 2 的区分优势的上界为

$$Pr_{\text{distinguish}_{1,2}} \leqslant query/p + ADV^{CR} + \varepsilon_{\text{dbdh}}.$$

引理 4 证毕.

为了将 FG-Auth 游戏的困难性最终归约至 Ex-DBDH 假设,我们需要构建一系列游戏,并用与引理 4 相似的方法证明游戏之间的不可区分性,其原理类似于混合参数(hybrid argument).

游戏 3.对比游戏 2,挑战密文改动为

$$C^*_{0,t} = (\overline{value_t}^*) Z_t^{s_t^*}, C^*_{1,t} = g^{s_t^*},$$

$$C^*_{3,t} = g_3^{\omega_t^*}, C^*_{4,t} = g^{\omega_t^* \theta_t} H_2(m_0^{\sharp}),$$

$$C^*_{5,t} = r_t^*, C^*_{2,t} = (u_t^{t_t^*} v_t^{r_t^*} d_t)^{s_t^*}.$$

$$C^*_{0,w} = (\overline{value_w}^*) Z_w^{s_w^*}, C^*_{1,w} = g^{s_w^*},$$

$$C^*_{3,w} = g_3^{\omega_w^*}, C^*_{4,w} = g_3^{\omega_w^* \theta_w} H_2(m_b),$$

$$C^*_{5,w} = r_w^*, C^*_{2,w} = (u_w^{t_w^*} v_w^{r_w^*} d_w)^{s_w^*}.$$

其中,$m_0^{\sharp} \leftarrow_r M$,$m_b$,$\overline{value_t}^*$,$\overline{value_w}^*$ 经由等式生成:

$$m_b = \begin{cases} m_1, \text{若 } b=1, \\ m_0^{\sharp}, \text{其他}, \end{cases}$$

$$\overline{value}_w^* = \begin{cases} (m_1 \parallel \omega_w^*), \text{若 } b=1, \\ invalid^*_{m\omega,w} \leftarrow_r G_T, \text{其他}, \end{cases}$$

$$\overline{value}_t^* = invalid^*_{m\omega,t} \leftarrow_r G_T.$$

当挑战者 $\mathcal{B}$ 被要求生成 C_t^* 关于 U_t 的一类令牌或二类令牌时,他将使用 $H_2(m_0^{\sharp})$ 来生成它,且不对其进行校验.当 $b=0$ 时,$\mathcal{B}$ 使用 $H_2(m_0^{\sharp})$ 生成 C_w^* 关于 U_w 的一类令牌或二类令牌而不对其进行校验.

我们将用引理 5 证明游戏 2 与游戏 3 的不可区分性:

引理 5. 如果 G 上的 DBDH 假设成立,则不存在能够以不可忽略概率区分游戏 2 与游戏 3 的 PPT 敌手.

证明. 令 $\mathcal{B}$ 为尝试攻破 DBDH 问题的敌手,其能够调用尝试区分游戏 2 和游戏 3 的敌手 $\mathcal{A}$.

1) 初始化. $\mathcal{B}$ 接收 DBDH 挑战五元组 (g, g^a, g^b, g^c, T),他使用 chg_{dbdh} 和随机生成的 $\theta_t \leftarrow_r \mathbb{Z}_q$ 扮演用户 U_t,并生成一个正常密钥对来扮演 U_w.

2) 查询阶段-1.

① 解密预言机.$\mathcal{B}$ 使用引理 1 中的方法处理关于 U_t 的密文解密请求,其余情况下使用 3.1 部分中的方法处理解密请求;

② 令牌预言机.与游戏 1 中所述的行为一致.

3) 挑战阶段. 敌手敌手 $\mathcal{B}$ 随机选择 $b \leftarrow_r \{0,1\}$.

如果 $b=1$,则通过正常加密过程加密 $m_1 \leftarrow_r M$ 得到 C_w^*.

对于挑战密文 C_t^*,首先生成 $m_0 \leftarrow_r M$,其次,根据引理 1 所述的方案,生成密文元组:

$$C_0^* = (m_0 \parallel \omega) T, C_1^* = g^c, C_3^* = g_3^{\omega},$$

$$C_4^* = H(m_0) g_3^{\omega \theta_t}, C_5^* = r^*, C_2^* = (u^{t^*} v^{r^*} d)^c.$$

如果 $b=0$,C_t^* 的生成方法不变,对于 C_w^*,则先使用 U_w 的密钥正常加密 $m_1 \leftarrow_r G_T$ 得到 C_w^*,并进行修改:

$$C^*_{4,w} = H_2(m_0) g_3^{\omega_w \theta_w}; invalid^*_{m\omega,w} \leftarrow_r G_T,$$

$$C^*_{0,w} = (invalid^*_{m\omega,w}) Z_w^{s_w^*}.$$

4) 查询阶段-2.

① 令牌预言机.与查询阶段-1 中的描述一致.但是当 $b=0$ 时,挑战者 $\mathcal{B}$ 使用 $H(m_0)$ 生成挑战密文 C_w^* 关于用户 U_w 的一类、二类令牌;无论 b 为何值时,均使用 $H(m_0)$ 生成挑战密文 C_t^* 关于用户 U_t 的一类、二类令牌;

② 解密预言机.$\mathcal{B}$ 会根据 FG-Auth 游戏中的描述拒绝特定的挑战密文解密请求,并使用引理 1 中查询阶段-2 的方法处理关于 U_t 的解密请求.

5) 输出阶段. 若 $\mathcal{A}$ 将游戏识别为游戏 2 则输出对 DBDH 元组的猜测 $T = e(g,g)^{abc}$,若不然,输出 $T = e(g,g)^z$.

与引理 1,引理 4 中的理由类似,当 $T = e(g,g)^z$ 时,敌手 $\mathcal{A}$ 的视角与其在游戏 2 中的视角一致,反之与其在游戏 3 中的视角一致.因此,敌手区分游戏 2 与游戏 3 的概率上界为

$$Adv_{\text{distinguish}_{2,3}} \leqslant query/p + ADV^{CR} + \varepsilon_{\text{dbdh}}.$$

对于 PPT 敌手而言其概率是可忽略的.

引理 5 证毕.

游戏 4.挑战密文如以下方式生成,其与游戏 5 中挑战密文的唯一区别在于 m_b 的赋值:

$$C^*_{0,t} = (\overline{value_t}^*) Z_t^{s_t^*}, C^*_{1,t} = g^{s_t^*},$$

$$C^*_{2,t} = g_3^{\omega_t^*}, C^*_{4,t} = g_3^{\omega_t^* \theta_t} H_2(m_0^{\sharp}),$$

$$C^*_{5,t} = r_t^*, C^*_{2,t} = (u_t^{t_t^*} v_t^{r_t^*} d_t)^{s_t^*},$$

$$C^*_{0,w} = (\overline{value_w}^*) Z_w^{s_w^*}, C^*_{1,w} = g^{s_w^*},$$

$$C^*_{3,w} = g_3^{\omega_w^*}, C^*_{4,w} = g_3^{\omega_w^* \theta_w} H_2(m_b),$$

$$C^*_{5,w} = r_w^*, C^*_{2,w} = (u_w^{t_w^*} v_w^{r_w^*} d_w)^{s_w^*},$$

其中 $m^{\sharp} \leftarrow_r M$.

该游戏中 m_b,$\overline{value_t}^*$,$\overline{value_w}^*$ 的产生:

$$m_b = \begin{cases} m_1^{\sharp}, \text{若 } b=1, \\ m_0^{\sharp}, \text{其他}, \end{cases}$$

$$\overline{value}_t^* = invalid_{m\omega,t}^* \leftarrow_r G_T,$$

$$\overline{value}_w^* = invalid_{m\omega,w}^* \leftarrow_r G_T.$$

引理 6 将用于证明游戏 3 与游戏 4 对于 PPT 敌手的不可区分性.

引理 6. 如果 G 上的 DBDH 假设成立,则以不可忽略概率区分游戏 3 与游戏 4 的 PPT 敌手不存在.

证明. 令 $\mathcal{B}$ 为尝试攻破 DBDH 问题的敌手,其能够调用尝试区分游戏 3 与游戏 4 的敌手 $\mathcal{A}$.

1) 初始化. 敌手 $\mathcal{B}$ 从 DBDH 挑战者处获取 DBDH 挑战五元组(g, g^a, g^b, g^c, T),同时为 U_t 生成一个正常的密钥对.$\mathcal{B}$ 随机生成 $bit \leftarrow_r \{0,1\}$.当 $bit=1$ 时,$\mathcal{B}$ 使用 DBDH 元组 $\theta_w \leftarrow_r \mathbb{Z}_q$,根据引理 1 中所述的方法产生 U_w 的密钥对,否则,为 U_w 生成正常的密钥对.

2) 查询阶段-1.

① 解密预言机.当 $bit=1$ 时,$\mathcal{B}$ 根据引理 1 中的方法处理关于 U_w 的解密请求,其余场合使用 3.1 节中的方法处理解密请求;

② 令牌预言机.与游戏 1 中所述的行为一致.

3) 挑战阶段.$\mathcal{B}$ 随机生成的明文 m_0, m_1.如果 $bit=0$,使用正常的密钥对对分别对 m_0, m_1,生成 C_t^*, C_w^*,并进行修改:

$$C_{4,t}^* = H_2(m_0^\sharp) g_3^{\omega_t \theta_t}, C_{4,w}^* = H_2(m_0^\sharp) g_3^{\omega_w \theta_w},$$

$$C_{0,t}^* = (invalid_{m\omega,t}^*) Z_t^{s_t^*}, C_{0,w}^* = (invalid_{m\omega,w}^*) Z_w^{s_w^*},$$

其中 $m_0^\sharp \leftarrow_r M$;$invalid_{m\omega,w}^*, invalid_{m\omega,t}^* \leftarrow_r G_T$.

如果 $bit=1$,C_t^* 的生成方式与 $bit=0$ 时一致.C_w^* 通过使用引理 1 中生成挑战密文的方法产生:

$$C_0^* = (m_1 \| \omega) T, C_1^* = g^c, C_3^* = g_3^\omega,$$

$$C_4^* = H(m_1) * g_3^{\omega\theta}, C_5^* = r^*, C_2^* = (u^t v^{r^*} d)^c.$$

4) 查询阶段-2.

① 令牌预言机.当 $b=1$ 时,$\mathcal{B}$ 使用 $H_2(m_1)$ 生成挑战密文 C_w^* 关于用户 U_w 的一类令牌、二类令牌,反之挑战者 $\mathcal{B}$ 使用 $H_2(m_0^\sharp)$ 生成挑战密文 C_w^* 关于用户 U_w;挑战者 $\mathcal{B}$ 使用 $H_2(m_0^\sharp)$ 生成 C_t^* 关于用户 U_t 的一类、二类令牌;剩余情况与查询阶段-1 描述一致;

② 解密预言机.$\mathcal{B}$ 会根据 FG-Auth 游戏中的描述拒绝特定的挑战密文解密请求.当 $bit=1$ 时,$\mathcal{B}$ 将使用引理 1 中查询阶段-2 的方法处理关于 U_w 的解密请求.

5) 输出阶段.如果 $bit=1$,$\mathcal{B}$ 保持 chg_{dbdh} 不变并重新调用敌手 $\mathcal{A}$ 作为子过程,否则,若 $\mathcal{A}$ 将游戏识别为游戏 3 则输出对 DBDH 元组的猜测 $T=e(g,g)^{abc}$,若不然,输出 $T=e(g,g)^z$.

与引理 1,引理 4 中的理由类似,当 $T=e(g,g)^z$ 时,敌手 $\mathcal{A}$ 的视角与其在游戏 3 中的视角一致,反之与其在游戏 4 中的视角一致.因此,敌手区分游戏 3 与游戏 4 的概率上界为

$$Adv_{\text{distinguish}_{3,4}} \leqslant query/p + ADV^{CR} + \varepsilon_{\text{dbdh}}.$$

最终,我们构建游戏 5,其与游戏 4 统计不可区分.

游戏 5.对比游戏 4 我们修改挑战密文组件 C_4^*,最终使得敌手攻破该假设的困难性能够被归约至 Ex-DBDH 问题,与 Tang 在文献[3]中的证明类似.

挑战密文对生成:

$$invalid_{m\omega,t}^* \leftarrow_r G_T, \alpha \leftarrow_r G_1, C_{0,t}^* = (invalid_{m\omega,t}^*) Z_t^{s_t^*},$$

$$C_{1,t}^* = g^{s_t^*}, C_{3,t}^* = g_3^{\omega_t^*}, C_{4,t}^* = g_3^{\omega_t^* \theta_t} \alpha,$$

$$C_{5,t}^* = r_t^*, C_{2,t}^* = (u^{t_t^*} v^{r_t^*} d)^{s_t^*},$$

$$invalid_{m\omega,w}^* \leftarrow_r G_T, C_{0,w}^* = (invalid_{m\omega,w}^*) Z_w^{s_w^*},$$

$$C_{1,w}^* = g^{s_w^*}, C_{3,w}^* = g_3^{\omega_w^*}, C_{4,w}^* = g_3^{\omega_w^* \theta_w} X,$$

$$C_{5,w}^* = r_w^*, C_{2,w}^* = (u^{t_w^*} v^{r_w^*} d)^{s_w^*},$$

其中,若挑战者随机生成的比特值 $b=0$,则赋值 $X=\alpha$,否则,$X \leftarrow_r G_1$.

显然,游戏 4 与游戏 5 不可区分.我们将证明如果 G_1, G_2 上的 Ex-DBDH 假设成立,则敌手在游戏 5 中的优势是可忽略的.

1) 初始化.$\mathcal{B}$ 使用 Ex-DBDH 挑战元组以及随机生成的 $invalid_{m\omega,t}^*, invalid_{m\omega,w}^*$ 来扮演游戏 5 的挑战者.以 2.1 节中提及的元组 X_0 为例,令 $X_0 = (g_a^{x_1}, g_b^{x_1}, g_d^{x_1}\alpha, g_a^{y_1}, g_c^{y_1}, g_e^{y_1}\alpha)$.

B 设置相应的密钥参数为

$$g_3 = g_a; g_4^{h_t} = g_b; g_4^{h_w} = g_c; g_3^{\omega_t^*} = g_d;$$

$$g_3^{\omega_w^*} = g_e; \theta_t = x_1; \theta_w = y_1.$$

其余公私钥参数则根据 3.1 节中述及的方法正常生成.

2) 查询阶段-1.

① 解密预言机.在游戏 5 中,$\mathcal{B}$ 能够校验关于用户 U_t, U_w 的非挑战密文 $C^\sharp$ 中包含的明文与 Hash 值是否相应,以 U_t 为例,$\mathcal{B}$ 可通过判定等式是否成立:

$$\tilde{e}(C_{t,4}^\sharp * (H_2(m'))^{-1}, g_4^{h_t}) =$$

$$\tilde{e}(C_{t,3}^\sharp, g_4^{h_t \theta_t}) \wedge C_{t,3}^\sharp = g_3^{\omega'},$$

其中,m', ω'可由非挑战密文组件 $C_0^\sharp, C_1^\sharp$ 提取而得.

② 令牌预言机.$\mathcal{B}$ 会根据 FG-Auth 游戏中的描述拒绝特定的令牌生成请求.

$\mathcal{B}$ 生成由非挑战密文 $C_t^\sharp$ 关于 U_t, U_w 的一类

令牌:

$$g_4^{h_w\gamma},\tilde{e}(H_2(m^{\sharp}),g_4^{h_w\gamma}),g_4^{h_w\gamma\theta_w},$$

其中,$m^{\sharp}$ 指代使用解密预言机解密 $C_t^{\sharp}$ 所得的明文.其余情况下,$\mathcal{B}$ 采用类似办法生成相应的一类令牌、二类令牌.

3) 挑战阶段.与游戏 5 中的描述一致.

4) 查询阶段-2.

① 解密预言机.与查询阶段-1 中的描述一致,但 $\mathcal{B}$ 会根据 FG-Auth 游戏中的描述拒绝特定的挑战密文解密请求.

② 令牌预言机.$\mathcal{B}$ 会根据 FG-Auth 游戏中的描述拒绝特定的令牌生成请求.$\mathcal{B}$ 生成关于挑战密文,参数为(i,t,ct_t^*)的一类令牌

$$\frac{\tilde{e}(g_3^{\theta_t\omega_t^*}\alpha,g_4^{h_t\gamma})}{\tilde{e}(g_3^{\omega_t^*},g_4^{h_t\gamma\theta_t})}=\tilde{e}(H_2(\hat{m}),g_4^{h_t\gamma}),$$

其中,$i\neq w,\gamma\leftarrow_r\mathbb{Z}_q,H_2(\hat{m})=\alpha$,元组 $g_4^{h_t\gamma},\tilde{e}(H_2(\hat{m}),g_4^{h_t\gamma}),g_4^{h_t\gamma\theta_i}$ 为该一类令牌的结构.

$\mathcal{B}$ 生成挑战密文与非挑战密文 $C_t^{\sharp},C_i$ 关于 U_t,U_i 的二类令牌

$$\frac{\tilde{e}(g_3^{\omega_t^*\theta_t}H_2(m_t^*),g_4^{h_t\gamma_t})}{\tilde{e}(g_3^{\omega_t^*},g_4^{h_t\gamma_t\theta_t})},\tilde{e}(H_2(m_i),g_4^{h_t\gamma_t})=\tilde{e}(H_2(m_t^*),g_4^{h_t\gamma_t}),\tilde{e}(H_2(m_i),g_4^{h_t\gamma_t}),$$

其中,$\gamma_t\leftarrow_r\mathbb{Z}_q$.

由于 $\mathcal{B}$ 能够以上述方式模拟游戏 5 的挑战者,因此我们可将游戏 5 的攻破困难性归约至 Ex-DBDH 假设. 引理 6 证毕.

最终,敌手攻破方案 3.1 的 FG-Auth 性质的优势上界可表示为

$$Adv_{\text{FG-SEC}}=Adv_{\text{distinguish}_{1,2}}+Adv_{\text{distinguish}_{2,3}}+Adv_{\text{distinguish}_{3,4}}+Adv_{Game_5}=3\times(\varepsilon_{\text{dbdh}}+Adv^{CR}+query/p)+\varepsilon_{\text{ex-dbdh}}.$$

显而易见,如果相应假设成立,则敌手的优势是可忽略的. 定理 3 证毕.

5 方案特性比对与效率比较

表 1 对各个 PKEET 方案的特性进行了描述和汇总,其中方案[1]不具备令牌机制,因而不具备针对二类敌手的 IND-CCA2 性质,在安全性方面逊于其余 PKEET 方案.由表 1 可见,我们所提出的方案是唯一集成了 FG-PKEET 与 PKEET-FA 方案功能性,与此同时,安全性基于标准模型的密文一致性检测公钥加密方案.

Table 1 Property Comparison of PKEET Schemes

表 1 PKEET 方案特性对比

方案	所基于假设/原语	安全模型	功能性
文献[1]	CDH	预言机模型	PKEET
文献[2]	CDH	预言机模型	PKEET
文献[6]	IBE+One-Time Sigunature	标准模型	PKEET
文献[7]	Hash Proof System	标准模型	PKEET
文献[5]	DBDH	标准模型	PKEET
文献[3]	XDDH+EX-DBDH	预言机模型	FG-PKEET
文献[4]	CDH	预言机模型	PKEET-FA
本文方案	XDDH+DBDH+EX-DBDH	标准模型	FG-PKEET+PKEET-FA

表 2、表 3 展示了各个 PKEET 方案的计算效率与存储开销.其中$|G|,|G_T|$分别代表双线性对 $G\times G\rightarrow G_T$ 中对应群中元素的大小.$|G_1|,|G_2|,|G_T^1|$ 分别代表第 3 类双线性对 $G_1\times G_2\rightarrow G_T^1$ 中对应群中元素的大小;EXP,BP,BP_3,DEC 分别表示实施一次幂运算、双线性对、第 3 类双线性对、解密操作所需的复杂度.

Table 2 Efficiency Comparison on Encryption/Decryption of PKEET Schemes

表 2 PKEET 方案加解密效率对比

方案	加密/解密				
	公钥长度	私钥长度	密文长度	加密时间复杂度	解密时间复杂度
文献[1]	$\|G\|$	$\|\mathbb{Z}_p\|$	$3\|G\|+\|\mathbb{Z}_p\|$	3EXP	3EXP
文献[2]	$2\|G\|$	$2\|\mathbb{Z}_p\|$	$4\|G\|+\|\mathbb{Z}_p\|+2\lambda$	5EXP	2EXP
文献[6]	$5\|G\|$	$\|G\|$	$(2\lambda+15)\|G\|+\|\mathbb{Z}_p\|$	1BP+14EXP	9BP+11EXP
文献[7]	$4\|G\|$	$8\|\mathbb{Z}_p\|$	$5\|G\|$	4EXP	4EXP
文献[5]	$3\|G\|+2\|G_T\|$	$2\|G\|+3\|\mathbb{Z}_p\|$	$2\|G\|+2\|G_T\|+\|\mathbb{Z}_p\|$	6EXP	2BP+1EXP
文献[3]	$4\|G\|$	$2\|\mathbb{Z}_p\|$	$\|G\|+2\|G_1\|+2\lambda$	4EXP	2EXP
文献[4]	$3\|G\|$	$3\|\mathbb{Z}_p\|$	$5\|G\|+\|\mathbb{Z}_p\|$	6EXP	5EXP
本文方案	$\|G_T\|+3\|G\|+\|G_1\|$	$\|G_2\|+3\|\mathbb{Z}_p\|+\|\mathbb{Z}_q\|$	$\|G_T\|+2\|G\|+\|\mathbb{Z}_p\|+2\|G_1\|$	3EXP+1BP_3	7EXP

Table 3 Efficiency Comparison on Token-Related Operation of PKEET Schemes

表 3 PKEET 方案令牌操作效率对比

方案	令牌长度	令牌生成时间复杂度	一致性检测时间复杂度	生成令牌所需交互轮数
文献[1]			2BP	
文献[2](令牌)	$\|\mathbf{Z}_p\|$	0	4EXP	1
文献[6](令牌)	$3\|G\|$	0	6BP+10EXP	1
文献[7](令牌)	$2\|\mathbf{Z}_p\|$	0	4EXP	1
文献[5](令牌)	$\|G\|$	0	2BP	1
文献[3](令牌)	$3\|G_1\|$	3EXP	$4BP_3$	2
文献[4](令牌)	$\|\mathbf{Z}_p\|$	0	2BP+2EXP	1
本文方案(令牌)	$3\|G_2\|$	3EXP	$4BP_3$	2
文献[4](一类令牌)	$\|G\|+\|\mathbf{Z}_p\|$	2EXP/0	2BP+2EXP	1
本文方案(一类令牌)	$\|G_T^{-1}\|+2\|G_2\|$	(2EXP+1DEC+$1BP_3$)/(2EXP)	$2BP_3$	2
文献[4](二类令牌)	$2\|G\|+2\|G_T\|$	2BP+2EXP	2BP+2EXP	1
本文方案(二类令牌)	$2\|G_T^{-1}\|$	DEC+3EXP+$1BP_3$	0	2

由于生成一类令牌时双方用户授权客体不同,即一方授权特定一条密文信息,另一方对自己公钥加密所得的所有密文进行授权,因此在令牌生成时间上有所区别,并用符号"/"划分.

某些场合下某些方案的令牌生成时间为 0,这是由于在这些情况中,令牌在公私钥对生成过程中一并生成,只需被重复地分发给代理即可完成授权.

由于 Lee 等人的方案[6]、Zeng 等人的方案[7]均为通用型方案,因此在进行效率计算时,我们特别声明文献[6]方案采用的一次性签名方案和 IBE 方案分别为文献[13]方案、文献[14]方案.文献[7]方案所基于的难题为离散对数难题.

在我们的方案中,当某个用户需生成关于同一密文的多个一次、二次令牌时,只需执行一次 Dec 操作.由表 2、表 3 可见,相比已有方案,我们方案在计算效率与存储开销方面相当.

6 结 论

针对当前 PKEET 方案在令牌授权机制方面无法兼顾弹性授权机制特性与细粒度授权机制特性的问题,本文提出了一种新的 PKEET 方案,该方案具备细粒度特性,即允许发放一致性检测令牌的用户能够对参与检测的另一客体进行约束.同时,该方案具备弹性授权机制,允许令牌授权的客体可在用户级别及密文级别间进行选择,结合上文提及的细粒度特性,方案的令牌类别总共可分为用户-用户令牌、密文-密文令牌及密文-用户令牌共计 3 类令牌,使该方案能够在令牌授权机制方面为用户提供了更细粒度的控制手段.

方案安全性方面,本文所提及方案在功能性层面的延展使我们需对方案的安全性质定义做出修改及补充,并最终证明了我们所设计方案具备适应性选择密文攻击安全性及授权细粒度安全性.相比于此前拥有细粒度特性或弹性授权机制的方案,本文所提出方案的安全性首次建立于标准模型之上在安全性方面更加具有可靠性.

参 考 文 献

[1] Yang Guomin, Tan C H, Huang Qiong, et al. Probabilistic public key encryption with equality test [G] //LNCS 5985: Topics in Cryptology-CT-RSA 2010. Berlin: Springer, 2010: 119-131

[2] Tang Qiang. Public key encryption supporting plaintext equality test and user-specified authorization [J]. Security and Communication Networks, 2012, 5(12): 1351-1362

[3] Tang Qiang. Towards public key encryption scheme supporting equality test with fine-grained authorization [G] // LNCS 6812: Proc of the 16th Australasian Conf Security and Priracy. Berlin: Springer, 2011: 389-406

[4] Ma Sha, Huang Qiong, Zhang Mingwu, et al. Efficient public key encryption with equality test supporting flexible authorization [J]. IEEE Transactions on Information Forensics and Security, 2015, 10(3): 458-470

[5] Zhang Kai, Chen Jie, Lee H T, et al. Efficient public key encryption with equality test in the standard model [J]. Theoretical Computer Science, 2019, 755(3): 65-80

[6] Lee H T, Ling San, Seo J H, et al. Public key encryption with equality test in the standard model [J]. Information Sciences, 2020, 516(11): 89-108

[7] Zeng Ming, Chen Jie, Zhang Kai, et al. Public key encryption with equality test via Hash proof system [J]. Theoretical Computer Science, 2019, 795(43): 20-35

[8] Canetti R, Goldreich O, Halevi S. The random oracle methodology, revisited [J]. Journal of the ACM, 2004, 51(4): 557-594

[9] Galbraith S D, Paterson K G, Smart N P. Pairings for cryptographers [J]. Discrete Applied Mathematics, 2008, 156(16): 3113-3121

[10] Zhou Jun, Dong Xiaolei, Cao Zhenfu. Research advances on privacy preserving in recommender systems [J]. Journal of Computer Research and Development, 2019, 56(10): 2033-2048 (in Chinese)
(周俊,董晓蕾,曹珍富. 推荐系统的隐私保护研究进展[J]. 计算机研究与发展, 2019, 56(10): 2033-2048)

[11] Lai Junzuo, Deng R H, Liu Shengli, et al. Efficient CCA-Secure PKE from identity-based techniques [G]//LNCS 5985: Topics in Cryptology—CT-RSA 2010. Berlin: Springer 2010: 132-147

[12] Ballard L, Green M, Medeiros B de, et al. Correlation-resistant storage via keyword-searchable encryption[EB/OL]. IACR Cryptology EPrint Archive, 2005 (2005-11-22). https://eprint.iacr.org/2005/417

[13] Boneh D, Shen E, Waters B. Strongly unforgeable signatures based on computational diffie-hellman [G] //LNCS 3958: Proc of Int Conf on Theory and Practice of Public-Key Cryptography 2006. Berlin: Springer 2006: 229-240

[14] Boneh D, Boyen X. Efficient selective-ID secure identity-based Encryption without random oracles [G] //LNCS 3027: Advances in Cryptology—EUROCRYPT 2004. Berlin: Springer, 2004, 223-238

Deng Xiangtian, born in 1996. Master. His main research interests include network security and cryptography.
邓翔天,1996 年生.硕士.主要研究方向为网络安全与密码学.

Qian Haifeng, born in 1977. PhD, professor. His main research interests include network security, cryptography, and algebraic geometry.
钱海峰,1977 年生.博士,教授.主要研究方向为网络安全、密码学与代数几何.