无双线性对的门限条件匿名代理重加密方案

李兆斌* 赵 洪 魏占祯
(北京电子科技学院 北京 100070)

摘 要：条件代理重加密(CPRE)可以根据条件对密文进行细粒度的授权，现有的CPRE方案只检查原密文的条件，忽略了重加密密钥的条件符合性，也不对条件信息进行保护，容易造成隐私泄露。该文构造了基于门限的无双线性对条件匿名代理重加密方案(TB-CAPRE)，对密文和重加密密钥的条件同时进行验证，并将敏感的条件信息进行匿名化处理，利用门限将重加密分布到多个代理节点完成，能够抵御合谋攻击。理论分析证明了该方案在随机预言模型下满足适应性选择密文攻击下的不可区分安全性(IND-CCA)。效率和计算量分析表明TB-CAPRE在增加安全性和相关功能后并没有引入过大的开销，可以应用到分布式环境中。

关键词：条件代理重加密；门限；条件匿名；无双线性对
中图分类号：TN918.4; TTP309.7 **文献标识码**：A **文章编号**：1009-5896(2021)11-3350-09
DOI: 10.11999/JEIT200946

Threshold- Based Pairing-free Conditional Anonymous Proxy Re-Encryption Scheme

LI Zhaobin ZHAO Hong WEI Zhanzhen
(*Beijing Electronic Science and Technology Institute*, *Beijing* 100070, *China*)

Abstract: Conditional Proxy Re-Encryption (CPRE) can grant fine-grained authorization to the original ciphertext according to the conditions. The existing CPRE schemes only check the conditions of the original ciphertext, but ignore the conditions of the re-encryption key. No effective measures are taken to protect the conditions in these CPRE schemes, which may lead to privacy disclosure of conditions. A Threshold-Based Conditional Anonymous Proxy Re-Encryption scheme (TB-CAPRE) is constructed, which can not only verify the conditions of ciphertext and re-encryption key at the same time, but also make sensitive conditional information anonymous. The re-encryption processes are completed by multiple agent nodes, so TB-CAPRE can resist the collusion attacks. The theoretical analysis proves that the new scheme is INDistinguishable against adaptive Chosen-Ciphertext Attack(simply donoted by IND-CCA) in the random oracle. The analysis of performance and computation shows that TB-CAPRE does not introduce excessive overhead while increasing security and related functions. It is possible that TB-CAPRE is applied to distributed environment.

Key words: Conditional Proxy Re-Encryption (CPRE); Threshold; Conditional anonymous; Pairing-free

1 引言

随着云计算、物联网等新技术的快速发展和应用，大量数据需要外包存储在各种开放环境中，数据安全尤其是数据机密性保护已经成为企业和个人关注的焦点。传统的解决方法是将数据加密后再存储到云服务器，但这种方法在密文授权和密钥管理时不够灵活。为了解决密文授权问题，Blaze 等人[1]首次提出了代理重加密方案，半可信服务器使用重加密密钥进行密文转换，将授权者的密文转换成被授权者可以解密的密文，重加密过程不会泄露明文和授权者的私钥信息。代理重加密可以实现单向或双向授权，也可以实现单次或多次重加密操作[2–5]，目前的重加密方案都是基于传统公钥或基于身份的公钥来实现的，在云计算环境中的应用已经有很多研究成果[6–11]。但这些代理重加密方案都存在缺陷，代理服务器在得到重加密密钥后可以将授权者的所有密文都进行重加密，授权者无法对密文进行细粒度的控制。因此Weng等人[12]提出了条件代理重加密方案(Conditional Proxy Re-Encryption, CPRE)，只有符合条件的密文才会被代理服务器重新加密，

收稿日期：2020-11-05；改回日期：2021-03-18；网络出版：2021-04-20
*通信作者： 李兆斌 bestibesti@163.com
基金项目：国家重点研发计划(2017YFB0802705)
Foundation Item: The National Key Research and Development Project (2017YFB0802705)

很多文献对CPRE进行了研究，大多数的CPRE方案都是基于双线性对[13–16]。由于双线性对运算效率不高，因此有研究者提出了无双线性对的CPRE方案[17–20]，但这些方案在重加密过程中只检查原始密文的条件符合性，而忽略了重加密密钥的条件符合性检查，导致攻击者可以发送不符合条件的重加密密钥来让代理服务器进行重加密操作，从而大量消耗代理服务器的资源。这些方案也没有对条件信息进行保护，容易造成敏感条件信息泄露。

为了解决单个代理服务器完成重加密操作带来的信任过度集中和单点失效问题，有文献提出了基于门限的代理重加密方案[21–24]。其中Patil等人[22]提出的方案在重加密时没有考虑原始密文的条件，并且需要授权者和被授权者将私钥信息提交给第三方来生成重加密密钥，存在很大的安全隐患。同时该方案中的被授权者与代理服务器合谋可以恢复出授权者私钥的哈希值，从而导致方案中授权者用该私钥哈希值对应公钥加密的所有原始密文都存在被恶意恢复的安全风险，方案只能达到选择明文攻击下的不可区分安全性。

本文是在条件代理重加密[20]和门限代理重加密方案[22]基础上提出的一种新的门限条件匿名代理重加密方案，本方案不依赖双线性对，在重加密过程中同时对密文和重加密密钥进行条件检查，对敏感的条件信息进行匿名化处理，将重加密过程分布到多个代理节点来执行，并能够防止代理服务节点与被授权者合谋恢复出授权者的私钥信息。

2　无双线性对的门限条件匿名代理重加密方案及安全性定义

2.1　方案定义

无双线性对的门限条件匿名代理重加密方案(Threshold-Based Conditional Anonymous Proxy Re-Encryption scheme, TB-CAPRE)基于有限域中的离散对数，在解决条件匿名化的同时也解决了Paul等人[20]所提方案中只检查密文的条件信息而忽略了重加密密钥有效性问题，具有更细粒度的密文授权能力。本文中的条件信息$t \in Z_q^*$可由用户根据实际应用需要进行定义，如密文生成时间、文件类型等。方案包含如下7个算法：

(1)系统参数生成(Setup)：输入安全参数λ，输出公开的系统参数param；

(2)密钥生成(KeyGen)：输入系统的公开参数集param，为用户i生成公私钥对$(\mathrm{PK}_i, \mathrm{SK}_i)$；

(3)加密(Encryption)：输入系统公开参数集param、用户的公钥PK_i、明文消息m和条件信息t，生成消息的原始密文C_i；

(4)重加密密钥生成与分发(ReKeyGen&Dist)：输入系统公开参数集param、授权者的私钥SK_i、被授权者的公钥PK_j和条件信息t，生成重加密密钥 $\mathrm{RK}_{i\to j}(z_\mu) = (\mathrm{RK}_{i\to j}^{(1)}, z_\mu, f(z_\mu), \mathrm{RK}_{i\to j}^{(3)}), u = \{1, 2, \cdots, n\}$ ，并分发到不同的代理节点；

(5)重加密(Rencryption)：输入系统公开参数集param、原始密文C_i、重加密密钥$\mathrm{RK}_{i\to j}(z_\mu)$，生成重加密密文$C_j$；

(6)原始密文(第2层密文)解密(Decryption2)：输入系统公开参数集param、原始密文C_i、授权者私钥SK_i以及密文条件信息t，返回明文消息m或无效标识$\perp$；

(7)重加密密文(第1层密文)解密(Decryption1)：输入系统公开参数集param、重加密密文C_j、被授权者私钥SK_j，返回明文消息m或无效标识$\perp$。

2.2　安全性定义

由于本方案中存在两种密文，即原始密文和重加密密文，因此在安全性定义时必须考虑两种密文的安全。

定义1 无双线性对的门限条件匿名代理重加密方案选择密文攻击下的不可区分性(TB-CPAE IND-Distinguishablity under Chosen Ciphertext Attack, TB-CAPRE-IND-CCA)，包括原始密文选择密文攻击下的不可区分性(Level-2 INDistinguishablity under Chosen Ciphertext Attack, L2-IND-CCA)和重加密密文选择密文攻击下的不可区分性 (Level-1 INDistinguishablity under Chosen Ciphertext Attack, L1-IND-CCA)。

下面通过两个安全游戏来描述原密文和重加密密文选择密文攻击下的不可区分性。

2.2.1　L2- IND- CCA游戏

该游戏考虑原始密文(第2层密文)选择密文攻击下的不可区分性。挑战者$\mathcal{C}$模拟TB-CAPRE运行环境，接收敌手$\mathcal{A}$的查询，游戏过程如下：

(1)初始化，挑战者$\mathcal{C}$运行Setup(λ)，获得系统参数param，并将param返回给敌手$\mathcal{A}$。

(2)查询阶段1，敌手$\mathcal{A}$可进行如下查询：

(a)诚实用户公钥查询$\mathcal{O}_{\mathrm{u}}(i)$：输入诚实的用户$i$，挑战者$\mathcal{C}$运行KeyGen$(i, \mathrm{param})$ 来获得用户的公私钥对$(\mathrm{PK}_i, \mathrm{SK}_i)$，并将公钥$\mathrm{PK}_i$发送给敌手$\mathcal{A}$；

(b)毁坏用户私钥查询$\mathcal{O}_{\mathrm{c}}(i)$：输入毁坏的用户$i$，挑战者$\mathcal{C}$运行KeyGen$(i, \mathrm{param})$ 来获得用户的公私钥对$(\mathrm{PK}_i, \mathrm{SK}_i)$，并将公私钥对$(\mathrm{PK}_i, \mathrm{SK}_i)$发送给敌手$\mathcal{A}$；

(c)重加密密钥查询$\mathcal{O}_{\mathrm{rk}}(\mathrm{PK}_i, \mathrm{PK}_j, t)$：挑战者$\mathcal{C}$运行ReKeyGen&Dist$(\mathrm{SK}_i, \mathrm{PK}_j, t, \mathrm{param})$来获得重

加密密钥$\mathrm{RK}_{i\to j}(z_\mu)=(\mathrm{RK}_{i\to j}^{(1)},z_\mu,f(z_\mu),\mathrm{RK}_{i\to j}^{(3)})$, $\mu=\{1,2,\cdots,k\}$，并将$\mathrm{RK}_{i\to j}(z_\mu)$发送给敌手$\mathcal{A}$；

(d)重加密查询$\mathcal{O}_{\mathrm{re}}(\mathrm{PK}_i,\mathrm{PK}_j,t,C_i)$：挑战者$\mathcal{C}$运行$C_j=\mathrm{Rencryption}(C_i,\mathrm{ReKeyGen}\ \&\mathrm{Dist}(\mathrm{SK}_i,\mathrm{PK}_j,t,\mathrm{param}),\mathrm{param})$来获得重加密密文，并发送给敌手$\mathcal{A}$；

(e)原始密文解密查询$\mathcal{O}_{d_2}(\mathrm{P}K_i,C_i)$：挑战者$\mathcal{C}$运行$\mathrm{Decryption2}(C_i,\mathrm{SK}_i,t,\mathrm{param})$，并将结果发送给敌手$\mathcal{A}$；

(f)重加密密文解密查询$\mathcal{O}_{d_1}(\mathrm{PK}_j,C_j)$：挑战者$\mathcal{C}$运行$\mathrm{Decryption1}(C_j,\mathrm{SK}_j,\mathrm{param})$，并将结果发送给敌手$\mathcal{A}$。

(3)挑战阶段，查询阶段1结束后敌手$\mathcal{A}$输出目标公钥PK_i^*、加密条件t^*，长度相同的明文$m_0,m_1\in M$，限制条件是PK_i^*不能属于毁坏用户，如果PK_j属于毁坏用户则敌手$\mathcal{A}$不能进行重加密密钥查询$\mathcal{O}_{\mathrm{rk}}(\mathrm{PK}_i^*,\mathrm{PK}_j,t^*)$。挑战者$\mathcal{C}$随机选择$\alpha\in\{0,1\}$，生成挑战密文$C_i^*=\mathrm{Encryption}(\mathrm{PK}_i^*,m_\alpha,t^*,\mathrm{param})$，并把$C_i^*$返回给敌手$\mathcal{A}$。

(4)查询阶段2，敌手$\mathcal{A}$继续进行与查询阶段1相同的查询，但限制是：如果PK_j属于毁坏用户则敌手$\mathcal{A}$不能进行重加密查询$\mathcal{O}_{\mathrm{rk}}(\mathrm{PK}_i^*,\mathrm{PK}_j,C^*,t^*)$；敌手$\mathcal{A}$不能进行重加密密文解密查询$\mathcal{O}_{d_1}(\mathrm{PK}_j,C_j)$；敌手$\mathcal{A}$不能进行原始密文解密查询$\mathcal{O}_{d_2}(C_i^*,\mathrm{PK}_i^*)$。

(5)猜测阶段，敌手$\mathcal{A}$输出对α的猜测α'。如果$\alpha'=\alpha$，则敌手$\mathcal{A}$赢得游戏，获胜的优势定义为$\mathrm{Adv}_{\mathcal{A}}^{\mathrm{L2-IND-CCA}}=|2\mathrm{Pr}\lceil\alpha'=\alpha\rfloor-1|$。

如果一个敌手$\mathcal{A}$在多项式时间内经过q_{u}次诚实用户公钥查询$\mathcal{O}_{\mathrm{u}}$, q_{c}次毁坏用户私钥查询$\mathcal{O}_{\mathrm{c}}$, q_{rk}次重加密密钥查询$\mathcal{O}_{\mathrm{rk}}$, q_{re}次重加密查询$\mathcal{O}_{\mathrm{re}}$, q_{d_2}次原始密文解密查询$\mathcal{O}_{d_2}$, q_{d_1}次重加密密文解密查询$\mathcal{O}_{d_1}$后赢得游戏的优势$\mathrm{Adv}_{\mathcal{A}}^{\mathrm{IND-L2-CCA}}\le\epsilon$，则称TB-CAPRE方案满足原始密文选择密文攻击下的不可区分性。

2.2.2 L1-IND-CCA游戏

该游戏考虑重加密密文(第1层密文)选择密文攻击下的不可区分性。使用第1层密文作为挑战密文，该密文不泄露第2层密文(原始密文)的任何信息。游戏过程如下：

(1)初始化，挑战者$\mathcal{C}$运行$\mathrm{Setup}(\lambda)$，获得系统参数param，并将param返回给敌手$\mathcal{A}$。

(2)查询阶段1，敌手$\mathcal{A}$可向挑战者$\mathcal{C}$发送各种查询(与L2-IND-CCA相同)。

(3)挑战阶段，敌手$\mathcal{A}$输出授权者的公钥PK_i'、被授权者的公钥PK_j^*、条件信息t^*以及两个等长的明文消息$m_0,m_1\in M$，限制条件是PK_j^*不能属于毁坏用户。挑战者$\mathcal{C}$随机选择$\alpha\in\{0,1\}$，生成挑战密文C_j^*返回给敌手$\mathcal{A}$。

(4)查询阶段2，敌手$\mathcal{A}$继续进行与查询阶段1相同的查询，但限制条件是敌手$\mathcal{A}$不能发送重加密密文解密查询$\mathcal{O}_{d_1}(\mathrm{PK}_j^*,C_j^*)$。

(5)猜测阶段，敌手$\mathcal{A}$输出对α的猜测α'。如果$\alpha'=\alpha$，则敌手$\mathcal{A}$赢得游戏，获胜的优势定义为$\mathrm{Adv}_{\mathcal{A}}^{\mathrm{L1-IND-CCA}}=|2\mathrm{Pr}\lceil\alpha'=\alpha\rfloor-1|$。

如果一个敌手$\mathcal{A}$在多项式时间内经过q_{u}次诚实用户公钥查询$\mathcal{O}_{\mathrm{u}}$, q_{c}次毁坏用户私钥查询$\mathcal{O}_{\mathrm{c}}$, q_{rk}次重加密密钥查询$\mathcal{O}_{\mathrm{rk}}$, q_{re}次重加密查询$\mathcal{O}_{\mathrm{re}}$, q_{d_2}次原始密文解密查询$\mathcal{O}_{d_2}$, q_{d_1}次重加密密文解密查询$\mathcal{O}_{d_1}$后赢得游戏的优势$\mathrm{Adv}_{\mathcal{A}}^{\mathrm{IND-L1-CCA}}\le\epsilon$，则称TB-CAPRE方案满足重加密密文选择密文攻击下的不可区分性。

3 方案的具体构造

(1)系统参数生成(Setup)：输入安全参数λ，选取λ比特素数q，g是q阶循环群G的生成元；H_1,H_2,H_3,H_4,H_5,H_6是6个单向哈希函数，其中H_1: $\{0,1\}^{l_0}\times\{0,1\}^{l_1}\to Z_q^*$, H_2: $G\to\{0,1\}^{l_0+l_1}$, H_3: $G\times\{0,1\}^*\to Z_q^*$, H_4 : $G^3\times\{0,1\}^{l_0+l_1}\to Z_q^*$, H_5 : $G^2\to Z_q^*$, H_6: $G^3\to Z_q^*$，消息空间$M=\{0,1\}^{l_0}$, (k,n)是门限参数，输出公共参数集$\mathrm{param}=\{q,g,G,H_1,H_2,H_3,H_4,H_5,H_6,l_0,l_1,(k,n)\}$。

(2)密钥生成(KeyGen)：输入param，随机选取$x_i,x_j\in Z_q^*$分别作为用户i和j的私钥：$\mathrm{SK}_i=s_i,\mathrm{SK}_j=s_j$，对应的公钥$\mathrm{PK}_i=g^{s_i},\mathrm{PK}_j=g^{s_j}$。

(3)加密(Encryption)：输入param、消息$m\in M$、用户i的公钥PK_i以及条件信息$t\in Z_q^*$，加密过程如下：

随机选取$\omega\in\{0,1\}^{l_1}$, $u\in Z_q^*, r=H_1(m,\omega)$, $T=g^t, D=(\mathrm{PK}_i)^{r\cdot H_3(t,T)}$, $E=(\mathrm{PK}_i)^{u\cdot H_3(t,T)}$, $F=H_2(g^r)\oplus(m||\omega)$, $\mathcal{S}=u\cdot H_3(t,T)+r\cdot H_3(t,T)\cdot H_4(D,E,F,T)$，输出原始密文$C_i=(D,E,F,T,\mathcal{S})$。

(4)重加密密钥生成与分发(ReKeyGen&Dist)：输入param、授权者i的私钥SK_i、被授权者j的公钥PK_j以及条件信息$t\in Z_q^*$，使用(k,n)门限秘密共享方式将重加密密钥分发到不同的代理节点。该过程只对特定的条件t生成重加密密钥，能够防止使用一个重加密密钥对授权者的所有密文进行密文转换，解决了被授权者与代理合谋实现密文的非授权访问问题。生成的重加密密钥不会泄露授权者的私钥和条件等敏感信息。生成过程如下：

随机选取$x_i\in Z_q^*$，然后计算$X_i=g^{x_i},\kappa=H_6(X_i,\mathrm{PK}_j,(\mathrm{PK}_j)^{x_i})$, $\mathrm{RK}_{i\to j}^{(1)}=X_i,\mathrm{RK}_{i\to j}^{(2)}=$

$\frac{\kappa}{\mathrm{SK}_i\cdot H_3(t,g^t)}=\frac{\kappa}{s_i\cdot H_3(t,T)}$，$\mathrm{RK}_{i\to j}^{(3)}=t+\mathrm{SK}_i\cdot H_5(g^t,\mathrm{PK}_i)=t+s_i\cdot H_5(T,\mathrm{PK}_i)$，分别安全地发送给$n$个代理节点。

(5)重加密(Rencryption)：输入param、原始密文C_i以及重加密密钥$\mathrm{RK}_{i\to j}(z_\mu)$，$k$个代理节点$R_\mu(\mu=\{1,2,\cdots,k\})$解析$C_i=(D,E,F,T,\mathcal{S})$，验证$g^{\mathrm{RK}_{i\to j}^{(3)}}\stackrel{?}{=}T\cdot\mathrm{PK}_i^{H_5(T,\mathrm{PK}_i)}$，若不成立则输出无效标志$\perp$；否则进一步验证$\mathrm{PK}_i^{\mathcal{S}}\stackrel{?}{=}E\cdot D^{H_4(D,E,F,T)}$，若不成立则输出无效标志$\perp$；否则进行运算$f'(z_\mu)=f(z_\mu)\cdot\prod_{\nu=1,\nu\neq\mu}^{k}\frac{0-z_\mu}{z_\mu-z_\nu}$，$D'_\mu=D^{f'(z_\mu)}$，每个代理节点$R_\mu$将$D'_\mu$发送给其他代理节点，其中任意一个代理节点计算$C_j^1=\prod_{\mu=1}^{k}D_{\mu'}=\prod_{\mu=1}^{k}g^{s_i\cdot r\cdot H_3(t,T)\cdot f'(z_\mu)}=g^{s_i\cdot r\cdot H_3(t,T)\cdot\sum_{\mu=1}^{k}f(z_\mu)\prod_{\nu=1,\nu\neq\mu}^{k}\frac{0-z_\mu}{z_\mu-z_\nu}}$，由拉格朗日插值公式可得$\sum_{\mu=1}^{k}f(z_\mu)\prod_{\nu=1,\nu\neq\mu}^{k}\frac{0-z_\mu}{z_\mu-z_\nu}=\frac{\kappa}{s_i\cdot H_3(t,T)}$，因此$C_j^1=g^{s_i\cdot r\cdot H_3(t,T)\cdot\frac{\kappa}{s_i\cdot H_3(t,T)}}=g^{r\cdot\kappa}$。令$C_j^2=\mathrm{RK}_{i\to j}^{(1)}=X_i$，$C_j^3=F=H_2(g^r)\oplus(m||\omega)$，输出重加密密文$C_j=(C_j^1,C_j^2,C_j^3)$。

(6)原始密文(第2层密文)解密(Decryption2)：输入系统公开参数集param、原始密文C_i、授权者私钥SK_i以及密文条件信息t，解析原始密文$C_i=(D,E,F,T,\mathcal{S})$，计算$\mathrm{PK}_i^{\mathcal{S}}\stackrel{?}{=}E\cdot D^{H_4(D,E,F,T)}$，若不成立则输出无效标志$\perp$；否则进行运算$m||\omega=F\oplus H_2\left(D^{\frac{1}{\mathrm{SK}_i\cdot H_3(t,T)}}\right)$，验证$D\stackrel{?}{=}PK_i^{H_1(m,\omega)\cdot H_3(t,T)}$，若成立则返回原始消息$m$，若不成立则输出无效标志$\perp$。

(7)重加密密文(第1层密文)解密(Decryption1)：输入系统公开参数集param、重加密密文C_j、被授权者私钥私钥SK_j，解析重加密密文$C_j=(C_j^1,C_j^2,C_j^3,)$，计算$\kappa'=H_6(C_j^2,\mathrm{PK}_j,(C_j^2)^{\mathrm{SK}_j})$，$m||\omega=C_j^3\oplus H_2\left((C_j^1)^{1/\kappa'}\right)$，验证$C_j^1\stackrel{?}{=}g^{\kappa'\cdot H_1(m,\omega)}$，若成立则返回原始消息$m$，若不成立则输出无效标志$\perp$。

4　方案的分析

4.1　正确性分析

(1)原始密文条件验证

$$\begin{aligned}E\cdot D^{H_4(D,E,F,T)}=&(\mathrm{PK}_i)^{u\cdot H_3(t,T)}\\&\cdot(\mathrm{PK}_i)^{r\cdot H_3(t,T)\cdot H_4(D,E,F,T)}\\=&\mathrm{PK}_i^{\mathcal{S}}\end{aligned}\tag{1}$$

(2)重加密密钥条件验证

$$T\cdot\mathrm{PK}_i^{H_5(T,\mathrm{PK}_i)}=g^{t+s_i\cdot H_5(T,\mathrm{PK}_i)}=g^{\mathrm{RK}_{i\to j}^{(3)}}\tag{2}$$

(3)原始密文解密验证

$$\begin{aligned}F\oplus H_2\left(D^{\frac{1}{\mathrm{SK}_i\cdot H_3(t,T)}}\right)=&(m||\omega)\oplus H_2(g^r)\oplus H_2\\&\cdot\left(\left((\mathrm{PK}_i)^{r\cdot H_3(t,T)}\right)^{\frac{1}{\mathrm{SK}_i\cdot H_3(t,T)}}\right)=(m||\omega)\end{aligned}\tag{3}$$

(4)重加密密文解密验证

$$\begin{aligned}\kappa'=&H_6\left(C_j^2,\mathrm{PK}_j,(C_j^2)^{\mathrm{SK}_j}\right)\\=&H_6\left(X_i,\mathrm{PK}_j,(X_i)^{\mathrm{SK}_j}\right)=\kappa\end{aligned}\tag{4}$$

$$\begin{aligned}C_j^3\oplus H_2\left((C_j^1)^{1/\kappa'}\right)=&(m||\omega)\oplus H_2(g^r)\\&\oplus H_2\left((g^{r\cdot\kappa})^{1/\kappa}\right)=(m||\omega)\end{aligned}\tag{5}$$

4.2　安全性分析

4.2.1　条件匿名性

本方案在原始消息加密和生成重加密密钥过程中使用条件信息t，生成的原始密文和重加密密钥中只包含$T=g^t$，所以即使攻击者得到T也无法恢复出条件信息，可以实现条件的匿名性。

4.2.2　重加密密钥保护

本方案使用门限技术将重加密密钥分发到n个不同的代理服务节点，在重加密过程中k个代理节点不需要直接恢复出重加密密钥，而是进行运算$f'(z_\mu)=f(z_\mu)\cdot\prod_{\nu=1,\nu\neq\mu}^{k}\frac{0-z_\mu}{z_\mu-z_\nu}$，$D'_\mu=D^{f'(z_\mu)}$，$C_j^1=\prod_{\mu=1}^{k}D'_\mu=\prod_{\mu=1}^{k}g^{s_i\cdot r\cdot H_3(t,T)\cdot f'(z_\mu)}$，这样就能够保护重加密密钥中的关键信息。

4.2.3　抗合谋攻击

当被授权者联合k个代理服务节点进行合谋攻击时，k个代理节点先通过分享各自掌握的部分重加密密钥，得到$\mathrm{RK}_{i\to j}^{(2)}=\frac{\kappa}{\mathrm{SK}_i\cdot H_3(t,g^t)}$，然后结合被授权者已经得到的一次性密钥$\kappa$，可以恢复出$\mathrm{SK}_i\cdot H_3(t,g^t)$。由于合谋攻击者没有条件信息$t$，因此无法计算$H_3(t,g^t)$，也就无法恢复出授权者的私钥$\mathrm{SK}_i$。

4.2.4　密文安全性

定义2 G是一个大素数q阶循环群，g是G的生成元，群G上的计算性Diffie-Hellman (Computational Diffie-Hellman, CDH)问题是：对于任意的$a,b\in Z_q^*$，给定$\{g,g^a,g^b\}\in G$，计算$g^{ab}\in G$。设概率多项式时间算法A_{CDH}解决CDH问题的概率为$\mathrm{Adv}(A_{\mathrm{CDH}})=\Pr[A_{\mathrm{CDH}}(G,q,g,g^a,g^b)=g^{ab}]$。

若对所有的概率多项式时间算法算法A_{CDH}，$\mathrm{Adv}(A_{\mathrm{CDH}})$都是可忽略的，则认为$G$上的CDH问题

是难解的。G上的除法计算性Diffie-Hellman (Divisible Computation Diffie-Hellman, DCDH)问题是对于任意的$a,b\in Z_q^*$，给定$\{g,g^a,g^b\}\in G$，计算$g^{\frac{b}{a}}\in G$，由文献[25]知DCDH问题与CDH问题等价。

定理 1 TB-CAPRE方案对于原始密文是基于DCDH假设，随机预言模型下IND-CCA安全的。

证明：利用文献[20]和文献[25]的方法，假设存在概率多项式时间敌手$\mathcal{A}$在随机预言模型下能以概率ε的优势攻破本方案，可以构建一个算法$\mathcal{C}$来模拟L2- IND- CCA游戏中的挑战者，以概率ε'的优势解决DCDH问题，且$\varepsilon'\geq\frac{1}{q_{H_2}}\left(\frac{\epsilon}{e\left(1+q_{rk}\right)}-\frac{q_{H_1}}{2^{l_1}}-\frac{q_{H_4}}{2^{l_0+l_1}}-q_{d_2}\left(\frac{q_{H_1}+q_{H_2}}{2^{l_0+l_1}}+\frac{2}{q}\right)\right)$，也就是说给定$\{g,g^a,g^b\}\in G,a,b\in Z_q^*$作为算法$\mathcal{C}$的输入，$\mathcal{C}$的目标是得到$g^{\frac{b}{a}}\in G$。算法$\mathcal{C}$将$\{q,g,G,H_1,H_2,H_3,H_4,H_5,H_6,l_0,l_1\}$发送给$\mathcal{A}$，其中$H_1,H_2,H_3,H_4,H_5,H_6$是随机预言，算法$\mathcal{C}$维护6个哈希列表$H_i^{\text{list}}$(初始为空)，$1\leq i\leq 6$。算法$\mathcal{C}$与敌手$\mathcal{A}$的随机预言查询交互如下：

(1)随机预言H_1查询：敌手$\mathcal{A}$输入(m,ω)，若H_1^{list}已有记录(m,ω,r)，$\mathcal{C}$输出r给敌手$\mathcal{A}$；否则，算法$\mathcal{C}$随机选$r\in Z_q^*$，把(m,ω,r)添加到H_1^{list}中，并将r返给敌手$\mathcal{A}$。

(2)随机预言H_2查询：敌手$\mathcal{A}$输入R，若H_2^{list}已有记录(R,h_2)，$\mathcal{C}$输出h_2给敌手$\mathcal{A}$；否则，$\mathcal{C}$随机选$h_2\in\{0,1\}^{l_0+l_1}$，把(R,h_2)添加到H_2^{list}中，并将h_2返给敌手$\mathcal{A}$。

(3)随机预言H_3查询：敌手$\mathcal{A}$输入(t,T)，若H_3^{list}已有记录(t,T,h_3)，$\mathcal{C}$输出h_3给敌手$\mathcal{A}$；否则，$\mathcal{C}$随机选$h_3\in Z_q^*$，把(t,T,h_3)添加到H_3^{list}中，并将h_3返给敌手$\mathcal{A}$。

(4)随机预言H_4查询：敌手$\mathcal{A}$输入(D,E,F,T)，若H_4^{list}已有记录(D,E,F,T,h_4)，$\mathcal{C}$输出h_4给敌手$\mathcal{A}$；否则，$\mathcal{C}$随机选$h_4\in Z_q^*$，把(D,E,F,T,h_4)添加到H_4^{list}中，并将h_4返给敌手$\mathcal{A}$。

(5)随机预言H_5查询：敌手$\mathcal{A}$输入(T,PK_i)，若H_5^{list}已有记录(T,PK_i,h_5)，$\mathcal{C}$输出h_5给敌手$\mathcal{A}$；否则，$\mathcal{C}$随机选$h_5\in Z_q^*$，把(T,PK_i,h_5)添加到H_5^{list}中，并将h_5返给敌手$\mathcal{A}$。

(6)随机预言H_6查询：敌手$\mathcal{A}$输入(X,PK_j,X')，若H_6^{list}已有记录(X,PK_j,X',h_6)，$\mathcal{C}$输出h_6给敌手$\mathcal{A}$；否则，$\mathcal{C}$随机选$h_6\in Z_q^*$，把(X,PK_j,X',h_6)添加到H_6^{list}中，并将h_6返给敌手$\mathcal{A}$。

算法$\mathcal{C}$维护2个初始为空的列表K^{list}和R^{list}来分别存储公私钥对和重加密密钥。

查询阶段1，敌手$\mathcal{A}$可进行各种查询，$\mathcal{C}$作如下响应：

(1)诚实用户公钥查询$\mathcal{O}_u(i)$：随机选取$s_i\in Z_q^*$，使用Coron方法[26]抛硬币$c_i\in\{0,1\}$，得到1的概率是ρ，得到0的概率是$1-\rho$。若$c_i=1$，则$\text{PK}_i=g^{s_i}$；若$c_i=0$，则$\text{PK}_i=g^{as_i}$。把(PK_i,s_i,c_i)更新到K^{list}，并将PK_i返回给$\mathcal{A}$。

(2)毁坏用户私钥查询$\mathcal{O}_c(i)$：随机选取$s_i\in Z_q^*$，$\text{PK}_i=g^{s_i},c_i=-$。把$(\text{PK}_i,s_i,c_i)$更新到$K^{\text{list}}$，并将$(\text{PK}_i,s_i)$返回给$\mathcal{A}$。

(3)重加密密钥查询$\mathcal{O}_{rk}(\text{PK}_i,\text{PK}_j,t)$：若表$R^{\text{list}}$中存在$\left(\text{PK}_i,\text{PK}_j,t,((\text{RK}_{i\to j}^{(1)},z_\mu,f(z_\mu),\text{RK}_{i\to j}^{(3)}),\mu=\{1,2,\cdots,k\}),\kappa,\tau\right)$，则返回重加密密钥$((\text{RK}_{i\to j}^{(1)},z_\mu,f(z_\mu),\text{RK}_{i\to j}^{(3)}),\mu=\{1,2,\cdots,k\})$；否则从$K^{\text{list}}$中得到$(\text{PK}_i,s_i,c_i)$和$(\text{PK}_j,s_j,c_j)$，生成重加密密钥：

(a)若$c_i=1$或$c_i=-$，随机选取$x_i\in Z_q^*$，然后计算$\text{RK}_{i\to j}^{(1)}=g^{x_i}=X_i,\kappa=H_6(X_i,\text{PK}_j,(\text{PK}_j)^{x_i})$，$\text{RK}_{i\to j}^{(2)}=\frac{\kappa}{\text{SK}_i\cdot H_3(t,g^t)}=\frac{\kappa}{s_i\cdot H_3(t,T)}$，$\text{RK}_{i\to j}^{(3)}=t+\text{SK}_i\cdot H_5(g^t,\text{PK}_i)=t+s_i\cdot H_5(T,\text{PK}_i)$，构造多项式$f(x)=\phi_0+\phi_1x+\phi_2x^2+\cdots+\phi_{k-1}x^{k-1}$，其中$\phi_0=\text{RK}_{i\to j}^{(2)}$，$\phi_1,\phi_2,\cdots,\phi_{k-1}\in Z_q^*$，然后再随机选取$z_1,z_2,\cdots,z_k$并计算$f(z_1),f(z_2),\cdots,f(z_k)$，将$\left(\text{PK}_i,\text{PK}_j,t,\left((\text{RK}_{i\to j}^{(1)},z_\mu,f(z_\mu),\text{RK}_{i\to j}^{(3)}),\mu=\{1,2,\cdots,k\}\right),\kappa,\tau=1\right)$更新到$R^{\text{list}}$，并将$\text{RK}_{i\to j}(z_\mu)=((\text{RK}_{i\to j}^{(1)},z_\mu,f(z_\mu),\text{RK}_{i\to j}^{(3)}),\mu=\{1,2,\cdots,k\})$返回给$\mathcal{A}$；

(b)若$c_i=0$且$c_j\in\{0,1\}$，随机选取$\text{RK}_{i\to j}^{(1)},\text{RK}_{i\to j}^{(2)},\text{RK}_{i\to j}^{(3)},\kappa\in Z_q^*$，构造多项式$f(x)=\phi_0+\phi_1x+\phi_2x^2+\cdots+\phi_{k-1}x^{k-1}$，其中$\phi_0=\text{RK}_{i\to j}^{(2)}$，$\phi_1,\phi_2,\cdots,\phi_{k-1}\in Z_q^*$，再随机选取$z_1,z_2,\cdots,z_k$并计算$f(z_1),f(z_2),\cdots,f(z_k)$，将$\left(\text{PK}_i,\text{PK}_j,t,\left((\text{RK}_{i\to j}^{(1)},z_\mu,f(z_\mu),\text{RK}_{i\to j}^{(3)}),\mu=\{1,2,\cdots,k\}\right),\kappa,\tau=0\right)$更新到$R^{\text{list}}$，并将$\text{RK}_{i\to j}(z_\mu)=\left((\text{RK}_{i\to j}^{(1)},z_\mu,f(z_\mu),\text{RK}_{i\to j}^{(3)})\right)$返回给$\mathcal{A}$；

(c)若$c_i=0$且$c_j=-$，中止游戏。

(4)重加密查询$\mathcal{O}_{\text{re}}(\text{PK}_i,\text{PK}_j,C_i)$：验证$g^{\text{RK}_{i\to j}^{(3)}}\stackrel{?}{=}T\cdot\text{PK}_i^{H_5(T,\text{PK}_i)}$，若不成立则输出无效标志；否则进一步验证$\text{PK}_i^{\mathcal{S}}\stackrel{?}{=}E\cdot D^{H_4(D,E,F,T)}$，若不成立则输出无效标志；以上两个验证通过之后，可以用如下方法得到重加密密文C_j：

(a)若$c_i=0$且$c_j=-$，从H_1^{list}中得到(m,ω,r)，检查$\left(\text{PK}_i^{H_3(t,T)}\right)^r\stackrel{?}{=}D$，从$R^{\text{list}}$中查找$\left(\text{PK}_i,\text{PK}_j,t,\left((\text{RK}_{i\to j}^{(1)},*,*,*),\mu=\{0,1,\cdots,k\}\right),\kappa,\tau=-\right)$，如果不存在则随机选取$x_i\in Z_q^*$，然后计算$\text{RK}_{i\to j}^{(1)}=g^{x_i}=X_i,\kappa=H_6(X_i,\text{PK}_j,(\text{PK}_j)^{x_i})$，更新$(\text{PK}_i,\text{PK}_j,$

$t,\left((\mathrm{RK}_{i\to j}^{(1)},*,*,*),\mu=\{1,2,\cdots,k\}\right),\kappa,\tau=-\big)$，计算$C_j^1=g^{r\cdot\kappa},C_j^2=\mathrm{RK}_{i\to j}^{(1)},C_j^3=F=H_2\left(g^r\right)\oplus(m||\omega)$，返回重加密密文$C_j=\left(C_j^1,C_j^2,C_j^3\right)$；

(b)对于其他的c_i和c_j，可以通过R^{list}找到$\mathrm{RK}_{i\to j}\left(z_\mu\right)$，并通过算法$C_j=\mathrm{Rencryption}(C_i,\mathrm{RK}_{i\to j}(z_\mu),\mathrm{param})$得到重加密密文，并返回给$\mathcal{A}$。

(5)原始密文解密查询$\mathcal{O}_{d_2}(\mathrm{PK}_i,C_i)$：$\mathcal{C}$查询$K^{\mathrm{list}}$得到$(\mathrm{PK}_i,s_i,c_i)$，计算$\mathrm{PK}_i^{\mathcal{S}}\stackrel{?}{=}E\cdot D^{H_4(D,E,F,T)}$，若不成立则输出无效标志$\perp$；否则，若$c_i=1$或$c_i=-$，$\mathcal{C}$运行$\mathrm{Decryption2}(C_i,\mathrm{SK}_i,t,\mathrm{param})$并将结果返回$m$给$\mathcal{A}$；若$c_i=0$并存在$(m,\omega,r)\in H_1^{\mathrm{list}}$、$(R,h_2)\in H_2^{\mathrm{list}},(t,T,h_3)\in H_3^{\mathrm{list}}$满足$(\mathrm{PK}_i)^{r\cdot h_3}=D$，$h_2\oplus(m||\omega)=F,g^r=R$，则返回$m$给$\mathcal{A}$。

(6)重加密密文解密查询$\mathcal{O}_{d_1}(\mathrm{PK}_j,C_j)$：查找$R^{\mathrm{list}}$是否存在$\Big(\mathrm{PK}_i,\mathrm{PK}_j,t,\left((\mathrm{RK}_{i\to j}^{(1)},z_\mu,f\left(z_\mu\right),\mathrm{RK}_{i\to j}^{(3)}),\mu=\{1,2,\cdots,k\}\right),\kappa,\tau=0\Big)$：

(a)若存在，计算$D=\left(C_j^1\right)^{\frac{1}{\sum_{\mu=1}^k f(z_\mu)\prod_{\nu=1,\nu\neq\mu}^k\frac{0-z_\mu}{z_\mu-z_\nu}}}=\left(C_j^1\right)^{\frac{s_i\cdot H_3(t,T)}{\kappa}}$，查找$(m,\omega,r)\in H_1^{\mathrm{list}},(R,h_2)\in H_2^{\mathrm{list}},(t,T,h_3)\in H_3^{\mathrm{list}}$是否满足$(\mathrm{PK}_i)^{r\cdot h_3}=D,h_2\oplus(m||\omega)=C_j^3,g^r=R$，若满足返回$m$给$\mathcal{A}$，若不满足则返回无效标志$\perp$；

(b)若不存在，则查找$(m,\omega,r)\in H_1^{\mathrm{list}}$，$(R,h_2)\in H_2^{\mathrm{list}},(g^{x_i},\mathrm{PK}_j,(\mathrm{PK}_j)^{x_i},h_6)\in H_6^{\mathrm{list}}$，验证是否满足$g^r=R,g^{rh_6}=C_j^1$，$h_2\oplus(m||\omega)=F$，若满足则返回$m$给$\mathcal{A}$，否则返回无效标志$\perp$。

挑战阶段，阶段1结束之后，敌手$\mathcal{A}$输出目标公钥PK_{i^*}，条件信息t^*以及两个等长的明文消息$m_0,m_1\in\{0,1\}^{l_0}$，算法$\mathcal{C}$从K^{list}得到$(\mathrm{PK}_i^*,s_i^*,c_i^*)$并随机取$\alpha\in\{0,1\}$，按如下方式生成挑战密文：

(1)若c_i^*=1，中止游戏；

(2)若c_i^*=0，计算$T^*=g^{t^*},D^*=(g^b)^{s_i^*\cdot H_3(t^*,T^*)}$；

(3)随机取$e^*,\mathcal{S}^*\in Z_q^*,E^*=\left(g^b\right)^{-s_i^*\cdot H_3(t^*,T^*)\cdot e^*}\cdot(g^a)^{-s_i^*\cdot\mathcal{S}^*}$；

(4)随机取$F^*\in\{0,1\}^{l_0+l_1},H_4\left(D^*,E^*,F^*,T^*\right)=e^*$；

(5)随机取$\omega^*\in\{0,1\}^{l_1},H_1\left(m_\alpha,\omega^*\right)=\dfrac{b}{a}$，$H_2\left(g^{\frac{b}{a}}\right)=(m_\delta||\omega^*)\oplus F^*$；

(6)将$C^*=(D^*,E^*,F^*,T^*,\mathcal{S}^*)$作为挑战密文发给$\mathcal{A}$。

显然，挑战密文C^*与实际密文具有相同的分布。不妨设$u^*\triangleq\dfrac{\mathcal{S}^*}{H_3\left(t^*,T^*\right)}-\dfrac{b}{a}e^*,r^*\triangleq\dfrac{b}{a}$，可以得到：

$$\begin{aligned}E^*&=(g^b)^{-s_i^*\cdot H_3(t^*,T^*)\cdot e^*}(g^a)^{s_i^*\cdot\mathcal{S}^*}\\&=(g^a)^{-s_i^*\cdot\frac{b}{a}H_3(t^*,T^*)\cdot e^*}(g^a)^{s_i^*\cdot H_3(t^*,T^*)\cdot\frac{\mathcal{S}^*}{H_3(t^*,T^*)}}\\&=(g^a)^{s_i^*\cdot H_3(t^*,T^*)(\frac{\mathcal{S}^*}{H_3(t^*,T^*)}-\frac{b}{a}e^*)}\\&=(\mathrm{PK}_i^*)^{u^*H_3(t^*,T^*)}\end{aligned}\tag{6}$$

$$\begin{aligned}D^*&=(g^b)^{s_i^*\cdot H_3(t^*,T^*)}=(g^{\frac{1}{a}})^{s_i^*\cdot H_3(t^*,T^*)\cdot ab}\\&=(\mathrm{PK}_i^*)^{r^*H_3(t^*,T^*)}\end{aligned}\tag{7}$$

$$F^*=H_2\left(g^{\frac{b}{a}}\right)\oplus(m_\alpha||\omega^*)=H_2\left(r^*\right)\oplus(m_\alpha||\omega^*)\tag{8}$$

$$\begin{aligned}\frac{\mathcal{S}^*}{H_3\left(t^*,T^*\right)}&=\frac{\mathcal{S}^*}{H_3\left(t^*,T^*\right)}-\frac{b}{a}e^*+\frac{b}{a}e^*\\&=u^*+r^*H_4\left(D^*,E^*,F^*,T^*\right)\end{aligned}\tag{9}$$

查询阶段2，敌手$\mathcal{A}$继续进行与查询阶段1相同的查询，但要遵循L2- IND- CCA游戏中的限制条件。

猜测阶段，$\mathcal{A}$输出对α的猜测$\alpha'\in\{0,1\}$，算法$\mathcal{C}$从H_2^{list}中随机取(R,h_2)并输出h_2作为DCDH的解。

借助文献[20]的方法先进行随机预言分析，定义如下事件：

AskH_1^*表示对(m_α,ω^*)做过随机预言H_1查询；AskH_2^*表示对$\left(g^{\frac{b}{a}}\right)$做过随机预言$H_2$查询；$\mathrm{AskH}_4^*$表示对$(D^*,E^*,F^*,T^*)$做过随机预言$H_4$查询；Abort表示在模拟过程中，$\mathcal{C}$中止游戏；$E_1$表示对$(m,\omega)$作随机预言$H_1$查询；$E_2$表示对$(g^r)$作随机预言$H_2$查询；$E_3$表示挑战阶段$c_i^*=0$；$E_4$表示重加密密钥查询时$c_i=1$；$E_{\mathrm{Valid}}$表示密文是合法密文。

在E_3和E_4的情况下$\mathcal{C}$不会中止游戏，因此$\Pr[\neg\mathrm{Abort}]\geq\rho^{q_{rk}}(1-\rho)$，当$\rho_{\mathrm{OPT}}=q_{\mathrm{rk}}/(1+q_{\mathrm{rk}})$时得到$\Pr\left[\neg\mathrm{Abort}\right]=\dfrac{1}{\mathrm{e}(1+q_{\mathrm{rk}})}$，e是自然对数的底。分析解密的模拟过程，在没有发生E_1或E_2的情况下，合法密文的概率$\Pr\left[E_{\mathrm{Valid}}|(\neg E_1\vee\neg E_2)\right]\leq\Pr[E_{\mathrm{Valid}}|\neg E_1]+\Pr[E_{\mathrm{Valid}}|\neg E_2]\leq\dfrac{2}{q}$，整个模拟过程中发生的$E_{\mathrm{Valid}}|(\neg E_1\vee\neg E_2)$事件用$E_{\mathrm{Derr}}$表示，由于敌手最多有$q_{d_2}$个解密查询，所以$\Pr\left[E_{\mathrm{Derr}}\right]\leq q_{d_2}\left(\dfrac{q_{H_1}+q_{H_2}}{2^{l_0+l_1}}+\dfrac{2}{q}\right)$。$(\mathrm{AskH}_1^*\vee\mathrm{AskH}_2^*\vee\mathrm{AskH}_4^*\vee E_{\mathrm{Derr}})|\neg\mathrm{Abort}$事件用$E_{\mathrm{Err}}$表示，当$E_{\mathrm{Err}}$没有发生时，由于随机预言$H_2$的输出具有随机性，所以敌手$\mathcal{A}$对$\alpha$的猜测优势不会大于$\dfrac{1}{2}$，其猜对$\alpha$的概率$\Pr\left[\alpha'=\alpha\right]=\Pr\left[\alpha'=\alpha|\neg E_{\mathrm{Err}}\right]\cdot\Pr\left[\neg E_{\mathrm{Err}}\right]+\Pr\left[\alpha'=\alpha|E_{\mathrm{Err}}\right]\Pr\left[E_{\mathrm{Err}}\right]=\dfrac{1}{2}(1+\Pr\left[E_{\mathrm{Err}}\right])$，$\Pr\left[\alpha'=\alpha\right]\geq\Pr\left[\alpha'=\alpha|\neg E_{\mathrm{Err}}\right]\Pr\left[\neg E_{\mathrm{Err}}\right]\geq\dfrac{1}{2}(1-\Pr\left[E_{\mathrm{Err}}\right])$。L2- IND- CCA游戏中敌手获胜的优势定义$\epsilon=|2\Pr\lceil\alpha=\alpha'\rfloor-1|\leq\Pr\left[E_{\mathrm{Err}}\right]$。由于$\mathcal{C}$随机选取$\omega\in$

$\{0,1\}^{l_1}$，所以$\Pr[\mathrm{AskH}_1^*]\le q_{H_1}/2^{l_1}$，可得$\Pr[\mathrm{AskH}_2^*]\ge \Pr[\neg\mathrm{Abort}]\cdot\epsilon-\Pr[\mathrm{AskH}_1^*]-\Pr[\mathrm{AskH}_4^*]-\Pr[\mathrm{E_{Derr}}]\ge \dfrac{\epsilon}{\mathrm{e}(1+q_{\mathrm{rk}})}-\dfrac{q_{H_1}}{2^{l_1}}-\dfrac{q_{H_4}}{2^{l_0+l_1}}-q_{d_2}\left(\dfrac{q_{H_1}+q_{H_2}}{2^{l_0+l_1}}+\dfrac{2}{q}\right)$。若$\mathrm{AskH}_2^*$事件发生，则$\mathcal{C}$解决DCDH问题的优势$\epsilon'\ge\dfrac{1}{q_{H_2}}\Pr[\mathrm{AskH}_2^*]\ge\dfrac{1}{q_{H_2}}\left(\dfrac{\epsilon}{\mathrm{e}(1+q_{\mathrm{rk}})}-\dfrac{q_{H_1}}{2^{l_1}}-\dfrac{q_{H_4}}{2^{l_0+l_1}}-q_{d_2}\left(\dfrac{q_{H_1}+q_{H_2}}{2^{l_0+l_1}}+\dfrac{2}{q}\right)\right)$。 证毕

定理 2 TB-CAPRE方案对于对于重加密密文是基于CDH假设，随机预言模型下IND-CCA安全的。

证明：利用定理1的方法，假设存在概率多项式时间敌手$\mathcal{A}$在随机预言模型下能以概率ε的优势攻破本方案，那可以构建一个算法$\mathcal{C}$来模拟L1- IND- CCA游戏中的挑战者，以概率ε'的优势解决CDH问题，且$\varepsilon'\ge\frac{1}{q_{H_2}}\left(\dfrac{\epsilon}{\mathrm{e}(2+q_{\mathrm{rk}})}-\dfrac{q_{H_1}}{2^{l_1}}-q_{d_2}\left(\dfrac{q_{H_1}+q_{H_2}}{2^{l_0+l_1}}+\dfrac{2}{q}\right)\right)$，也就是说给定$\{g,g^a,g^b\}\in G,a,b\in Z_q^*$作为算法$\mathcal{C}$的输入，$\mathcal{C}$的目标是得到$g^{ab}\in G$。

查询阶段1，与定理1中的描述相同。

挑战阶段，敌手$\mathcal{A}$输出授权者的公钥$\mathrm{PK}_{i'}$、被授权者公钥PK_i^*、条件信息t以及两个等长的明文消息$m_0,m_1\in\{0,1\}^{l_0}$，算法$\mathcal{C}$从K^{list}得到$(\mathrm{PK}_{i'},s_{i'},c_{i'})$和$(\mathrm{PK}_i^*,s_i^*,c_i^*)$，若$c_i^*=0$且$c_{i'}\in\{1,-\}$，随机取$\alpha\in\{0,1\}$，按如下方式生成挑战密文：

(1)定义$\kappa=\dfrac{\kappa_1}{as_{i'}\cdot H_3(t,g^t)},\kappa_1\in Z_q^*$，$\mathcal{C}$不知道$a$的值；

(2)随机取$\omega^*\in\{0,1\}^{l_1},H_1(m_\alpha,\omega^*)=ab$；

(3)随机取$F^*\in\{0,1\}^{l_0+l_1},F^*=H_2\left(g^{ab}\right)\oplus(m_\delta||\omega^*)$；

(4)随机选取$x_i\in Z_q^*$，计算$X_i=g^{x_i},\kappa_1=H_6(X_i,\mathrm{PK}_i^*,(\mathrm{PK}_i^*)^{x_i})$；

(5)计算$D'^*=(g^b)^{\frac{\kappa_1}{s_{i'}\cdot H_3(t,g^t)}}$；

(6)返回挑战密文$C^*=(D'^*,X_i,F^*)$给$\mathcal{A}$。

显然，挑战密文C^*与重加密算法生成的密文具有相同的分布：不妨设$r^*\triangleq ab$，则$D'^*=(g^b)^{\frac{\kappa_1}{s_{i'}\cdot H_3(t,g^t)}}=(g^{ab})^{\frac{\kappa_1}{as_{i'}\cdot H_3(t,g^t)}}=g^{r^*\kappa},F^*=H_2\left(g^{ab}\right)\oplus(m_\delta||\omega^*)=H_2\left(g^{r^*}\right)\oplus(m_\delta||\omega^*)$。

查询阶段2，敌手$\mathcal{A}$继续进行与查询阶段1相同的查询，但要遵循L1- IND- CCA游戏中的限制条件。

猜测阶段，$\mathcal{A}$输出对α的猜测$\alpha'\in\{0,1\}$，算法$\mathcal{C}$从H_2^{list}中随机取(R,h_2)并输出h_2作为CDH的解。定义如下事件：

AskH_1^*表示对(m_α,ω^*)做过随机预言H_1查询；AskH_2^*表示对$\left(g^{ab}\right)$做过随机预言H_2查询；Abort表示在模拟过程中，$\mathcal{C}$中止游戏；E_1表示对(m,ω)作随机预言H_1查询；E_2表示对(g^r)作随机预言H_2查询；E_3表示挑战阶段$c_i^*=0$且$c_i'\ne0$；E_4表示重加密密钥查询时$c_i=1$；E_{Valid}表示密文是合法密文。

在E_3和E_4的情况下$\mathcal{C}$不会中止游戏，因此$\Pr[\neg\mathrm{Abort}]\ge\rho^{1+q_{\mathrm{rk}}}(1-\rho)$，当$\rho_{\mathrm{OPT}}=\dfrac{1+q_{\mathrm{rk}}}{2+q_{\mathrm{rk}}}$时得到$\Pr[\neg\mathrm{Abort}]=\dfrac{1}{\mathrm{e}(2+q_{\mathrm{rk}})}$，e是自然对数的底。解密的模拟过程与定理1的过程类似，由于敌手最多有q_{d_2}个解密查询，所以$\Pr[E_{\mathrm{Derr}}]\le q_{d_2}\left(\dfrac{q_{H_1}+q_{H_2}}{2^{l_0+l_1}}+\dfrac{2}{q}\right)$。$(\mathrm{AskH}_1^*\vee\mathrm{AskH}_2^*\vee E_{\mathrm{Derr}})|\neg\mathrm{Abort}$事件用$E_{\mathrm{Err}}$表示，由原密文安全性分析过程，可得$\Pr[\mathrm{AskH}_2^*]\ge\Pr[\neg\mathrm{Abort}]\cdot\epsilon-\Pr[\mathrm{AskH}_1^*]-\Pr[E_{\mathrm{Derr}}]\ge\dfrac{1}{q_{H_2}}\left(\dfrac{\epsilon}{\mathrm{e}(2+q_{\mathrm{rk}})}-\dfrac{q_{H_1}}{2^{l_1}}-q_{d_2}\left(\dfrac{q_{H_1}+q_{H_2}}{2^{l_0+l_1}}+\dfrac{2}{q}\right)\right)$。若$\mathrm{AskH}_2^*$事件发生，则$\mathcal{C}$解决CDH问题的优势$\epsilon'\ge\dfrac{1}{q_{H_2}}\Pr[\mathrm{AskH}_2^*]\ge\dfrac{1}{q_{H_2}}\left(\dfrac{\epsilon}{\mathrm{e}(2+q_{\mathrm{rk}})}-\dfrac{q_{H_1}}{2^{l_1}}-q_{d_2}\left(\dfrac{q_{H_1}+q_{H_2}}{2^{l_0+l_1}}+\dfrac{2}{q}\right)\right)$。 证毕

4.3 效率分析

下面对本方案进行计算效率方面的分析，表1为本方案与其他方案计算效率和特点的比较。由于

表 1 计算效率与特点对比

对比项目	文献[22]	文献[20]	文献[23]	本文方案
KeyGen	e	$2e$	e	e
Enc	$2e$	$6e$	$2e+p$	$4e$
ReKeyGen	0	$2e$	ne	$3e$
ReEnc	ke	$5e$	$ke+kp$	$5ke$
Dec1	e	$6e$	e	$3e$
Dec2	e	$5e$	$e+p$	$4e$
条件	×	√	×	√
门限	√	×	√	√

方案的加解密、重加密和条件匿名等功能主要是由指数和哈希运算实现的，因此表2统计了本方案各过程中包含的指数运算和哈希运算计算量。其中p,e,h分别表示双线性对运算、群G中的指数运算以及哈希运算，系数为运算次数，k是门限值，n是代理服务节点总数量。

表 2　本方案计算量

	ReKeyGen	Encrypt	ReEncrypt	Dcrypt1	Decypt2
计算量	$3e+3h$	$2e+4h$	$5ke+2kh$	$3e+2h$	$4e+3h$

从表1可以看出，除了重加密过程(ReEnc)外，本文方案的性能与文献[20]相当，但本文方案使用门限的方法来解决重加密过程中信任过度集中和单点失效问题，因而ReEnc过程计算效率会有所下降。文献[23]的方案是基于双线性对且不支持条件重加密，考虑到双线性对运算效率一般只有指数运算的一半，在代理重加密服务节点数量较多时((k,n)门限中的n较大)，本文方案的计算效率要优于文献[23]。文献[22]的计算效率比较高，主要原因是在重加密密钥生成过程(ReKeyGen)中没有进行指数运算和双线性对运算，而是直接将授权者和被授权者的私钥信息发送给一个第三方来进行简单的代数运算生成重加密密钥，这带来了私钥泄露的风险，同时该方案也不支持条件重加密，只能达到IND-CPA安全。本文方案实现了随机预言下的IND-CCA安全，从表2可以看到，为了实现IND-CCA安全和条件信息的匿名化，本方案的计算量中引入了一定的哈希运算。由于哈希运算的效率比指数运算要高得多，因此本方案在提高安全性的同时并没有增加太大的计算量。

5　结束语

本文提出了无双线对的门限条件匿名代理重加密方案的定义及安全模型，该方案在CDH问题困难性假设下能够满足随机预言模型中的IND-CCA安全，本文方案在原始密文和重加密密钥中加入条件信息，能够对重加密密文进行细粒度的控制。方案在重加密密钥生成和重加密过程中引入了门限技术，为重加密应用于分布式系统提供了基础。方案不依赖双线性对操作，可以对敏感的条件信息进行匿名化处理，并能够防止代理服务节点与被授权者合谋恢复授权者的私钥信息。方案目前只能支持确定的单条件信息，无法实现模糊条件和多条件，这是下一步需要重点解决的问题。

参 考 文 献

[1] BLAZE M, BLEUMER G, and STRAUSS M. Divertible protocols and atomic proxy cryptography[C]. International Conference on the Theory and Application of Cryptographic Techniques, Espoo, Finland, 1998: 127–144. doi: 10.1007/BFb0054122.

[2] CANETTI R and HOHENBERGER S. Chosen-ciphertext secure proxy re-encryption[C]. The 14th ACM Conference on Computer and Communications Security, Alexandria, USA, 2007: 185–194. doi: 10.1145/1315245.1315269.

[3] JIANG M M, HU Y P, WANG B C, *et al*. Lattice-based multi-use unidirectional proxy re-encryption[J]. *Security and Communication Networks*, 2015, 8(18): 3796–3803. doi: 10.1002/sec.1300.

[4] WENG Jian, DENG R H, LIU Shengli, *et al*. Chosen-ciphertext secure bidirectional proxy re-encryption schemes without pairings[J]. *Information Sciences*, 2010, 180(24): 5077–5089. doi: 10.1016/j.ins.2010.08.017.

[5] LIBERT B and VERGNAUD D. Unidirectional chosen-ciphertext secure proxy re-encryption[J]. *IEEE Transactions on Information Theory*, 2011, 57(3): 1786–1802. doi: 10.1109/TIT.2011.2104470.

[6] RAWAL B S. Proxy re-encryption architect for storing and sharing of cloud contents[J]. *International Journal of Parallel, Emergent and Distributed Systems*, 2020, 35(3): 219–235. doi: 10.1080/17445760.2018.1439491.

[7] VIJAYAKUMAR V, PRIYAN M K, USHADEVI G, *et al*. E-health cloud security using timing enabled proxy re-encryption[J]. *Mobile Networks and Applications*, 2019, 24(3): 1034–1045. doi: 10.1007/s11036-018-1060-9.

[8] SU Mang and WANG Liangchen. PreBAC: A novel access control scheme based proxy re-encryption for cloud computing[J]. *KSII Transactions on Internet and Information Systems*, 2019, 13(5): 2754–2767. doi: 10.3837/tiis.2019.05.028.

[9] QIAN Xin, YANG Zhen, WANG Shihui, *et al*. A no-pairing proxy re-encryption scheme for data sharing in untrusted cloud[C]. The 5th International Conference on Artificial Intelligence and Security, New York, USA, 2019: 85–96. doi: 10.1007/978-3-030-24274-9_8.

[10] WANG Xu'an, YANG Xiaoyuan, LI Cong, *et al*. Improved functional proxy re-encryption schemes for secure cloud data sharing[J]. *Computer Science and Information Systems*, 2018, 15(3): 585–614. doi: 10.2298/CSIS171218024W.

[11] 苏铓, 曹梦元, 谢绒娜, 等. 基于代理重加密的物联网云节点授权可信更新机制[J]. 计算机研究与发展, 2018, 55(7): 1479–1487. doi: 10.7544/issn1000-1239.2018.20180056.
SU Mang, CAO Mengyuan, XIE Rongna, *et al*. PRE-TUAN: Proxy re-encryption based trusted update scheme of authorization for nodes on IoT cloud[J]. *Journal of Computer Research and Development*, 2018, 55(7): 1479–1487. doi: 10.7544/issn1000-1239.2018.20180056.

[12] WENG Jian, DENG R H, DING Xuhua, *et al*. Conditional proxy re-encryption secure against chosen-ciphertext attack[C]. The 4th International Symposium on Information, Computer, and Communications Security, Sydney, Australia, 2009: 322–332. doi: 10.1145/1533057.1533100.

[13] ZENG Peng and CHOO K K R. A new kind of conditional proxy re-encryption for secure cloud storage[J]. *IEEE Access*, 2018, 6: 70017–70024. doi: 10.1109/ACCESS.2018.2879479.

[14] SUN Maosheng, GE Chunpeng, FANG Liming, *et al*. Conditional proxy broadcast re-encryption with fine grain policy for cloud data sharing[J]. *International Journal of Embedded Systems*, 2019, 11(2): 115–124. doi: 10.1504/IJES.2019.098296.

[15] HUANG Qinlong, YANG Yixian, and FU Jingyi. PRECISE: Identity-based private data sharing with conditional proxy re-encryption in online social networks[J]. *Future Generation Computer Systems*, 2018, 86: 1523–1533. doi: 10.1016/j.future.2017.05.026.

[16] LIU Yepeng, REN Yongjun, GE Chunpeng, *et al*. A CCA-secure multi-conditional proxy broadcast re-encryption scheme for cloud storage system[J]. *Journal of Information Security and Applications*, 2019, 47: 125–131. doi: 10.1016/j.jisa.2019.05.002.

[17] 徐洁如, 陈克非, 沈忠华, 等. 无双线性对的基于证书多域条件代理重加密方案[J]. 密码学报, 2018, 5(1): 55–67. doi: 10.13868/j.cnki.jcr.000218.
XU Jieru, CHEN Kefei, SHEN Zhonghua, *et al*. Pairing-free certificate-based multi-domain conditional proxy re-encryption scheme[J]. *Journal of Cryptologic Research*, 2018, 5(1): 55–67. doi: 10.13868/j.cnki.jcr.000218.

[18] LI Jiguo, ZHAO Xuexia, ZHANG Yichen, *et al*. Provably secure certificate-based conditional proxy re-encryption[J]. *Journal of Information Science and Engineering*, 2016, 32(4): 813–830.

[19] LU Yang. Efficient certificate-based proxy re-encryption scheme for data sharing in public clouds[J]. *KSII Transactions on Internet and Information Systems*, 2015, 9(7): 2703–2718. doi: 10.3837/tiis.2015.07.021.

[20] PAUL A, SELVI S S D, and RANGAN C P. A provably secure conditional proxy re-encryption scheme without pairing[J]. *IACR Cryptology ePrint Archive*, 2019, 2019: 1135.

[21] JAKOBSSON M. On quorum controlled asymmetric proxy re-encryption[C]. The 2nd International Workshop on Practice and Theory in Public Key Cryptography, Kamakura, Japan, 1999: 112–121. doi: 10.1007/3-540-49162-7_9.

[22] PATIL S M and PURUSHOTHAMA B R. Non-transitive and collusion resistant quorum controlled proxy re-encryption scheme for resource constrained networks[J]. *Journal of Information Security and Applications*, 2020, 50: 102411. doi: 10.1016/j.jisa.2019.102411.

[23] CHEN Xi, LIU Yong, LI Yong, *et al*. Threshold proxy re-encryption and its application in blockchain[C]. The 4th International Conference on Cloud Computing and Security, Haikou, China, 2018: 16–25. doi: 10.1007/978-3-030-00015-8_2.

[24] PATIL S M and PURUSHOTHAMA B R. RSA-based collusion resistant quorum controlled proxy re-encryption scheme for distributed secure communication[C]. The 15th International Conference on Distributed Computing and Internet Technology, Bhubaneswar, India, 2019: 349–363. doi: 10.1007/978-3-030-05366-6_29.

[25] CHOW S S M, WENG Jian, YANG Yanjiang, *et al*. Efficient unidirectional proxy re-encryption[C]. The 3rd International Conference on Cryptology in Africa, Stellenbosch, South Africa, 2010: 316–332. doi: 10.1007/978-3-642-12678-9_19.

[26] CORON J S. On the exact security of full domain hash[C]. The 20th Annual International Cryptology Conference, Santa Barbara, USA, 2000: 229–235.

李兆斌：男，1977年生，副研究员，研究方向为下一代网络安全、密码算法实现与测评.

赵 洪：男，1978年生，讲师，研究方向为量子密码、密码协议设计与实现.

魏占祯：男，1971年生，研究员级高工，研究方向为密码测评、软件定义网络安全.

责任编辑：余 蓉