支持关键字搜索的无证书密文等值测试加密方案

张玉磊[①] 陈文娟*[②] 张永洁[②] 张雪微[②] 王彩芬[③]

[①](西北师范大学 兰州 730070)

[②](甘肃卫生职业学院 兰州 730000)

[③](深圳技术大学 深圳 518118)

摘 要：公钥加密等值测试(PKEET)可以实现云环境下不同公钥加密数据之间的密文等值比较，即不对密文解密的情况下测试两个密文对应的明文是否一致。但是，密文等值测试加密不提供关键字密文搜索功能。已有密文等值测试加密方案直接以消息生成陷门作为等值测试的凭证，测试的准确度不高，搜索效率较低。针对此问题，该文首先提出了支持关键字搜索的无证书密文等值测试加密(CLEETS)方案。方案通过关键字检索判断是否包含自己需要的信息，根据判断结果选择执行等值测试，从而避免无效测试。然后，在随机预言机模型下证明该方案满足适应性选择关键词不可区分性。最后，对方案进行功能和效率对比。对比结果表明，该文方案的计算代价略高，但是方案在密文等值测试加密中实现了关键字的检索功能，弥补了效率低的不足。

关键词：无证书公钥密码；可搜索加密；密文等值测试

中图分类号：TP309 **文献标识码**：A **文章编号**：1009-5896(2020)11-2713-07

DOI: 10.11999/JEIT190752

Certificateless Puklic Key Encryption With Equality Test of Supporting Keyword Search

ZHANG Yulei[①] CHEN Wenjuan[②] ZHANG Yongjie[②]
ZHANG Xuewei[②] WANG Caifen[③]

[①](*Northwest Normal University*, *Lanzhou* 730070, *China*)

[②](*Gansu Health Vocational*, *Lanzhou* 730000, *China*)

[③](*Shenzhen Technology University*, *Shenzhen* 518118, *China*)

Abstract: Public Key Encryption with Equality Test (PKEET) is an important method to achieve the equality test of ciphertexts which are generated by the different public key aiming to the same plaintext in cloud environment. In other words, it can tests the plaintext corresponding to the two ciphertext's equivalence without decrypting the ciphertext, but does not supply the searchable function. Nowadays, the existing PKEET scheme takes directly the message to generate a trapdoor as the proof of equality test, which has low test accuracy and search efficiency. To solve the above problems, a certificateless public key encryption with equality test scheme supporting keyword search (CertificateLess Equality test EncrypTion with keyword Search, CLEETS) is proposed. The scheme determines whether it contains information needed by the user through the keyword search, then performs the equality test according to the search result, which can avoid invalid test. Then, it is proved that the scheme satisfies the indistinguishability of adaptive selection of keywords under the random oracle model. Finally, the comparison analyses of function and efficiency are performed. The results indicate the computation cost of CLEETS scheme is less efficient. Fortunately, it can realizes the function of keyword search in encryption with equality test, which can remedies the inefficiency.

Key words: Certificateless public key cryptography; Encryption with keyword search; Equality test

收稿日期：2019-09-29；改回日期：2020-04-23；网络出版：2020-05-28

*通信作者： 陈文娟 497604722@qq.com

基金项目：国家自然科学基金(61662069)，甘肃省高等学校科研项目(2017A-003, 2018A-207)

Foundation Items: The National Natural Science Foundation of China(61662069), The Higher Educational Scientific Research Foundation of Gansu Province (2017A-003, 2018A-207)

1 引言

云计算[1]能够提供广泛的计算、分析、存储、部署和支持应用程序等服务，帮助用户降低成本。考虑到云数据的敏感性[2]，引入密码系统来加密私有数据。可搜索加密[3]既可以保护用户的隐私，又支持搜索加密数据，在云数据存储中有广泛应用。

2004年，Boneh等人[4]提出了第1个公钥可搜索加密方案。该方案既支持密文检索，又能实现用户数据共享。传统公钥可搜索加密中，证书颁发机构是公钥基础设施系统的核心部分，负责发布和管理用户的证书，这样带来了证书管理问题。基于身份的公钥可搜索加密方案[5]可以解决传统公钥可搜索加密的证书管理问题，但是又引入了密钥托管问题。2014年，Peng等人[6]提出了无证书可搜索加密方案。该方案既解决了证书管理问题，又解决了密钥托管问题。2017年，Ma等人[7,8]提出了分别应用于工业互联网和医疗健康网络的无证书可搜索加密方案。但是，以上方案[6–8]只能对使用了相同公钥加密的数据进行操作。云环境中用户可能会使用不同的公钥对数据进行加密，但是，服务器不能直接对不同公钥加密的数据进行比较，因此公钥可搜索加密技术[9,10]存在一定的局限性。

2010年，Yang等人[11]提出了公钥加密等值测试(Public Key Encryption with Equality Test, PKEET)，该技术可以判断不同公钥加密的密文是否是同一明文所产生。也就是说，若C和C'是用两个不同公钥加密的密文，其中$C=\mathrm{Encrypt}(M,\mathrm{PK})$和$C'=\mathrm{Encrypt}(M',\mathrm{PK}')$，该算法通过比较$C=C'$是否成立来判断$M=M'$是否成立。但是，Yang方案不提供授权机制，任何用户都可以进行等值测试，暴露了数据所有者的隐私。2011年，Tang[12]在PKEET中引入授权机制，提出了一个支持细粒度授权的公钥加密等值测试方案。随后，Tang[13,14]又提出了仅指定用户可以进行密文等值测试和抵抗离线消息恢复攻击的密文等值测试方案。2015年，Ma等人[15]提出了支持灵活授权的公钥加密等值测试方案。2016年，Ma[16]提出了身份等值测试方案，该方案简化了证书管理问题并支持用户级授权。2018年，Qu等人[17]提出了无证书密文等值测试加密方案，该方案可以解决传统PKEET方案[18]和身份PKEET方案[19]中的证书管理和密钥托管问题。但是，当前已有的PKEET方案不提供关键字检索功能。如果云服务器存储的密文中根本没有用户需要的信息，而用户只有执行密文等值测试后才能得知，这无形中消耗了大量的网络带宽。

针对上述问题，本文提出了支持关键字搜索的无证书密文等值测试加密(CertificateLess Equality test EncrypTion with keyword Search, CLEETS)方案。方案中用户首先根据关键字生成一个陷门，利用该陷门判断云服务器的密文中是否包含关键字对应的密文信息。如果包含则进行等值测试；否则不进行等值测试，这样，将提高用户的检索效率。在随机预言模型下证明方案满足适应性选择关键词不可区分性。同时将方案与已有的PKEET方案进行功能和效率对比。对比结果表明，虽然本文方案的计算代价略高，但是它在密文等值测试中增加了关键字检索功能，使得方案的适用性更强。

2 CLEETS方案安全模型

2.1 CLEETS方案系统模型

支持关键字搜索的无证书密文等值测试加密(CLEETS)方案包括云服务器、公钥服务器、密钥生成中心(KGC)、数据拥有者和用户5类实体。系统模型如图1所示。

(1) 云服务器：存储密文，并且通过接收者的陷门和授权执行密文搜索与等值测试。

(2) 公钥服务器：存储所有用户的公钥。

(3) 密钥生成中心(KGC)：生成部分私钥。

(4) 数据拥有者：生成、上传密文到云服务器。

(5) 用户：提出搜索和等值测试请求，生成搜索陷门和等值测试陷门。

2.2 CLEETS方案形式化定义

CLEETS方案包括以下算法：

(1) 系统设置：KGC安全参数K，生成系统公共参数PP和主密钥$\hat{g}$。

(2) 生成部分私钥：KGC输入参数PP、主密钥$\hat{g}$和用户的身份，生成用户的部分私钥D_i。

(3) 生成秘密值法：用户输入参数PP，生成用户的秘密值X_i。

(4) 生成私钥：用户输入参数PP、部分私密钥D_i和秘密值X_i，生成用户的(完整)私钥SK_i。

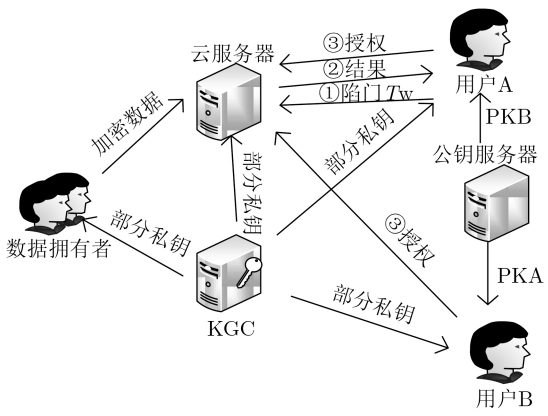

图 1 系统模型图

(5) 生成公钥：用户运行，输入参数PP和秘密值X_i，生成用户的公钥PK_i。

(6) 加密：数据拥有者输入参数PP、关键字w_i、接收者的公钥PK_i和身份ID_i，输出密文C。

(7) 生成搜索陷门：接收者输入参数PP、关键字w_i'和接收者的私钥$\mathrm{SK_R}$，输出搜索陷门T_{W}。

(8) 生成测试陷门：接收者运行，输入参数PP和接收者的私钥$\mathrm{SK_R}$，输出测试陷门td_i。

(9) Test_1搜索测试算法：服务器输入参数PP、服务器私钥$\mathrm{SK_R}$和搜索陷门T_{W}。如果$w_i = w_i'$，输出“1”并执行Test_2算法；否则输出“0”，终止操作。

(10) Test_2等价测试算法：服务器运行，设C_{A}和$\mathrm{td_A}$为接收者A的密文和测试陷门，C_{B}和$\mathrm{td_B}$为接收者B的密文和测试陷门。输入参数PP和两个密文/测试陷门对$(C_{\mathrm{A}}, \mathrm{td_A})$和$(C_{\mathrm{B}}, \mathrm{td_B})$。如果二者为同一消息的密文，则输出“1”；否则输出“0”。

2.3 CLEETS方案安全模型

与文献[8,17]一样，本文定义敌手$\mathrm{A_I}$和$\mathrm{A_{II}}$。其中，敌手$\mathrm{A_I}$模仿恶意用户，用户的公钥可以被任意替换，但是无法获得用户的部分私钥。敌手$\mathrm{A_{II}}$模仿恶意KGC，它不能得到用户的公钥，但是知道主密钥及部分私钥。CLEETS方案的安全性由挑战者C与敌手$\mathrm{A_I}$及$\mathrm{A_{II}}$之间的游戏来完成。

游戏1：挑战者C和$\mathrm{A_I}$之间的游戏过程如下：

(1) 系统建立阶段：C输入安全参数K，输出主密钥s和参数PP。C返回PP给$\mathrm{A_I}$，保存s；

(2) 询问阶段：$\mathrm{A_I}$进行Hash、生成部分私钥、生成私钥、生成公钥、公钥替换、生成搜索陷门、生成测试陷门等询问，C将返回相应值给$\mathrm{A_I}$；

(3) 挑战阶段：$\mathrm{A_I}$输出两个挑战关键字w_0和w_1，$w_0 \neq w_1$且未被$\mathrm{A_I}$询问过。C任意选择$b \in \{0,1\}$，运行加密算法生成密文，发送给$\mathrm{A_I}$；

(4) 猜测阶段：$\mathrm{A_I}$以$b \in \{0,1\}$作为猜测值输出。如果$b' = b$，则$\mathrm{A_I}$获胜。

$\mathrm{A_I}$获胜的优势为$\mathrm{Adv}_{\mathrm{A_I}}(k) = \left|\Pr[b' = b] - \frac{1}{2}\right|$。

游戏2：挑战者C和$\mathrm{A_{II}}$的游戏过程如下：

(1) 系统建立阶段：C输入安全参数K，输出主密钥s和参数PP并返回给$\mathrm{A_{II}}$；

(2) 询问阶段：与游戏1中过程相同，除了不进行部分私钥询问和公钥替换询问；

(3) 挑战阶段：与游戏1中过程相同；

(4) 猜测阶段：$\mathrm{A_{II}}$输出$b' \in \{0,1\}$作为猜测值，如果$b' = b$，则$\mathrm{A_{II}}$获胜。

$\mathrm{A_{II}}$获胜的优势为：$\mathrm{Adv}_{\mathrm{A_{II}}}(k) = \left|\Pr[b' = b] - \frac{1}{2}\right|$。

3 CLEETS方案

3.1 具体方案

(1) 系统设置。输入安全参数K，生成系统参数PP。KGC执行以下过程：

(a)选取两个循环群G_1和G_2，其阶为q，g为G_1的生成元，再选择双线性映射$e: G_1 \times G_1 \to G_2$; (b)随机选择$s \in Z_q^*$作为主密钥并秘密保存，并计算$\hat{g} = g^s$；(c)选择4个抗碰撞的哈希函数$H_1, H_2: \{0,1\}^* \to G_1$, $H_3: \{0,1\}^* \to Z_q^*$, $H_4: G_2 \to Z_q^*$; (d)公开系统参数$\mathrm{PP} = \{G_1, G_2, e, q, g, \hat{g}, H_1, H_2, H_3, H_4\}$。

(2) 生成部分私钥。KGC执行以下步骤生成接收者和云服务器的部分私钥：

(a)输入接收者的身份信息$\mathrm{ID_R} \in \{0,1\}^*$，计算$Q_{\mathrm{R}} = H_1(\mathrm{ID_R})$，计算接收者的部分私钥$D_{\mathrm{S}} = Q_{\mathrm{R}}^s$; (b)输入云服务器的身份信息$\mathrm{ID_S} \in \{0,1\}^*$，计算$Q_{\mathrm{S}} = H_1(\mathrm{ID_S})$，计算云服务器的部分私钥$D_{\mathrm{S}} = Q_{\mathrm{S}}^s$。

(3) 生成秘密值。选择一个随机数作为秘密值。

(a)接收者任意选取$X_{\mathrm{R}} \in Z_q^*$作为其秘密值；(b)云服务器任意选取$X_S \in Z_q^*$作为其秘密值。

(4) 生成私钥。用户输入秘密值和部分私钥，产生自己的完整私钥。

(a)接收者输入$X_{\mathrm{R}} \in Z_q^*$和D_{R}，生成私钥$\mathrm{SK_R} = D_{\mathrm{R}}^{X_{\mathrm{R}}}$；(b)云服务器输入$X_{\mathrm{S}} \in Z_q^*$和$D_{\mathrm{S}}$，生成私钥$\mathrm{SK_S} = {D_{\mathrm{S}}}^{X_{\mathrm{S}}}$。

(5) 生成公钥。输入秘密值生成用户的公钥。

(a)接收者输入秘密值$\mathrm{X_R}$和系统公钥$\hat{g}$，生成接收者的公钥$\mathrm{PK_R} = \hat{g}^{\mathrm{X_R}}$; (b)云服务器输入$X_{\mathrm{S}}$和系统公钥$\hat{g}$，生成接收者的公钥$\mathrm{PK_S} = \hat{g}^{\mathrm{X_S}}$。

(6) 加密。设一组关键字集$W = \{w_i \mid 1 \leq i \leq n\}$。输入PP, $\mathrm{ID_S}$, $\mathrm{ID_R}$, $\mathrm{PK_R}$, $\mathrm{PK_S}$, w_i和消息M，数据拥有者执行以下计算过程：

(a)随机选择$\alpha, \beta \in Z_q^*$，计算$C_{1i} = g^{\alpha}$和$C_{2i} = g^{\beta}$；(b)计算$C_{3i} = e(H_2(w_i)^{\alpha}, \mathrm{PK_R}\mathrm{PK_S})\, e(Q_R{}^{\alpha}, \hat{g})$；(c)计算$C_{4i} = H_3(M)^{\alpha} \cdot H_4(e(\mathrm{PK_R}, Q_{\mathrm{R}})^{\alpha})$。则目标密文$C = \{C_1, C_2, \cdots, C_n\}$，其中$C_i = (C_{1i}, C_{2i}, C_{3i}, C_{4i})$。

(7) 生成搜索陷门。接收者输入PP, w_i'和$\mathrm{SK_R}$，计算出搜索陷门$T_{\mathrm{W}} = H_2(w_i')^{X_{\mathrm{R}}} D_{\mathrm{R}}$。

(8) 生成测试陷门。接收者输入PP和$\mathrm{SK_R}$，接收者计算出测试陷门$T_{\mathrm{d}} = {D_{\mathrm{R}}}^{X_{\mathrm{R}}}$。

(9) Test_1算法。云服务器输入PP, $\mathrm{SK_S}$和T_{W_i}，验证$e(T_{\mathrm{W}}, C_1)\, e\left(H_2(w_i)^{X_{\mathrm{S}}}, C_1\right) = C_3$是否成立。若等式成立则继续执行以下算法，否则返回“0”。

(10) Test_2算法。云服务器输入PP, $T_{d_{\mathrm{A}}}$, $T_{d_{\mathrm{B}}}$,

$C_{\mathrm{A}}=(C_{\mathrm{A}1},C_{\mathrm{A}2},C_{\mathrm{A}3},C_{\mathrm{A}4})$ ， $C_{\mathrm{B}}=(C_{\mathrm{B}1},C_{\mathrm{B}2},C_{\mathrm{B}3}$ $C_{\mathrm{B}4})$ ， 计 算$K_{\mathrm{A}}=\dfrac{C_{\mathrm{A}4}}{H_4\left(e\left(C_{\mathrm{A}2},T_{d\mathrm{A}}\right)\right)}$ ， $K_{\mathrm{B}}=\dfrac{C_{\mathrm{B}4}}{H_4\left(e\left(C_{\mathrm{B}2},T_{d\mathrm{B}}\right)\right)}$， 验 证 等 式$e\left(C_{\mathrm{A}1},K_{\mathrm{B}}\right)=e\left(C_{\mathrm{B}1},K_{\mathrm{A}}\right)$。若等式成立则给用户返回“1”，否则返回“0”。

3.2 正确性

CLEETS方案正确当且仅当Test_1算法和Test_2算法中的验证等式成立。

(1) Test_1中的验证等式成立，如果$w=w_i$：

$$\begin{aligned}&e\left(T_{\mathrm{W}},C_1\right)e\left(H_2(w_i)^{X_{\mathrm{S}}},C_1\right)\\&=e\left(H_2(w_i)^{X_{\mathrm{R}}}D_{\mathrm{R}},g^{\alpha}\right)e\left(H_2(w_i)^{X_{\mathrm{S}}},g^{\alpha}\right)\\&=e\left(H_2(w_i)^{X_{\mathrm{R}}},g^{\alpha}\right)e\left(D_{\mathrm{R}},g^{\alpha}\right)e\left(H_2(w_i)^{X_{\mathrm{S}}},g^{\alpha}\right)\\&=e\left(H_2(w_i)^{X_{\mathrm{R}}+X_{\mathrm{S}}},g^{\alpha}\right)e\left({Q_{\mathrm{R}}}^{s},g^{\alpha}\right)\\&=e\left(H_2(w_i)^{\alpha},\mathrm{PK_R}\mathrm{PK_S}\right)e\left({Q_{\mathrm{R}}}^{\alpha},\hat{g}\right)=C_3\end{aligned}$$

即等式$e\left(T_{\mathrm{W}},C_1\right)e\left(H_2(w_i)^{X_{\mathrm{S}}},C_1\right)=C_3$成立。

(2) Test_2中的验证等式成立：

$$\begin{aligned}K_{\mathrm{A}}&=\frac{C_{\mathrm{A}4}}{H_4\left(e\left(C_{\mathrm{A}2},T_{d\mathrm{A}}\right)\right)}\\&=\frac{H_3(M_{\mathrm{A}})^{\alpha_{\mathrm{A}}}\cdot H_5\left(e(\mathrm{PK_{RA}},{Q_{\mathrm{RA}}}^{\beta_{\mathrm{A}}}\right)}{H_4\left(e\left(g^{\beta_{\mathrm{A}}},{D_{\mathrm{RA}}}^{X_{\mathrm{RA}}}\right)\right)}\\&=\frac{H_3(M_{\mathrm{A}})^{\alpha_{\mathrm{A}}}\cdot H_4\left(e\left(g^{sX_{\mathrm{RA}}},{Q_{\mathrm{RA}}}^{\beta_{\mathrm{A}}}\right)\right.}{H_4\left(e\left(g^{\beta_{\mathrm{A}}},{Q_{\mathrm{RA}}}^{sX_{\mathrm{RA}}}\right)\right)}\\&=H_3(M_{\mathrm{A}})^{\alpha_{\mathrm{A}}},\\K_{\mathrm{B}}&=\frac{C_{\mathrm{B}4}}{H_4\left(e\left(C_{\mathrm{B}2},T_{d\mathrm{B}}\right)\right)}\\&=\frac{H_3(M_{\mathrm{B}})^{\alpha_{\mathrm{B}}}\cdot H_4\left(e(\mathrm{PK_{RB}},{Q_{\mathrm{RB}}}^{\beta_{\mathrm{B}}}\right)}{H_4\left(e\left(g^{\beta_{\mathrm{B}}},{D_{\mathrm{RB}}}^{X_{\mathrm{RB}}}\right)\right)}\\&=\frac{H_3(M_{\mathrm{B}})^{\alpha_{\mathrm{B}}}\cdot H_4\left(e\left(g^{sX_{\mathrm{RB}}},{Q_{\mathrm{RB}}}^{\beta_{\mathrm{B}}}\right)\right.}{H_4\left(e\left(g^{\beta_{\mathrm{B}}},{Q_{\mathrm{RB}}}^{sX_{\mathrm{RB}}}\right)\right)}\\&=H_3(M_{\mathrm{B}})^{\alpha_{\mathrm{B}}}。\end{aligned}$$

如 果 $M_{\mathrm{A}}=M_{\mathrm{B}}$， 则$e\left(C_{\mathrm{A}1},K_{\mathrm{B}}\right)=e\left(C_{\mathrm{B}1},K_{\mathrm{A}}\right)$成立。

4 安全性证明

假设挑战者C无法以一定的优势解决BDH (Bilinear Diffie-Hellman)困难问题，则在随机预言模型下，方案在适应性选择关键字攻击下具有关键字不可区分性安全。

引理1 假设敌手$\mathrm{A_I}$以一定的优势ε攻破CLEETS方案，则存在一个算法B以一定的优势解决BDH困难问题。设q_{H_1}, q_{PP}, q_{P}, q_{T}和q_{A}分别表示H_1哈希询问、部分私钥询问、私钥询问、搜索陷门询问和测试陷门询问，则算法B解决BDH困难问题的优势为

$$\varepsilon'\geq\frac{\varepsilon}{q_{H_1}}\left(1-\frac{1}{q_{H_1}}\right)^{q_{\mathrm{PP}}+q_{\mathrm{P}}+q_{\mathrm{T}}+q_{\mathrm{A}}}$$

证明 给出BDH问题实例$\left(g,g^a,g^b,g^c\right)$, C的主要任务是通过$\mathrm{A_I}$的回答，计算出$e(g,g)^{abc}$。

(1) 系统设置。C选择一个$\mathrm{ID_I}\,(1\leq\mathrm{ID_I}\leq q_{H_1})$作为挑战身份，计算$\hat{g}=g^a$,将系统参数$\mathrm{PP}=\{K,G_1,G_2,e,q,g,\hat{g},H_1,H_2,H_3,H_4\}$发送给敌手$\mathrm{A_I}$。

(2) H_1哈希询问：C维持L_{H_1}数据表，该表包含元组$\langle\mathrm{ID}_i,\alpha_i,Q_i\rangle$。$\mathrm{A_I}$询问$\mathrm{ID}_i$时，C执行以下操作：

(a)如果ID_i已经在$\langle\mathrm{ID}_i,\alpha_i,Q_i\rangle$中，C输出$Q_i$; (b)否则，判断$\mathrm{ID}_i$是否等于$\mathrm{ID_I}$。如果$\mathrm{ID}_i=\mathrm{ID_I}$, C随机选取$\alpha_i\in Z_q^*$并计算$Q_i=g^{\alpha_i b}$; (c)否则，C随机选取$\alpha_i\in Z_q^*$并计算$Q_i=g^{\alpha_i}$，添加$\langle\mathrm{ID}_i,\alpha_i,Q_i\rangle$到表$L_{H_1}$中并输出$Q_i$。

(3)H_2哈希询问：C维持L_{H_2}数据表，该表包含元组$\langle w_i,\beta_i,H_2\left(w_i\right)\rangle$。$\mathrm{A}_I$询问$w_i$时，C执行以下操作：

(a)如果$H_2\left(w_i\right)$已经在$\langle w_i,\beta_i,H_2\left(w_i\right)\rangle$中，C输出$H_2\left(w_i\right)$; (b)否则，C随机选取$\beta_i\in Z_q^*$并计算$H_2\left(w_i\right)=g^{\beta_i}$，添加$\langle w_i,\beta_i,H_2\left(w_i\right)\rangle$到表$L_{H_2}$中并输出$H_2\left(w_i\right)$。

(4)H_3哈希询问：C维持L_{H_3}数据表，该表包含元组$\langle M_j,\beta_j,H_3\left(M_j\right)\rangle$。当$\mathrm{A_I}$对$M_j$进行询问时，C执行以下操作：

(a)如果$H_3\left(M_j\right)$已经在$\langle M_j,\beta_j,H_3\left(M_j\right)\rangle$中，C输出$H_3\left(M_j\right)$; (b)否则，C随机选取$\gamma_i\in Z_q^*$并计算$H_3\left(M_j\right)=g^{\gamma_i}$，添加$\langle w_i,\beta_i,H_2\left(w_i\right)\rangle$到表$L_{H_3}$中并输出$H_3\left(M_j\right)$。

(5)H_4哈希询问：C维持L_{H_4}数据表，该表包含元组$\langle\mathrm{PK}_i,Q_i,H_4\rangle$。当$\mathrm{A_I}$对$\mathrm{PK}_i$, Q_i进行询问时，C执行以下操作：

(a)如果PK_i, Q_i已经在L_{H_1},L_{PK}在中，C输出H_4; (b)否则，C随机选取$x_i,\alpha_i\in Z_q^*$并计算$\mathrm{PK}_i=g^{x_i}$, $Q_i=g^{\alpha_i}$，添加$\langle\mathrm{PK_i},Q_i,H_4\rangle$到表$L_{H_4}$中并输出$H_4$。

(6)部分私钥提取询问。C维持为L_{PSK}数据表，该表包含元组$\langle\mathrm{ID}_i,Q_i,D_i\rangle$。$\mathrm{A_I}$询问$\mathrm{ID}_i$的部分私钥时，C执行$H_1$询问获得$\langle\mathrm{ID}_i,\ \alpha_i,Q_i\rangle$，然后执行：

(a)如果$\mathrm{ID}_i\neq\mathrm{ID_I}$，先计算$D_i=\hat{g}^{\alpha_i}$，然后添加$\langle\mathrm{ID}_i,Q_i,D_i\rangle$到表$L_{\mathrm{PSK}}$中并输出$D_i$; (b)否则，C终止操作，记录该事件为$E_1$。

(7)公钥提取询问。C维持L_{PK}数据表，该表包

括元组$\langle \mathrm{ID}_i, \mathrm{X}_i, \mathrm{PK}_i\rangle$。$\mathrm{A_I}$询问$\mathrm{ID}_i$的公钥时，C执行如下操作：

(a)如果PK_i存在于L_{PK}中则输出PK_i; (b)否则C随机选取$X_i \in Z_q^*$计算$\mathrm{PK}_i = g^{\frac{1}{X_i}}$，C添加$\langle \mathrm{ID}_i, \mathrm{X}_i, \mathrm{PK}_i\rangle$到表$L_{\mathrm{PK}}$中并输出$\mathrm{PK}_i$。

(8)公钥替换询问。$\mathrm{A_I}$以一个随机值代替任意用户的公钥。

(9)私钥询问。输入用户身份ID_i，如果$\mathrm{ID}_i = \mathrm{ID}_I$, C终止查询，此事件记为$E_2$；否则C执行以下操作：

(a)如果$\langle \mathrm{ID}_i, Q_i, D_i\rangle$和$\langle \mathrm{ID}_i, X_i, \mathrm{PK}_i\rangle$分别位于表$L_{\mathrm{PSK}}$和$L_{\mathrm{PK}}$中，则C设置私钥$\mathrm{SK}_i = (\mathrm{X}_i, D_i)$并输出此值；(b)否则，C以身份$\mathrm{ID}_i$进行查询，通过上述过程获得$\mathrm{SK}_i = (X_i, D_i)$并输出。

(10)搜索陷门询问。当$\mathrm{A_I}$对关键字w_i对应的T_{W_i}进行陷门查询时，挑战者C执行以下操作:

(a)如果$\mathrm{ID}_i = \mathrm{ID_I}$，则C终止查询，此事件记为$E_3$；(b)否则，C从表$L_{\mathrm{PK}}$中取出$\langle \mathrm{ID}_i, X_i, \mathrm{PK}_i\rangle$，从表$L_{\mathrm{PSK}}$中取出$\langle \mathrm{ID}_i, Q_i, D_i\rangle$，从表$L_{H_2}$中取出$\langle w_i, \beta_i, H_2(w_i)\rangle$，计算$T_{W_i} = H_2(w_i')^{X_{\mathrm{R}}} D_{\mathrm{R}}$并输出。

(11)测试陷门询问。当$\mathrm{A_I}$对消息M_j进行测试陷门查询时，挑战者C执行以下操作:

(a)如果$\mathrm{ID}_i = \mathrm{ID}_I$，则C终止查询，此事件记为$E_4$；(b)否则，C从表$L_{\mathrm{PK}}$中取出$\langle \mathrm{ID}_i, X_i, \mathrm{PK}_i\rangle$，从表$L_{\mathrm{PSK}}$中取出$\langle \mathrm{ID}_i, Q_i, D_i\rangle$，计算$t_{M_j} = {D_{\mathrm{R}}}^{X_{\mathrm{R}}}$。

(12)挑战。利用身份ID^*, $\mathrm{A_I}$对两个关键字w_0和w_1发起挑战，C执行以下操作：

(a)如果$\mathrm{ID}^* \neq \mathrm{ID}_i$，C终止查询，此事件记为$E_5$；(b)否则，随机选择$b \in \{0,1\}$, $r \in Z_q^*$，输出$C^* = (g^{\mathrm{rc}}, C_{3\mathrm{b}})$。如果$C^*$是有效密文，则$C^* = e\left(g^{\beta_i}, g^{X_{\mathrm{R}}}\mathrm{PK_S}\right)^{\mathrm{rc}} e\left(g^{\alpha_i b}, g^{a}\right)^{\mathrm{rc}} = e(g,g)^{\beta_i X_{\mathrm{R}} \mathrm{rc}} e(g, \mathrm{PK_S})^{\beta_i \mathrm{rc}} e(g, g)^{\alpha_i abrc}$。

(13)更多陷门询问。$\mathrm{A_I}$通过发送w_i给挑战者C，进行更多的陷门询问，其中$w_i \neq w_0$且$w_i \neq w_1$, C像上面一样输出，以事件E_6表示$\mathrm{A_I}$既没有对w_0询问也没有对w_1询问。

(14)猜测。$\mathrm{A_I}$输出$b' \in \{0,1\}$作为猜测值，此时，

$$\begin{aligned}
&\left(\frac{C^*}{e(g^c, g)^{r\beta_i X_{\mathrm{i}}} e(g^c, \mathrm{PK_S})^{r\beta_i}}\right)^{\frac{1}{r\alpha_i}} \\
&= \left(\frac{e(g,g)^{cr\beta_i X_{\mathrm{i}}} e(g, \mathrm{PK_S})^{cr\beta_i} e(g,g)^{rabc\alpha_i}}{e(g^c, g)^{r\beta_i X_i} e(g^c, \mathrm{PK_S})^{r\beta_i}}\right)^{\frac{1}{r\alpha_i}} \\
&= \left(\frac{e(g,g)^{cr\beta_i X_{\mathrm{i}}} e(g, \mathrm{PK_S})^{cr\beta_i} e(g,g)^{rabc\alpha_i}}{e(g^c, g)^{r\beta_i X_i} e(g^c, \mathrm{PK_S})^{r\beta_i}}\right)^{\frac{1}{r\alpha_i}} \\
&= \left(e(g,g)^{rabc\alpha_i}\right)^{\frac{1}{r\alpha_i}} = e(g,g)^{abc}
\end{aligned}$$

以下分析C获胜的优势ε'：

(a)当$\mathrm{A_I}$执行$H_i\,(1 \le i \le 4)$询问时，C以随机值回应；(b)如果$E_i\,(1 \le i \le 6)$都不发生，C不终止查询。显然，$\Pr[\neg E_1 \wedge \neg E_2 \wedge \neg E_3 \wedge \neg E_4 \wedge \neg E_5] = (1 - 1/q_{H_1})^{q_{\mathrm{PP}}+q_{\mathrm{P}}+q_{\mathrm{T}}+q_{\mathrm{A}}}\,(1/q_{H_1})$。

根据文献[8]可知解决困难问题的优势为

$$\begin{aligned}
\varepsilon' &\ge \frac{1}{2}\Pr[\neg E_6]\Pr[\neg E_1 \wedge \neg E_2 \wedge \neg E_3 \wedge \neg E_4 \wedge \neg E_5] \\
&\ge \frac{1}{2}\cdot 2\varepsilon \cdot \left(1 - \frac{1}{q_{H_1}}\right)^{q_{\mathrm{PP}}+q_{\mathrm{P}}+q_{\mathrm{T}}+q_{\mathrm{A}}} \cdot \left(\frac{1}{q_{H_1}}\right) \\
&= \left(\frac{\varepsilon}{q_{H_1}}\right)\left(1 - \frac{1}{q_{H_1}}\right)^{q_{\mathrm{PP}}+q_{\mathrm{P}}+q_{\mathrm{T}}+q_{\mathrm{A}}}
\end{aligned}$$

证毕

引理2　假设敌手$\mathrm{A_{II}}$以一定的优势ε攻破CLEETS方案，则存在一个算法B以一定的优势解决BDH困难问题。设q_{H_1}, q_{P}, q_{T}和q_{A}分别表示H_1哈希询问、私钥询问、搜索陷门询问和测试陷门询问。则算法B解决BDH困难问题的优势为

$$\varepsilon' \ge \frac{\varepsilon}{q_{H_1}}\left(1 - \frac{1}{q_{H_1}}\right)^{q_P+q_T+q_A}$$

证明　给出BDH问题实例$\left(g, g^a, g^b, g^c\right)$, C的主要任务是通过$\mathrm{A_{II}}$的回答，计算$e(g,g)^{abc}$。

(1)系统设置。C选择$\mathrm{ID_I}\,(1 \le \mathrm{ID_I} \le q_{H_1})$作为挑战身份，并计算$\hat{g} = g^s$，最后将系统参数$\mathrm{PP} = \{\mathrm{K}, G_1, G_2, e, q, g, \hat{g}, H_1, H_2, H_3, H_4\}$和$s$发送给敌手$\mathrm{A_{II}}$。

(2)H_1哈希询问：C维持L_{H_1}数据表，该表包含元组$\langle \mathrm{ID}_i, Q_i\rangle$。当$\mathrm{A_{II}}$询问$\mathrm{ID}_i$时，C执行以下操作：

(a)如果$\mathrm{ID_I}$已经在$\langle \mathrm{ID}_i, \alpha_i, Q_i\rangle$中，C输出$Q_i$；(b)否则，C随机选取$Q_i \in G_1$，添加$\langle \mathrm{ID}_i, Q_i\rangle$到表$L_{H_1}$中并输出$Q_i$。

(3)H_2哈希询问：与定理1中证明过程相同。

(4)H_3哈希询问：与定理1中证明过程相同。

(5)H_4哈希询问：与定理1中证明过程相同。

(6)公钥提取询问。C维持L_{PK}数据表，该表包含元组$\langle \mathrm{ID}_i, X_i, \mathrm{PK}_i\rangle$。当$\mathrm{A_{II}}$询问$\mathrm{ID}_i$的公钥时，C执行如下操作：

(a)如果PK_i已经存在于L_{PK}中，输出PK_i；(b)否则，C随机选取$X_i \in Z_q^*$，如果$\mathrm{ID}_i = \mathrm{ID_I}$，计算$\mathrm{PK}_i = g^{b\frac{1}{X_i}}$。否则，$\mathrm{PK}_i = g^{\frac{1}{X_i}}$，添加$\langle \mathrm{ID}_i, X_i, \mathrm{PK}_i\rangle$到表$L_{\mathrm{PK}}$中并输出$\mathrm{PK}_i$。

(7)私钥询问。输入用户身份ID_i，如果$\mathrm{ID}_i = \mathrm{ID_I}$, C终止查询，此事件记为$E_1$。否则，C查询$L_{H_1}$和$L_{\mathrm{PK}}$获得$\langle \mathrm{ID}_i, Q_i\rangle$和$\langle \mathrm{ID}_i, X_i, \mathrm{PK}_i\rangle$。最后，输出$\mathrm{SK}_i = (X_i, sQ_i)$。

(8)搜索陷门询问。与定理1中相同，除了计算$T_{\mathrm{W}_i} = H_2(w_i')^{\mathrm{D_R}} sQ_{\mathrm{R}}$。

(9)测试陷门询问。与定理1中相同，除了计算$t_{M_j}=sQ_{\mathrm{R}}{}^{X_{\mathrm{R}}}$。

(10)挑战。与定理1中挑战过程相同。

(11)更多陷门询问。与定理1中证明过程相同。

(12)猜测。$\mathrm{A_{II}}$输出$b'\in\{0,1\}$作为猜测值，此时：

$$\left(\frac{C^*}{e(g^a,g^{cX_S})^{r\beta_i}e(Q_i,g^{cs})^r}\right)^{\frac{1}{r\beta_iX_i}}=\left(\frac{e(g,g)^{a\beta_ibX_irc}e\left(g^a,g^{cX_S}\right)^{\beta_ir}e(Q_i,g^{cs})^r}{e(g^a,g^{cX_S})^{r\beta_i}e(Q_i,g^{cs})^r}\right)^{\frac{1}{r\beta_iX_i}}=\left(e(g,g)^{a\beta_ibX_irc}\right)^{\frac{1}{r\beta_iX_i}}=e(g,g)^{abc}$$

以下分析C获胜的优势ε'：

(a)当$\mathrm{A_{II}}$执行$H_i\,(1\le i\le 4)$询问时，C以随机值回应；(b)如果$E_i\,(1\le i\le 5)$都不发生，则C不终止查询。显然：$\Pr[\neg E_1\wedge\neg E_2\wedge\neg E_3\wedge\neg E_4]=\left(1-\frac{1}{q_{H_1}}\right)^{q_{\mathrm{P}}+q_{\mathrm{T}}+q_{\mathrm{A}}}\left(\frac{1}{q_{H_1}}\right)$根据文献[8]可知解决困难问题的优势为：

$$\begin{aligned}\varepsilon'&\ge\frac{1}{2}\Pr[\neg E_5]\Pr[\neg E_1\wedge\neg E_2\wedge\neg E_3\wedge\neg E_4]\\&\ge\frac{1}{2}\cdot 2\varepsilon\cdot\left(1-\frac{1}{q_{H_1}}\right)^{q_{\mathrm{P}}+q_{\mathrm{T}}+q_{\mathrm{A}}}\cdot\left(\frac{1}{q_{H_1}}\right)\\&=\left(\frac{\varepsilon}{q_{H_1}}\right)\left(1-\frac{1}{q_{H_1}}\right)^{q_{\mathrm{P}}+q_{\mathrm{T}}+q_{\mathrm{A}}}\end{aligned}$$

证毕

5 性能分析

5.1 功能对比

从表1可以看出，文献[6]方案支持关键字搜索功能但不支持密文等值测试；文献[16]方案支持密文等值测试但不支持关键字搜索功能，其搜索准确度不高。本文方案同时支持密文等值测试和关键字搜索功能。先通过关键字搜索判断是否有用户所需的密文信息，再执行密文等值测试，有效地提高了检索效率。

5.2 效率分析

文献[6]方案和本文方案均能实现关键字搜索加密，因此仅对本文方案和文献[6]方案的通信开销进行比较。$|G_1|$和$|G_2|$代表群G_1和G_2上元素的长度，$\lg_q$代表二进制比特数，$|Z_q|$代表Z_q上数的长度。从表2可以看出，与文献[6]方案相比，本文方案秘密值长度和公钥长度相同，部分私钥长度和密文长度稍高。

5.3 计算代价

以下模拟方案的计算代价。假设T_{sm}为生成随机数时间；T_{b}为双线性对运算时间；T_{H}为哈希函数运算时间；T_{ex}为指数运算时间；T_{mul}为乘法运算时间。仿真环境为戴尔笔记本电脑(I7-4700CPU@ 3.2 GHz，16 GB和Ubuntu Linux)和PBC[20]函数库。方案的执行时间如表3所示，其中，T_{sm}=2.142 ms，T_{b}=0.671 ms，T_{H}=5.762 ms，T_{ex}=5.611ms，T_{mul}=0.1 ms。

6 结束语

本文提出了一种支持关键字搜索的无证书密文等值测试加密方案。该方案可以实现未对密文解密的情况下，先执行关键字检索，再判断是否执行密文等值测试，进而判断两个密文是否是同一明文产生。该方案不仅克服了传统可搜索加密的密文等值测试问题，也解决了等值测试中检索效率低的问题。通过安全性分析，证明本文方案满足关键字不可区分性。由于方案在密文等值测试的基础上增加了密文关键字搜索功能，使服务器免于执行遍历密文的等值测试，但是，增加了两个双线性对运算和

表 1 功能对比

方案	等值测试	关键字搜索	搜索陷门生成	测试内容
文献[6]方案	不支持	支持	与关键字绑定	判断密文对应明文是否相等
文献[16]方案	支持	不支持	与密文或加密者绑定	判断密文是否包含关键字
本文方案	支持	支持	以上两者都具备	以上两者都具备

表 2 通信开销

方案	部分私钥长度	秘密值长度	公钥长度	密文长度
文献[6]方案	$\|2Z_q\|$	$\|2Z_q\|$	$\|2G_1\|$	$n(\|G_1\|+lg_q)$
本文方案	$\|2G_1\|$	$\|2Z_q\|$	$\|2G_1\|$	$n(\|G_1\|+\|G_2\|+lg_q)$

表 3 计算代价(ms)

密钥生成	加密	陷门	授权	测试算法1	测试算法2
$4T_{\mathrm{ex}}+2T_{\mathrm{H}}+2T_{\mathrm{sm}}$=38.252	$4T_{\mathrm{ex}}+3T_{\mathrm{b}}+T_{\mathrm{mul}}+2T_{\mathrm{H}}+2T_{\mathrm{sm}}$=40.365	$T_{\mathrm{H}}+T_{\mathrm{ex}}+T_{\mathrm{mul}}$=11.473	T_{ex}=5.611	$2T_{\mathrm{b}}+T_{\mathrm{ex}}$=6.953	$4T_{\mathrm{b}}+2T_{\mathrm{H}}$=14.208

一个映射到循环群的哈希运算，因此，具有相对较大的计算开销。未来，我们会进一步解决计算和通信开销问题。

参 考 文 献

[1] 张键红, 李鹏燕. 一种有效的云存储数据完整性验证方案[J]. 信息网络安全, 2017(3): 1–5. doi: 10.3969/j.issn.1671-1122.2017.03.001.
ZHANG Jianhong and LI Pengyan. An efficient data integrity verification scheme for cloud storage[J]. *Netinfo Security*, 2017(3): 1–5. doi: 10.3969/j.issn.1671-1122.2017.03.001.

[2] MAYER-SCHONBERGER V and CUKIER K. Big data: A Revolution that Will Transform How We Live, Work and Think[M]. London: John Murray, 2013: 94–98.

[3] SONG D X, WAGNER D, and PERRIG A. Practical techniques for searches on encrypted data[C]. 2000 IEEE Symposium on Security and Privacy, Berkeley, USA, 2000: 44–55. doi: 10.1109/SECPRI.2000.848445.

[4] BONEH D, DI CRESCENZO G, OSTROVSKY R, *et al*. Public key encryption with keyword search[C]. International Conference on the Theory and Applications of Cryptographic Techniques, Interlaken, Switzerland, 2004: 506–522. doi: 10.1007/978-3-540-24676-3_30.

[5] 江明明, 郭宇燕, 余磊, 等. 有效的标准模型下格上基于身份的代理重加密[J]. 电子与信息学报, 2019, 41(1): 61–66. doi: 10.11999/JEIT180146.
JIANG Mingming, GUO Yuyan, YU Lei, *et al*. Efficient identity-based proxy re-encryption on lattice in the standard model[J]. *Journal of Electronics & Information Technology*, 2019, 41(1): 61–66. doi: 10.11999/JEIT180146.

[6] PENG Yanguo, CUI Jiangtao, PENG Changgen, *et al*. Certificateless public key encryption with keyword search[J]. *China Communications*, 2014, 11(11): 100–113. doi: 10.1109/CC.2014.7004528.

[7] MA Mimi, HE Debiao, KUMAR N, *et al*. Certificateless searchable public key encryption scheme for industrial internet of things[J]. *IEEE Transactions on Industrial Informatics*, 2018, 14(2): 759–767. doi: 10.1109/TII.2017.2703922.

[8] MA Mimi, HE Debiao, KHAN M K, *et al*. Certificateless searchable public key encryption scheme for mobile healthcare system[J]. *Computers & Electrical Engineering*, 2018, 65: 413–424. doi: 10.1016/j.compeleceng.2017.05.014.

[9] 张玉磊, 刘祥震, 郎晓丽, 等. 云存储环境下多服务器的密钥聚合可搜索加密方案[J]. 电子与信息学报, 2019, 41(3): 674–679. doi: 10.11999/JEIT180418.
ZHANG Yulei, LIU Xiangzhen, LANG Xiaoli, *et al*. Multi-server key aggregation searchable encryption scheme in cloud environment[J]. *Journal of Electronics & Information Technology*, 2019, 41(3): 674–679. doi: 10.11999/JEIT180418.

[10] 张玉磊, 刘文静, 刘祥震, 等. 基于授权的多服务器可搜索密文策略属性基加密方案[J]. 电子与信息学报, 2019, 41(8): 1808–1814. doi: 10.11999/JEIT180944.
ZHANG Yulei, LIU Wenjing, LIU Xiangzhen, *et al*. Searchable multi-server CP-ABE scheme based on authorization[J]. *Journal of Electronics & Information Technology*, 2019, 41(8): 1808–1814. doi: 10.11999/JEIT180944.

[11] YANG Guomin, TAN C H, HUANG Qiong, *et al*. Probabilistic public key encryption with equality test[C]. Cryptographers' Track at the RSA Conference 2010, San Francisco, USA, 2010: 119–131. doi: 10.1007/978-3-642-11925-5_9.

[12] TANG Qiang. Towards public key encryption scheme supporting equality test with fine-grained authorization[C]. The 16th Australasian Conference on Information Security and Privacy, Melbourne, Australia, 2011: 389–406.

[13] TANG Qiang. Public key encryption supporting plaintext equality test and user-specified authorization[J]. *Security and Communication Networks*, 2012, 5(12): 1351–1362. doi: 10.1002/sec.418.

[14] TANG Qiang. Public key encryption schemes supporting equality test with authorisation of different granularity[J]. *International Journal of Applied Cryptography*, 2012, 2(4): 304–321. doi: 10.1504/IJACT.2012.048079.

[15] 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. doi: 10.1109/TIFS.2014.2378592.

[16] MA Sha. Identity-based encryption with outsourced equality test in cloud computing[J]. *Information Sciences*, 2016, 328: 389–402. doi: 10.1016/j.ins.2015.08.053.

[17] QU Haipeng, YAN Zhen, LIN Xijun, *et al*. Certificateless public key encryption with equality test[J]. *Information Sciences*, 2018, 462: 76–92. doi: 10.1016/j.ins.2018.06.025.

[18] HUANG Kaibin, TSO R, CHEN Yuchi, *et al*. PKE-AET: Public key encryption with authorized equality test[J]. *The Computer Journal*, 2015, 58(10): 2686–2697. doi: 10.1093/comjnl/bxv025.

[19] LEE H T, LING San, SEO J H, *et al*. Semi-generic construction of public key encryption and identity-based encryption with equality test[J]. *Information Sciences*, 2016, 373: 419–440. doi: 10.1016/j.ins.2016.09.013.

[20] The pairing-based cryptography library[EB/OL]. http://crypto.stanford.edu/pbc/, 2015.

张玉磊：男，1979年生，博士，副教授，研究方向为密码学与信息安全.

陈文娟：女，1993年生，硕士生，研究方向为密码学与信息安全.

张永洁：女，1978年生，硕士，副教授，研究方向为密码学与信息安全.

张雪微：女，1995年生，硕士生，研究方向为密码学与信息安全.

王彩芬：女，1963年生，博士，教授，研究方向为密码学与信息安全.

责任编辑：马秀强