支持属性撤销的可验证多关键词搜索加密方案

孙 瑾 王小静* 王尚平 任利利

(西安理工大学 西安 710054)

摘 要：近年来，可搜索加密技术及细粒度访问控制的属性加密在云存储环境下得到广泛应用。考虑到现存的基于属性的可搜索加密方案存在仅支持单关键词搜索而不支持属性撤销的问题，以及单关键词搜索可能造成返回搜索结果部分错误并导致计算和宽带资源浪费的缺陷，该文提出一种支持属性撤销的可验证多关键词搜索加密方案。该方案允许用户检测云服务器搜索结果的正确性，同时在细粒度访问控制结构中支持用户属性的撤销，且在属性撤销过程中不需要更新密钥和重加密密文。该文在随机预言机模型下基于判定性线性假设被证明具有抵抗选择关键词集攻击安全性及关键词隐私性，同时从理论和实验两方面分析验证了该方案具有较高的计算效率与存储效率。

关键词：可搜索加密；属性撤销；多关键词搜索；可证明安全

中图分类号：TN918.1 **文献标识码**：A **文章编号**：1009-5896(2019)01-0053-08

DOI: 10.11999/JEIT180237

Verifiable Multi-keyword Search Encryption Scheme with Attribute Revocation

SUN Jin WANG Xiaojing WANG Shangping REN Lili

(*Xi'an University of Technology*, *Xi'an* 710054, *China*)

Abstract: In recent years, searchable encryption technology and fine-grained access control attribute encryption is widely used in cloud storage environment. Considering that the existing searchable attribute-based encryption schemes have some flaws: It only support single-keyword search without attribute revocation. The single-keyword search may result in the waste of computing and broadband resources due to the partial retrieval from search results. A verifiable multi-keyword search encryption scheme that supports revocation of attributes is proposed. The scheme allows users to detect the correctness of cloud server search results while supporting the revocation of user attributes in a fine-grained access control structure without updating the key or re-encrypting the ciphertext during revocation stage. The aforementioned scheme is proved by the deterministic linearity hypothesis, and the relevant analysis results indicate that it can resist the attacks of keyword selection and the privacy of keywords in the random oracle model with high computational efficiency and storage effectiveness.

Key words: Searchable encryption; Attribute revocation; Multi-keyword search; Provable security

1 引言

云存储是供用户进行数据存储和访问的一种新兴存储技术。为节省本地资源，更多的用户将数据交给云存储服务器托管；为确保数据不被篡改、丢失、非法访问或任意窃取，云存储数据的安全性和隐私性近年来也被作为研究热点进行讨论，其中最常用的技术是先加密共享内容后将其存储在云服务器上，然而待加密数据文件的巨大会影响用户的搜索和应用。可搜索加密技术要求授权用户具有关键词密文查询的能力，使加密数据的安全性和隐私性得到保证。例如某用户检索他想获取的文件时，云服务器仅返回与关键词相关的搜索结果。在实际应用系统中，基于属性的可搜索加密技术仍存在用户属性变更、属性到期、密钥泄露等问题，可通过引入属性撤销机制来变更授权用户的访问权限。

2000年，Song等人[1]为了解决客户端只检索包含某些内容文档的问题，首次在对称密钥的基础下

收稿日期：2018-03-14；改回日期：2018-08-21；网络出版：2018-08-31
*通信作者： 王小静 wxjdyb@163.com
基金项目：国家自然科学青年基金(61303223)，国家自然科学基金(61572019)

Foundation Items: The National Natural Science Youth Foundation of China (61303223), The National Natural Science Foundation of China (61572019)

提出用于搜索加密数据的实用性技术。该方案仅支持将单个文件加密成一系列单词的密文，在搜索阶段关键词陷门和索引需进行异或运算，不足之处在于其搜索效率低下。2004年，Boneh等人[2]定义了公钥可搜索加密的概念，提出的几种构建方案每次运行加密算法都要使用对运算，计算效率较低。2006年，Curtmola等人[3]采用反向索引技术提出了有效的单关键词搜索方案。2014年，李双等人[4]为了实现加密数据的共享性和节省信息的存储空间，提出了基于密钥策略的可搜索加密方案，使得共享数据可被多方查询。2017年，Yang等人[5]设计了一种新型的支持连接关键词搜索和认证授权功能的加密方案。同年，Qiu等人[6]提出了隐藏策略的关键词搜索方案，但该方案的安全性是在一般群模型下被证明的。为了准确、快速地检索到感兴趣的文件，允许用户在一次搜索请求中提交多个关键词的可搜索加密方案[7–10]相继被提出。

在实际环境中，用户的属性是不断变化的，失去部分属性的非法用户依然能够检索和解密被加密的消息，易造成数据泄露，使得云环境中密文的安全性不能得到保障。2016年，王尚平等人[11]构造了具有两个撤销属性列表的属性加密方案，该方案不具有搜索功能。Sun等人[12]利用代理重加密技术构造了具有可验证的关键词搜索且支持用户属性撤销的方案，但该方案计算开销巨大。随后，陈燕俐等人[13]提出了支持属性撤销的可搜索加密方案，该方案安全性较低且属性撤销时需要更新密钥和密文。

本文主要贡献和创新点如下：(1)允许用户进行多关键词密文查询；(2)结合基于密钥策略的属性加密和审计思想提出了可验证的搜索加密方案，提高了搜索的精准度；(3)提出具有撤销功能的加密方案，在密文中绑定撤销信息，不需更新用户密钥和原始密文；(4)基于实际数据集的仿真实验表明，本方案在实际应用场景中是可行的、高效的。

2 基础知识

2.1 双线性群

定义 1(双线性对运算)[2]： 令G_0和G_1是两个阶为素数p的乘法循环群。则称映射$e: G_0 \times G_1 \to G_T$为一个双线性对运算，映射$e$满足以下性质：(1)双线性：对于$\forall g_0 \in G_0, g_1 \in G_1$和$\forall a, b \in Z_p$，满足等式$e\left(g_0^a, g_1^b\right) = e(g_0, g_1)^{ab}$；(2)非退化性：$\exists g_0 \in G_0$, $g_1 \in G_1$，使得$e(g_0, g_1) \neq 1$；(3)可计算性：对$\forall g_0 \in G_0, g_1 \in G_1$，存在计算$e(g_0, g_1)$的有效算法。如果$G_0 = G_1$，称为对称对，否则这个对是非对称的。

2.2 访问树[14]

访问树用T来描述，树中的每一个内节点x代表一个阈值门(k_x, num_x)，其中k_x表示阈值且$0 < k_x \leq \mathrm{num}_x$, num_x表示孩子节点个数。当$k_x = 1$时，阈值门是“OR”，当$k_x = \mathrm{num}_x$时，阈值门是“AND”。为了便于描述定义如下函数：函数parent(x)返回节点x的父节点；对访问树T中每个节点的子节点从1到num_x编号，index(x)返回节点x的编号；att(x)返回与叶子节点x相关的属性值。

2.3 秘密共享[9]

Shamir's秘密共享方案是一种在域F中由n个参与者共享秘密s的方法，可以构造一个t次的随机多项式$P \in F[x]$，其常数项为$P(0) = s$。第i个参与者拥有秘密参数$(i, P(i))$，获得任意$t+1$个参与者的共享份额$P(x_0), P(x_1), \cdots, P(x_t)$后，通过Lagrange插值公式可以恢复出共享秘密s。$P(0) = \sum_{i=0}^{t} \lambda_i \cdot P(x_i)$，其中$\lambda_i = \prod_{j \neq i} x_j/(x_j - x_i)$。

2.4 困难性假设

定义 2(判定性线性(decisional linear)假设)[15]： 根据安全参数选择一个阶为素数p的群G_0，挑战者选定$g, \hbar, f, Q \in G_0$，随机选择$r_1, r_2 \in Z_p$。任意多项式时间敌手A_{DL}在得到$(g, \hbar, f, f^{r_1}, g^{r_2})$后，必须将$G_0$中的随机元素$Q$和$\hbar^{r_1+r_2} \in G_0$分开。定义$A_{\mathrm{DL}}$攻破DL问题的优势为$\varepsilon$，若等式(1)成立，则有定义3：

$$\left| \Pr\left[A\left(g, \hbar, f, f^{r_1}, g^{r_2}, \hbar^{r_1+r_2}\right) = 1\right] - \Pr\left[A\left(g, \hbar, f, f^{r_1}, g^{r_2}, Q\right) = 1\right] \right| \geq \varepsilon \tag{1}$$

定义 3 如果没有多项式时间敌手以不可忽略的优势解决DL问题，则称DL假设是困难的。

3 系统模型和安全模型

本文方案共包含6个参与方，数据拥有者(DO)、数据使用者(DU)、云服务提供者(CSP)、第3方审计(TPA)、授权中心(AC)、撤销代理服务器(RPS)。DO把数据上传到CSP之前对文件集进行加密并生成签名，同时生成关键词索引；半可信的CSP负责文件的存储和检索；授权的DU提交自身属性集和陷门给CSP以发起搜索请求；RPS生成用于撤销运算的密文；完全可信的TPA验证搜索结果的正确性。图1给出本文方案的系统模型。

为保证加密文件的完整性和机密性，一个安全的可搜索加密方案必须保证关键词隐私性和选择关键词集安全。攻击游戏参考文献[15]。

4 方案的构造

4.1 方案具体构造如下

Setup(1^l)：该算法输入安全参数l，设p是一个素数，G_0, G_1是两个阶为p的循环群，g, h分别为

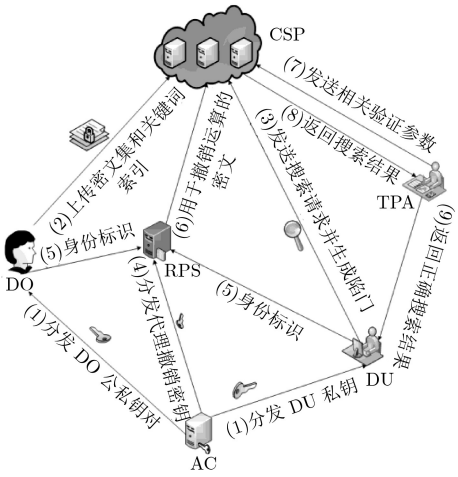

图 1 系统模型

G_0, G_1的生成元。$e: G_0 \times G_1 \to G_T$表示一个双线性映射。用$Z_p^*$中的元素表示系统属性集$S$中的属性。AC首先在$Z_p^*$上为属性$i \in S$随机生成一个$t_i$阶的多项式$P_i$。其次，$H_1: \{0,1\}^* \to G_0$, $H_2: \{0,1\}^* \to Z_p^*$是两个抗冲突的哈希函数，随机选取$a, b, c \in {Z_p}^*$, $g_0 \in G_0$，计算g^a, g^b, g^c。最后，算法设置系统公共参数pk，并且设置主密钥msk如式(2)：

$$\begin{aligned} &\mathrm{pk} = \left(g, h, g_0, g^a, g^b, g^c\right), \\ &\mathrm{msk} = \left(a, b, c, \{P_i\}_{i \in S}\right) \end{aligned} \tag{2}$$

KeyGen(T, msk, pk)：该算法输入访问树T，主密钥msk，系统公共参数pk。AC为访问树T中的每个节点x选取一个阶为d_x的多项式q_x，其中d_x的值为阈值减1，即$d_x = k_x - 1$。从根节点r开始，令$q_r(0) = ac$，随机选取剩余的d_r个点的值来完成多项式q_r的选定。往下其他的节点x，令$q_x(0) = q_{\mathrm{parent}(x)}(\mathrm{index}(x))$，通过随机选取剩余的$d_x$个点的值来完成多项式$q_x$的选定。若每个叶子节点$x \in \Phi$($\Phi$表示叶子节点集合)的多项式被完全确定，则随机选择$t \in Z_p^*$。在叶子节点对应属性$i = \mathrm{att}(x)$的密钥中绑定$P_i(0)$，密钥$D''_x$中嵌入用户$u_k$的相关参数$P_i(u_k)$。随机选择$r' \in Z_p^*$，计算$g^{r'}$。算法输出DU的私钥为$\mathrm{sk}_{u_k} = (T, \{(D_x, D'_x, D''_x) | \forall\, x \in T\})$：

$$\left.\begin{aligned} &D_x = g^{q_x(0)} H_1(\mathrm{att}(x))^{t \cdot P_i(0)} \\ &D'_x = h^t \\ &D''_x = {D'_x}^{P_i(u_k)} = h^{t \cdot P_i(u_k)} \end{aligned}\right\} \tag{3}$$

并且设置DO的公私钥对为$(\mathrm{pk}_0, \mathrm{sk}_0) = (g^{r'}, r')$。

Enc(pk, W, D, sk_0, S)：给定要加密的文件集$D = (d_1, d_2, \cdots, d_p)$及相对应的身份集ID = $(\mathrm{id}_1, \mathrm{id}_2, \cdots, \mathrm{id}_p)$($p$表示文件的个数)和关键词集合$W = \{w_1, w_2, \cdots, w_m\}$($m$表示关键词集合中关键词的个数)。DO采用对称密钥$k$(密钥空间中随机选择$k$)加密$D$得到密文集$C = \mathrm{Enc}_k(D)$, $C = (c_1, c_2, \cdots, c_p)$。随机选择$r_1, r_2 \in Z_p^*$，为文件$d_{\bar{j}}$建立索引$I_{\bar{j}} = \left(E_0, E_1, \{E_i\}_{i \in S}, \{E_j\}_{j \in [1,m]}\right)$：

$$\left.\begin{aligned} &E_0 = h^{cr_1}, E_1 = h^{r_2}, E_i = H_1(i)^{r_2} \\ &E_j = g^{a(r_1+r_2)} g^{b \cdot r_1 \cdot H_2(w_j)},\ (1 \le j \le m) \end{aligned}\right\} \tag{4}$$

另外，DO为每个文件$d_{\bar{j}}\left(\bar{j} \in [1, p]\right)$生成签名$\mathrm{sig}_{\bar{j}} = \left(H_1(\mathrm{id}_{\bar{j}}) g_0^{H_2(c_{\bar{j}})}\right)^{r'}$，其中$\mathrm{id}_{\bar{j}}$表示文件$d_{\bar{j}}$的身份。最后DO上传密文CT = (I, C)和签名sig = $\left\{\mathrm{sig}_1, \mathrm{sig}_2, \cdots, \mathrm{sig}_p\right\}$到CSP。$\left(I = \{I_1, I_2, \cdots, I_p\}\right)$。

Trap(W', sk_{u_k}, pk)：当DU对关键词集合$W' = \{{w'}_1, {w'}_2, \cdots, {w'}_t\}$($t$表示DU查询关键词的个数)进行搜索询问时，随机选择$s \in Z_p^*$，计算W'的陷门。最后DU将元组($T_{W'}, T$)发送给CSP。算法设置陷门$T_{W'} = \left(\mathrm{tok}_1, \mathrm{tok}_2, \left\{\tilde{D}_x, \tilde{D}'_x, \tilde{D}''_x\right\}_{x \in T}\right)$：

$$\left.\begin{aligned} &\mathrm{tok}_1 = \prod_{j=1}^{t} \left(g^a g^{bH_2(w'_j)}\right)^s, \mathrm{tok}_2 = h^{cs} \\ &\tilde{D}_x = D_x^s = \left(g^{q_x(0)} H_1(\mathrm{att}(x))^{t \cdot P_i(0)}\right)^s \\ &\tilde{D}'_x = D'^s_x = h^{ts}, \tilde{D}''_x = D''^s_x = h^{tsP_i(u_k)} \end{aligned}\right\} \tag{5}$$

ProxyRekey (msk, $\{\mathrm{RL}_i\}_{i \in \Phi}$)：当AC撤销用户某一属性时，为系统中每个属性$i \in \Phi$建立相应大小为$t_i$的用户撤销列表$\mathrm{RL}_i = \{u_{i,1}, u_{i,2}, \cdots, u_{i,t_i}\}$。由于多项式的阶数固定，因此限制每个属性对应撤销用户的最大数量为t_i。当属性i的撤销数量小于t_i时，随机选取$(x, P_i(x))$将RL_i填充至t_i个用户，其中假设撤销的用户集没有重叠。AC依次计算出用户$u_i \in \mathrm{RL}_i$对应的$P_i(u_i)$值。最后，计算代理撤销密钥$\mathrm{PRK} = \{\forall i \in \Phi, \forall u_i \in \mathrm{RL}_i : (u_i, P_i(u_i))\}$并发送RPS。

Convert (PRK, $\{E_i\}_{i \in S}, u_k$)：算法输入代理撤销密钥PRK，部分密文$\{E_i\}_{i \in S}$和用户标识u_k。RPS生成用于撤销运算的密文$E'_i = E_i^{\sum_i^{t_i} \lambda_{i,u} P_i(u_i)}$。另外，RPS需要计算$\lambda_{i,u_k} = \prod_{u_j \in \mathrm{RL}_i} u_j/(u_j - u_k)$并发送给CSP。其中$\lambda_{i,u} = u_k/(u_k - u_i) \cdot \prod_{j \ne i} u_j/(u_j - u_i)$ ($\forall\, u_i, u_j \in \mathrm{RL_i}, u_k \notin \mathrm{RL_i}$)。

Search(pk, $S, T_{W'}, T$, CT)：当收到DU的搜索请求后，CSP首先检测S是否满足T，满足时执行以下操作，否则输出$\perp$。

(1)定义一个递归算法DecryptNode(CT, $T_{W'}, x$)，该算法输入密文CT，陷门$T_{W'}$和节点x，输出群G_T中的一个元素或“$\perp$”。具体操作如下：如果x是叶子节点，其属性$i \in S$且未被撤销，则有等式：

$$\begin{aligned}&\text{DecryptNode}(CT, T_{W'}, x)\\&= e\left(\tilde{D}_x, E_1\right) / e\left(\tilde{D}'_x, E'_i\right) e(\tilde{D}''_x, E_i)^{\lambda_{i,u}}\\&= e(g,h)^{q_x(0) sr_2}\end{aligned} \tag{6}$$

如果用户u的属性i被撤销，则不能求出$P_i(0)$，即完成对属性i的撤销。

(2)如果x是非叶子节点，则考虑以下递归情况。对于节点x的所有子节点z，调用算法DecryptNode($CT, T_{W'}, z$)计算E_z。设存在一个随机的大小为k_x的节点集合w_x，如果不存在则$E_x = \perp$。否则E_x计算如下：令$i = \text{index}(z)$，$w'_x = \{\text{index}(z) : z \in w_x\}$

$$E_x = \prod_{z \in w_x} E_z^{\Delta_{i,w'_x}(0)} = e(g,h)^{sr_2 q_x(0)} \tag{7}$$

因此得出DecryptNode的完整定义，计算

$$E_{\text{root}} = e(g,h)^{q_r(0) sr_2} = e(g,h)^{acsr_2} \tag{8}$$

(3)CSP通过式(9)判断索引和陷门是否匹配。若等式成立，CSP返回相关搜索结果$C'' = \left(c'_1, c'_2, \cdots, c'_\phi\right)$($\phi$表示搜索结果的数量)和对应身份集$\text{ID} = \left(\text{id}'_1, \text{d}'_2, \cdots, \text{id}'_\phi\right)$给TPA；否则输出$\perp$。

$$e(E_0, \text{tok}_1) E_{\text{root}} = e\left(\prod_{j=1}^{t} E_j, \text{tok}_2\right) \tag{9}$$

$\text{Verify}\left(\text{sig}, \text{pk}, \text{pk}_0, C''\right)$：收到搜索结果$C''$后，TPA为身份为$\text{id}'_\rho$的密文$c'_\rho (\rho \in [1, \phi])$选取$\mu_\rho \in Z_p^*$，与CSP具体交互如下：(a)TPA将$(\rho, \mu_\rho)$发送给CSP；(b)CSP首先计算$\zeta = \sum_{\rho=1}^{\phi} \mu_\rho H_2\left(c'_\rho\right)$和$\sigma = \prod_{\rho=1}^{\phi} \left(\text{sig}'_\rho\right)^{\mu_\rho}$，随后把$(\sigma, \zeta)$发送给TPA；(c)TPA根据式(10)判断搜索结果C''是否正确。

$$e\left(\prod_{\rho=1}^{\phi} H_1(\text{id}')^{\mu_\rho} \cdot g_0^{\zeta}, \text{pk}_0\right) = e(\sigma, g) \tag{10}$$

4.2 正确性分析

不难验证上述方案是正确的，限于篇幅关系，此处省略推导细节。

5 安全性分析与证明

定理 1 如果随机预言机和DL假设成立，则存在概率多项式时间敌手A以优势$(1/|M| - \tau) + \xi$赢得关键词隐私性游戏。

证明 系统建立：挑战者C首先随机选择$a, b, c \in Z_p^*$，$f \in Z_p^*$，然后选择一个哈希函数$H_1 : \{0,1\}^* \to G_0$和$H_2 : \{0,1\}^* \to Z_p^*$。最后设置公共参数$\text{pk} = (e, g, g^a, g^b, g^c, f)$，主密钥$\text{msk} = (a, b, c)$。下面$C$将模拟随机预言机$O_{H_1}(j)$如下：

如果属性j没有被询问过，C首先随机选择$\alpha_j \in Z_p^*$，然后将元组(α_j, j)添加到哈希列表O_{H_1}并输出g^{α_j}；否则C从O_{H_1}中检索α_j并输出g^{α_j}。

询问阶段1：C响应私钥和陷门提取查询。

$O_{\text{KeyGen}}(T)$：C运行算法$\text{KeyGen}(T, \text{msk}, \text{pk})$，使用产生的$\text{sk}_{u_k}$回应$A$。添加访问策略$T$至一个初始为空的集合$D$中。

$O_{\text{Trap}}(T, W)$：挑战者C首先运行密钥生成算法产生$\text{sk}_{u_k} = \left(T, \left\{\left(D_x, D'_x, D''_x\right) | \forall\, x \in T\right\}\right)$，然后调用算法$\text{Trap}(W, \text{sk}, pk, T)$计算陷门，并将$T_{W'}$返回给$A$。

挑战阶段：C选定一个访问策略T^*。敌手A选择一个属性集合S^*且满足$T^*(S^*) = 1$。运行算法$\text{KeyGen}(T^*, \text{msk}, \text{pk})$生成私钥$sk^*_{u_k}$，$C$随机选择一个关键词集$W^*$，调用加密和陷门生成算法，计算密文$\text{CT}^*$和陷门$T^*_{W^*}$。

猜测阶段：A输出$\hat{W}$发送给C。C调用算法$\text{Enc}(W, S)$计算密文CT^*。在搜索阶段如果陷门和索引匹配$\text{Search}\left(T^*_{W^*}, \text{CT}^*\right) = 1$，则$A$获胜。

假设敌手A在返回$\hat{W}$之前，询问了τ个不同的关键词集，则敌手A最多以概率优势$\text{Adv}_\text{A} = (1/|M| - \tau) + \xi$赢得以上安全游戏。这里$|M| - \tau$为执行查询后剩余关键词集空间的大小，$H_2$是一个单向抗碰撞的哈希函数。$\xi$意味着敌手$A$以可忽略的优势从$H_2\left(\hat{W}\right)$中析出$\hat{W}$。 证毕

定理 2 给定DL假设和一个单向的哈希函数H_2。本方案在随机预言机模型下是选择安全的可抵抗选择关键词集攻击。其中，H_1是一个随机预言机。

证明 挑战者C选择随机元素$g, \hbar, f, Q \in G_0$，$r_1, r_2 \in Z_p$，$\upsilon \in \{0,1\}$和随机元素$R \in G_0$。如果$\upsilon = 0$则$Q = h^{r_1 + r_2}$，否则$Q = R$。挑战者C将元组$(g, \hbar, f, f^{r_1}, g^{r_2}, Q)$发送给$B$。挑战者$C$的角色由算法$B$扮演，敌手$A$和算法$B$之间的安全游戏如下：

系统建立：为了给敌手A提供公共参数pk，算法B首先设置$\hbar = g^a$，$f = g^c$(a, c未知)。然后随机选择$d \in Z_p$，计算$g^b = f^d = (g^c)^d$，隐含设置$b = cd$，$f = h^c$。B随机选择一个t_i阶的多项式$P_i \in Z_p^*$，令主密钥$\text{msk} = (d, P_i)$。最后算法B将公共参数$\text{pk} = (e, p, g, \hbar, f, f^d)$发送给敌手$A$。

敌手A选择属性集S^*并将它发送给算法B。下面将模拟随机预言机$O_{H_1}(j)$如下：

如果属性j没有被询问过：若$j\in S^*$，B选择$\beta_j\in Z_p$，然后将元组$(\alpha_j=0,\beta_j,j)$添加到哈希列表O_{H_1}中并输出g^{β_j}；否则，B选择$\alpha_j,\beta_j\in Z_p$，将元组(α_j,β_j,j)添加到O_{H_1}中并输出$f^{\alpha_j}g^{\beta_j}$。如果属性j被询问过，算法B从O_{H_1}中检索(α_j,β_j)并输出$f^{\alpha_j}g^{\beta_j}$。

询问阶段1：敌手A进行密钥询问和陷门询问。算法B持有初始值为空的关键词集列表L_W。

$O_{\mathrm{KeyGen}}(T)$：敌手A提交访问策略T给算法B且属性集不满足访问策略，即$T(S^*)=0$。为生成密钥，B必须为访问树中每个非叶子节点确定一个d_x次的多项式$Q_x(x)$。首先定义两个函数：PolySat和PolyUnsat[14]。算法B调用函数PolyUnSat$(S^*,T,\hbar)$为访问树T中每个节点x确定一个多项式q_x，隐含设置$q_r(0)=a$。如果T中的每个叶子节点x满足属性集S^*，则可完全确定多项式q_x。若x不满足S^*，则至少得到$g^{q_x(0)}$。B为访问树T中每个叶子节点x定义最终多项式$Q_x(\cdot)=cq_x(\cdot)$，$Q_x(0)=cq_r(0)=ac$，其中$j=\mathrm{att}(x)$。若$j\in S^*$，B随机选择$t\in Z_p$，令$D_x=f^{q_x(0)}g^{\beta_j t}=g^{cq_x(0)}H_1(\mathrm{att}(x))^t$，$D'_x=h^t$，$D''_x=(h^t)^{P_i(u_k)}=h^{t\cdot P_i(u_k)}$；若$j\notin S^*$，$B$随机选择$t'\in Z_p$，设置

$$\left.\begin{aligned}D_x&=g^{(-\beta_j/\alpha_j)\cdot q_x(0)}\cdot\left(f^{\alpha_j}g^{\beta_j}\right)^{t'}\\&=H_1(\mathrm{att}(x))^{t'-(q_x(0)/\alpha_j)}\cdot g^{c\cdot q_x(0)}\\D'_x&=h^{-q_x(0)/\alpha_j}\cdot h^{t'}\\&=h^{t'-(-q_x(0)/\alpha_j)}\\D''_x&=\left(h^{-q_x(0)/\alpha_j}\cdot h^{t'}\right)^{P_i(u_k)}\\&=h^{(t'-(-q_x(0)/\alpha_j))\cdot P_i(u_k)}\end{aligned}\right\}\tag{11}$$

其中，隐含设置$t=t'-(q_x(0)/\alpha_j)$。最后B将私钥$\mathrm{sk}_{u_k}=(D_x,D'_x,D''_x)$发送给敌手$A$。

$O_{\mathrm{Trap}}(T,W^*)$：B运行算法$O_{\mathrm{KeyGen}}(T)$产生私钥sk_{u_k}。B随机选择$s\in Z_p$，调用算法Trap$(W',\mathrm{sk_u},\mathrm{pk})$计算关键词陷门：

$$\left.\begin{aligned}\mathrm{tok}_1&=\prod_{j=1}^{t}\left(g^a g^{bH_2(w'_j)}\right)^s\\\mathrm{tok}_2&=h^{cs}\\\tilde{D}_x&=D_x^s=\left(g^{q_x(0)}H_1\big(\mathrm{att}(x)\big)^{t\cdot P_i(0)}\right)^s\\\tilde{D}'_x&=D'^s_x=h^{ts}\\\tilde{D}''_x&=D''^s_x=h^{tsP_i(u_k)}\end{aligned}\right\}\tag{12}$$

B返回陷门$T_{W^*}=\left(\mathrm{tok}_1,\mathrm{tok}_2,\left\{\tilde{D}_x,\tilde{D}'_x,\tilde{D}''_x\right\}_{x\in T}\right)$给敌手$A$。如果$T(S^*)=1$，则$B$添加关键词集$W^*$到关键词列表$L_W$中。

挑战阶段：敌手A给出两个等长的关键词集$W_0,W_1\notin L_W$，算法B随机选择比特$\upsilon\in\{0,1\}$，运行加密算法计算挑战密文CT^*：

$$\left.\begin{aligned}&E_0=f^{r_1},E_1=h^{r_2}\\&\{E_i=(g^{r_2})^{\beta_j}|\forall i\in S^*\}\\&\{E_j=Q(f^{r_1})^{dH_2(w_j^{\upsilon})}|1\le j\le m\}\end{aligned}\right\}\tag{13}$$

B将挑战密文$\mathrm{CT}^*=(E_0,E_1,\{E_j\},\{E_i\}_{i\in S^*})$发送给敌手$A$。当$Q=\hbar^{r_1+r_2}$，$\mathrm{CT}^*$是明文消息$W_\upsilon$的合法密文；当$Q$是$G_T$中随机元素时，在敌手$A$看来$\mathrm{CT}^*$是随机消息的密文。

询问阶段2：A继续进行类似于Phase1的询问且$W^*\neq W_0,W_1$。

猜测阶段：敌手A输出猜测υ'。若$\upsilon'=\upsilon$，B输出0，则$Q=\hbar^{r_1+r_2}$，否则输出1，表明$Q=R$。

在安全游戏中算法B的优势为

$$\begin{aligned}&\mathrm{Adv_B}\\&=1/2\Big(\Pr\left[B\Big(g,\hbar,f,f^{r_1},g^{r_2},Q=\hbar^{r_1+r_2}\Big)=0\right]\\&\quad+\Pr\left[B\Big(g,\hbar,f,f^{r_1},g^{r_2},Q=R\Big)=0\right]\Big)-1/2\\&=1/2(1/2+\varepsilon+1/2)-1/2=\varepsilon/2\end{aligned}\tag{14}$$

本方案还可达到以下安全性：(1)抗属性共谋：利用Shamir's秘密共享机制，在D_x中嵌入秘密ac，密钥与随机多项式q绑定，使属性不同的DU之间不能结合私钥，因此实现抗属性共谋的目标。(2)抗撤销共谋：$D''_x=h^{t\cdot P_i(u_k)}$中包含与属性相关的随机值$t$和撤销多项式，属性撤销的不同DU不能结合其私钥恢复出$P_i(0)$，实现抗撤销共谋。(3)CSP不可伪造：CSP是诚实但好奇的，可能为了经济利益返回部分错误的搜索结果。CSP基于错误的搜索结果伪造有效的信息通过TPA是不可行的。

证毕

6　方案分析

6.1　功能比较

表1将提出的方案与已有的具有代表性的方案进行对比。从表中可以看出，本文方案的功能性更强。这些特点使得本文所提方案更加符合实际应用需求。

6.2　存储代价比较

群中元素的个数决定着被占有的空间。为了便于比较，定义一些用于存储代价的符号。$|Z_p|$和$|G|$分别表示在域Z_p、群G中元素的比特长度；

表 1 功能比较

方案	多关键词搜索	属性撤销	密钥和密文不需更新	结果可验证
文献[6]	×	×	×	×
文献[9]	×	×	×	×
文献[10]	√	×	×	×
文献[12]	×	×	×	√
文献[13]	×	√	×	×
本文方案	√	√	√	√

$|S|$表示系统中某用户所拥有属性的个数；$|N|$表示系统中所有属性的数量；$|M|$表示访问结构的大小。t表示用户提交搜索关键词的个数；m和l分别表示文件和访问结构中关键词的个数；n_i表示属性i可能值的个数；φ表示搜索结果的个数。接下来表2给出存储代价。

6.3 计算代价比较

分析计算代价之前，给出一些主要耗时的操作符号：双线性对运算P、指数操作E。从表3可以看出本文方案的搜索计算代价为常量，而其他方案的搜索代价和属性个数成正相关。此外由于结果认证机制执行验证算法，本文方案带来额外的计算和存储代价。

6.4 实验模拟

为了分析本文方案和相关文献的实际性能，本文使用真实的数据集和PBC(Pairing-Based Cryptography)库进行了一系列的仿真实验。在实验中，主要考虑指数运算E和对运算P, $|\mathbb{Z}_p|$ =160 bit, $|G|$ = 1024 bit。为了便于描述，本文假设属性数量$|S| = |N| \in [10, 60]$。给出主要算法的实验结果如图2(a)–图2(f)所示。

如图2(a)–图2(c)所示，分别描述了系统建立、加密和陷门生成算法的存储代价。如图2(a)和图2(c)所示，本文方案在建立和陷门生成算法存储代价远远低于文献[6]和文献[12]。从图2(b)和图2(e)可以发现，本文方案和文献[12]存储代价和加密时间相同，事实上当$|S| << |N|$时，本文方案的效率更低。因此本文方案更适合于资源有限的实体，特别是移动终端和传感器节点。

如图2(d)–图2(f)所示，分别描述了系统建立、加密和密文搜索时间。可以发现本文方案在系统建立和密文搜索阶段所需时间是个极小的常量，不受属性数量的影响，而文献[6]，文献[12]与属性数量成线性增长的关系。如图2(e)所示，文献[6]的加密时间较长，约是本文方案和文献[6]的2倍，而本文方案和文献[12]加密时间相差不大。综上所述，本文方案在实际应用中是高效的、可行的。

7 结束语

相比以往的搜索加密方案，本文方案不仅在不降低搜索效率的条件下实现了多关键词搜索，而且引入了TPA来验证搜索结果的正确性，同时实现了用户属性撤销的功能。本文在随机预言机模型下基于DL假设进行了安全性证明，随后通过理论和实验两方面验证方案的可行性。当然，随着实际应用的需要和进一步的研究，设计更安全、高搜索效率、支持动态数据集搜索方案将是未来工作中需要考虑的问题。

表 2 存储代价比较

方案	系统建立算法	密钥生成算法	加密算法	陷门生成算法
文献[6]	$\left(4+\sum_{i=1}^{N} n_i\right)\lvert G\rvert + \left(2+\sum_{i=1}^{N} n_i\right)\lvert Z_p\rvert$	$(2N+2)\lvert G\rvert$	$(2N+2)\lvert G\rvert$	$(2N+1)\lvert G\rvert + \lvert Z_p\rvert$
文献[9]	$9\lvert G\rvert + 5\lvert Z_p\rvert$	$\lvert G\rvert + \lvert Z_p\rvert$	$(5m+2)\lvert G\rvert$	$(6l+2)\lvert G\rvert + \lvert M\rvert$
文献[12]	$(3N+2)\lvert G\rvert + (3N+1)\lvert Z_p\rvert$	$(2N+1)\lvert G\rvert + \lvert Z_p\rvert$	$(N+2)\lvert G\rvert$	$(2N+1)\lvert G\rvert + \lvert Z_p\rvert$
本文方案	$7\lvert G\rvert + (\lvert S\rvert+3)\lvert Z_p\rvert$	$(2\lvert S\rvert+2)\lvert G\rvert + \lvert Z_p\rvert$	$(\lvert S\rvert+m+2)\lvert G\rvert$	$(\lvert S\rvert+4)\lvert G\rvert$

表 3 计算代价比较

方案	系统建立算法	密钥生成算法	加密算法	陷门生成算法	搜索算法	验证算法
文献[6]	$\left(2+\sum_{i=1}^{N} n_i\right)E$	$(2N+2)E$	$(2N+2)E$	$(2N+1)E$	$E+(2N+1)P$	—
文献[9]	$5E$	E	$(6m+3)E$	$(15l+3)E$	$(l+1)E+(6l+1)P$	—
文献[12]	$(3N+1)E+P$	$(2N+3)E$	$(N+2)E$	$(2N+1)E$	$E+(N+1)P$	—
本文方案	$3E$	$(2\lvert S\rvert+2)E$	$(\lvert S\rvert+3)E$	$(2\lvert S\rvert+3)E$	$E+3P$	$(\varphi+1)E+2P$

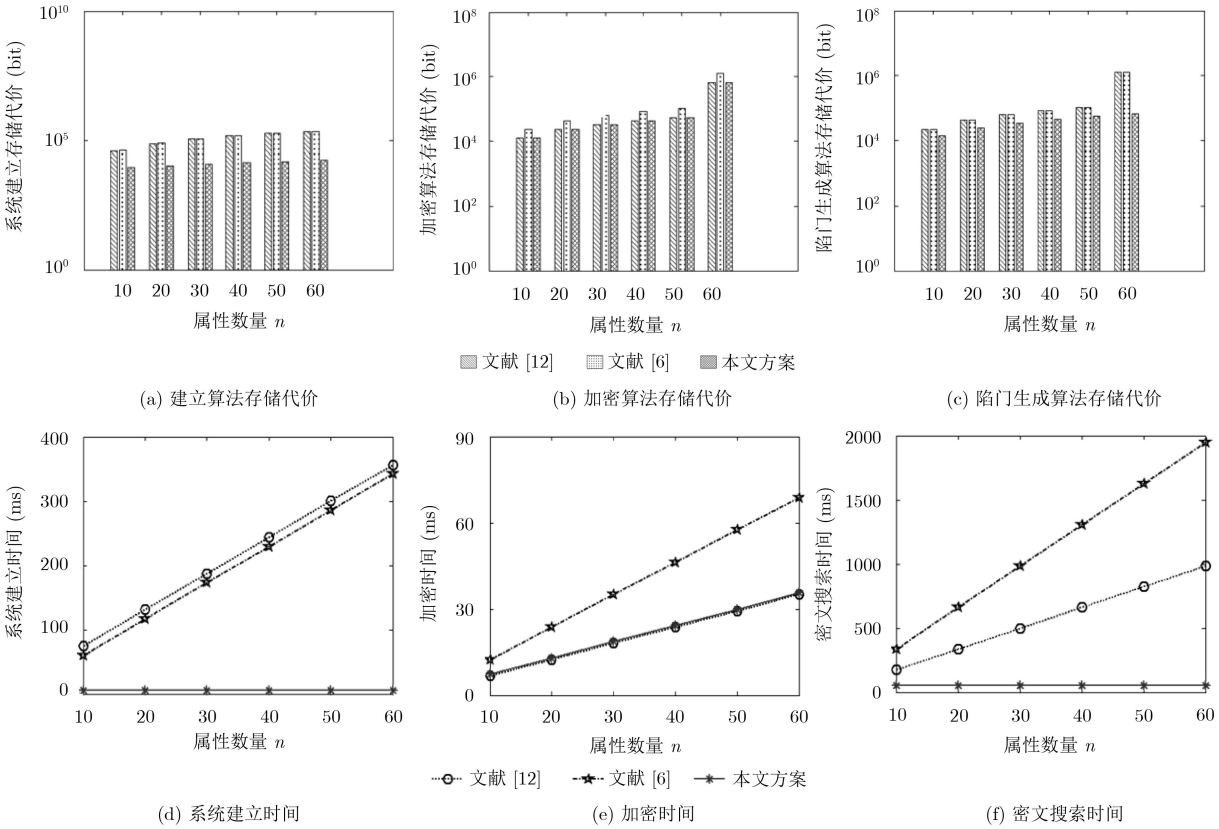

图 2 存储代价与仿真时间对比

参 考 文 献

[1] 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, 2008: 44–55. doi: 10.1109/SECPRI.2000.848445.

[2] BONEH D, CRESCENZO G D, 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.

[3] CURTMOLA R, GARAY J, KAMARA S, *et al*. Searchable symmetric encryption: Improved definitions and efficient constructions[C]. Proceedings of the 13th ACM Conference on Computer and Communications Security, Alexandria, USA, 2006: 79–88. doi: 10.1145/1180405.1180417.

[4] 李双, 徐智茂. 基于属性的可搜索加密方案[J]. 计算机学报, 2014, 37(5): 1018–1024. doi: 10.3724/SP.J.1016.2014.01017.
LI Shuang and XU Zhimao. Attribute-based public encryption with keyword search[J]. *Chinese Journal of Computers*, 2014, 37(5): 1018–1024. doi: 10.3724/SP.J.1016.2014.01017.

[5] YANG Yang and MA Maode. Conjunctive keyword search with designated tester and timing enabled proxy re-encryption function for E-Health clouds[J]. *IEEE Transactions on Information Forensics and Security*, 2017, 11(4): 746–759. doi: 10.1109/TIFS.2015.2509912.

[6] QIU Shuo, LIU Jiqiang, SHI Yanfeng, *et al*. Hidden policy ciphertext-policy attribute-based encryption with keyword search against keyword guessing attack[J]. *Science China* (*Information Sciences*), 2017, 60(5): 1–12. doi: 10.1007/s11432-015-5449-9.

[7] MIAO Yinbin, MA Jianfeng, WEI Fushan, *et al*. VCSE: Verifiable conjunctive keywords search over encrypted data without secure-channel[J]. *Peer-to-Peer Networking and Applications*, 2017, 10(4): 995–1007. doi: 10.1007/s12083-016-0458-z.

[8] MIAO Yinbin, MA Jianfeng, JIANG Qi, *et al*. Verifiable keyword search over encrypted cloud data in smart city[J]. *Computers and Electrical Engineering*, 2017, 65(1): 90–101. doi: 10.1016/j.compeleceng.2017.06.021.

[9] CUI Hui, WAN Zhiguo, DENG R H, *et al*. Efficient and expressive keyword search over encrypted data in the cloud[J]. *IEEE Transactions on Dependable and Secure Computing*, 2016, 15(3): 409–422. doi: 10.1109/TDSC.2016.2599883.

[10] LI Runhe, ZHENG Dong, ZHANG Yinghui, *et al*. Attribute-based encryption with multi-keyword search[C]. IEEE Second International Conference on Data Science in Cyberspace, Shenzhen, China, 2017: 172–177. doi: 10.1109/DSC.2017.97.

[11] 王尚平, 余小娟, 张亚玲. 具有两个可撤销属性列表的密钥策略的属性加密方案[J]. 电子与信息学报, 2016, 38(6): 1406–1411. doi: 10.11999/JEIT150845.
WANG Shangping, YU Xiaojuan, and ZHANG Yaling. Revocable key-policy attribute-based encryption scheme with two revocation lists[J]. *Journal of Electronics & Information Technology*, 2016, 38(6): 1406–1411. doi: 10.11999/JEIT150845.

[12] SUN Wenhai, YU Shucheng, LOU Wenjing, *et al*. Protecting your right: Verifiable attribute-based keyword search with fine-grained owner-enforced search authorization in the cloud[J]. *IEEE Transactions on Parallel and Distributed Systems*, 2016, 27(4): 1187–1198. doi: 10.1109/TPDS.2014.2355202.

[13] 陈燕俐, 杨华山. 可支持属性撤销的基于CP-ABE可搜索加密方案[J]. 重庆邮电大学学报(自然科学版), 2016, 28(4): 545–554. doi: 10.3979/j.issn.1673-825X.2016.04.016.
CHEN Yanli and YANG Huashan. CP-ABE based searchable encryption with attribute revocation[J]. *Journal of Chongqing University of Posts and Telecommunications* (*Natural Science Edition*), 2016, 28(4): 545–554. doi: 10.3979/j.issn.1673-825X.2016.04.016.

[14] GOYAL V, PANDEY O, SAHAI A, *et al*. Attribute-based encryption for fine-grained access control of encrypted data[C]. Proceedings of the 13th ACM Conference on Computer and Communications Security, Alexandria, USA, 2006: 89–98. doi: 10.1145/1180405.1180418.

[15] ZHENG Qingji, XU Shouhuai, and ATENIESE G. VABKS: Verifiable attribute-based keyword search over outsourced encrypted data[C]. IEEE INFOCOM, Toronto, Canada, 2014: 522–530. doi: 10.1109/INFOCOM.2014.6847976.

孙 瑾：女，1977年生，博士，副教授，硕士生导师，研究方向为密码理论与网络安全.

王小静：女，1992年生，硕士，研究方向为密码理论与网络安全.

王尚平：男，1963年生，博士，教授，博士生导师，研究方向为密码理论与网络安全.

任利利：女，1994年生，硕士，研究方向为密码理论与网络安全.