云存储环境下多服务器的密钥聚合可搜索加密方案

张玉磊[①] 刘祥震*[①] 郎晓丽[①] 张永洁[②] 陈文娟[①] 王彩芬[①]

[①](西北师范大学计算机科学与工程学院 兰州 730070)

[②](甘肃卫生职业学院 兰州 730070)

摘 要：密钥聚合可搜索加密不仅可以通过关键字检索密文，还可以减少用户密钥管理的代价和安全风险。该文分析了一个可验证的密钥聚合可搜索加密方案，指出该方案不满足关键字猜测攻击，未经授权的内部用户可以猜测其他用户的私钥。为了提高原方案的安全性，提出了云存储环境下多服务器的密钥聚合可搜索加密方案。所提方案不仅改进了原方案的安全性问题，还增加了多服务的特性，提高了上传和存储的效率，更适合一对多的用户环境。

关键词：密钥聚合；可搜索加密；多服务器；关键字猜测

中图分类号：TP309 **文献标识码**：A **文章编号**：1009-5896(2019)03-0674-06

DOI: 10.11999/JEIT180418

Multi-server Key Aggregation Searchable Encryption Scheme in Cloud Environment

ZHANG Yulei[①] LIU Xiangzhen[①] LANG Xiaoli[①] ZHANG Yongjie[②]
CHEN Wenjuan[①] WANG Caifen[①]

[①](*College of Computer Science and Engineering, Northwest Normal University, Lanzhou* 730070, *China*)

[②](*Gansu Health Vocational College, Lanzhou* 730070, *China*)

Abstract: Key aggregation searchable encryption can not only retrieve ciphertext through keywords, but also can reduce user key management costs and security risks. This paper analyzes a verifiable key aggregation searchable encryption scheme, noting that the scheme does not satisfy keyword guessing attacks, and that unauthorized internal users can guess the private keys of other users. In order to improve the security of the original scheme, a multi-server key aggregation searchable encryption scheme is proposed in the cloud environment. The new scheme not only improves the security of the original solution, but also adds multi-service features, and improves the storage and search efficiency. Therefore, it is more suitable for a one-to-many user environment.

Key words: Searchable encryption; Key aggregation; Multiple servers; Keyword guessing

1 引言

伴随着云存储技术[1]的日益成熟，越来越多的用户将数据存储到云服务器上。为了确保用户数据不被泄漏或者篡改，数据拥有者在上传数据之前可以先对数据加密。这样，用户需要在云服务器端搜索加密的数据，目前常见的方法是使用可搜索加密技术[2]。

2000年，Song等人[3]提出了对称可搜索加密，但是该方案不适用于多用户环境。2004年，文献[4]提出了第1个关键字公钥可搜索加密方案。自此，可搜索加密成为密码学研究的一个热点[5–7]。2014年，Peng等人[8]首次提出无证书公钥可搜索加密方案。Wu等人[9]指出Peng方案存在恶意KGC攻击，并给出了改进方案。2017年，Ma等人[10,11]设计了工业物联网环境下和移动医疗环境下的可搜索加密方案，但是这两个方案都是基于单服务器。同年，黄海平等人[12]提出了一种多服务器的可搜索加密方案，但是该方案的多服务器模型效率较低，数据拥有者上传数据之前都需要自行对数据加密。为此，

收稿日期：2018-05-03；改回日期：2018-11-27；网络出版：2018-12-07
*通信作者： 刘祥震 woliuxiangzhen@foxmail.com
基金项目：国家自然科学基金(61163038, 61262056, 61262057)，甘肃省高等学校科研项目(2017A-003, 2018A-207)

Foundation Items: The National Natural Science Foundation of China (61163038, 61262056, 61262057), The Higher Educational Scientific Research Foundation of Gansu Province (2017A-003, 2018A-207)

需要提出新的多服务器模型。将数据加密托付给子加密服务器，从而提高了上传密文数据的效率。

在传统的可搜索加密中，为了考虑保密性和效率，不同的文档使用不同的密钥，这样，用户持有的密钥数量将随着他们可以检索的文档的数量而变化。这势必带来传输和密钥管理问题。2014年，Chu等人[13]提出密钥聚合的观点，并且应用于云环境。密钥聚合可以将多个密钥合为一个密钥，用户使用一个密钥就可以解密多个文档。2015年，Cui等人[14]提出云存储环境下数据集群密钥聚合可搜索加密，首次将密钥聚合应用于可搜索加密的环境。

2017年，Liu等人[15]提出了一个可验证的密钥聚合可搜索加密方案。经分析，发现该方案不满足关键词不可区分性，通过暴力攻击可以猜测出用户私钥。为了解决该方案中存在的安全性问题，本文首先描述了具体的攻击过程，然后提出了一种云存储环境下多服务器的密钥聚合可搜索加密方案，最后在安全性分析中证明了本文方案的安全性。虽然本文方案在陷门生成阶段效率略低于Liu等人方案，但是克服了Liu等人方案的安全性问题，提高了方案的实用性。

2　基础知识及系统模型

2.1　基础知识

设n为整数，(q, G, G_T, e)是一个双线性映射群。设g是G的生成元，$h_1=g^{\gamma_1}$，$h_2=g^{\gamma_2}$，$\alpha, \beta_2, \gamma_1, \gamma_2 \in Z_q^*$。集合$f$与$\sigma$互素，$\deg(f)=1$，$s'$表示每次询问中没有重复的一次随机数集合，$Z \in G$，以下为群元素序列：

$$\begin{pmatrix} g & g^{\beta_2} & g^{x} \\ g^{\beta_2\alpha^1} & g^{\alpha^{n-\#-1}} & g^{\beta_2\alpha^{n-\#}} \\ g^{\beta_2\alpha^{n+2-\#}} & . & g^{\beta_2\alpha^{n}} \\ g^{\gamma_1\alpha^1} & g^{\gamma_1\beta_2 f} & g^{\gamma_1\alpha^{n}} \\ g^{\gamma_1 f} & . & g^{\beta_2} \\ g^{(\beta_2 H_2+\gamma_2\alpha^1)f} & . & g^{(\beta_2 H_2+\gamma_2\alpha^{\#-1})f} \\ g^{(\beta_2 H_2+\gamma_2\alpha^{\#+1})f} & . & g^{(\beta_2 H_2+\gamma_2\alpha^{n})f} \\ g^{(\beta_2 H_1+\gamma_2\alpha^1)f} & . & g^{(\beta_2 H_1+\gamma_2\alpha^{n})f} \\ g^{\gamma_2\alpha^1} & g^{\gamma_2\alpha^{2n-1}} & g^{\gamma_2\alpha^{2n}} \end{pmatrix}$$

其中，#为$\{1, 2, \cdots, n\}$随机选择的。问题是要区分Z是否等于$g^{\beta_2(\alpha^{n+1-\#}H_2+x)}$或$G$中的某个随机元素。

$(2n, n, 2)$-MSE-DDH困难问题：g为G的生成元。P独立于(D, E)，其中D, E, P定义如下，$E=1$, $P=\beta_2\alpha^{n+1-\#}H_2+x$, D为以下序列

$$\begin{pmatrix} 1 & \beta_2 & x \\ \beta_2\alpha^1 & \alpha^{n-\#-1} & \beta_2\alpha^{n-\#} \\ \beta_2\alpha^{n+2-\#} & . & \beta_2\alpha^{n} \\ \gamma_1\alpha^1 & \beta_2 f & \gamma_1\alpha^{n} \\ \gamma_1 f & . & \beta_2 \\ (\beta_2 H_2+\gamma_2\alpha^1)f & . & (\beta_2 H_2+\gamma_2\alpha^{\#-1})f \\ (\beta_2 H_2+\gamma_2\alpha^{\#+1})f & . & (\beta_2 H_2+\gamma_1\alpha^{n})f \\ (\beta_2 H_1+\gamma_2\alpha^1)f & . & (\beta_2 H_1+\gamma_1\alpha^{n})f \\ \gamma_2\alpha^1 & \gamma_2\alpha^{2n-1} & \gamma_2\alpha^{2n} \end{pmatrix}$$

表明P与(D, E)无关，设$\{x_{i,j}\}$, $\{y_k\}$, z_1为无系数。使得方程$\sum x_{i,j}d_i d_j = \sum y_k d_k p_1 + z_1$，其中多项式$d_i$, d_j, d_k选自D，并且p_1来自P。由于p_1具有α的分数，所以必须从D中具有α分数的元素中选择d_i和d_j中的一个。由于p_1具有f的一部分，因此d_i和d_j中的一个是$\gamma_1\beta_1 f$或$\gamma_1\beta_2 f$，并且$d_k \neq \gamma_1\beta_1 f$和$\gamma_1\beta_2 f$。令$d_k=\gamma_1\alpha^t (t=1, 2, \cdots, n)$，计算$d_k p_1 = \gamma_1\alpha^t((\beta_2 H_2+\alpha^{\#}\gamma_2)f) = \gamma_1\alpha^t\beta_2 H_2 f + \gamma_1\alpha^{t+\#}\gamma_2 f$，所以，$\gamma_1\alpha^t\beta_2 H_2 f = (\gamma_1\gamma_2)(\alpha^t)f$，因此，$d_i$和$d_j$之中，必须有一个$\alpha^t$。然而，$\alpha^t$不在$D$中。因此，无系数$\{x_{i,j}\}$, $\{y_k\}$, z_1，使得方程$\sum x_{i,j}d_i d_j = \sum y_k d_k p_1 + z_1$成立。

因此，$(2n, n, 2)$-MSE-DDH问题困难。

2.2　系统模型

云存储环境下多服务器的密钥聚合可搜索加密方案包括云服务器，加密服务器，数据拥有者和用户4类角色，如图1所示。

(1)云服务器：存储和搜索加密的数据。

(2)加密服务器：将数据拥有者上传的加密数据上传给云服务器。利用数据拥有者提供的密钥对数据加密。

(3)数据拥有者：当数据拥有者需要上传数据时，先将密钥发送给加密服务器，由加密服务器负责加密并上传。

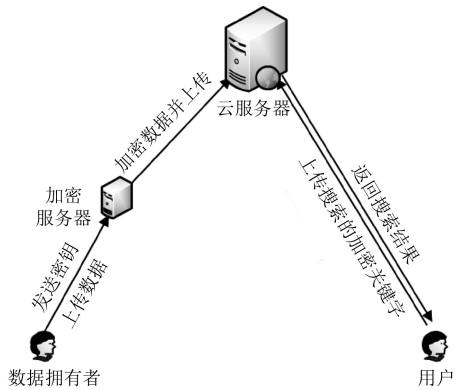

图 1 系统模型

(4)用户：提供搜索的关键字并获取数据。当用户需要获取数据时，首先使用关键字生成用于检索的陷门发送给云服务器，云服务器根据陷门搜索完毕后会返回给用户搜索结果。

3 Liu等人方案及安全性分析

限于篇幅，略去对Liu等人方案的描述，具体算法见文献[15]。

假设A为被攻击的用户，授权私钥为$\mathrm{ak}_{\triangle}$，访问集为S_A。B为内部攻击者，访问集为S_B。本文假定$S_A \neq S_B$, $S_A \cap S_B \neq \varnothing$且$S_A \subseteq S_B$。为了简化描述，文件$F_l \in (S_A \cap S_B)$, F_l中的关键词为$\{w_1, w_2, w_3\}$。

A对搜索的关键词提交陷门$\mathrm{Tr} = \mathrm{ak}_{\triangle} \cdot H_0(w_{\triangle})$，云服务器在测试后返回测试结果。$F_l$是结果之一。

B窃听A提交的陷门Tr后，计算一个聚合密钥集合$k = \{\mathrm{ak}_1, \mathrm{ak}_2, \mathrm{ak}_3\} = \left\{ \dfrac{\mathrm{Tr}}{H_0(w_1)}, \dfrac{\mathrm{Tr}}{H_0(w_2)}, \dfrac{\mathrm{Tr}}{H_0(w_3)} \right\}$, $\mathrm{ak}_{\triangle} \in \{\mathrm{ak}_1, \mathrm{ak}_2, \mathrm{ak}_3\}$。$B$可以正确地猜测$\mathrm{ak}_{\triangle}$。首先，$B$猜测$w_{\triangle} = w_1$和$\mathrm{ak}_{\triangle} = \mathrm{ak}_1$，计算$\mathrm{Tr}' = \mathrm{ak}_1 \cdot H_0(w_2)$或者$\mathrm{Tr}' = \mathrm{ak}_1 \cdot H_0(w_3)$。之后，$B$把$\mathrm{Tr}'$发送给云服务器进行搜索测试。不幸的是，云服务器无法区分搜索是由A提交的还是由B提交的。因此，它会诚实地返回搜索测试结果。若F_l在返回的结果中，陷门Tr'就是有效的，即$\mathrm{ak}_{\triangle} = \mathrm{ak}_1$；否则，陷门$\mathrm{Tr}'$无效。$B$继续对$\mathrm{ak}_2$和$\mathrm{ak}_3$进行类似的测试，直到找到正确的$\mathrm{ak}_{\triangle}$。$B$最多进行3次测试以获得$A$的授权私钥$\mathrm{ak}_{\triangle}$。

在上述攻击中，攻击者B只有在共享F_l的访问权限时才能成功。在获得A的授权私钥$\mathrm{ak}_{\triangle}$后，B可以进一步对与A共享文件访问的人进行相同攻击。

4 云存储环境下多服务器的密钥聚合可搜索加密方案

(1)系统建立算法(Setup)：系统运行此算法来生成系统参数：

(a)生成一个双线性映射群系统(q, G, G_T, e)，其中q为G的安全大素数阶。

(b)选择生成元$g \in G$和单向散列函数$H: \{0,1\}^* \to Z_q^*$。选择k个独立的通用散列函数：$H_1', H_2', \cdots, H_k'$，用于构造一个m位Bloom过滤器，并让另一个单向散列函数$H': G \to \{0,1\}^m$为一个安全的伪随机生成器。

(c)n为数据拥有者拥有文档的最大可能数量，关键词空间为m。公开系统参数为$(p, G, G_T, e, g, n, m, H, H', \{H_1', H_2', \cdots, H_k'\})$。

(2)数据拥有者密钥生成算法(KeyGen)：数据拥有者运行。随机选择$\alpha, \beta_2, \gamma_1, \gamma_2 \in Z_q^*$，计算$h_1 = g^{\gamma_1}$, $h_2 = g^{\gamma_2}$, $v = g^{\beta_2}$, $g_i = g^{\alpha^i}$；其中$i = (1, 2, \cdots, n)$。之后，计算公钥为$h_{1,i} = h_1^{\alpha^i}$，其中$i = (1, 2, \cdots, n)$，$h_{2,i} = h_2^{\alpha^i}$，其中$i = (1, 2, \cdots, n, n+1, \cdots, 2n)$；数据拥有者计算加密钥$(\mathrm{ek}_1, \mathrm{ek}_2) = (g^{\beta_1\gamma_1}, g^{\beta_2\gamma_1})$通过安全通道发送给加密服务器。

因此，数据拥有者的公钥为$\mathrm{pk} = \{v, h_{1,i}, h_{2,i}\}$，私钥为$\mathrm{sk} = \{\beta_2, \gamma_1, \gamma_2, g_i\}$。加密钥为$(\mathrm{ek}_1, \mathrm{ek}_2)$。

(3)云服务器密钥生成算法(KeyGen1)：云服务器执行。随机选择$\beta_1 \in Z_q^*$，计算$u = g^{\beta_1}$。云服务器的公钥$\mathrm{pk}_c = u = g^{\beta_1}$，私钥$\mathrm{sk}_c = \beta_1$。

(4)授权算法(Authorize) $(S) \to \mathrm{ak}$：数据拥有者对新用户进行授权。对任意的子集$S \subseteq \{1, 2, \cdots, n\}$，数据拥有者产生授权密钥$\mathrm{ak} = \prod_{j \in S} g_{n+1-j}^{\beta_2}$，利用安全通道发送给用户。

(5)加密算法(Encryption) $(i, W_i) \to (\triangle_i, \mathrm{CW}_i)$：该算法以索引$i \in \{1, 2, \cdots, n\}$作为输入，加密服务器计算：

(a)随机选择$t_i \in Zq^*$。通过计算生成该文档关键字集W_i的Bloom过滤器：

$$\mathrm{BF}_i = \mathrm{BFGen}(\{H_1', H_2', \cdots, H_k'\}, W_i) \tag{1}$$

(b)随机选择$M \in G$，计算消息密文$\triangle_i$: $c_1 = \mathrm{ek}_1^{t_i} = g^{\gamma_1\beta_1 t_i}$, $c_2 = \mathrm{ek}_2^{t_i} = g^{\gamma_1\beta_2 t_i}$, $c_3 = H'(M) \oplus \mathrm{BF}_i$。

(c)对于该集合W_i中的每个关键字w，计算其密文CW_i: $\mathrm{cw} = \left(v^{H(w)} h_{2,i}\right)^{t_i} = \left(g^{\beta_2 H(w)} g_i^{\gamma_2}\right)^{t_i}$。

最后，算法输出$(\triangle_i, \mathrm{CW}_i)$。

(6)陷门生成(Trapdoor) $(w, \mathrm{ak}) \to (\mathrm{Tr}, S)$：用户运行算法来生成关键字$w$的陷门。随机选择$x \in Z_q^*$，计算$\mathrm{Tr}_1 = \mathrm{ak}^{H(w)} v^x$, $\mathrm{Tr}_2 = u^x = g^{\beta_1 x}$，用户把$(\mathrm{Tr}_1, \mathrm{Tr}_2, S)$提交给云服务器进行搜索询问。

(7)检索(Retrieve) $(\mathrm{Tr}, S, \mathrm{CW}_i, \triangle_i) \to (\mathrm{RST}, \mathrm{PRF})$：云服务器通过以下两步进行测试：

(a)计算Direct $(\mathrm{Tr}, i, S) \to \mathrm{Tr}_i$：对$i \in S$的文档计算陷门$\mathrm{Tr}_{1,i}$, $\mathrm{Tr}_{1,i} = \mathrm{Tr}_1 \cdot \prod_{j \in S, j \neq i} h_{2,(n+1-j+i)} = \mathrm{Tr}_1 \cdot \prod_{j \in S, j \neq i} g_{n+1-j+i}^{\gamma_2}$。

(b)测试(Test) $(\mathrm{Tr}_i, \mathrm{cw}, \triangle_i) \to \delta$：判断$\dfrac{e(\mathrm{pub}, \mathrm{cw})^{\beta_1} e(c_2, \mathrm{Tr}_2)}{e(\mathrm{Tr}_{1,i}, c_1)} = e(h_{2,n+1}, c_1)$是否相等，其中$\mathrm{pub} = \prod_{j \in S} h_{1,(n+1-j)} = \prod_{j \in S} g_{n+1-j}^{\gamma_1}$。若相等，输出“True”；否则，输出“False”。

对于S中的关键字密文集$\{\mathrm{CW}_i\}$和消息密文集$\{\triangle_i\}$, Retrieve算法执行如下：

①对于每个$i \in S$，计算$\mathrm{Tr}_i \leftarrow \mathrm{Direct}(\mathrm{Tr}, i, S)$。

②对于每个$i \in S$，设置$p_1 = c_1$, $p_2 = c_3$和$\mathrm{prf}_i = (p_1, p_2)$。

③重置集合RST，并且对于每个$i \in S$，计算rst_i：对于每个关键字密文$\mathrm{cw} \in \mathrm{CW}_i$，计算$\mathrm{Test}(\mathrm{Tr}_i, \mathrm{cw}, \triangle_i)$，若$\delta$为真，则将相应文档的标识添加到$\mathrm{rst}_i$。

最后，该算法输出S中每个文档集的搜索结果和证明的集合$(\mathrm{RST}, \mathrm{PRF})$。在该方案中，为了效率考虑，集合$S$的PRF和RST只能计算1次。

(8)验证(Verify) $(w, S, \mathrm{RST}, \mathrm{PRF}) \to \mathrm{ACC}$：该算法输入集合$S$，测试关键字$w$和$(\mathrm{RST}, \mathrm{PRF})$。

(a)通过计算来恢复第i个Bloom过滤器：$\mathrm{BF}'_i = H'(M) \oplus p_3$，一旦$\mathrm{BF}'_i$不能恢复，输出$\perp$。

(b)验证关键字w是否存在：

$$\mathrm{acc}_i \leftarrow \mathrm{BFVerify}(\{H'_1, H'_2, \cdots, H'_k\}, \mathrm{BF}'_i, w) \quad (2)$$

5　新方案的安全性分析

5.1　正确性证明

$$\frac{e(\mathrm{pub}, \mathrm{cw})^{\beta_1} e(c_2, \mathrm{Tr}_2)}{e(\mathrm{Tr}_{1,i}, c_1)}$$

$$= \frac{e\left(\prod_{j\in S} g_{n+1-j}^{\gamma_1}, \left(g^{\beta_2 H(w)} g_i^{\gamma_2}\right)^{t_i}\right)^{\beta_1} e\left(g^{\gamma_1\beta_2 t_i}, g^{\beta_1 x}\right)}{e\left(\mathrm{Tr}_1 \cdot \prod_{j\in S, j\neq i} g_{n+1-j+i}^{\gamma_2}, g^{\gamma_1\beta_1 t_i}\right)} = \frac{e\left(\prod_{j\in S} g_{n+1-j}^{\gamma_1}, g^{\beta_2 H(w) t_i \beta_1} g_i^{\gamma_2 t_i \beta_1}\right) e\left(g^{\gamma_1\beta_2 t_i}, g^{\beta_1 x}\right)}{e\left(\mathrm{ak}^{H(w)} v^x \cdot \prod_{j\in S, j\neq i} g_{n+1-j+i}^{\gamma_2}, g^{\gamma_1\beta_1 t_i}\right)}$$

$$= \frac{e\left(\prod_{j\in S} g_{n+1-j}^{\gamma_1}, g^{\beta_2 H(w) t_i \beta_1} g_i^{\gamma_2 t_i \beta_1}\right) e\left(g^{\gamma_1\beta_2 t_i}, g^{\beta_1 x}\right)}{e\left(\left(\prod_{j\in S} g_{n+1-j}^{\beta_2}\right)^{H(w)} \cdot g^{\beta_2 x} \cdot \prod_{j\in S, j\neq i} g_{n+1-j+i}^{\gamma_2}, g^{\gamma_1\beta_1 t_i}\right)} = \frac{e\left(\prod_{j\in S} g_{n+1-j}^{\gamma_1}, g^{\beta_2 H(w) t_i \beta_1} g_i^{\gamma_2 t_i \beta_1}\right)}{e\left(\left(\prod_{j\in S} g_{n+1-j}^{\beta_2}\right)^{H(w)} g^{\gamma_1\beta_1 t_i}\right) e\left(\prod_{j\in S, j\neq i} g_{n+1-j+i}^{\gamma_2}, g^{\gamma_1\beta_1 t_i}\right)}$$

$$= \frac{e\left(\prod_{j\in S} g_{n+1-j}^{\gamma_1}, g_i^{\gamma_2 t_i \beta_1}\right)}{e\left(\prod_{j\in S, j\neq i} g_{n+1-j+i}^{\gamma_2}, g^{\gamma_1\beta_1 t_i}\right)} = \frac{e\left(\prod_{j\in S} g_{n+1-j+i}, g\right)^{\gamma_1\gamma_2 t_i \beta_1}}{e\left(\prod_{j\in S, j\neq i} g_{n+1-j+i}, g\right)^{\gamma_1\gamma_2\beta_1 t_i}} = e\left(g_{n+1}^{\gamma_2}, g^{\gamma_1\beta_1 t_i}\right) = e(h_{2,n+1}, c_1). \quad (3)$$

在数据分享系统中，控制加密服务器的数据拥有者不构成攻击者。在本文方案的框架中，云服务器好奇而诚实。尽管它不会更改云中的任何信息并且诚实地返回搜索结果，但可能会与未经授权的用户勾结，并猜测查询中的关键字。新方案的密文保密性证明过程与原方案一致，以下仅对关键词不可区分性进行分析。

5.2　关键词不可区分性

假设整个文件集为U，文件$F^{\#}$作为搜索请求的挑战文件。对$F^{\#}$没有搜索权限为攻击者。设λ为安全参数，A为敌手，B为挑战者，KS为关键字空间。

定理　若$(2n, n, 2)$-MSE-DDH问题困难，则所提方案在随机预言模型中是语义安全的。

证明　假设存在一个多项式时间敌手A以ε的优势攻击本文方案。B模拟挑战者，以$\dfrac{\varepsilon}{eq_c}$解决$(2n, n, 2)$-MSE-DDH问题。通用关键字空间和文件集被定为大小为m的W和大小为n的U。

(1)开始(Init)：敌手A的挑战文件为$F^{\#}(F^{\#} \subseteq U)$。

(2)系统建立(Setup)：B以群参数(q, G, G_T, e)和$(2n, n, 2)$-MSE-DDH的实例作为输入。对于任何子集$F_{\mathrm{at}} \subset U - F^{\#}$，设置$H_1 = H(w_l)$ $(w_l \neq w_\theta)$且$H_2 = H(w_\theta)$。s'表示查询中没有重复的一次随机数集合。B进一步给出$Z \in G$。如果$Z = g^{\beta_2(\alpha^{n+1-\#H_2+x})}$，则$B$输出1；否则，$B$输出0。

(3)Hash询问：B维护列表$L(w_l, y_l)$，初始为空，当B收到对w_l的询问时：

(a)如果w_l在表L中，B就把相应的元组y_l返回给敌手A。如果$w_l = w_\theta$时，B设置$H(w_l) = y_l = a_\theta$。

(b)否则，$H(w_l) = y_l = a_l$。B将(w_l, y_l)添加到表L中，并把相应的y_l发送给A。这里隐含的设置为$H_1 = H(w_l)$ $(w_l \neq w_\theta)$和$H_2 = H(w_\theta)$。

阶段1　攻击者A按以下方式对授权算法Authorize询问和加密算法Encryption询问：

(4)授权算法(Authorize)询问：A对F_{at}进行授权密钥查询，其中$F_{\mathrm{at}} \subseteq U - F^{\#}$。$B$用相应的授权密钥$\mathrm{ak}_1$答复$A$。授权密钥查询$\mathrm{ak}_1$的结果可以在$(2n, n, 2)$-MSE-DDH实例中组合而成。

(5)加密算法(Encryption)询问：A对指定的文件$F_t(F_t \subseteq U)$的关键字(w_l, y_l)的密文进行询问。B运行Encryption算法并按以下方式回复：

(a)若$F^{\#} \not\subset F_t$，B随机选取$s' \in Z_q^*$并计算密文$\mathrm{cw}_l=(c_1,c_2,c_3)$，其中$c_1=g^{\beta_1\gamma_1 s'}$，$c_2=g^{\beta_2\gamma_1 s'}$，$c_3=H'(M)\oplus \mathrm{BF}_i$，$\mathrm{cw}=(g^{\beta_2 H(w_l)+\gamma_2\alpha^1})^{s'}$。这里$H(w_l)=H_1\cdot\mathrm{cw}$，可以从$(2n,n,2)$-MSE-DDH实例中的元素$g^{\beta_1\gamma_1 s'}$和$g^{\beta_2\gamma_1 s'}$计算出$c_1$与$c_2$。

(b)若$F^{\#} \subseteq F_t$，B随机选取$s' \in Z_q^*$并计算密文$\mathrm{cw}_\theta=(c_1,c_2,c_3)$作为回应：其中$c_1=g^{\beta_1\gamma_1 s'}$，$c_2=g^{\beta_2\gamma_1 s'}$，$c_3=H'(M)\oplus \mathrm{BF}_i$，$\mathrm{cw}=\left(g^{\beta_2 H(w_l)+\gamma_2\alpha^{\#+1}}\right)^{s'}$。这里$H(w_l)=H_1\cdot\mathrm{cw}$，可以从$(2n,n,2)$-MSE-DDH实例中的元素$g^{\beta_1\gamma_1 s'}$和$g^{\beta_2\gamma_1 s'}$计算出$c_1$与$c_2$。

(6)挑战(Challenge)：挑战文件集$F^{\#}$中有两个相同长度的挑战密文cw_0和cw_1。A先前没有对F^*的授权私钥进行询问，也没有对(F_t,w_0)或(F_t,w_1)的密文进行询问，其中$F^{\#} \subseteq F_t$。B的回应如下：

(a)如果$w_\theta \notin \{w_0,w_1\}$，$B$失败并中止游戏。

(b)否则，B随机选取cw_0和$\sigma \in Z_q^*$并用挑战陷门$\mathrm{Tr}=(\mathrm{Tr}_1,\mathrm{Tr}_2)$回应$A$：其中$\mathrm{Tr}_1=\mathrm{ak}^{H(w)}\cdot g^{\beta_2 x}=g^{\beta_2\alpha^{n+1-\#}H_2+\beta_2\sigma}$，$\mathrm{Tr}_2=g^{\beta_1\sigma}$。其中$H(w_\theta)=H_2\cdot\mathrm{Tr}_1$，$\mathrm{Tr}_2$可以从$(2n,n,2)$-MSE-DDH中的元素$g^{\beta_2\alpha^{n+1-\#}H_2+\beta_2\sigma}$和$g^{\beta_1\sigma}$计算得出。

若$Z=g^{\beta_2(\alpha^{n+1-\#}H_2+x)}$，如果$Z$是$G$中的一个随机元素，则挑战陷门Tr从$A$的角度来看是随机的。

阶段2 A以阶段1的方式继续对F_{at}进行私钥询问，对$(F_t,w_l)\neq(F_t,w_0),(F_t,w_1)$进行密文询问，其中$F^{\#} \subseteq F_t$或$(F_t,w_l)$，其中$F^{\#} \not\subset F_t$。$B$响应如下：

(a)若查询的文件集合F_t，其中$F^{\#} \subseteq F_t$和$w_l \in \{w_0,w_1\}$，失败并中止游戏。

(b)若查询的文件为$F^{\#} \subseteq F_t$，其中$w_l \notin \{w_0,w_1\}$，B随机选取随机数$s' \in Z_q^*$，计算密文$\mathrm{cw}_l=(c_1,c_2,c_3)$，其中$c_1=g^{\beta_1\gamma_1 s'}$，$c_2=g^{\beta_2\gamma_1 s'}$，$c_3=H'(M)\oplus \mathrm{BF}_i$，$\mathrm{cw}=(g^{\gamma_2\alpha^1})^{s'}$。可以从$(2n,n,1)$-MSE-DDH实例中的元素$g^{\beta_1\gamma_1 s'}$和$g^{\beta_2\gamma_1 s'}$计算出$c_1$与$c_2$。

(c)如果查询的文件集合F_t，其中$F^{\#} \not\subset F_t$和$w_l \notin \{w_0,w_1\}$，B随机选取$s' \in Z_q^*$，计算密文$\mathrm{cw}_l=(c_1,c_2,c_3)$，其中$c_1=g^{\beta_1\gamma_1 s'}$，$c_2=g^{\beta_2\gamma_1 s'}$，$c_3=H'(M)\oplus \mathrm{BF}_i$，$\mathrm{cw}=(g^{\beta_2 H(w_l)+\gamma_2\alpha^1})^{s'}$。并且$H(w_l)=H_1\cdot\mathrm{cw}$，可以从$(2n,n,2)$-MSE-DDH实例中的元素$g^{\beta_1\gamma_1 s'}$和$g^{\beta_2\gamma_1 s'}$计算得出$c_1$与$c_2$。

(d)如果询问的文件集合F_t，其中$F^{\#} \not\subset F_t$和$w_l \in \{w_0,w_1\}$，让$(w_l,y_l)=(w_\theta,y_\theta)$为列表$L$的元组，$B$随机选择$s' \in Z_q^*$，计算密文，其中$c_1=g^{\beta_1\gamma_1 s'}$，$c_2=g^{\beta_2\gamma_1 s'}$，$c_3=H'(M)\oplus \mathrm{BF}_i$，$\mathrm{cw}=(g^{\beta_2 H(w_l)+\beta_1\alpha^{\#+1}})^{s'}$。这里$H(w_l)=H_2\cdot\mathrm{cw}$，可以从$(2n,n,2)$-MSE-DDH实例中的元素$g^{\beta_1\gamma_1 s'}$和$g^{\beta_2\gamma_1 s'}$计算得出$c_1$与$c_2$。

(7)猜测(Guess)：A输出它的猜测θ'。如果$\theta'=\theta$，B输出1；否则，输出0。若$Z=g^{\beta_2(\alpha^{n+1-\#}H_2+x)}$，挑战成功。因此，当$Z=Z^{\#}=g^{\beta_2(\alpha^{n+1-\#}H_2+x)}$时，正确猜测的概率是$\Pr[\theta'=\theta|Z=Z^{\#}]=\varepsilon$。$Z$是从$G$随机选取的，挑战密文则独立于$A$。那么，$\theta'$等于$\theta$的概率是$\Pr[\theta'=\theta|Z\text{是随机的}]=1/2$。若$B$不中止，则$|\Pr[\theta'=\theta]=1/2|\geq\varepsilon$。假设$A$进行密文询问和授权密钥询问的次数为$q_c$和$q_a$。如果对挑战文件集$F_t(F^{\#}\subseteq F_t)$和挑战密文$\mathrm{cw}_\theta$进行密文询问，则$B$中止。对文件集$F_t(F^{\#}\subseteq F_t)$和挑战密文$\mathrm{cw}_\theta$进行密文查询的概率是$(1/q_c)^{2((2^m-1)-(2^{m-1}-1))}$。对授权密钥进行询问不会使$B$中止。因此，阶段1或阶段2中，$B$不中止的概率至少为

$$1-(1/q_c)^{2((2^m-1)-(2^{m-1}-1))}\geq 1/e \tag{4}$$

在挑战阶段，若A没有选择w_0或w_1，则B中止。由于$\Pr[w_\theta=w_0]=\Pr[w_\theta=w_1]=1/q_c$，则$\Pr[\mathrm{cw}_\theta\neq\mathrm{cw}_0,\mathrm{cw}_1]=(1-1/q_c)^2\leq 1-1/q_c$。

因此，若A在阶段1和阶段2中没有对挑战文件$F^{\#}\subseteq F_t$和挑战密文w_θ进行密文询问，且在挑战阶段没有选择w_0或w_1时，则B就不中止。在阶段1和阶段2中B不中止的事件和在挑战阶段B不中止的事件之间是独立的。因此，当B不中止的概率至少为ε/eq_c。 证毕

6 效率分析

将本文方案与Liu等人方案进行效率对比。其中，T_{sm}代表随机数生成运算耗费的时间，T_{a}代表Hash运算耗费的时间，T_{exp}代表指数运算耗费的时间，T_{mul}代表乘法运算耗费的时间，T_{p}代表双线性对运算耗费的时间。

本文方案使用的基本运算耗费的时间如表1所示。实验环境为戴尔笔记本(I7-4700 CPU @2.60 GHz, 16 GB内存和Ubuntu Linux操作系统)。同时使用了密码函数库(Pairing-Based Cryptography, PBC)。

表2将本文方案与Liu等人方案的陷门生成阶段与加密阶段的效率进行了对比。可以看出，本文方案加密阶段的效率略高于Liu等人方案，陷门生成效率低于Liu等人方案，这是由于为了克服Liu等人

表 1 基本运算耗费的时间(ms)

T_{sm}	T_{a}	T_{exp}	T_{mul}	T_p
0.756	0.267	3.756	0.185	3.611

表 2 效率对比

方案	加密	陷门生成	安全性
Liu等人方案	$2T_{\mathrm{sm}}+2T_{\mathrm{a}}+4T_{\mathrm{exp}}+2T_{\mathrm{mul}}+3T_{\mathrm{p}}\geq 28.273$	$T_{\mathrm{a}}+T_{\mathrm{mul}}\geq 0.452$	低
本文方案	$2T_{\mathrm{sm}}+2T_{\mathrm{a}}+5T_{\mathrm{exp}}+6T_{\mathrm{mul}}\geq 21.936$	$T_{\mathrm{sm}}+T_{\mathrm{a}}+3T_{\mathrm{exp}}+2T_{\mathrm{mul}}\geq 12.661$	高

方案的不安全性，不可避免地增加了计算的开销。

7　结束语

本文首先分析了Liu等人方案提出的可验证的密钥聚合可搜索加密方案的安全性，指出该方案不满足关键词不可区分性，存在安全问题。鉴于此，本文基于密钥聚合技术，提出了一个云存储环境下多服务器的密钥聚合可搜索加密方案，在分析并改进了Liu等人方案的安全性问题的基础上，添加了多服务器属性，提高了上传密文的效率。由于本文方案在陷门生成阶段计算开销略高于Liu等人方案，如何提高陷门生成效率是我们下一步研究的重点。

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] 陆海宁. 可隐藏搜索模式的对称可搜索加密方案[J]. 信息网络安全, 2017(1): 38–42. doi: 10.3969/j.issn.1671-1122.017.01.006.
LU Haining. Searchable symmetric encryption with hidden search pattern[J]. *Netinfo Security*, 2017(1): 38–42. doi: 10.3969/j.issn.1671-1122.017.01.006.

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

[4] DAN B, CRESCENZO G D, OSTROVSKY R, *et al*. Public key encryption with keyword search[J]. *Lecture Notes in Computer Science*, 2004, 3027: 506–522. doi: 10.1007/978-3-540-24676-3_30.

[5] 王尚平, 刘利军, 张亚玲. 一个高效的基于连接关键词的可搜索加密方案[J]. 电子与信息学报, 2013, 35(9): 2266–2271. doi: 10.3724/SP.J.1146.2012.01036.
WANG Shangping, LIU Lijun, and ZHANG Yaling. An efficient conjunctive keyword searchable encryption scheme[J]. *Journal of Electronics & Information Technology*, 2013, 35(9): 2266–2271. doi: 10.3724/SP.J.1146.2012.01036.

[6] CHANG Yujui and WU Jaling. Multi user searchable encryption scheme with constant size keys[C]. IEEE International Symposium on Cloud and Service Computing, Kanazawa, Japan, 2017: 98–103. doi: 10.1109/SC2.2017.22.

[7] 刘振华, 周佩琳, 段淑红. 支持关键词搜索的属性代理重加密方案[J]. 电子与信息学报, 2018, 40(3): 683–689. doi: 10.11999/JEIT170448.
LIU Zhenhua, ZHOU Peilin, and DUAN Shuhong. Attribute based proxy reencryption scheme with keyword search[J]. *Journal of Electronics & Information Technology*, 2018, 40(3): 683–689. doi: 10.11999/JEIT170448.

[8] 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.

[9] WU Tsuyang, MENG Fanya, CHEN Chienming, *et al*. On the security of a certificateless searchable public key encryption scheme[C]. International Conference on Genetic and Evolutionary Computing, Fuzhou, China, 2016: 113–119. doi: 10.1007/978-3-319-48490-7_14.

[10] 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*, 2017, 14(2): 759–767. doi: 10.1109/TII.2017.2703922.

[11] MA Mimi, HE Debiao, KHAN M K, *et al*. Certificateless searchable public key encryption scheme for mobile healthcare system[J]. *Computers & Electrical Engineering*, 2017, 65(5): 413–424. doi: 10.1016/j.compeleceng.2017.05.014.

[12] 黄海平, 杜建澎, 戴华, 等. 一种基于云存储的多服务器多关键词可搜索加密方案[J]. 电子与信息学报, 2017, 39(2): 389–396. doi: 10.11999/JEIT160338.
HUANG Haiping, DU Jianpeng, DAI Hua, *et al*. Multi sever multi keyword searchable encryption scheme based on cloud storage[J]. *Journal of Electronics & Information Technology*, 2017, 39(2): 389–396. doi: 10.11999/JEIT160338.

[13] CHU C K, CHOW S S M, TZENG W G, *et al*. Key aggregate cryptosystem for scalable data sharing in cloud storage[J]. *IEEE Transactions on Parallel and Distributed Systems*, 2014, 25(2): 468–477. doi: 10.1109/TPDS.2013.112.

[14] CUI Baojiang, LIU Zheli, and WANG Lingyu. Key-Aggregate Searchable Encryption (KASE) for group data sharing via cloud storage[J]. *IEEE Transactions on Computers*, 2016, 65(8): 2374–2385. doi: 10.1109/TC.2015.2389959.

[15] LIU Zheli, LI Tong, LI Ping, *et al*. Verifiable searchable encryption with aggregate keys for data sharing system[J]. *Future Generation Computer Systems*, 2017, 78(2): 778–788. doi: 10.1016/j.future.2017.02.024.

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.

刘祥震：男，1991年生，硕士生，研究方向为密码学与信息安全.

郎晓丽：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.

陈文娟：女，1993年生，硕士生，研究方向为密码学与信息安全.