云存储中一种支持可验证的模糊查询加密方案

朱小玉[①] 刘 琴[②] 王国军*[③]
[①](中南大学信息科学与工程学院 长沙 410083)
[②](湖南大学信息科学与工程学院 长沙 410082)
[③](广州大学计算机科学与教育软件学院 广州 510006)

摘 要：针对当前可查询加密方案大多不支持模糊查询的不足，并且无法应对恶意服务器的威胁，云计算亟需为用户提供一种允许拼写错误并且可以验证查询结果正确性的加密方案。同时考虑到云存储中数据经常更新，提出一种动态云存储中支持可验证的模糊查询加密方案。该方案通过编辑距离生成模糊关键词集，并基于伪随机函数、随机排列函数等技术构建安全索引，从而保护用户的数据隐私。通过 RSA 累加器和哈希函数验证查询结果的正确性，用于检测恶意攻击者的非法行为。安全分析证明该方案能够保护用户的隐私安全，并具有可验证性。实验结果表明该方案具有可行性与有效性。

关键词：云存储；隐私保护；模糊查询；可验证查询；加密方案
中图分类号： TP393 **文献标识码**：A **文章编号**：1009-5896(2017)07-1741-07
DOI: 10.11999/JEIT160971

Fuzzy Searchable Encryption Scheme Supporting Verification in Cloud Storage

ZHU Xiaoyu[①] LIU Qin[②] WANG Guojun[③]
[①](*School of Information Science and Engineering*, *Central South University*, *Changsha* 410083, *China*)
[②](*College of Computer Science and Electronic Engineering*, *Hunan University*, *Changsha* 410082, *China*)
[③](*School of Computer Science and Educational Software*, *Guangzhou University*, *Guangzhou* 510006, *China*)

Abstract: Most of the existing searchable encryption schemes can not support fuzzy keyword search, and can not resist threats from malicious server, that the cloud computing needs to provide an encryption scheme, which can support typos and verification of the search result. Considering the data is updated frequently in cloud computing, a verifiable fuzzy searchable encryption scheme in dynamic cloud storage is presented. The proposed scheme constructs the fuzzy keyword set based on edit distance technique, and builds secure index based on pseudorandom function and random permutation, so as to protect the users' data privacy. The RSA accumulator and Hash function are used to verify the correctness of the search result, in order to detect the cheating behavior of the malicious attacker. The security analysis proves that the proposed scheme is privacy preserving and verifiable. The experiment results show that the proposed scheme is efficient.

Key words: Cloud storage; Privacy preserving; Fuzzy search; Verifiable search; Encryption scheme

1 引言

随着云存储的快速发展，可查询加密方案逐渐获得人们的关注和认可。大量的用户通过云盘上传个人文件，然而人们在享用云存储服务带来便利的同时，也面临着敏感信息泄露的风险。在云存储服务中，用户失去了对数据的直接控制权。为了保护用户的隐私信息不被云破解，很多用户会选择将个人的数据加密之后上传。然而，用户经常需要在加密的数据集上进行查询。因此，目前支持加密查询的方案成为了一个亟待解决的问题。

可查询加密方案最早由Song等人[1]提出，随后国内外学者基于不同的功能和设定条件提出了一些可查询加密方案[2–6]。用户在输入查询请求时，经常会出现拼写错误，而精确关键词的可查询加密方案此时就无法返回正确的查询结果。Li等人[7]提出了一种模糊关键词可查询加密方案。Chai等人[8]针对

收稿日期：2016-09-26；改回日期：2017-02-20；网络出版：2017-04-14
*通信作者：王国军 csgjwang@163.com
基金项目：国家自然科学基金(61632009, 61472451, 61272151, 61402161)
Foundation Items: The National Natural Science Foundation of China (61632009, 61472451, 61272151, 61402161)

半诚实且好奇的服务器提出了一种支持验证查询结果完整性的加密方案。Wang等人[9]提出了一种混合云模型下的可验证查询加密方案。Sun等人[10]提出了一种支持验证连接关键词查询结果的加密方案。Zhang等人[11]针对不可信服务器提出了一种验证排序查询结果的加密方案。Kurosawa等人[12]针对恶意云服务器提出了一种可验证查询加密方案。Kamara等人[13]构建了一种动态可查询加密方案，当添加、修改或删除文件后，系统依旧保留了查询的功能，而无需完全重新构建索引。紧接着，将原有方案扩展为并行的动态可查询加密方案[14]。Kurosawa等人[15]提出动态可验证查询加密方案，数据使用者可以动态地更新文件，并且可以检测任何恶意服务器的作弊行为。

尽管以往研究者们分别提出了模糊查询方案和动态可验证查询方案，但目前还没有一个方案可以支持动态云存储中加密数据的可验证模糊查询，这很大程度上减少了将可查询加密方案应用于实际云存储系统中的机会。针对上述问题，本文结合模糊查询方案和动态可验证查询方案，针对恶意云服务器，提出一种动态云存储中支持可验证模糊查询的加密方案。本方案通过编辑距离来定义关键词之间的相似度，利用通配符构造模糊关键词集，基于倒排索引构造安全索引，构造可验证集合验证服务器是否篡改查询结果。本文给出了方案的定义、算法构造、安全分析以及实验分析。本方案支持用户进行模糊查询、动态更新加密数据、验证查询结果的正确性，有利于提升用户的查询体验，促进可查询加密方案的研究与发展。

2 系统模型和相关定义

2.1 系统模型

本文提出支持动态可验证模糊查询的加密方案，该方案由3类实体组成：数据拥有者、数据使用者和云服务器。假设给定包含 n 个文档的明文集合 $D=(d_1,d_2,\cdots,d_n)$，数据拥有者通过一个安全的加密算法将其加密为密文集合 $\widetilde{D}=(\tilde{d}_1,\tilde{d}_2,\cdots,\tilde{d}_n)$。给定预先定义好的 m 个不同的关键词 $W=(w_1,w_2,\cdots,w_m)$，数据拥有者生成安全查询索引 Index 和验证累加值 acc，然后数据拥有者将密文集合 $\widetilde{D}$、安全查询索引 Index 和验证累加值 acc 一起上传到云服务器。数据拥有者可以对云服务器中的文档集合进行任意的增删改查操作。数据使用者发出查询请求 w_a，通过访问控制策略从数据拥有者处生成查询陷门 T_a 和其他的辅助信息。接收到数据拥有者发来的查询陷门后，云服务器计算出查询结果和验证证据，并一起发给数据使用者。数据使用者收到查询结果后，在本地验证查询结果的完整性和正确性，如果验证结果失败，则返回失败；如果验证结果通过，则通过解密算法解密返回的加密文档，获得加密文档对应的明文。支持可验证模糊查询的加密方案模型如图1所示。

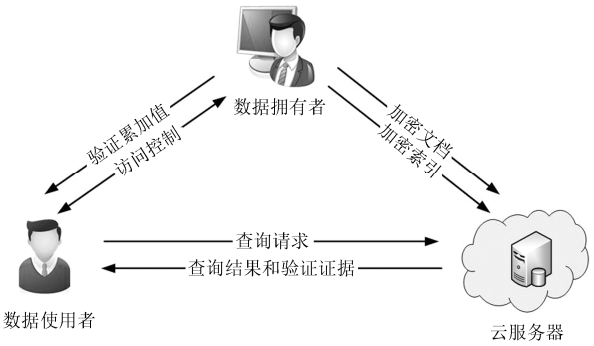

图 1 支持可验证的模糊查询加密方案架构

2.2 方案定义

定义 1 支持动态可验证模糊查询的加密方案：包含 11 个算法 (Keygen, FuzzySet, Enc, SecIndex, Acc, Trapdoor, Search, Proof, Verify, Upd, Dec)，算法定义如下：

$(k,\mathrm{pk})\leftarrow\mathrm{Keygen}(\sigma)$：密钥生成算法，输入安全参数 σ，输出私钥 k 和公钥 pk。

$S_w\leftarrow\mathrm{FuzzySet}(w,\mathrm{ed})$：模糊关键词集生成算法，输入关键词 w 和编辑距离 ed，输出模糊关键词集 S_w。

$\tilde{d}\leftarrow\mathrm{Enc}(k,d)$：加密算法，输入私钥 k 和明文文档 d，输出密文文档 $\tilde{d}$。

$(\widetilde{D},\mathrm{Index})\leftarrow\mathrm{SecIndex}(k,D,\boldsymbol{I})$：安全查询索引生成算法，输入私钥 k、明文文档集合 D 和索引 $\boldsymbol{I}$，输出密文文档集合 $\widetilde{D}$ 和安全查询索引 Index。

$\mathrm{acc}\leftarrow\mathrm{Acc}(\mathrm{pk},\widetilde{D},\mathrm{Index})$：验证累加值生成算法，输入公钥 pk、密文文档集合 $\widetilde{D}$ 和安全查询索引 Index，输出验证累加值 acc。

$T_i\leftarrow\mathrm{Trapdoor}(k,w_i)$：查询陷门生成算法，输入私钥 k 和关键词 w_i，输出查询陷门 T_i。

$\widetilde{D}(w_a)\leftarrow\mathrm{Search}(\mathrm{Index},T_a)$：查询算法，输入安全查询索引 Index 和查询陷门 T_a，输出查询结果 $\widetilde{D}(w_a)$。

$\mathrm{pf}\leftarrow\mathrm{Proof}(\mathrm{pk},\widetilde{D}(w_a),\mathrm{Index},T_a)$：验证证据生成算法，输入公钥 pk、查询结果 $\widetilde{D}(w_a)$、安全查询索引 Index 和查询陷门 T_a，输出验证证据 pf。

$(\mathrm{accept},\mathrm{reject})\leftarrow\mathrm{Verify}(\mathrm{pk},\widetilde{D}(w_a),\mathrm{acc},\mathrm{pf})$：验证算法，输入公钥 pk、查询结果 $\widetilde{D}(w_a)$、累加值 acc 和

验证证据 pf。如果所有的验证都通过，输出验证结果 accept；不然输出 reject。

$(\text{Index}', \widetilde{D}', \text{acc}') \leftarrow \text{Upd}(\text{pk}, \text{Index}, \widetilde{D}, \text{acc}, \text{op}, j, d_j)$：更新算法，输入公钥 pk、安全查询索引 Index、密文文档集合 $\widetilde{D}$、验证累加值 acc、更新操作 $\text{op} \in \{\text{mod},\text{ins},\text{del}\}$、文档编号 j 和文档 d_j，输出新的 Index', $\widetilde{D}'$ 和 acc'。

$d \leftarrow \text{Dec}(k, \tilde{d})$：解密算法，输入私钥 k 和密文文档 $\tilde{d}$，输出明文文档 d。

2.3 安全定义

定义 2　隐私安全：在整个可查询加密过程中，云服务器仅获取上传的加密文档、安全查询索引、验证累加值、查询陷门、查询结果和验证证据。除此之外，云服务器无法获取文档对应的明文、查询陷门对应的查询请求等其他任何信息，从而做到隐私保护。

定义 3　可验证安全：在整个可查询加密过程中，假定恶意攻击者存在篡改用户查询结果等恶意行为，那么用户能够快速识别。

基于定义2、定义3，本文设计的方案应能保证用户的隐私信息在整个查询过程中不被泄露，保证用户查询结果的完整性和正确性。

3　预备知识

3.1 编辑距离

编辑距离可以测量两个字符串之间的相似度，w_1 和 w_2 之间相互转换需要的步骤数用编辑距离 $d(w_1, w_2)$ 来表示。3 个基本操作为：

(1)替换：将字符串中的一个字母替换为另一个。

(2)删除：将字符串中的一个字母删除。

(3)插入：将一个字母插入到一个字符串中。

给定一个关键词 w 和编辑距离 ed，定义 S_w 来表示与关键词 w 满足 $d(w, w') \le \text{ed}$ 关系的 w' 所组成的模糊关键词集合。

3.2 RSA 累加器

RSA 累加器[16]是一种具有时间戳功能的数据认证机制，输入任意大集合，可以输出一个固定大小的摘要，并为集合中的任意元素计算证据来验证元素是否属于这个集合。

假设 G 为循环群，q 为 G 的一个生成元。假设 x', y' 为素数，$x = 2x' + 1$ 和 $y = 2y' + 1$ 为安全大素数。假设 $N = xy$, $G = \{u = v^2 \bmod N, v \in \mathbb{Z}_N^*\}$。对于集合 $S = \{s_1, s_2, \cdots, s_n\}$，计算出一个累加值 $\text{acc}(S) = g^{\prod_{i=1}^{n} P(s_i)} \bmod N$，其中对于 s_i，通过质数生成函数 $P(\cdot)$ 计算出质数 $P(s_i)$。对于集合 S 中的任意元素 $s_j \in S$，生成一个验证证据 $\text{pf}(S) = g^{\prod_{i \ne j} P(s_i)} \bmod N$。紧接着，通过式(1)验证元素 s_j 是否在集合 S 中。

$$\text{acc}(S) \stackrel{?}{=} \text{pf}(S)^{P(s_j)} \bmod N \tag{1}$$

RSA累加器的安全定义来源于强RSA假设。

4　支持可验证的模糊查询加密方案

在支持可验证模糊查询的加密方案中采用倒排索引作为索引结构，为模糊查询结果提供验证机制。首先构造一个 $m \times n$ 的二元矩阵 $\boldsymbol{I} = \{I_{i,j}\}$ 作为索引，如果关键词 w_i 包含在文档 d_j 中，则 $I_{i,j} = 1$；否则 $I_{i,j} = 0$。其中 $\boldsymbol{I}_i$ 代表 $\boldsymbol{I}$ 的第 i 行，d_{n+1} 代表插入的第 $n+1$ 个文档。支持可验证模糊查询的加密方案分为 4 个部分：数据上传、查询、数据下载、数据更新，其中每个部分包含定义 1 中的一个或多个算法。

4.1 数据上传

在系统初始化阶段，数据拥有者通过密钥生成算法 Keygen，生成私钥集合 $k = (k_0, k_1, \text{sk})$，其中 k_0 和 k_1 是伪随机函数 f_k 的密钥，sk 是加密算法 Enc 和解密算法 Dec 的密钥，该密钥可以与数据使用者共享。另外，Keygen 算法也输出公钥集合 $\text{pk} = (N, g)$。数据拥有者使用对称加密算法 Enc 和对称加密密钥 sk 将明文文档集合 D 加密成为加密文档集合 $\widetilde{D}$。

为实现加密数据的模糊查询功能，采用通配符方法来生成模糊关键词集合，使用通配符 * 表示对关键词的基本操作，定义 S_w 为关键词 w 编辑距离为 ed 的关键词模糊集合。例如，通过 FuzzySet 算法，预定义编辑距离为 1，构造出关键词 set 的模糊关键词集 $S_{\text{set},1} = \{\text{set}, *\text{set}, *\text{et}, s*\text{et}, s*t, \text{se}*t, \text{se}*, \text{set}*\}$。

数据拥有者通过调用 SecIndex 算法生成安全查询索引，算法包含以下步骤：

(1)对于所有的关键词 $w_i \in W$，通过 FuzzySet 算法为它们分别构造一个模糊集 S_{w_i}。

(2)对于所有的关键词 $w_i \in W$，通过 Trapdoor 算法为它们分别构造一个查询陷门 T_i。首先计算 $\widetilde{F}_i = \left\{\left[f_{k_0}(w_i')\right]_{1\cdots128}\right\}_{w_i' \in S_{w_i}}$，再计算辅助信息 $\left[f_{k_1}(w_i)\right]_{1\cdots n}$，其中 $\left[f_{k_1}(w_i)\right]_{1\cdots n}$ 指的是 $f_{k_1}(w_i)$ 的前 n 位，最后生成关键词 w_i 对应的陷门 $T_i = \left\{\widetilde{F}_i, \left[f_{k_1}(w_i)\right]_{1\cdots n}\right\}$。

(3)计算 $\tilde{\boldsymbol{I}}_i = \boldsymbol{I}_i \oplus [f_{k_1}(w_i)]_{1\cdots n}$，定义加密索引为 $\tilde{\boldsymbol{I}} = \{\tilde{\boldsymbol{I}}_i\}_{w_i \in W}$。

(4)使用一个随机排列函数 γ 作用于 $\{1,2,\cdots, m\}$，最终获得安全查询索引 $\text{Index} = \{(\widetilde{F}_{\gamma(i)}, \tilde{\boldsymbol{I}}_{\gamma(i)})_{1 \le i \le m}\}$。

文档集合为 $\{(j,\tilde{d}_j)\,|\,1\le j\le n\}$，索引集合为 $\{(j,[\tilde{\boldsymbol{I}}_i]_j)\,|\,1\le i\le m,1\le j\le n\}$，将 RSA 累加器作用于文档集合和索引集合，分别生成累加值 $\mathrm{acc}(D)$ 和 $\mathrm{acc}(\boldsymbol{I})$。数据拥有者通过调用 Acc 算法生成验证累加值 $\mathrm{acc}=(\mathrm{acc}(D),\mathrm{acc}(\boldsymbol{I}))$。

文档验证累加值 $\mathrm{acc}(D)$ 通过式(2)计算：

$$\mathrm{acc}(D)=g^{\prod_{j=1}^{n}P(H(j,H(\tilde{d}_j)))}\bmod N \tag{2}$$

索引验证累加值 $\mathrm{acc}(\boldsymbol{I})$ 通过式(3)计算：

$$\mathrm{acc}(\boldsymbol{I})=g^{\prod_{i=1}^{m}\prod_{j=1}^{n}P(H(j,[\tilde{\boldsymbol{I}}_i]_j))}\bmod N \tag{3}$$

其中，$P(\cdot)$ 是一个质数生成函数，$H:\{0,1\}^*\to\{0,1\}^\sigma$ 是一个无碰撞哈希函数。数据拥有者保存累加值 acc，并将密文集合 $\widetilde{D}$、安全查询索引 Index 和公钥集合 pk 一起上传到云服务器。

4.2 查询

对于查询请求 w_a，首先数据使用者通过 FuzzySet 算法计算出 w_a 对应的模糊关键词集 S_{w_a}，并将 S_{w_a} 发送到数据拥有者。接收 S_{w_a} 后，数据拥有者通过 Trapdoor 算法计算查询陷门 $T_a=\{\widetilde{F}_a,[f_{k_1}(w_a)]_{1\cdots n}\}$，并将 T_a 返回给数据使用者。

云服务器从数据使用者处接收到查询陷门 T_a 后，服务器将 $T_a=\{\widetilde{F}_a,[f_{k_1}(w_a)]_{1\cdots n}\}$ 与查询索引 $\mathrm{Index}=\{(\widetilde{F}_{\gamma(i)},\tilde{\boldsymbol{I}}_{\gamma(i)})_{1\le i\le m}\}$ 进行匹配，查找到 $(\widetilde{F}_a,\tilde{\boldsymbol{I}}_a)\in\mathrm{Index}$，获得索引 $\tilde{\boldsymbol{I}}_a$。然后服务器再利用辅助值 $[f_{k_1}(w_a)]_{1\cdots n}$ 计算 $\boldsymbol{I}_a=\tilde{\boldsymbol{I}}_a\oplus[f_{k_1}(w_a)]_{1\cdots n}$，解密得到 $\boldsymbol{I}_a$。令 $\boldsymbol{I}_a=(e_1,e_2,\cdots,e_n)$，最终服务器计算出查询结果 $\widetilde{D}(w_a)=\{\tilde{d}_j\,|\,e_j=1\}$。

云服务器紧接着通过 Proof 算法计算验证证据 $\mathrm{pf}=(\mathrm{pf}(D),\mathrm{pf}(\boldsymbol{I}))$。

文档验证证据 $\mathrm{pf}(D)$ 通过式(4)计算：

$$\mathrm{pf}(D)=g^{\prod_{e_j=0}P(H(j,H(\tilde{d}_j)))}\bmod N \tag{4}$$

索引验证证据 $\mathrm{pf}(\boldsymbol{I})$ 通过式(5)计算：

$$\mathrm{pf}(\boldsymbol{I})=g^{\prod_{i\ne a}\{\prod_{j=1}^{n}P(H(j,[\tilde{\boldsymbol{I}}_i]_j))\}}\bmod N \tag{5}$$

云服务器最后将查询结果和验证证据 $\{\widetilde{D}(w_a),\mathrm{pf}\}$ 返回给数据使用者。

4.3 数据下载

数据使用者通过 Verify 算法验证云服务器返回的查询结果是否正确。数据使用者验证过程示于表 1。

如果 Verify 算法输出 accept，则数据拥有者可以下载 $\widetilde{D}(w_a)$，并通过 Dec 算法解密成明文文档。如果输出 reject，说明云服务器返回的查询结果不正确。

表 1 数据使用者验证过程

算法 1 数据使用者验证过程

输入：公钥 pk、查询结果 $\widetilde{D}(w_a)$、累加值 acc 和验证证据 pf

输出：accept或reject

(1)对于所有的 $(j,\tilde{d}_j)\in\widetilde{D}(w_a)$，计算 $x_j=P(H(j,H(\tilde{d}_j)))$；

(2)验证 $\mathrm{acc}(D)\overset{?}{=}\mathrm{pf}(D)^{\prod_{e_j=1}x_j}\bmod N$；

(3)如果步骤(2)不通过，则输出reject，否则继续；

(4)根据查询结果 $\widetilde{D}(w_a)$ 重建 $\boldsymbol{I}_a=\{e_j\,|\,e_1,e_2,\cdots,e_n\}$；

(5)计算 $\tilde{\boldsymbol{I}}_a=\boldsymbol{I}_a\oplus[f_{k_1}(w_a)]_{1\cdots n}$；

(6)对于所有的 $1\le j\le n$，计算 $z_j=P(H(j,[\tilde{\boldsymbol{I}}_a]_j))$；

(7)验证 $\mathrm{acc}(\boldsymbol{I})\overset{?}{=}\mathrm{pf}(\boldsymbol{I})^{\prod_{j=1}^{n}z_j}\bmod N$；

(8)如果步骤(7)不通过，则输出reject，否则输出accept。

4.4 数据更新

在动态云存储环境中，数据拥有者可以任意地增加、删除或修改文档。

(1)增加：假设数据拥有者希望添加一个文档 d_{n+1}，首先通过调用 Upd 算法为矩阵索引新增一列。如果文档 d_{n+1} 中包含关键词文档 w_i，令 $I_{i,n+1}=1$，否则令 $I_{i,n+1}=0$。

数据拥有者首先使用加密算法将文档 d_{n+1} 加密成 $\tilde{d}_{n+1}=\mathrm{Enc}_{\mathrm{sk}}(d_{n+1})$。对于 $1\le i\le m$，计算 $b_i=[f_{k_i}(w_i)]_{n+1}\oplus I_{i,n+1}$。计算出 $b_{n+1}=(b_{\gamma(1)},b_{\gamma(2)},\cdots,b_{\gamma(m)})$，然后数据拥有者将 $(\tilde{d}_{n+1},b_{n+1})$ 发送到云服务器。

对于 $1\le i\le m$，云服务器将索引 $\boldsymbol{I}_{\gamma(i)}$ 更新为 $\boldsymbol{I}'_{\gamma(i)}=\boldsymbol{I}_{\gamma(i)}\,||\,b_{\gamma(i)}$，其中 $\|$ 代表连接词。另外，计算出 $\mathrm{acc}(D)'=\mathrm{acc}(D)^{P(H(n+1,H(\tilde{d}_{n+1})))}\bmod N$，$\mathrm{acc}(\boldsymbol{I})'=\mathrm{acc}(\boldsymbol{I})^{\prod_{i=1}^{m}P(H(n+1,b_i))}\bmod N$。然后将 $\mathrm{acc}(D)$ 更新为 $\mathrm{acc}(D)'$，$\mathrm{acc}(\boldsymbol{I})$ 更新为 $\mathrm{acc}(\boldsymbol{I})'$。

最后，云服务器更新密文集合、安全查询索引和验证累加值。

(2)删除：假设数据拥有者希望删除文档 d_j，云服务器收到数据拥有者发出的删除请求后，调用 Upd 算法首先计算 $x_j=P(H(j,H(\tilde{d}_j)))$，计算累加值 $\mathrm{acc}(D)'=\mathrm{acc}(D)^{(x_j)^{-1}}\bmod N$。最后，云服务器删除密文 $\tilde{d}_j$，将累加值 $\mathrm{acc}(D)$ 更新为 $\mathrm{acc}(D)'$。

(3)修改：假设数据拥有者希望将文档 d_j 修改为 d'_j，而 d_j 和 d'_j 拥有相同的关键词。因此，数据拥有者不需要更新安全查询索引。云服务器收到数据拥有者发出的修改请求后，调用 Upd 算法首先计算 $x_j=P(H(j,H(\tilde{d}_j)))$ 和 $x'_j=P(H(j,H(\tilde{d}'_j)))$，其中 $\tilde{d}'_j$ 是 d'_j 的密文。然后计算 $\mathrm{acc}(D)'=\mathrm{acc}(D)^{(x_j)^{-1}d'_j}\bmod N$，将累加值 $\mathrm{acc}(D)$ 更新为 $\mathrm{acc}(D)'$。

5　安全性分析

定理 1　本文提出的支持可验证模糊查询的加密方案可以满足定义2中的隐私安全。

证明　假定 $\mathcal{S}$ 是一个模拟器，$\mathcal{S}$ 首先从敌手 $\mathcal{A}$ 处接收到 $|d_1|,|d_2|,\cdots,|d_n|$ 和 m。对于 $1\le j\le n$，$\mathcal{S}$ 可以模拟出密文文档 $\tilde{d}_j=\mathrm{Enc}_{\mathrm{sk}}\left(0^{|d_j|}\right)$，其中 sk 在 Enc 算法中随机选取，然后生成 $\widetilde{D}'=\{\tilde{d}_1,\tilde{d}_2,\cdots,\tilde{d}_n\}$。对于 $1\le i\le m$，$\mathcal{S}$ 随机选择随机数为 $\widetilde{F}_i$，随机选择 $\tilde{\boldsymbol{I}}_i\in\{0,1\}^n$。用一个随机排列函数 γ 作用于 $\{1,2,\cdots,m\}$，生成 $\mathrm{Index}'=\left\{\left(\widetilde{F}_{\gamma(i)},\tilde{\boldsymbol{I}}_{\gamma(i)}\right)_{1\le i\le m}\right\}$，最后将 $\left\{\widetilde{D}',\mathrm{Index}'\right\}$ 发送给 $\mathcal{A}$。

$\mathcal{A}$ 发出查询请求 w_a，$\mathcal{S}$ 得知查询结果 $\widetilde{D}(w_a)=\left\{\tilde{d}_j\mid e_j=1\right\}$。首先计算 $\left[f'_{k_1}(w_a)\right]_{1\cdots n}=\tilde{\boldsymbol{I}}_a\oplus(e_1,e_2,\cdots,e_n)$，关键词 w_i 对应的陷门为 $T'_a=\left(\widetilde{F}'_a,\left[f'_{k_1}(w_a)\right]_{1\cdots n}\right)$，$\mathcal{S}$ 将 T'_a 发送给 $\mathcal{A}$。

$\mathcal{A}$ 发出增加文档的请求，$\mathcal{S}$ 模拟出 $\tilde{d}_{n+1}=\mathrm{Enc}_{\mathrm{sk}}(0^{|d_{n+1}|})$，对于 $1\le i\le m$，随机选取 $b'_i\in\{0,1\}$，使用一个随机排列函数 γ 作用于 $\{1,2,\cdots,m\}$，并计算出 $b'_{n+1}=\left(b'_{\gamma(1)},b'_{\gamma(2)},\cdots,b'_{\gamma(m)}\right)$，$\mathcal{S}$ 将 $(\mathrm{ins},n+1,\tilde{d}_{n+1},b'_{n+1})$ 发送给 $\mathcal{A}$。

$\mathcal{A}$ 发出删除文档的请求，$\mathcal{S}$ 将 (del,j) 发送给 $\mathcal{A}$。

$\mathcal{A}$ 发出修改文档的请求，$\mathcal{S}$ 模拟出 $\tilde{d}'_j=\mathrm{Enc}_{\mathrm{sk}}\left(0^{\left|\tilde{d}'_j\right|}\right)$，然后将 $\left(\mathrm{mod},j,\tilde{d}'_j\right)$ 发送给 $\mathcal{A}$。

由于加密算法 Enc 是CPA安全的，所以 $\mathcal{A}$ 无法区分密文 $\widetilde{D}$ 和 $\widetilde{D}'$。由于伪随机函数 f 和随机排列函数 γ，导致 $\left(T'_a,\tilde{d}'_j,\tilde{d}'_{n+1},b'_{n+1}\right)$ 和 $\left(T_a,\tilde{d}_j,\tilde{d}_{n+1},b_{n+1}\right)$ 也是不可区分的。所以 $\mathcal{A}$ 无法获知更多的信息，所以保护了隐私安全。　　证毕

定理 2　本文提出的支持可验证模糊查询的加密方案可以满足定义3中的可验证安全。

证明　为了证明本文方案是可验证安全的，需要证明攻击者无法伪造正确的查询结果和验证证据。

假设 $(\widetilde{D}(w_a),\mathrm{pf}(D),\mathrm{pf}(\boldsymbol{I}))$ 是正确的查询结果和验证证据，需要证明攻击者伪造的查询结果和验证证据 $(\widetilde{D}'(w_a),\mathrm{pf}'(D),\mathrm{pf}'(\boldsymbol{I}))$ 无法通过数据使用者的验证算法，需要证明伪造的查询结果和证据与原有的证据不符，即 $(\widetilde{D}(w_a),\mathrm{pf}(D),\mathrm{pf}(\boldsymbol{I}))\ne(\widetilde{D}'(w_a),\mathrm{pf}'(D),\mathrm{pf}'(\boldsymbol{I}))$。分为3种可能的情况：(1) $\widetilde{D}(w_a)=\widetilde{D}'(w_a)$ 且 $(\mathrm{pf}(D),\mathrm{pf}(\boldsymbol{I}))\ne(\mathrm{pf}'(D),\mathrm{pf}'(\boldsymbol{I}))$；(2) $\widetilde{D}(w_a)=\widetilde{D}'(w_a)$ 且 $\{z_j\}\ne\{z'_j\}$；(3) $\widetilde{D}(w_a)\ne\widetilde{D}'(w_a)$ 且 $\{z_j\}=\{z'_j\}$。

接下来证明这3种情况下，验证过程失败的概率可以忽略不计。(1) 因为 $(\mathrm{pf}(D),\mathrm{pf}(\boldsymbol{I}))\ne(\mathrm{pf}'(D),\mathrm{pf}'(\boldsymbol{I}))$，因此验证失败的概率可以忽略不计；(2)因为 $\{z_j\}\ne\{z'_j\}$，在强RSA假设下，$\mathrm{pf}(\boldsymbol{I})$ 验证失败的概率可以忽略不计；(3)因为 $\widetilde{D}(w_a)\ne\widetilde{D}'(w_a)$，这说明存在两种情况 $(j,\tilde{d}_j)\in\widetilde{D}(w_a)$ 和 $(j,\tilde{d}_j)\in\widetilde{D}'(w_a)$ 可以使得 $\tilde{d}_j\ne\tilde{d}'_j$。对于这种情况，由于哈希函数 H 的无冲撞特性，导致 $H(j,H(\tilde{d}_j))\ne H(j,H(\tilde{d}'_j))$。因此，在强 RSA 假设下，由于 $P(H(j,H(\tilde{d}_j)))\ne P(H(j,H(\tilde{d}'_j)))$，$\mathrm{pf}(D)$ 验证失败的概率可以忽略不计。

基于以上分析，攻击者不能伪造出真实可信的查询结果和验证证据，因此本文方案是可以满足可验证安全的。　　证毕

6　实验分析

实验硬件环境为Windows 7操作系统，CPU为Intel Core i5-4590(3.30 GHz)，内存为4 GB，采用Java编程语言实现。数据集为近10年的IEEE INFOCOM论文集，包含超过3500篇文章，通过提取文档中包含的关键词，形成关键词集合。实验采用256位AES对称加密算法来加密和解密文档，采用密钥长度1024位的RSA累加器生成验证证据，采用SHA-256作为哈希函数。

生成模糊关键词集合的时间开销如图 2 所示。在编辑距离变化时，时间开销与关键词数都几乎呈线性增长，而编辑距离为 2 比编辑距离为 1 的时间开销大很多，因为编辑距离越大，生成的模糊关键词集的数目将呈指数级增长。编辑距离是影响模糊查询效率的一个非常重要的因子。

生成安全查询索引的时间开销如图 3 所示。设定编辑距离为 1，安全查询索引的生成时间与文件数呈正相关性。随着文件数增加，关键词数量也不断增加，构造安全查询索引的时间开销随之增加。安全查询索引只需要构造一次，在文档增加、更新、删除时，只需给服务器发送请求，服务器在原有的安全查询索引上进行更新，而无需再次重新构造索引，节省了数据拥有者的时间开销。

查询的时间开销如图 4 所示。查询时间随着文件数的增加呈线性增长，查询陷门由数据拥有者生成并发送给云服务器，云服务器将查询陷门与安全查询索引匹配得到查询结果。模糊关键词集由数据拥有者完成，因而在云服务器端的查询时间开销与文件数呈正相关性，与生成模糊关键词集的开销无关。

验证的时间开销如图 5 所示。验证时间随着文

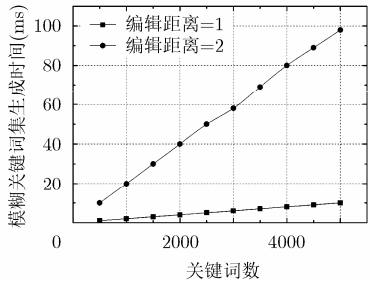

图 2 模糊关键词集生成时间

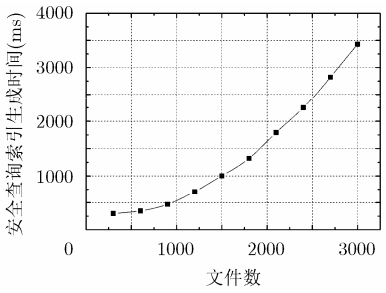

图 3 安全查询索引生成时间

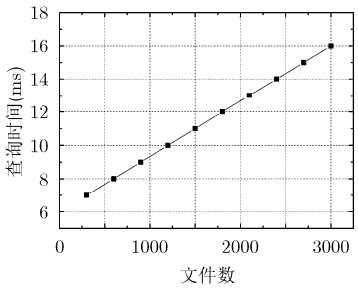

图 4 查询时间

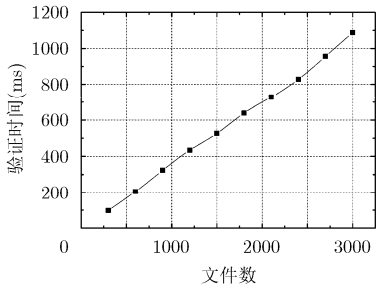

图 5 验证时间

件数的增加而增加，数据使用者首先验证文档的完整性，然后根据查询结果重建索引并验证查询结果的完整性。验证时间与文件数呈正相关性。

综上所述，生成模糊关键词集需要较大的时间开销，因而扩展后的安全查询索引耗时较多，但是构造索引只需一次，而查询和文档更新操作较为频繁，本方案在搜索、更新操作上有较高的效率，可以满足实际环境的需求。

7 结束语

本文在动态云存储环境中提出了一种支持可验证模糊查询的加密方案，该方案不仅支持模糊查询，验证查询结果的正确性，并能够高效地更新文档。通过使用基于编辑距离的方法构造模糊关键词集，再基于 RSA 累加器验证查询结果的正确性。通过安全分析，证明该方案在对抗恶意服务器时是隐私保护和可验证安全的。通过模拟实验表明该方案具有有效性和可行性。当然，该方案在效率上仍存在一定的提升空间，例如模糊关键词集会随着编辑距离的增大而增大。下一步将寻找更为高效的方法来提升方案的效率。

参 考 文 献

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

[2] 林鹏, 江颉, 陈铁明. 云环境下关键词搜索加密算法研究[J]. 通信学报, 2015, 36(Z1): 1–7. doi: 10.11959/j.issn.1000-436x.2015307.

LIN Peng, JIANG Jie, and CHEN Tieming. Application of keyword searchable encryption in cloud[J]. *Journal on Communications*, 2015, 36(Z1): 1–7. doi: 10.11959/j.issn.1000-436x.2015307.

[3] CAO Ning, WANG Cong, LI Ming, *et al.* Privacy-preserving multi-keyword ranked search over encrypted cloud data[J]. *IEEE Transactions on Parallel and Distributed Systems*, 2014, 25(1): 222–233. doi: 10.1109/TPDS.2013.45.

[4] XIA Zhihua, WANG Xinhui, SUN Xingming, *et al.* A secure and dynamic multi-keyword ranked search scheme over encrypted cloud data[J]. *IEEE Transactions on Parallel and Distributed Systems*, 2016, 27(2): 340–352. doi: 10.1109/TPDS.2015.2401003.

[5] FU Zhangjie, SUN Xingming, LIU Qi, *et al.* Achieving efficient cloud search services: Multi-keyword ranked search over encrypted cloud data supporting parallel computing[J]. *IEICE Transactions on Communications*, 2015, 98(1): 190–200. doi: 10.1587/transcom.E98.B.190.

[6] 李真, 蒋瀚, 赵明昊. 一个自主授权的多用户可搜索加密方案[J]. 计算机研究与发展, 2015, 52(10): 2313–2322. doi: 10.7544/issn1000-1239.2015.20150504.

LI Zhen, JIANG Han, and ZHAO Minghao. A discretionary searchable encryption scheme in multi-user settings[J]. *Journal of Computer Research and Development*, 2015, 52(10): 2313–2322. doi: 10.7544/issn1000-1239.2015.20150504.

[7] LI Jin, WANG Qian, WANG Cong, *et al.* Fuzzy keyword search over encrypted data in cloud computing[C]. IEEE

International Conference on Computer Communications, San Diego, CA, USA, 2010: 1–5.

[8] CHAI Qi and GONG Guang. Verifiable symmetric searchable encryption for semi-honest-but-curious cloud servers[C]. IEEE International Conference on Communications, Ottawa, Canada, 2012: 917–922.

[9] WANG J, YU X, and ZHAO M. Fault-tolerant verifiable keyword symmetric searchable encryption in hybrid cloud[J]. *International Journal of Network Security*, 2015, 17(4): 471–483.

[10] SUN W, LIU X, LOU W, *et al*. Catch you if you lie to me: Efficient verifiable conjunctive keyword search over large dynamic encrypted cloud data[C]. IEEE Conference on Computer Communications, Hong Kong, China, 2015: 2110–2118.

[11] ZHANG W, LIN Y, and GU Q. Catch you if you misbehave: Ranked keyword search results verification in cloud computing[J]. *IEEE Transactions on Cloud Computing*, 2015, 6(1): 1–14. doi: 10.1109/TCC.2015.2481389.

[12] KUROSAWA K and OHTAKI Y. UC-secure searchable symmetric encryption[C]. International Conference on Financial Cryptography and Data Security, Kralendijk, Bonaire, 2012: 285–298.

[13] KAMARA S, PAPAMANTHOU C, and ROEDER T. Dynamic searchable symmetric encryption[C]. ACM Conference on Computer and Communications Security, Raleigh, NC, USA, 2012: 965–976.

[14] KAMARA S and PAPAMANTHOU C. Parallel and dynamic searchable symmetric encryption[C]. International Conference on Financial Cryptography and Data Security, Okinawa, Japan, 2013: 258–274.

[15] KUROSAWA K and OHTAKI Y. How to update documents verifiably in searchable symmetric encryption[C]. International Conference on Cryptology and Network Security, Paraty, Brazil, 2013: 309–328.

[16] GENNARO R, HALEVI S, and RABIN T. Secure hash-and-sign signatures without the random oracle[C]. International Conference on the Theory and Applications of Cryptographic Techniques, Prague, Czech Republic, 1999: 123–139.

朱小玉：　女，1989 年生，博士生，研究方向为云计算安全与隐私保护.

刘　琴：　女，1982 年生，博士，助理教授，研究方向为云计算、大数据和隐私保护.

王国军：　男，1970 年生，教授，博士生导师，研究方向为网络计算、可信计算、大数据、隐私保护.