多用户关键词合取密文搜索方案

伍 琦 万常选

(江西财经大学信息管理学院 南昌 330013)

摘 要 随着云计算下数据外包的流行,可搜索加密的重要性日益凸显。针对Yang等提出的多用户关键词合取搜索方案中,用户必须给全所有关键词值这一局限,设计了一个改进方案。随后,详细分析了新方案的正确性、安全性及运行效率。分析表明,新方案在维持原方案安全性的基础上,实现了用户仅需提交部分关键词这一实用功能。新方案运行效率出色,在云计算"瘦客户机"应用背景下有一定的实践价值。最后附带给出了新方案的一个简易实现。

关键词 云计算,可搜索加密,关键词合取搜索

中图法分类号 TN918.1 **文献标识码** A

Multi-user Conjunctive Keyword Search Scheme over Ciphertext

WU Qi WAN Chang-xuan

(School of Information Technology, Jiangxi University of Finance and Economics, Nanchang 330013, China)

Abstract Searchable encryption becomes increasingly significant with the prevalence of data outsourcing in cloud computing. To overcome the flaw in Yang et al.'s scheme that users have to provide all keywords, an improved one was proposed in this paper. The correctness, security and performance of the proposed scheme were analyzed in detail. It shows that the new scheme maintains the security properties of Yang et al.'s scheme but allows users to provide part of keywords. The new scheme performs well in terms of computational complexity, and can be applied to cloud computing where "thin client" is demanded. At last, a simple implementation of the new scheme was given.

Keywords Cloud computing, Searchable encryption, Conjunctive keyword search

1 引言

近年来,云计算逐渐成为业界和学术界的热点。它是IT界继网格计算、分布式计算、并行计算、效用计算、网络存储、虚拟化、负载均衡等之后,一项新兴的革命性技术。Amazon、Google、Yahoo、HP、Intel、Dell等企业相继推出了各自的云计算产品,学术界相关研究成果也不断涌现[1-5]。

随着数据量的日益增大以及计算任务的日益繁重,资源和专业知识均十分有限的云用户往往选择将数据外包。然而,与以往观念不同,人们通常认为云服务器端并不一定为可信的,其存储的数据可能被窃取或被泄露给其他无权限用户,因此,云用户需要将数据加密。但传统的数据加密算法,如DES、AES、RSA等,均未提供搜索功能。无论从带宽还是存储空间来看,当数据量庞大时,将所有密文数据传给云用户令其解密,再抽取想要的部分,是不可行的。

因此,设计对密文数据进行搜索的方案十分必要。Song等[6]首次给出可证机密性、搜索控制、查询隐藏、查询隔离等重要安全性质的定义,并提出相应的可搜索加密方案。从此,可搜索加密引起了人们的广泛关注[7-17]。美国国防部先进研究项目局(DARPA)认为:可搜索加密是可用来在信息聚合系统中,平衡隐私性和国家安全需求的技术进步之一[18]。

Song等[6]在顺序扫描模式下,利用伪随机函数和伪随机数发生器,首先构造了满足可证机密性的基本方案;随后,为了解决该方案中的密钥泄露问题,引入了伪随机函数,提出了一个满足搜索可控制的改进方案。但上述方案存在着关键词泄露问题,Song等使用分组加密函数,构造了一个满足查询隐藏的改进方案;并且提出了具有查询隔离功能的可搜索加密方案。Goh[7]建立了自适应选择关键词攻击下的语义安全模型(IND-CKA),并借助伪随机函数和Bloom过滤器实现了一个IND-CKA安全的索引(Z-IDX),详细分析了其效率和特性;最后,Goh附带给出了Bloom过滤器参数选择的要点及Z-IDX在实际应用中的注意事项。Chang等[8]建立了远程加密数据隐私保护关键词搜索模型(PPSED),并针对用户本地存储字典与否这两种情况分别构造了安全方案,最后给出了对追加数据的支持。Ballard等[9]为实现关键词连结搜索,基于Shamir秘密共享提出了一个安全方案;随后,为解决该方案中的陷门占用空间过大这一问题,提出了一个基于双线性的改进方案。Golle等[10]指出前人对关键词连结搜索的两种解决方法的不足之处,建立关键词连结搜索的安全模型,基于DDH假设提出了一个安全方案;随后,为解决该方案中的通信开销过大问题,基于BDDH假设提出了一个固定通信开销的安全方案。Curtmola等[11]指出文献[7,8]中安全模型的缺

到稿日期:2012-11-07 返修日期:2013-01-07 本文受国家自然科学基金(61163053),江西省教育厅科学技术研究项目(GJJ11730)资助。

伍 琦(1984-),男,博士生,软件设计师,主要研究方向为信息安全,E-mail:wuqiocjzd@126.com;万常选(1962-),男,博士,教授,博士生导师,CCF高级会员,主要研究方向为Web数据管理、XML信息检索、金融数据挖掘和情感计算。

陷，建立可搜索对称加密模型(SSE)，并利用伪随机函数和伪随机排列构造了两个方案。这两个方案具有较高的计算效率，能大大减少服务器端计算量。此外，Curtmola 等给出了多用户下的可搜索对称加密模型和方案。Yang 等[12]首先提出一个仅泄露少量信息的基本方案，随后分别提出一个不泄露任何信息的强安全方案以及一个利用元数据实现查询提速的方案。

以上方案均为对称可搜索加密，即其查询请求和索引均用同样的密钥。Boneh 等[13]以邮件系统为例，建立关键词搜索公钥加密模型(PEKS)，分别基于 BDH 假设和陷门排列各提出了一个方案，最后附带给出了利用 Jacobi 符号的构造。Abdalla 等[14]详细分析了 PEKS 的一致性问题，证明文献[13]中的方案是计算一致的，再提出了一个统计一致的新方案，最后给出把匿名 IBE 方案转为安全 PEKS 方案的方法。Hwang 等[15]建立关键词连结搜索公钥加密模型(PECK)，随后基于 DLDH 假设提出了一个方案；与前人方案比较得到，新方案的密文和私钥占用空间最少；最后，Hwang 等把模型和方案扩展到多用户情况下。Yang 等[16]建立多用户下的可搜索加密模型，基于 BLS 短签名的安全性提出了一个方案；分析表明，新方案实现了查询机密性、查询不可伪造性和用户可撤销性；最后，Yang 等附带给出新方案的查询提速技巧。Yang 等[17]在与文献[16]略有不同的应用背景下提出了一个类似的方案，它具有同等的安全性；最后附带给出该方案在责任隔离、关键词合取及模糊搜索方面的拓展。

目前来说，文献[16,17]中方案对多用户背景考虑得最为详尽，运行效率也最高。但经笔者研究发现，文献[17]中最后对关键词合取的拓展功能不全：仅当用户提交所需记录的全部关关键词值时，才可取到相应记录。为此，本文提出一个改进方案。分析表明，新方案在维持原有安全性的基础上，实现了用户提交所需记录的关键词集的任意子集，均可取到相应记录。新方案运行效率同样出色，具有一定的应用前景。

本文第 2 节介绍预备知识；第 3 节回顾文献[17]中的方案；第 4 节提出新方案；第 5 节分析新方案的正确性、安全性和运行效率；第 6 节给出新方案的一个简易实例；最后是总结。

2 预备知识

下面给出本文将要用到的双线性映射的有关知识。

定义 1 给定质数 p 及 p 阶群 G_1 和 G_2，若二元映射 $\hat{e}: G_1\times G_1\to G_2$ 满足以下条件，则称 $\hat{e}$ 为双线性映射。

1)双线性：对 $\forall g_1, g_2\in G_1$，$\forall x_1, x_2\in Z_p^*$，有 $\hat{e}(x_1g_1, x_2g_2)==\hat{e}(g_1,g_2)^{x_1x_2}$；

2)非退化性：若 g 是 G_1 中的生成元，则 $\hat{e}(g,g)$ 是 G_2 中的生成元；

3)可计算性：对任意 $g_1, g_2\in G_1$，$\hat{e}(g_1,g_2)$ 可在多项式时间内算得。

3 文献[17]的方案

该方案包含 3 个实体：企业、云服务器、用户集。其中，企业是数据拥有者，也是用户管理者。它负责设置系统参数，添加/删除用户，以及向云服务器端写入记录。用户集中任一用户均可递交查询。云服务器负责存储数据以及处理用户的查询请求。

该方案包含 6 个算法：*Setup*、*AddUser*、*RemoveUser*、*WriteRecord*、*GenQuery*、*Search*。一开始企业调用 *Setup* 设置系统参数，再反复调用 *AddUser* 将各合法用户添加进来。企业随时可调用 *RemoveUser* 将任一用户移除。随后，企业多次调用 *WriteRecord* 向云服务器端写入记录。任一合法用户随时可调用 *GenQuery* 生成搜索请求。收到搜索请求后，云服务器调用 *Search* 以搜索相关记录。

各算法具体如下：

(1)$Setup(1^K)$：企业设置公共参数 G_1, G_2 和 $\hat{e}(\hat{e}: G_1\times G_1\to G_2$ 是双线性映射)；选 G_1 中生成元 g；选带密钥散列函数 $h(\cdot)$，散列函数 $h_1(\cdot)$ 和 $h_2(\cdot)$；选 $x\in_R Z_p^*$，并设 $MK_{ENT}=x$；为对称加密算法 $Enc(\cdot)$ 选随机记录加密密钥 ek。

(2)$AddUser(MK_{ENT}, ek, u)$：企业选 $x_u\in_R Z_p^*$，设 $qk_u=x_u$；计算 $hk_u=\frac{MK_{ENT}}{x_u}g$；把 qk_u 和 ek 通过安全渠道发给用户 u；把 hk_u 通过安全渠道发给云服务器，令其在 U-Hkey 列表中加一项 (u, hk_u)。

(3)$RemoveUser(u)$：企业指示云服务器从 U-Hkey 列表中删掉 (u, hk_u) 项。

(4)$WriteRecord(MK_{ENT}, ek, d_i)$：为写入一条记录 d_i 到数据库 D'，企业首先生成该记录关键词 $d_i.w_1, d_i.w_2, \cdots, d_i.w_t$ 对应的索引，步骤如下：企业计算 $e_w=\hat{e}(h_1(d_i.w_1)\oplus(h_1(d_i.w_2)\oplus\cdots\oplus(h_1(d_i.w_t), MK_{ENT}g)$ 和 $k=h_2(e_w)$，设 $Indx(d_i.w)=\langle m, h_k(m)\rangle$(其中 m 是随机值)。随后，企业用 ek 计算 $Enc(d_i)$。最后，企业把 $d_i'=\langle Indx(d_i.w), Enc(d_i)\rangle$ 传给云服务器。

(5)$GenQuery(qk_u, w_1', w_2', \cdots, w_t')$：用户 u 计算 $q_u(w_1', w_2', \cdots, w_t')=qk_u(h_1(w_1')\oplus h_1(w_2')\oplus\cdots\oplus h_1(w_t'))$，输出 $(u, q_u(w_1', w_2', \cdots, w_t'))$ 作为他/她对关键词 $w_1', w_2', \cdots, w_t'$ 的搜索请求。

(6)$Search(q_u(w_1', w_2', \cdots, w_t'), hk_u, D')$ 收到查询请求 $(u, q_u(w_1', w_2', \cdots, w_t'))$ 后，云服务器在 U-Hkey 列表中查找 (u, hk_u)。若没找到，输出 $\perp$。否则，它计算 $k'=h_2(\hat{e}(q_u(w_1', w_2', \cdots, w_t'), hk_u))$ 并设 $RPLY_{q_u(w_1', w_2', \cdots, w_t')}=\phi$。随后，云服务器扫描整个数据库 D'，对每项形如 $\langle m_i, c_i\rangle$ 的索引项 $Indx(d_i.w)$，若 $c_i=h_{k'}(m_i)$，则令 $RPLY_{q_u(w_1', w_2', \cdots, w_t')}=RPLY_{q_u(w_1', w_2', \cdots, w_t')}\cup\{Enc(d_i)\}$。

该方案的问题在于：企业调用 $WriteRecord(MK_{ENT}, ek, d_i)$ 时，记录所有关键词值 $d_i.w_1, d_i.w_2, \cdots, d_i.w_t$ 对应的索引值被合在一起。因此，用户调用 $GenQuery(qk_u, w_1', w_2', \cdots, w_t')$ 时，也必须给出所需记录的所有关键词值，才能找到相应记录。

例如，某员工信息数据库有 3 个关键词域，分别为“姓名”、“职位”、“出生年月”。那么，用户为了查询 1970 年 3 月出生的总经理张三的信息，必须给全 3 个关键词“张三”、“总经理”、“1970.3”才行。显然，这与实际需求严重不符。实际应用中，只要用户给出{“张三”，“总经理”，“1970.3”}的任意非空子集，就能取到该员工的信息。

为此，本文提出一个新的多用户关键词合取搜索方案(Multi-user Conjunctive Keyword Search，MCKS)。

4 MCKS 方案

为与笔者相关工作匹配，本文对应用背景略作调整。在 MCKS 中，有 3 个实体：项目经理、云服务器和程序员群。项目经理是程序员管理者，它负责设置系统参数及招募/撤消程序员。任一程序员均可上传文件和提交查询。云服务器负责存储数据以及处理程序员的查询请求。

MCKS 包含 7 个算法：*Setup*、*Enroll*、*Revoke*、*GenIndex*、*WriteFile*、*GenQuery*、*Search*。一开始项目经理调用 *Setup* 设置系统参数，再反复调用 *Enroll* 将各程序员招募进来。项目经理随时可调用 *Revoke* 来撤消因离职、调动等原因脱离项目组的程序员。程序员上传文件时，首先调用 *GenIndex* 来生成相应关键词的索引，再调用 *WriteFile* 向云服务器端写入文件。程序员随时可调用 *GenQuery* 生成搜索请求。收到搜索请求后，云服务器调用 *Search* 来搜索相关文件。

各算法具体如下：

(1) $Setup(1^K)$：项目经理选安全质数 p 及 Z_p^* 下生成元 g；选群 W，G_1，G_2，以及 G_1 下的生成元 g'；设置双线性映射 $\hat{e}: G_1 \times G_1 \to G_2$；选密码学散列函数 $h: W \to G_1$；选非对称加密算法 $Enc(\cdot): Z_p^* \times Z_p^* \to Z_p^* \times Z_p^*$；选 $x \in_R Z_p^*$，令 $k_{PM}=x$；设用户集 $U=\phi$；令云服务器为其建一个根目录。

(2) $Enroll(k_{PM}, u)$：项目经理为程序员 u 生成密钥对 $(k_u, pk_u=g^{k_u})$，计算 $k_u'=\frac{k_{PM}}{k_u}g'$。项目经理设 $U=U\cup\{u\}$，把 u 和 k_u' 一并通过安全渠道发给云服务器。随后，云服务器在根目录下建一个名为 u 的子目录，把 k_u' 放在里面。

(3) $Revoke(u)$：项目经理设 $U=U-\{u\}$，令云服务器删掉目录 u 下的 k_u'。

(4) $GenIndex(k_u, w=\{w_1,\cdots,w_l\}; k_u')$：为生成关键词集 w 的索引，程序员 u 选 $rw \in_R Z_p^*$，把 u 和 $r_w h(w_i)$ $(1\leqslant i\leqslant l)$ 发给云服务器。若目录 u 下无 k_u'，云服务器输出 $\perp$。否则，它把 $e_{w,i}=\hat{e}(r_w h(w_i), k_u')$ $(1\leqslant i\leqslant l)$ 发给程序员 u。程序员 u 计算 $a_{w,i}=e_{w,i}^{k_u}$ $(1\leqslant i\leqslant l)$，设索引 $Ind(w)=(r_w, a_{w,1}, a_{w,2},\cdots,a_{w,l})$。程序员 u 输出 $Ind(w)$。

(5) $WriteFile(k_u, m, pk_c; k_u')$：为了向云服务器端写文件 $m\in Z_p^*$，程序员 u 调用 $GenIndex(k_u, m.w=\{w_1,\cdots,w_l\}; k_u')$ 来得到 $Ind(m.w)$。随后程序员 u 用 pk_u 加密 m，把 $(u, Enc_{pk_u}(m), Ind(m.w))$ 发给云服务器。云服务器把 $(Enc_{pk_u}(m), Ind(m.w))$ 放在目录 u 下。

(6) $GenQuery(k_u, Q=\{I_1,\cdots,I_n, w_{I_1}',\cdots,w_{I_n}'\})$：程序员 u 计算 $query_u(Q)=k_u\sum_{j=1}^{n}h(w_{I_j}')$，输出 $q=(u, query_u(Q), I_1,\cdots,I_n)$ 作为对关键词集 Q 的查询请求。

(7) $Search(q, k_u')$：收到查询请求 $q=(u, query_u(Q), I_1,\cdots,I_n)$ 后，若目录 u 下无 k_u'，云服务器输出 $\perp$。否则，它计算 $k'=\hat{e}(query_u(Q), k_u')$ 并设 $Reply_q=\phi$。对目录 u 下的每个元组 $(Enc_{pk_u}(m), Ind(m.w))$，云服务器设 $Reply_q=Reply_q\cup\{Enc_{pk_u}(m)\}$ 当且仅当 $k'^{r_w}==\prod_{j=1}^{n}a_{w,I_j}$。云服务器输出 $Reply_q$。

不难看出，若把 MCKS 应用到前文所述的员工信息数据库下，用户 u 调用 $GenQuery(k_u, Q=\{1, \text{“张三”}\})$、$GenQuery(k_u, Q=\{2,3, \text{“总经理”}, \text{“1970.3.”}\})$、$GenQuery(k_u, Q=\{1,2, \text{“张三”}, \text{“总经理”}\})\cdots$，均可取到该员工信息。

5 MCKS 方案分析

5.1 正确性分析

定理 1 对任意合法程序员 $u(Enroll(k_{PM}, u)$ 被调用且 $Revoke(u)$ 未被调用)，任意 $n\leqslant l$，任意 $1\leqslant I_j\leqslant l$ $(1\leqslant j\leqslant n)$，任意 w_{I_j}' $(1\leqslant j\leqslant n)$，有

$$Seqrch(GenQuery(k_u, Q=\{I_1,\cdots,I_n, w_{i_1}',\cdots,w_{i_n}'\}), k_u')$$
$$=\{Enc_{pk_u}(m) \mid m.w_{I_j}=w_{I_j}' (1\leqslant j\leqslant n)\}$$

证明：云服务器执行 $Search(u, query_u(Q), I_1,\cdots,I_n, k_u')$ 时，对目录 u 下的每个元组 $(Enc_{pk_u}(m), Ind(m.w))$，有

$$k'^{r_w}=\hat{e}(query_u(Q), k_u')^{r_w}=\hat{e}(k_u\sum_{j=1}^{n}h(w_{I_j}'), k_u')^{r_w}$$

由双线性映射的性质，有

$$\prod_{j=1}^{n}a_{w,I_j}=\prod_{j=1}^{n}e_{w,I_j}^{k_u}=\prod_{j=1}^{n}\hat{e}(r_w h(m.w_{I_j}), k_u')^{k_u}$$
$$=\prod_{j=1}^{n}\hat{e}(k_u h(m.w_{I_j}), k_u')^{r_w}=\hat{e}(\sum_{j=1}^{n}k_u h(m.w_{I_j}), k_u')^{r_w}$$
$$=\hat{e})(k_u\sum_{j=1}^{n}h(m.w_{I_j}), k_u')^{r_w}$$

不难看出，当 $m.w_{I_j}=w_{I_j}'$ $(1\leqslant j\leqslant n)$ 时，必有 $k'^{r_w}=\prod_{j=1}^{n}a_{w,I_j}$，$Enc_{pk_u}(m)$ 将被放入输出集；当 $\exists j$，使得 $m.w_{I_j}\neq w_{I_j}'$ 时，由于 h 的碰撞抵抗性，$k'^{r_w}=\prod_{j=1}^{n}a_{w,I_j}$ 的概率可忽略。

由此，定理 1 得证，MCKS 方案正确。

5.2 安全性分析

就目前来看，文献[17]在多用户单关键词背景下给出的查询隐私性、查询不可伪造性和用户可撤销性的定义，是可搜索加密领域最强的安全性定义。笔者参照文献[17]，分别给出 MCKS 下的相应定义。

不可避免地，程序员递交的查询会泄露给服务器一些信息。例如，两次返回结果相同的查询极有可能是针对相同的关键词集。查询隐私性确保除了服务器明显可观察到的信息外，查询不再泄露任何其他信息，其具体定义如下：

定义 2 设整个目录树为 D，令 $Id(Enc_{pk_u}(m))$ 表示 $Enc_{pk_u}(m)$ 在目录树中的唯一标识，$Id(u)$ 表示目录 u 在目录树中的唯一标识，t 为多项式限制的正整数，令 $q_j=(u_j, query_{u_j}(I_1,\cdots,I_{n_j}, w_{I_1}',\cdots,w_{I_{n_j}}'), I_1,\cdots,I_{n_j})$ $(1\leqslant j\leqslant t)$ 是程序员 $U_t=(u_1,\cdots,u_t)$ 发出的 t 个查询请求，令 $Reply_t$ 是相应的回复。则云服务器对这 t 个查询的视图为 $V_t=(D, \{q_j\}_{1\leqslant j\leqslant t}, Reply_t)$，这 t 个查询的迹为 $T_t=(\{|D_u|\}_{u\in U}, |U|, Id(U_t), Id(Reply_{q_1}),\cdots, Id(Reply_{q_t}))$，其中 $|D_u|$ 为目录 u 下的形如 $Enc_{pk_u}(m)$ 的文件个数，$Id(U_t)=\{Id(u_j)\}_{1\leqslant j\leqslant t}$，$Id(Reply_{q_j})$ 为其内所有的标识信息集合。对于任意信息，若云服务器可从 V_t 中求得，则其均可仅由 T_t 算得，称方案满足查询隐私性。

除了程序员本人 u 之外，任何其他实体都应无法捏造 u 的查询请求，即方案应满足查询不可伪造性。

定义 3 设程序员 u 的合法查询请求集为 $q_u=\{q \mid q=GenQuery(k_u, I_1,\cdots,I_n, w_{i_1}',\cdots,w_{i_n}')\}$，其中 $I_j\in_R\{1,2,\cdots,l\}$，$w_{I_j}'\in_R W$ $(1\leqslant j\leqslant n)$。对任意程序员 u，若任意概率多项式时间敌手 T 不知道 k_u，则 $Prob[T(.)\in q_u]$ 可忽略，那么，称方案满足查询不可伪造性。

定义 4 对任意概率多项式时间敌手 T，若其不知道 k_u'，令其与云服务器之间进行如下游戏：

1）T 任选 $w=\{w_1,\cdots,w_l\}$，调用 $GenIndex(k_u,w=\{w_1,\cdots,w_l\};k_u')$ 得到 $Ind(w)$。该步骤可重复任意多次。

2）T 任选 1）中未出现过的 $w_1^*=\{w_{1,1},\cdots,w_{1,l}\}$ 和 $w_2^*=\{w_{2,1},\cdots,w_{2,l}\}$，分别调用 $GenIndex(k_u,w_1^*;k_u')$ 和 $GenIndex(k_u,w_2^*;k_u')$。云服务器取 $b\in_R\{1,2\}$，只对 $GenIndex(k_u,w_b^*;k_u')$ 作出响应。最后 T 输出对 b 的猜测 b'。

若必有 $Prob[b'==b]-\frac{1}{2}$ 可忽略，则称方案具有用户可撤消性。

下面分别证明新方案的相应安全性质。

定理 2 当 $h(\cdot)$ 是伪随机函数，$Enc(\cdot)$ 是伪随机排列时，MCKS 方案满足查询隐私性。

证明：要证明 V_t=所有信息均包含在 T_t 中，可尝试用 T_t 模拟出一个与 V_t 计算上不可区分的视图 V_t^*，步骤如下：

首先，模拟者选 $x_j^*\in Z_p^*\ (1\leqslant j\leqslant|U|)$，设 $x^*=\prod_{j=1}^{|U|}x_j^*$。

·模拟 $\{q_i\}_{1\leqslant i\leqslant t}$：对 $(1\leqslant i\leqslant t)$，若 $\neg\exists j(j<i\wedge Id(Reply_{q_j})==Id(Reply_{q_i}))$，选 $u_i^*\in_R U$，取 x_i^* 作为其私钥，选 $n_i\in_R Z_{l+1}^*$，选 $I_1,\cdots,I_{n_i}\in_R\{1,2,\cdots,l\}$，取 $r_{I_1},\cdots,r_{I_{n_i}}\in_R G_1$，设 $q_i^*=(u_i^*,x_i^*\sum_{j=1}^{n}r_{I_j},I_1,\cdots,I_{n_i})$；否则，设 $q_i^*=q_j^*$。易见，若 $h(\cdot)$ 是伪随机函数，则 q_i^* 与实际的 q_i 是计算上不可区分的。

·模拟 D：模拟者建根目录并在其下建立 U 中所有元素相应的子目录。对于 $\{q_i^*\}_{1\leqslant i\leqslant t}$ 中涉及到的 u_i^*，在其下放 $k_{u_i^*}'=\frac{x^*}{x_i^*}g'$；其余子目录下的 $k_u'\in_R G_1$。由于 x_i^* 的随机性，易知 $\frac{x^*}{x_i^*}g'$ 与实际的 $\frac{k_{PM}}{k_u}g'$ 是不可区分的。所有子目录 u 下形如 $Enc_{pk_u}(m)$ 的元素均设为 $E_m^*\in_R Z_p^*$。易见，当 $Enc(.)$ 是伪随机排列时，E_m^* 与实际密文数据是不可区分的。当 $1\leqslant i\leqslant t$ 时，对 $Id(Reply_{q_i})$ 标识出的所有文件，选 $r_w^*\in_R Z_p^*$，对 q_i^* 涉及到的 $I_j\ (1\leqslant j\leqslant n_i)$，设 $a_{w,I_j}^*=\hat{e}(r_{I_j},x^*g')^{r_w^*}$，其余位置的 $a_{w,l'}^*$ 设为随机值。所有 $Id(Reply_{q_i})$ 均未涉及到的文件，其索引设为随机值。

·模拟 $Reply_t$：当 $1\leqslant i\leqslant t$ 时，对 $Id(Reply_{q_i})$ 涉及到的所有标识，输出其相应 E_m^* 的集合。

由此，MCKS 方案满足查询隐私性。

定理 3 当 $h(\cdot)$ 是随机预言机时，MCKS 方案满足查询不可伪造性。

证明：此处借用文献[19]中的 BLS 短签名，其定义略述如下：

$G_1,G_2,\hat{e},g'$ 定义同上，$h:\{0,1\}^*\to G_1$ 为碰撞抵抗散列函数，用户 u 的密钥对为 $(k_u\in Z_p^*,pk_u=k_ug')$。用户 u 对消息 m 的签名为 $\sigma=k_uh(m)$。接收方验证 $\hat{e}(g',\sigma)\stackrel{?}{=}\hat{e}(pk_u,h(m))$。

文献[19]中已证明，当 $h(\cdot)$ 是随机预言机时，BLS 短签名具有存在不可伪造性。易知，MCKS 方案的查询不可伪造性可平凡归约到 BLS 短签名的存在不可伪造性，此处不再赘述。

项目经理调用 $Revoke(u)$ 后，必须剥夺程序员 u 的一切搜索能力，即将程序员 u 撤消。

定理 4 MCKS 方案满足用户可撤消性。

证明：对任意关键词 w_i，其相应

$$a_{w,i}=e_{w,i}^{k_u}=\hat{e}(r_wh(w_i),k_u')^{k_u}=\hat{e}(h(w_i),k_{PM}g')^{r_w}$$

显然，若 T 不知道 k_u'，仅由 k_u 无法求得 $k_{PM}g'$。而为了确保 $h(w_i)^{r_w}$ 的随机性（避免云服务器观察出 w_i 之间的异同），T 必须保证 r_w 的随机性。因此，$a_{w,i}$ 对于 T，与 G_2 下的随机元素是不可区分的。因此，T 仅能凭空猜测 b'，即 $Prob[b'==b]-\frac{1}{2}$ 可忽略。所以，MCKS 方案满足用户可撤消性。

5.3 运行效率分析

Setup 阶段的计算涉及质量，此处不作讨论。以下仅讨论各实体的计算时间复杂度，其通信时间复杂度及空间复杂度不作赘述。

·项目经理在招募程序员时，进行 1 次模乘、1 次模逆、1 次点乘、1 次模幂。

·云服务器在生成索引和写文件时均进行 l 次双线性映射；在执行搜索时，进行 1 次双线性映射、$|D_u|$ 次点乘、$|D_u|*(n-1)$ 次模乘。

·程序员在生成索引时，进行 $2l$ 次点乘、l 次散列；在写文件时，进行 $2l$ 次点乘、l 次散列、1 次加密；生成查询请求时，进行 1 次点乘、n 次散列、$(n-1)$ 次模加。

各实体计算量如表 1 所列。

表 1 各实体计算量表

计算量 / 实体 / 阶段	项目经理	云服务器	程序员
Enroll	1Mul+1Inv+1Pt+1Pow	\	\
Revoke	\	\	\
GenIndex	\	lBL	2lPt+lHash
WriteFile	\	1BL	2lPt+lHash+1Enc
GenQuery	\	\	nHash+1Pt+(n−1)Add
Search	\	1BL+|Du|Pt+|Du|*(n−1)Mul	\

文献[20]中显示，最耗时的运算是双线性，其次是点乘和模幂，再次是模乘和模加，最后是散列函数。由表中可见，双线性运算全部由服务器端承担，且集中在数据写入阶段，而搜索时仅需执行 1 次双线性运算。可见，MCKS 非常符合云计算的“瘦客户机”理念，具有良好的应用前景。

6 MCKS 实例

限于篇幅，仅给出 MCKS 的一个简易示例。MCKS 应用于实际系统时，需要更庞大的参数和更严密的组织结构。

笔者在 JavaSE1.6 下，以 Eclipse3.7.2 为 IDE 完成了 MCKS 的全部 7 个算法，具体如下：

首先，项目经理调用 $Setup(1^9)$ 生成安全质数 $p=359$，选取 z_p^* 下生成元 $g=117$；选群 $W=\{0|1\}^*$，G_1 为超奇异椭圆曲线 $E/F_{5743}:y^2=x^3+x$ 中所有阶为 p 的点形成的子群，$G_2=F_{5743}$，G_1 下生成元 $g'=(101,4204)$；$\hat{e}$ 为变形映射 ϕ：

$$\begin{cases}(x,y)\to(-x,yi)\\ O\to O\end{cases}$$

下的 Tate 对；密码学散列函数 h 为 $h(w)=GI[SHA\text{-}1(w) \bmod (p-1)+1]$（即将关键词 w 的 *SHA*-1 散列结果模到 G_1 中的某个点上）；非对称加密算法 *Enc*(.)取为 Elgamal 算法；随机选 $k_{PM}=173$；用户集 U 置为空集。

随后，项目经理分别调用 *Enroll*（"Alice"），*Enroll*（"Bob"），*Enroll*（"Carl"）将程序员 Alice、Bob 和 Carl 分别招募进来。其中，Alice 的密钥对为(104，282)，k_{Alice}'为(1570，5144)；Bob 的密钥对为(180，242)，k_{Bob}'为(2594，1875)；Carl 的密钥对为(113，183)，k_{Carl}'为(5724，3262)。

本实例中，所有程序文件均有 3 个关键词域：子项目名、主要功能、版本号。任一程序文件的所有关键词域均填满，并且不同关键词域下的值不同。

随后，Alice 4 次调用 *WriteFile*(·)，分别向服务器端写入 *a*. c、*b*. c、*c*. c、*d*. c 这 4 个程序文件，其关键词集分别为{"project1"，"print"，"1. 0"}、{"project1"，"display"，"1. 0"}、{"project1"，"display"，"2. 0"}、{"project2"，"test"，"1. 0"}。

此时，服务器端的 Alice 目录下文件如图 1 所示。

名称	修改日期	类型	大小
a.a1	2013/3/2 17:27	A1 文件	1 KB
a.a2	2013/3/2 17:27	A2 文件	1 KB
a.a3	2013/3/2 17:27	A3 文件	1 KB
a.c	2013/3/2 17:27	c_file	1 KB
a.rw	2013/3/2 17:27	RW 文件	1 KB
b.a1	2013/3/2 17:27	A1 文件	1 KB
b.a2	2013/3/2 17:27	A2 文件	1 KB
b.a3	2013/3/2 17:27	A3 文件	1 KB
b.c	2013/3/2 17:27	c_file	1 KB
b.rw	2013/3/2 17:27	RW 文件	1 KB
c.a1	2013/3/2 17:27	A1 文件	1 KB
c.a2	2013/3/2 17:27	A2 文件	1 KB
c.a3	2013/3/2 17:27	A3 文件	1 KB
c.c	2013/3/2 17:27	c_file	1 KB
c.rw	2013/3/2 17:27	RW 文件	1 KB
d.a1	2013/3/2 17:27	A1 文件	1 KB
d.a2	2013/3/2 17:27	A2 文件	1 KB
d.a3	2013/3/2 17:27	A3 文件	1 KB
d.c	2013/3/2 17:27	c_file	1 KB
d.rw	2013/3/2 17:27	RW 文件	1 KB
ku'	2013/3/2 17:08	文件	1 KB

图 1　服务器端 Alice 目录下文件列表

其中，*a*. *c* 为原程序文件 *a*. *c* 的密文，*a*. *rw* 为 r_w，*a*. *a*1、*a*. *a*2、*a*. *a*3 依次为 $a_{w,1}$、$a_{w,2}$、$a_{w,3}$。其余文件类似，不再赘述。

同样地，Bob 3 次调用 *WriteFile*(·)，分别向服务器端写入 *e*. java、*f*. java、*g*. java 这 3 个程序文件，其关键词集分别为{"project1"，"test"，"1. 0"}、{"project2"，"calc"，"1. 0"}、{"project2"，"calc"，"1. 1"}；Carl 3 次调用 *WriteFile*(·)，分别向服务器端写入 *h*. cpp、*i*. cpp、*j*. cpp 这 3 个程序文件，其关键词集分别为{"project1"，"mining"，"1. 0"}、{"project1"，"mining"，"1. 1"}、{"project2"，"mining"，"1. 0"}。

Alice 可调用得到服务器端回复密文 *b*. c 和 *c*. c。随后，Alice 将两文件解密，其内容同原明文，如图 2 所示。

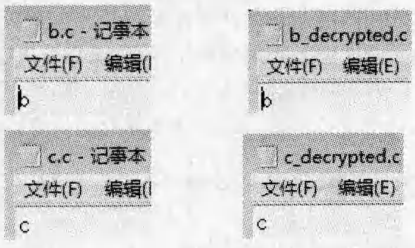

图 2　明密文比对图

同样地，Bob 可调用 *GenQuery*(k_{Bob}，Q={1，2，"*project2*"，"*calc*"})得到服务器端回复密文 *f*. java 和 *g*. java；Carl 可调用 *GenQuery*(k_{Carl}，Q={2，3，"*mining*"，"1. 0"})得到服务器端回复密文 *h*. cpp 和 *j*. cpp。

现项目经理调用 *Revoke*("Bob")来撤消程序员 Bob。此后，若 Bob 再调用 *GenQuery*(·)，服务器将拒绝，如图 3 所示。

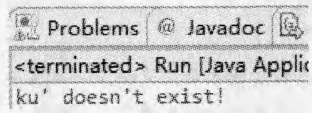

图 3　服务器拒绝响应图

结束语　本文针对文献[17]中的方案必须给全记录所有关键词这一缺陷，提出了一个改进方案。经分析看到，该方案能满足用户仅给出部分关键词这一需求，并维持了原方案的 3 个安全性质。由实验结果看到，新方案的运算量较小，且集中在服务器端，对日益强调便携设备的云计算服务，具有一定的实践价值。最后，笔者编程实现了新方案的一个简易实例。

参 考 文 献

[1] 冯登国，张敏，张妍，等. 云计算安全研究[J]. 软件学报，2011，22(1)：71-83

[2] 张逢喆，陈进，陈海波，等. 云计算中的数据隐私性保护与自我销毁[J]. 计算机研究与发展，2011，48(7)：1155-1167

[3] 李乔，郑啸. 云计算研究现状综述[J]. 计算机科学，2011，38(4)：32-37

[4] 杨健，汪海航，王剑，等. 云计算安全问题研究综述[J]. 小型微型计算机系统，2012，33(3)：472-479

[5] 伍琦，万常选，李国林. 一个改进型云存储共享方案[J]. 计算机科学，2012，39(8)：99-103

[6] Song D X，Wagner D，Perrig A. Practical Techniques for Searches on Encrypted Data[C]// Proceedings of IEEE Symposium on Security and Privacy (S&P'00). 2000：44-55

[7] Goh E J. Secure Indexes[OL]. http://crypto. stanford. edu/ eujin/papers/ secureindex/secureindex. pdf，2003

[8] Chang Y C，Mitzenmacher M. Privacy Preserving Keyword Searches on Remote Encrypted Data[C]// Proceedings of Applied Cryptography and Network Security(ACNS'05). LNCS 3531，2005：442-455

[9] Ballard L，Kamara S，Monrose F. Achieving Efficient Conjunctive Keyword Searches over Encrypted Data[C]// Proceedings of International Conference on Information and Communications Security(ICICS'05). 2005：414-426

[10] Golle P，Staddon J，Waters B. Secure Conjunctive Keyword Search over Encrypted Data[C]// Proceedings of Applied Cryptography and Network Security (ACNS'04). 2004：31-45

[11] Curtmola R，Garay J，Kamara S，et al. Searchable symmetric encryption：improved definitions and efficient constructions[C]// Proceedings of ACM Conference on Computer and Communications Security (CCS'06). 2006：79-88

[12] Yang Z Q，Zhong S，Wright R. Privacy-Preserving Queries on Encrypted Data[C]// Proceedings of European Symposium on Research in Computer Security (ESORICS'06). LNCS 4189，2006：479-495

(下转第 173 页)

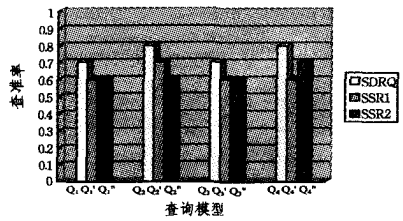

图 6 查准率

综上可以看出，本文提出的方法在权重分配和语义距离评估方面是合理的。一方面，通过对查询及 RDF 的预处理，可以在很大程度上减少由于查询松弛及同源词替换导致的时间性能上的不足，另一方面，在查准率以及查全率上，本文的方法均有良好的表现。

结束语 为了解决 RDF 查询返回结果为空或少量的问题，本文提出基于语义的 RDF 近似查询的处理方法。首先，初始查询通过 RDFS 蕴含规则进行松弛，之后利用权重对松弛查询进行语义选择并进行同源词替换，最后利用语义距离选取与初始查询在语义上相近的结果。在此基础上，给出基于语义的 RDF 近似查询处理的算法。实验结果表明，本文提出的方法能够为用户提高更多更为准确的查询结果，并且有较好的查询响应时间。目前 RDF 查询(包括近似查询)方法主要关注的是 RDF 显式表示的信息，基于推理机制的 RDF 查询能够抽取出 RDF 非显式表示、但能由 RDF 显式表示信息推导出的信息。未来我们将为本文提出的近似查询方法提供推理机制。

参 考 文 献

[1] Berners-Lee T, Handler J, Lassila O. The Semantic Web[M]. Scientific American, 2001, 184: 34-43

[2] Miller E, Swick R, Brickley D. Resource Description Framework RDF[C]// Recommendation, W3C. 2004

[3] Clark K G. RDF Data Access Use Cases and Requirements[C]// W3C Working Draft. March 2005

[4] Hurtado C A, Poulovassilis A, Wood P T. A relaxed approach to RDF querying[C]// Proceedings of the 5th International Semantic Web Conference. LNCS, 2006, 4273: 314-328

[5] Hurtado C A, Poulovassilis A, Wood P T. Query relaxation in RDF[J]. Journal of Data Semantics, 2008, 10: 31-61

[6] Poulovassilis A, Wood P T. Combining approximation and relaxation in semantic web path queries[C]// Proceedings of the 2010 International Semantic Web Conference. LNCS, 2010, 6496: 631-646

[7] Huang H, Liu C F, Zhou X F. Computing relaxed answers on RDF databases[C]// Proceedings of the 9th International Conference on Web Information Systems Engineering. LNCS, 2008, 5175: 163-175

[8] Andreasen T, Bulskov H, Knappe R. From ontology over similarity to query evaluation[C]// Proceedings of the 2nd CoLogNETElsNET Symposium-Questions and Answers: Theoretical and Applied Perspectives. 2003: 39-50

[9] Maedche A, Staab S. Measuring similarity between ontologies [C]// Proceedings of the 13th International Conference on Knowledge Engineering and Knowledge Management. LNCS, 2002, 2473: 251-263

[10] Guo Y, Pan Z, Hein J. An evaluation of knowledge base systems for large OWL datasets[C]// Proceedings of the 3rd International Semantic Web Conference. LNCS, 2004, 3298: 274-288

[11] Jena S D B[OL]. http://jena.hpl.hp.com/wiki/SDB

[12] Zhu L, Ma Q, Liu C N. Semantic-distance based evaluation of ranking queries over relational databases[J]. Journal of Intelligent Information Systems, 2010, 31: 415-445

(上接第 151 页)

[13] Boneh D, Crescenzo G, Ostrovsky R, et al. Public key encryption with keyword search[C]// Proceedings of Annual International Conference on the Theory and Applications of Cryptographic Techniques (Eurocrypt'04). 2004: 506-522

[14] Abdalla M, Bellare M, Catalano D, et al. Searchable Encryption Revisited: Consistency Properties, Ration to Anonymous IBE, and Extensions[C]// Proceedings of International Cryptology Conference (CRYPTO'05). LNCS 3621, 2005: 205-222

[15] Hwang Y H, Lee P J. Public Key Encryption with Conjunctive Keyword Search and Its Extension to a Multi-User System[C]// Proceedings of International Conference on Pairing-Based Cryptography(Pairing'07). LNCS 4575, 2007: 2-22

[16] Yang Y J, Bao F, Ding X H, et al. Multiuser private queries over encrypted databases[J]. Journal of Applied Cryptography, 2009, 1(4): 309-319

[17] Yang Y J, Lu H B, Weng J. Multi-user private keyword search for cloud computing[C]// Proceedings of Third IEEE International Conference on Cloud Computing Technology and Science (CloudCom'11). 2011: 264-271

[18] DARPA Information Science and Technology Study Group. Privacy with security[R]. Technical report. http://www.cs.berkeley.edu/-tygar/papers/ISAT-final- briefing.pdf, 2002-12

[19] Boneh D, Lynn B, Shacham H. Short signatures from the Weil pairing[J]. Journal of Cryptology, 2004, 17(4): 297-319

[20] Zhu R, Yang G M, Wong D. An efficient identity-based key exchange protocol with KGS forward secrecy for low-power devices[C]// Proceedings of Internet and Network Economics First International workshop (WINE'05). LNCS 3828, 2005: 500-509

(上接第 155 页)

[15] Shamus Software Ltd., Miracl library [OL]. http://www.shamus.ie/index.php? page=home

[16] Ren K, Lou W, Zeng K, et al. On broadcast authentication in wireless sensornetworks[J]. IEEE Trans. on Wireless Commun., 2007, 6(11): 4136-4144

[17] Boneh D, Franklin M. Identity-based encryption from the Weil pairing[A]// CRYPTO 2001, 2001[C]. New York: Springer-Verlag, 2001: 213-229

[18] Barreto P, Kim H, Bynn B, et al. Efficient algorithms for pairing-based cryptosystems[A]// CRYPTO 2002, 2002[C]. New York: Springer-Verlag, 2002: 354-368

[19] Bao F, Deng R H, Zhu H. Variations of Diffie-Hellman problem [A]// Proc. ICS, 2003[C]. New York: Springer-Verlag, 2003: 301-312