计算机研究与发展
Journal of Computer Research and Development
DOI:10.7544/issn1000-1239.2015.20150504
52(10):2313-2322,2015

一个自主授权的多用户可搜索加密方案

李 真[1,2] 蒋 瀚[1] 赵明昊[1]
[1](山东大学计算机科学与技术学院 济南 250101)
[2](山东财经大学计算机科学与技术学院 济南 250014)
(sdufelizhen@126.com)

A Discretionary Searchable Encryption Scheme in Multi-User Settings

Li Zhen[1,2], Jiang Han[1], and Zhao Minghao[1]
[1](*School of Computer Science and Technology, Shandong University, Jinan* 250101)
[2](*School of Computer Science and Technology, Shandong University of Finance and Economics, Jinan* 250014)

Abstract Searchable encryption (SE) allows a client to store a collection of encrypted documents on a server and later quickly carry out keyword searches on these encrypted documents, while revealing minimal information to the server. Searchable encryption is an active area of research and a number of schemes with different efficiency and security characteristics have been proposed in the literature. In terms of the multi-user setting, most existing schemes involve a fully-trusted third-party to assign permission among users. In this paper, based on bilinear pairing, we propose a multi-user searchable encryption scheme without the trusted third-party. Specifically, we allow users to discretionarily authorize the documents which other users can access, by maintaining rights assignment matrix to the cloud service provider(CSP) which is honest but curious. Moreover in our scheme, in the searching phase the user can search the documents he wants meanwhile has access to, and accordingly reduce the search scopes of the cloud server. In addition, based on bilinear pairing, we solve the problem of symmetric key distribution, which is neglected in most existing schemes. Actually it implies security risks if the symmetric key is shared among the users. Lastly, we provide formal security proof of our scheme in random oracle model.

Key words cloud computing; searchable encryption (SE); multi-user; provable security; bilinear maps

摘 要 可搜索加密(searchable encryption, SE)允许用户将数据加密后存储到云服务器上,然后在密文数据中按关键词进行搜索,且保证隐私泄漏的最小化.现已提出了针对效率和安全性方面的多种SE方案,但对于多方用户的可搜索加密,目前绝大多数方案都需要用到完全可信的第三方来进行用户授权.针对这一问题,提出让半诚实的云服务器来维护一个权限分配矩阵,允许用户按自己的意愿控制其他用户对自己文件的访问权限,从而弱化了可信第三方的功能.而且,搜索者可指定用户并且服务器只在对其授权的用户文档中进行搜索,从而缩小了搜索范围.同时,利用双线性对的性质,在不增加额外交互的前提下解决了加密文档的密钥分发问题.最后给出该方案在随机预言机模型下安全性的形式化证明.

收稿日期:2015-06-12;**修回日期**:2015-08-10
基金项目:国家自然科学基金面上项目(61173139,61572294);教育部高等学校博士学科点专项科研基金项目(20110131110027)
通信作者:蒋 瀚(jianghan@sdu.edu.cn)

关键词 云计算;可搜索加密;多用户;可证明安全;双线性映射

中图法分类号 TP309

云计算能够为用户提供强大而便捷的数据存储、处理和共享服务,这种新兴商业模式迅速获得企业、服务商和用户的青睐.但云计算/云服务模式的普及与应用受到安全性和隐私问题的制约[1],特别是对于医疗、金融等领域的敏感数据,一旦用户将数据上传给云端就失去了对数据的控制,信息的泄漏和滥用将对用户造成巨大损失,因而用户更倾向于不向任何其他用户以及云服务供应商(cloud service provider, CSP)泄漏任何信息.

一种可以考虑的解决方案是将数据加密后发送到云端,但是这对用户的数据处理和应用造成极大困难.理论上,安全多方计算(secure multi-party computation, SMPC)[2]、全同态加密[3]等能够在某种程度上对加密数据进行有限的处理,但是目前这些密码学原语效率极其低下,难以在现实环境中使用.通用的加密数据处理目前难以找到有效方法,本文针对加密数据的搜索问题进行讨论,设计可证明安全的可搜索加密[4-5]方案.可搜索加密(searchable encryption, SE)能够使用户高效地根据关键词检索出包含该关键词的文档,同时保证服务器仅获得极少的关于所存储文档的信息.

根据数据拥有者(向 CSP 上传数据的一方, owner)和数据使用者(从 CSP 下载数据的一方, user)是否相同或具有相同的行为与权限,可将可搜索加密方案分为单写单读(S/S)、单写多读(S/M)、多写单读(M/S)和多写多读(M/M)四种应用架构[6].目前较多的研究集中于单写单读和身份基加密(identity-based encryption, IBE)的多写单读[7-8]方案的构建.单写多读[9]实现了一个 owner 为多个 user 共享文档的功能,多写多读则可以实现多个 owner 多个 user 的文档共享.多写多读是最一般的情形,同时也是实现最为复杂的一种模型,这种模型最近被提出并逐渐得到学者们的关注[10-18];具体地说,在该系统模型下任何用户都可以向云服务器提供数据,也都可以在特定的权限下,获取其他用户上传的数据;该模型更准确而全面地涵盖了当前云环境的信息存储与共享模式,在现实中有广阔的应用前景[19-21].

本文在多写多读的架构模型下,提出了一种高效的可搜索加密方案.和以前工作相比,该方案在安全模型上弱化了可信第三方的要求,同时免去了用户之间用于加密文档的对称密钥的分发工作;在功能方面,用户可以动态自主分配权限和指定搜索范围;在效率方面,由于每次搜索避免了服务器遍历所有文档,所以效率取得了一定的提升.

1 相关工作及主要贡献

1.1 相关工作

Curtmola 等人[22]首先提出多用户可搜索加密并利用广播加密对单用户方案进行扩展,实现了一个高效的单写多读的方案.随后各种多写多读模式的解决方案被提出.文献[10-16]均采用了完全可信的用户管理中心,负责用户权限的添加和撤销.文献[10]利用双线性映射的性质,由可信中心向每个用户发送唯一的密钥,用来生成索引及查询陷门;向服务器端发送用户的对应密钥,保证只有合法用户才能进行写和读.文献[11-12]同样采用服务器端保留用户的对应密钥的方式,分别利用基于 RSA 和 ElGamal 的代理重加密[23]构建了多写多读的可搜索加密方案.文献[13]利用基于密文策略的属性加密将用户私钥关联到一个属性集,而将密文关联到一棵访问结构树,若属性集满足该访问结构树,则用户具有解密该数据的能力;但是用户在搜索时需要先生成自己基于属性的签名,且增加了管理的复杂度.同样利用属性基加密(attribute-based encryption, ABE)实现方案的还有文献[14-15]等.文献[16]采用广播加密的方法实现了粗粒度访问控制,类似于文献[22]中 S/M 方案的扩展,只需要一个随机数来验证身份,但需要借助完全可信的私有云,且用户每次查询之前要与公有云和私有云进行交互,通信复杂度过高.

目前,借助可信第三方来实现多写多读的可搜索加密研究较多,但是这类方案对第三方的依赖较强,用户密钥的分发、用户权限的添加和撤销等全都交给可信第三方来处理,不能实现用户的动态自主授权.而且,现实中往往不存在被多方用户都信赖的第三方,因此,有些文献开始研究不使用可信第三方的方案[17-18].文献[17-18]采用了每文档一密的思想来去掉可信第三方,直接将搜索权限附在每篇文档的索引之后,这样可以由用户来指定每篇文档的不同授权;但是每篇文档的索引后面加入所有可以授

权的用户信息,在一定程度上加大了索引表的复杂性,且如果数据拥有者需要更改某文档的授权信息时需要重传整个索引.

1.2　主要贡献

本文考虑类似这样一种应用场景:一些商业合作伙伴,作为数据拥有者为了寻求别人与自己合作,需要将自己特定类别的产品文档共享给特定用户;作为数据使用者为了更精准地寻求合作伙伴,在搜索某产品时,可选择具有一定特性,如合作过的或是口碑高的 owner 等,从而节约搜索时间和范围(注:这些信息由 user 自己掌握).所以,方案需要实现 owner 按文档指定访问权限;user 在进行按关键词搜索的同时可指定搜索对象.另外还需满足动态授权,即支持 owner 随时撤销或者添加 user 的权限.而且由于这些用户隶属于不同机构,所以不易在现实中找到所有人都信任的第三方,因此不使用可信第三方来管理用户的方式具有更好的实用性.

我们设计针对上述问题的多方用户加密数据查询方案,具体贡献如下:

1) 弱化可信第三方的功能.本方案中,只需要用户从一个可信的密钥中心获取公钥和私钥即可,这可由现有的认证中心(certificate authority, CA)实现,不需要其他的可信中心.

2) 动态权限分配.由 owner 定义自己文档的权限分配向量,这些权限可以动态添加和撤销,只需要将对应的权限元素发给服务器,而不需要重传索引表.

3) 自主选择搜索范围.user 可指定搜索一个或多个 owner 的文档.

4) 非交互式查询.用户既可以作为 owner,又可以作为 user,且在分配权限以及生成搜索陷门时不需要跟其他用户交互.

5) 提供了高效的密钥分配方案.本方案不需要使用专用的密钥交换协议或密钥分发中心来获得用于解密文档的密钥,而是利用 CSP 在返回搜索结果的同时发送给 user 一个解密陷门,使 user 可使用自己的私钥获取解密密钥,将 owner 的文档解密.

2　基础知识

2.1　双线性映射

令 G_1,G_2,G_T 是 3 个 q 阶循环群,g_1 为 G_1 的生成元,g_2 为 G_2 的生成元,$e:G_1\times G_2\rightarrow G_T$ 若满足以下性质则称 e 为一个双线性映射[24]:

1) 双线性性.对于任意的 $u\in G_1,v\in G_2,a,b\in\mathbb{Z}_q$,有 $e(u^a,v^b)=e(u,v)^{ab}$.

2) 非退化性.存在 $u_1\in G_1,v_1\in G_2$,使得 $e(u_1,v_1)$ 在 G_T 中的阶为 q.

3) 可计算性.对任意的 $u\in G_1,v\in G_2$,存在多项式时间算法计算 $e(u,v)$.

2.2　DL 假设

已知 $a\in_R\mathbb{Z}_q,g$ 是 q 阶循环群 G 的生成元,则给定 g^a,不存在概率多项式时间(probabilistic polynomial time, PPT)算法 $P(\cdot)$,能够以不可忽略的概率正确计算出 a.

2.3　双线性 DH 变体(bilinear Diffie-Hellman variant, BDHV)假设

BDHV 假设[25]是指,对于所有 PPT 敌手以及足够大的安全参数 k,有:

$$|Pr[param\leftarrow\mathrm{CSetup}(1^k);a,b,c\leftarrow\mathbb{Z}_q:adv(params,g_1^a,g_2^b,g_2^{1/a},g_1^c,e(g_1,g_2)^{abc})=1]-Pr[param\leftarrow\mathrm{CSetup}(1^k);a,b,c\leftarrow\mathbb{Z}_q,R\leftarrow G_T:adv(params,g_1^a,g_2^b,g_2^{1/a},g_1^c,R)=1]|=negl(k),$$

其中,$negl(k)$表示可忽略函数.

2.4　拓展的 DH 变体(external Diffie-Hellman variant, XDHV)假设

XDHV 假设[24]是指,对于所有 PPT 敌手以及足够大的安全参数 k,有:

$$|Pr[param\leftarrow\mathrm{CSetup}(1^k);a,b,c,m\leftarrow\mathbb{Z}_q:adv(params,g_1^a,g_1^b,g_1^{ab},g_2^{ca},g_2^{cd},g_1^d,g_2^{1/d})=1]-Pr[param\leftarrow\mathrm{CSetup}(1^k);a,b,c,m\leftarrow\mathbb{Z}_q,R\leftarrow G_T:adv(params,g_1^a,g_1^b,R,g_2^{ca},g_2^{cd},g_1^d,g_2^{1/d})=1]|=negl(k).$$

3　多用户自主授权的可搜索加密方案

3.1　系统描述

在多用户可搜索加密系统中,含有以下 3 个实体:CSP、数据拥有者和数据使用者.CSP 是半可信(也称为诚实但好奇)的,即它会诚实执行协议,但会观察所有交互的数据以便获得额外信息.每个用户都可以既是 owner 又是 user;但在这个方案中,我们不考虑搜索自己的文档这种情况,因为这种情况可以使用更高效的对称加密方案而实现.如果在这个方案中兼顾实现这种情况,要么方案过于复杂,要么存在一定信息泄漏的风险[18].

主要符号说明:

U 为用户集合,每个用户的标识为 $u_i(1\leqslant i\leqslant n,$

n 为用户数);

W 为关键词集合,w_s 代表单一关键词($1\leqslant s\leqslant m$,m 为关键词个数);

D 为文档集合;

D_i 表示用户 u_i 拥有的文档集合;

D_{ilk} 表示用户 u_i 拥有的属于第 l 个级别的第 k 篇文档,其中 $1\leqslant l\leqslant\rho$,$\rho$ 为用户的文档级别数,$1\leqslant k\leqslant\sigma$,$\sigma$ 为每个级别中的文档数;

$id(D_{ilk})$ 表示文档 D_{ilk} 的标识;

DK_{ilk} 表示用户 u_i 为自己文档 D_{ilk} 生成的加密密钥;

D' 表示加密后的文档集合,D'_{ilk} 表示加密后的每篇文档;

CK_{il} 表示用户 u_i 为自己第 l 个级别的文档生成的索引密钥.

3.2 系统构造

一个可由用户自主授权的多用户可搜索加密系统由系统建立、数据上传、增删权限以及数据查询这 4 个算法构成.

1) 系统建立

系统建立阶段主要包括系统参数生成算法、密钥生成算法以及权限矩阵的初始化算法.

① 系统参数生成算法($param\leftarrow\text{setup}(\lambda)$):输入安全参数 λ,输出公共参数 $param$. 记 $\psi=(q,G_1,G_2,G_T,e,g_1,g_2)$,其中 G_1,G_2,G_T 是 3 个 q 阶循环群,g_1 为 G_1 的生成元,g_2 为 G_2 的生成元,e 是双线性映射;$H_1:\{0,1\}^*\rightarrow G_1$,$H_2:\{0,1\}^*\rightarrow G_2$ 是 2 个抗碰撞散列函数,则 $param=(\psi,H_1,H_2)$.

② 密钥生成算法($PK_i,SK_i\leftarrow\text{KeyGen}(param)$):用户 $u_i(1\leqslant i\leqslant n)$ 调用此算法,输入安全参数,得到自己的公钥 PK_i 和 SK_i. 其中 $SK_i=x_i$,$PK_i=g_2^{1/x_i}$,$x_i\in_R\mathbb{Z}_q$.

③ 权限矩阵的初始化:CSP 初始化一个 n 行 n 列的权限矩阵 $\boldsymbol{\Lambda}$,其每个元素 $\pi_{ji}=\{A_{jil}\mid l\in\{1,2,\cdots,\rho\}\}$ 为一个集合,用来表示作为 owner 的用户 u_i 对作为 user 的用户 u_j 授权访问 u_i 的级别为 l 的文档,其初始值为$\varnothing$. 如下所示:

$$\begin{array}{c} \\ u_1 \\ \vdots \\ u_j \\ \vdots \\ u_n \end{array}\begin{array}{c} \begin{array}{ccccc} u_1 & \cdots & u_i & \cdots & u_n \end{array} \\ \begin{bmatrix} \pi_{11} & \cdots & \pi_{1i} & \cdots & \pi_{1n} \\ \vdots & & \vdots & & \vdots \\ \pi_{j1} & \cdots & \pi_{ji} & \cdots & \pi_{jn} \\ \vdots & & \vdots & & \vdots \\ \pi_{n1} & \cdots & \pi_{ni} & \cdots & \pi_{nn} \end{bmatrix} \end{array},$$

其中,$\pi_{ji}=\begin{cases}\{A_{jil}\mid l=1,2,\cdots,\rho\}, & \text{有授权.}\\ \varnothing, & \text{无授权.}\end{cases}$

也就是说,权限矩阵的每行表示作为 user 的用户已拥有的所有权限,每列表示作为 owner 的用户已分配给其他用户的所有权限.

2) 数据上传

这个阶段的算法由 owner 执行,包括索引密钥生成算法、文档密钥生成算法、索引生成算法及文档加密算法.

① 索引密钥生成算法($CK_{il}\leftarrow\text{CKGen}(param)$):由每个作为 owner 的用户 u_i 随机选择每个级别为 l 的文档索引密钥 $CK_{il}(1\leqslant i\leqslant n,1\leqslant l\leqslant\rho)$,$CK_{il}\in_R\mathbb{Z}_q$.

② 文档密钥生成算法($DK_{ilk},r_{ilk}\leftarrow\text{DKGen}(SK_i,CK_{il})$):由每个作为 *owner* 的用户 u_i 为自己每篇文档 D_{ilk} 生成加密密钥 $DK_{ilk}=H_2(e(r_{ilk},g_2^{CK_{il}/x_i}))$,$r_{ilk}\in_R G_1$,其中 $1\leqslant i\leqslant n,1\leqslant l\leqslant\rho,1\leqslant k\leqslant\sigma$.

③ 索引生成算法($C_{ilk}\leftarrow\text{GenIndex}(CK_{il},w_s,SK_i,r_{ilk})$):由作为 owner 的用户 u_i 为自己的每篇文档 D_{ilk} 生成索引项,$w_s(1\leqslant s\leqslant m)$ 为此文档包含的一个关键词. r_{ilk} 与每篇文档的加密密钥 DK_{ilk} 生成算法中的随机数一致,并且同一文档中的多个关键词使用同一个随机数以及同样的方法. 对文档的每个关键词计算 $X_s=H_2(r_{ilk},e(H_1(w_s),g_2)^{CK_{il}/x_i})$,记 $C_{ilk}=(r_{ilk},X)$,其中 X 为一个集合 $X=\{X_s\mid s\in\{1,\cdots,m\}\}$. 用户对自己的所有文档执行 GenIndex 算法,最后将由 $id(D_{ilk})$ 和 C_{ilk} 组成的索引表 C_i 发给 CSP.

④ 文档加密算法($D'_{ilk}\leftarrow\text{EncDoc}(DK_{ilk},D_{ilk})$):由作为 owner 的用户 u_i 为自己的每篇文档 D_{ilk} 加密,加密后的文档记为 D'_{ilk},$D'_{ilk}\leftarrow\text{AES}_{DK_{ilk}}(D_{ilk})$.

3) 增删权限

这个阶段由 owner 对自己的不同级别的文档进行按用户的授权,并将权限信息发送给 CSP,由 CSP 存入权限矩阵. 这个过程的 2 个算法都满足动态更新性,也就是说用户可随时改变某些用户的访问权限,包括用户授权算法及权限撤销算法.

① 用户授权算法($A_{jil}\leftarrow\text{Grant}(SK_i,PK_j,CK_{il})$):由作为 owner 的 u_i 对作为 user 的 u_j 授予对他的 l 级别的文档的搜索权限,此权限值为 $g_2^{CK_{il}/x_ix_j}$,记为 A_{jil}. 重复调用 Grant 算法,可对 u_j 进行多个级别文档的授权,记集合 $A_{ji}=\{A_{jil}\mid l\in\{1,\cdots,\rho\}\}$ 来表示 u_j 拥有的所有权限,u_i 发送 $\text{Grant}(u_j,u_i,A_{ji})$ 给 CSP,CSP 更新 π_{ji} 为 $\pi_{ji}\cup A_{ji}$.

② 权限撤销算法($A_{jil}\leftarrow\text{Revoke}(SK_i,PK_j,$

CK_{il})):由作为 owner 的 u_i 对作为 user 的 u_j 删除对他的 l 级别的文档的搜索权限,此权限值为 $g_2^{CK_{il}/x_ix_j}$,记为 A_{jil}.重复调用 Revoke 算法,可对 u_j 进行多个权限的删除,记集合 $A_{ji}=\{A_{jil}\mid l\in\{1,\cdots,\rho\}\}$ 来表示需要删除的 u_j 的所有权限,u_i 发送 Revoke(u_j,u_i,A_{ji})给 CSP,CSP 更新 π_{ji} 为 $\pi_{ji}=\pi_{ji}/A_{ji}$,其中"/"运算表示集合相减.

更新权限的操作可由先执行 Revoke、再执行 Grant 这 2 个算法来完成,所以不需额外定义.并且,由于对文档的增删操作和对用户的授权操作互不影响,所以 owner 可随时更新自己的文档而不影响已有的授权,同时也可随时更新授权而不影响已有的文档.

4) 数据查询

此阶段由 user 向 CSP 发送访问请求,CSP 执行搜索算法返回结果以及相应文档的解密陷门.包括搜索陷门生成算法、搜索执行算法及解密文档算法.

① 搜索陷门生成算法($Tr(w_s)\leftarrow$Query(SK_j,w_s)):由作为 user 的用户 u_j 利用自己的私钥生成对关键词 w_s 的搜索陷门:$Tr(w_s)=H_1(w_s)^{x_j}$,记 $Tw=(u_j,U_j,Tr(w_s))$,其中 $U_j=\{u_i\mid i\in\{1,2,\cdots,n\}\wedge i\neq j\}$.也就是说,$u_j$ 可以指定一个或多个想要搜索的用户,如想搜索所有给自己权限的用户,就用 U 来表示.由 u_j 发送 Tw 给 CSP.

② 搜索执行算法($Z\leftarrow$Search(Tw,$\boldsymbol{\Lambda}$,C_i)):由 CSP 执行,算法描述如下:

算法 1. 搜索执行算法.

输入:user 的访问请求 Tw、权限矩阵 $\boldsymbol{\Lambda}$ 和所有目标索引表 C_i;

输出:目标文档及相应的解密陷门构成的集合 Z.

① 初始化 $Z=\varnothing$;
② for every $u_i\in U_j$ do /* 对 u_j 想搜索的每个对象 u_i */
③　for every $A_{jil}\in\pi_{ji}$ do /* 对 u_i 赋予 u_j 的所有权限 */
④　　for each $D_{ilk}\in D_i$ do /* 遍历 u_i 的所有文档 */
⑤　　$Tr'(w_s)=e(Tr(w_s),A_{jil})$;
⑥　　if $H_2(r_{ilk},Tr'(w_s))=X_s$
⑦　　$DK'_{ilk}=e(r_{ilk},g_2^{CK_{il}/x_ix_j})$;$Z=Z\cup\{(D'_{ilk},DK'_{ilk})\}$;
⑧　　end if
⑨　　end for
⑩　end for
⑪ end for

算法执行完毕后,CSP 发送 Z 给 u_j.

③ 解密文档算法($D_{ilk}\leftarrow$DecDoc(SK_j,Z)):用户 u_j 计算 $H_2((DK'_{ilk})^{x_j})$,此值即为 DK_{ilk},用此密钥解密 D'_{ilk} 得到 D_{ilk}.

3.3　系统正确性分析

1) 查询正确性.由双线性对的性质可知,$Tr'(w_s)=e(Tr(w_s),A_{jil})=e(H_1(w_s)^{x_j},g_2^{CK_{il}/x_ix_j})=e(H_1(w_s),g_2^{CK_{il}/x_i})$,所以,CSP 用索引表中每个文档名后的随机数联合 $Tr'(w_s)$ 进行 Hash 运算,其结果 $H_2(r_{ilk},Tr'(w_s))$ 若等于这个索引项中的某个 X_s,由散列函数的抗碰撞性,则该文档即为目标文档之一.

2) 解密正确性.同样地,由双线性对的性质,$H_2((DK'_{ilk})^{x_j})=H_2(e(r_{ilk},g_2^{CK_{il}/x_ix_j})^{x_j})=H_2(e(r_{ilk},g_2^{CK_{il}/x_i}))$,此值就是 u_i 用来加密文档 D_{ilk} 的密钥 DK_{ilk}.由 AES 算法的对称性,此密钥可正确解密.

4　安全性分析与证明

假设此方案中的 CSP 是半诚实的,并且不与任何用户合谋,则方案的安全性主要考虑 4 个方面:1)存储在 CSP 中的文档具有保密性;2)文档与关键词对应关系的索引表具有保密性;3)用于查询的陷门具有保密性;4)当用户被撤销权限后将不能搜索相应的文档(即可撤销性).

4.1　文档保密性

在云端存储的文档不会泄漏文档内容的任何信息.由 DKGen 算法以及 EncDoc 算法可知,用户为自己的每篇文档生成唯一的加密密钥,再利用 AES 算法进行加密.由 AES 的安全性,将文档加密存储于服务器后,云服务商无法从文档密文中获取任何关于明文的信息.在解密阶段,因为解密密钥需要结合用户私钥生成,所以由 DL 假设,保证了服务器不能获得解密密钥,从而保证了文档保密性.

4.2　索引保密性

用户在云端存储的索引不会泄漏文档对应的关键词信息,所以这里考虑的是作为 owner 的用户的安全性.需要满足选择关键词攻击下的不可区分性安全(indistinguishability under chosen keyword attack,IND-CKA).为了证明方案达到这个安全性,我们定义一个挑战者与敌手之间的游戏.

1) 建立阶段. 挑战者运行建立算法,将公共参数 $param$ 给敌手. 敌手选择某个作为 owner 的用户(例如 u_1)的一些索引密钥 $CK_{1l}(l>0)$,然后构造一些可访问 l 级别的文档的用户 $u_j(j\geqslant 2)$,并利用他们的公钥,再使用敌手自己的私钥 x_{Adv} 生成这些用户的权限:$g_2^{CK_{1l}/x_{\mathrm{Adv}}x_j}$. 0 级文档用于挑战,所以由挑战者生成 CK_{10},且挑战者掌握 u_1 的私钥 x_1,再选择一些能访问 0 级文档的用户 $u_j(j\geqslant 2)$,并为他们生成私钥:$x_j\leftarrow \mathrm{KeyGen}(param)$,计算其对应的权限 $g_2^{CK_{10}/x_1x_j}$ 发给敌手,即敌手拿到权限矩阵中 u_1 对应的列的所有值.

2) 挑战阶段. 敌手任选 2 个关键词 $w_0^*, w_1^*\leftarrow\{0,1\}^l$ 发给挑战者. 挑战者进行随机掷币,任选一个将其作为 0 级文档 k(即 D_{10k})的关键词,并对其加密,将生成的 $H_2(r_{10k}, e(H_1(w_b^*), g_2)^{CK_{10}/x_1})$ 以及 r_{10k} 发给敌手.

3) 问询阶段. 敌手自适应地问询挑战者以下查询,其中第 l 次为

① "将 w_l 加密后作为文档 D_{10l} 的关键词":挑战者返回$(r_{10l}, H_2(r_{10l}, e(H_1(w_l), g_2)^{CK_{10}/x_1}))$.

② "作为 user 的用户 u_j 搜索 w_l 的陷门":挑战者返回 $H_1(w_l)^{x_j}$.

4) 猜测阶段. 敌手输出 b',如果 $b'=b$,则敌手赢得游戏.

① 问询阶段的步骤②对敌手的限制是如果 u_j 有搜索 0 级文档的权限,则这里的 $w_l\notin\{w_0^*, w_1^*\}$. 若不加此限制,敌手可通过计算 $H_2(r_{10l}, e(H_1(w_l)^{x_j}, g_2^{CK_{10}/x_1x_j}))=H_2(r_{10l}, e(H_1(w_l), g_2)^{CK_{10}/x_1})$ 来区分 w_0^* 和 w_1^*. 但是敌手可以问询对 0 级文档无搜索权限的用户关于 w_0^* 和 w_1^* 的陷门,这给出了保证 u_j 无权限时不能搜索文档的情况.

② 敌手能生成 u_1 的其他级别的文档的索引密钥,模拟的是现实中敌手可以自己造一些文档,在这种假设下仍不会影响现有文档的安全性.

用 $win_{\mathrm{Adv}}(k)$ 表示敌手赢得游戏,下面给出索引保密性的安全定义:

定理 1. 一个多用户可搜索加密系统满足索引保密性,当且仅当对于所有的 PPT 敌手,在安全参数 k 下,$Pr[win_{\mathrm{Adv}}(k)]<1/2+negl(k)$.

证明. 方案的安全性证明依赖于 2.2 节和 2.3 节给出的假设. 借助一系列混合游戏(hybrid games),其中 $Game_0$ 就是本节中所描述的游戏,也就是真实的游戏,后面的游戏将条件逐渐简化,在 $Game_4$ 中可直接看出敌手赢得游戏的概率为 1/2(以下证明中 $Game_i$ 中的敌手记为 Adv_i,挑战者记为 ch_i).

$Game_1$ 与 $Game_0$ 仅有一处区别,即去掉了敌手选择 0 级以上文档的索引密钥的步骤. 下面证明如果方案在 $Game_1$ 中安全,则在 $Game_0$ 中也安全. 采用反证法,如果方案在 $Game_0$ 中不安全,则存在 PPT 敌手 Adv_0 可赢得 $Game_0$,那么如果能构造出 Adv_1 通过调用 Adv_0 以不可忽略的概率去赢得 $Game_1$,则得证. 由 DL 假设知,Adv_1 可选择随机数 $R_l(l>0)$,并将 $g_2^{R_l}$ 作为权限存入矩阵,则可成功模拟出 Adv_0 的输入,剩下的跟 $Game_0$ 完全一致,则 Adv_1 输出 Adv_0 的输出.

$Game_2$ 与 $Game_1$ 的区别是去掉了无权访问 0 级文档的用户. 下面证明方案在 $Game_2$ 中安全则在 $Game_1$ 中安全,依然采用反证法. 若 Adv_1 可赢得 $Game_1$,则可构造出 Adv_2 来赢得 $Game_2$. 这样在问询阶段,Adv_2 的策略是:对于那些可访问 0 级文档的用户,直接用 ch_2 返回搜索陷门 $H_1(w_l)^{x_j}$,对于那些不可访问 0 级文档的用户,Adv_2 用随机数作为他们的私钥,返回 $H_1(w_l)^R$. 由 DL 假设,这 2 个值是计算不可区分的,所以成功模拟出 Adv_1 的输入.

$Game_3$ 与 $Game_2$ 的区别是把 $e(H_1(w_b^*), g_2)$ 用随机数 R 来替换. 问询阶段的步骤 1)变为如下内容:"将 w_l 加密后作为文档 D_{10l} 的关键词":如果 $w_l=w_b^*$,挑战者返回$(r_{10l}, H_2(r_{10l}, R))$,否则挑战者返回 $(r_{10l}, H_2(r_{10l}, R^\alpha))$,其中 α 满足 $g_1^\alpha=H(w_{1-b}^*)/H(w_b^*)$. $Game_2$ 与 $Game_3$ 的不可区分性通过引理 1 来证明.

由于在 $Game_3$ 中,所有的私钥和用户权限已经都不再使用了,所以 $Game_4$ 可进一步简化,去掉建立阶段的所有步骤即可. 由 Random Oracle $\mathcal{H}$ 的安全性可知,没有敌手可以以不可忽略的概率赢得游戏. 证毕.

引理 1. 如果 BDHV 假设成立并且 $\mathcal{H}$ 是一个可编程的 Random Oracle,则 $Game_3$ 与 $Game_2$ 是计算不可区分的.

证明. 采用反证法. 假设存在一个 PPT 敌手 $\mathcal{D}$ 能区分这 2 个游戏,则可以构造一个 PPT 敌手 $\mathcal{B}$ 来攻破 BDHV.

$\mathcal{B}$ 接收的输入是:$g_1^a, g_2^b, g_2^{1/a}, g_1^c, T$. $T=e(g_1, g_2)^{abc}$ 或者 $T=R$. 为了区分 T,$\mathcal{B}$ 进行如下操作:$\mathcal{B}$ 将其输入作为 $\mathcal{D}$ 在询问 w_0^* 或 w_1^* 时访问 Random Oracle 的挑战结果.

$\mathcal{B}$ 要猜测哪次对 Random Oracle $\mathcal{H}$ 的问询是挑战值. 不失一般性,假设 $\mathcal{D}$ 每次问询 $\mathcal{H}$ 是独立的,则有 3 种情况:

1) $\mathcal{B}$ 在挑战阶段之前没有对 w_0^* 或 w_1^* 问询过 $\mathcal{H}$;

2) $\mathcal{B}$ 在挑战阶段之前对 w_0^* 或 w_1^* 中的一个问询过 $\mathcal{H}$;

3) $\mathcal{B}$ 在挑战阶段之前对 w_0^* 和 w_1^* 全都问询过 $\mathcal{H}$.

用 i_0 表示 $\mathcal{B}$ 问询 w_0^* 时猜测的对应索引,则如果 $\mathcal{B}$ 在挑战阶段之前没有问询过 w_0^*,则 i_0 记为 $\perp$.

假设 $\mathcal{D}$ 可询问多项式次 $\mathcal{H}$,则 $\mathcal{B}$ 有 $p(k)/3$ 的机会得到 i_0 或 i_1,其中 p 为多项式约束,k 为安全参数. 当 $\mathcal{D}$ 在挑战阶段提供 w_0^* 或 w_1^* 给 $\mathcal{B}$ 时,$\mathcal{B}$ 可验证猜测的 i_0 或 i_1 是否正确. 如果不正确,$\mathcal{B}$ 输出一个随机的猜测.

1) 初始化. $\mathcal{B}$ 生成参数并且发送给 $\mathcal{D}$,$\mathcal{B}$ 选择 $\alpha \in_R \mathbb{Z}_q$.

$\mathcal{H}$ 的构造:初始化 $oracle$,对 $\mathcal{D}$ 的每个对 $\mathcal{H}$ 的查询 w,$\mathcal{B}$ 进行如下操作:

如果这个是 i_0 的查询,返回 g_1^c.

如果这个是 i_1 的查询,返回 $g_1^{c\alpha}$.

否则,选择 $q \in_R \mathbb{Z}_q$,将 w 对应的输出记为 q 存入 $oracle$,即 $oracle[w] := q$ 并且返回 g_1^q.

$\mathcal{B}$ 收到 $\mathcal{D}$ 发过来的权限表,对于表中的用户,用 g_2^{b/ω_j} 代替 $g_2^{CK_{10}/x_1 x_j}$.

2) 挑战阶段. $\mathcal{B}$ 从 $\mathcal{D}$ 收到 w_0^* 和 w_1^*. 验证 i_0 和 i_1 是否是正确的猜测,如果不是,输出一个随机位并终止;否则,把 r_{10l},$H_2(r_{10l}, T)$ 发给 $\mathcal{D}$.

3) 在自适应问询阶段. 对"加密 w_l":如果 $w_l = w_b^*$,返回 r_l,$H_2(r_l, T)$;否则,返回 r_l,$H_2(r_l, T^\alpha)$. 对"u_j 搜索 w_l 的陷门":挑战者返回 $g_1^{\alpha\omega_j \times oracle[w_l]}$,输出 $\mathcal{D}$ 的输出. 则 $\mathcal{B}$ 正确模拟了 $\mathcal{D}$ 的输入. $\mathcal{H}$ 的所有输入正确分布且具有以下等价性:$a \leftrightarrow x_1$,$b \leftrightarrow CK_{10}/x_1^2$,$g_1^c \leftrightarrow H_1(w_b^*)$,$g_1^{c\alpha} \leftrightarrow H(w_{1-b}^*)$,$\omega_j = x_j/x_1$.

如果 $T = e(g_1, g_2)^{abc}$,则 $T = e(H_1(w_b^*), g_2)^{CK_{10}/x_1}$ 就是在 $Game_2$ 的分布,而 $T = R$ 则是在 $Game_3$ 的分布.

对于输出陷门阶段,$g_1^{\alpha\omega_j \times oracle[w_l]}$ 就是 $H_1(w_l)^{x_j}$.

因此,$\mathcal{B}$ 成功模拟了 $\mathcal{D}$ 的输入,若 $\mathcal{D}$ 可以以不可忽略的概率区分 $Game_3$ 和 $Game_2$,则 $\mathcal{B}$ 可以同样以不可忽略的概率攻破 BDHV 假设. 证毕.

4.3 陷门保密性

用户在执行查询操作时提供给服务器的查询陷门不会泄漏正在查询的关键词信息. 同样需要满足选择关键词攻击下的不可区分性安全:IND-CKA,所以也定义一个挑战者与敌手之间的游戏.

1) 建立阶段. 挑战者运行建立算法,将公共参数 $param$ 给敌手. 敌手选择某个作为 user 的用户(这里使用 u_0)的一些查询权限:$g_2^{CK_{j0}/x_j x_0}$ ($i > 0$,且为简化起见,假设所有用户都把自己的 0 级文档共享给 u_0). 由挑战者生成 u_0 的私钥:$x_0 \leftarrow \text{KeyGen}(param)$,并生成自己 0 级文档的索引密钥 CK_{00},敌手选择一些用户 u_j 让其有访问 u_0 的 0 级文档的权限,挑战者将他们的权限 $g_2^{CK_{00}/x_0 x_j}$ 发给敌手,即敌手拿到权限矩阵中的 u_0 对应的行的所有值以及 u_0 对应的列的部分值.

2) 挑战阶段. 敌手任选 2 个关键词 w_0^*,$w_1^* \leftarrow \{0,1\}^l$ 发给挑战者. 挑战者进行随机掷币,任选一个将其作为 u_0 要查询的关键词,将 $H_1(w_b^*)^{x_0}$ 发给敌手.

3) 问询阶段. 敌手自适应地问询挑战者以下查询,其中第 l 次包括 2 步:

① "将 w_l 加密后作为文档 D_{00l} 的关键词":挑战者返回 $(r_{00l}, H_2(r_{00l}, e(H_1(w_l), g_2)^{CK_{00}/x_0}))$.

② "作为 user 的用户 u_j ($j > 0$) 搜索 w_l 的陷门":挑战者返回 $H_1(w_l)^{x_j}$.

4) 猜测阶段. 敌手输出 b',如果 $b' = b$,则敌手赢得游戏.

$win_{\text{Adv}}^{\text{token}}(k)$ 表示敌手赢得游戏,$negl(k)$ 表示一个可忽略函数. 下面给出陷门保密性的安全定义:

定理 2. 一个多用户可搜索加密系统满足陷门保密性,当且仅当对于所有的 PPT 敌手,在安全参数 k 下,$Pr[win_{\text{Adv}}^{\text{token}}(k)] < 1/2 + negl(k)$.

证明. 证明过程与索引安全性证明类似. 安全性依赖于 2.2 节和 2.4 节给出的假设. 定义一系列混合游戏,其中 $Game_0$ 就是本节中所描述的游戏,也就是真实的游戏.

$Game_1$ 与 $Game_0$ 仅有一处区别,即去掉了问询阶段的步骤①,加密关键词. 因为这种加密可以通过陷门和权限而得到,所以并不影响陷门的安全性.

下面证明如果方案在 $Game_1$ 中安全则在 $Game_0$ 中也安全. 采用反证法,若存在 PPT 敌手 Adv_0 可赢得 $Game_0$,那么可以构造出 Adv_1 通过调用 Adv_0 以不可忽略的概率去赢得 $Game_1$. 构造如下:Adv_1 可重复问询阶段的步骤②,直到问询到一个能访问 u_0 的 0 级文档的用户,利用其陷门以及对应于权限矩阵中的值,可求得 $e(H_1(w_l)^{x_j}, g_2^{CK_{00}/x_0 x_j}) = e(H_1(w_l), g_2)^{CK_{00}/x_0}$,再自选一个随机数,即可获得此关键词的索引加密. 则可成功模拟出 Adv_0 的输入,剩下的跟 $Game_0$ 完全一致,则 Adv_1 输出 Adv_0 的输出.

$Game_2$ 与 $Game_1$ 的区别是去掉了"敌手选择一些用户 u_j 让其有访问 u_0 的 0 级文档的权限,挑战

者将他们的权限 $g_2^{CK_{00}/x_0x_j}$ 发给敌手"这一步.类似之前的证明,用反证法.由 DL 假设知,Adv_2 可选择随机数 $R_l(l>0)$,将 $g_2^{R_l}$ 作为权限存入矩阵.则可成功模拟出 Adv_1 的输入,剩下的跟 $Game_1$ 完全一致,则 Adv_2 输出 Adv_1 的输出.

$Game_3$ 在 $Game_2$ 的基础上继续简化,使得挑战者 ch_3 收到敌手 Adv_3 发来,w_0^* 和 w_1^* 后,通过掷币协议选择 b 后,从 G_T 中任选随机数 R 返回给 Adv_3.这样,Adv_3 收到的东西跟 b 完全无关,所以赢得游戏的概率为 1/2. 证毕.

下面通过引理 2 证明 $Game_3$ 和 $Game_2$ 是计算不可区分的.

引理 2. 如果满足 XDHV 假设,则在 Random Oracle $\mathcal{H}$ 下,$Game_3$ 和 $Game_2$ 是计算不可区分的.

证明. 反证法.假设存在 PPT 敌手 $\mathcal{D}$ 能区分 $Game_3$ 和 $Game_2$,则能够构造 PPT 敌手 $\mathcal{B}$ 来攻破 XDHV 假设.

$\mathcal{B}$ 收到 $g_1^a, g_1^b, T, g_2^{\alpha}, g_2^{\alpha d}, g_1^d, g_2^{1/d}$ 作为输入,需要确定 $T=g_1^{ab}$ 还是 $T=R$.

$\mathcal{H}$ 的构造:$\mathcal{B}$ 通过掷币协议,决定一种情况是 $\mathcal{B}$ 预言 $\mathcal{D}$ 永远不会对 w_b^* 询问 $\mathcal{H}$,另一种情况是 $\mathcal{D}$ 会对 w_b^* 询问 $\mathcal{H}$,在这种情况下 $\mathcal{D}$ 构造一个查询记录来表示在这些查询中 $\mathcal{D}$ 会询问到 w_b^*,其中的第 I 次查询记为 $I\in\{0,\cdots,p(k)\}$.则对于 $\mathcal{D}$ 询问 $\mathcal{H}$ 的每个 w,$\mathcal{B}$ 进行如下操作:

如果这是第 I 次查询,返回 g_1^b;否则,选择 $q\in_R\mathbb{Z}_p$,将 w 对应的输出记为 q 存入 $oracle$,即 $oracle[w]:=q$ 并且返回 g_1^q.

自适应阶段:如果是 u_0 的陷门,则 $\mathcal{B}$ 返回 $g_1^{a\times oracle[w]}=H(w)^{x_0}$ $(w\neq w_b^*)$.若是 $u_i(i>0)$ 的陷门,则返回 $g_1^{d\times oracle[w]}=H(w)^{x_i}$.并且有如下等价性:$a\leftrightarrow x_0$,$g_1^b\leftrightarrow H(w_b^*)$,$c\leftrightarrow CK_{i0}/x_ix_0$,$d\leftrightarrow x_i$,$\omega_i\leftrightarrow x_i/x_0$,$\alpha_j\leftrightarrow 1/x_j$,则:

对于 $g_2^{CK_{i0}/x_ix_0}$ $(i\geqslant 1)$,有 $g_2^{d\alpha_i}=g_2^{x_i\times CK_{i0}/x_ix_0\times 1/x_i}=g_2^{CK_{i0}/x_ix_0}$;

对于 $g_2^{CK_{i0}/x_ix_j}$,有 $g_2^{d\alpha_j/\omega_i}=g_2^{(x_i\times CK_{i0}/x_ix_0\times 1/x_j)/(x_i/x_0)}=g_2^{CK_{i0}/x_ix_j}$;

对于 g_2^{1/x_0},有 $g_2^{\omega_i/d}=g_2^{\omega_i/x_i}=g_2^{1/x_0}$;

这些都跟预期结果一样.

所以,如果 T 是随机数,则挑战阶段和 $Game_3$ 中的一样;若 $T=g_1^{ab}=H(w_b^*)^{x_0}$,则挑战阶段跟 $Game_2$ 中的一样.因此 $\mathcal{B}$ 成功模拟了 $\mathcal{D}$ 的输入,若 $\mathcal{D}$ 可以以不可忽略的概率区分 $Game_3$ 和 $Game_2$,则 $\mathcal{B}$ 可以同样以不可忽略的概率攻破 XDHV. 证毕.

4.4 可撤销性

多用户的可搜索加密方案必须要考虑的一个安全性就是用户权限的可撤销性.保证一个用户权限撤销后不能再进行后续的查询操作,也不能影响其他合法用户的查询操作.

本方案没有一个可信的用户管理中心,所以文档的更新管理以及文档对应的用户权限的更新管理都是由文档的拥有者即 owner 自主决定.

对于权限的更新,因为是按照 user 进行的,所以只需要更新权限矩阵中对应的元素即可,不会影响其他 user 的权限.同样,将某个 user 对某级别文档的访问权限删除后,服务器在执行搜索算法时将不会生成与这个级别的索引密钥相关的搜索陷门,也就不会检索出这个级别的文档.所以,此方案满足可撤销性.

5 方案性能比较

由于文献[10-13,18]的方案应用场景与本方案类似,所以与本方案在实现功能和安全性方面做了比较,如表 1 所示:

Table 1 Comparison between Our Scheme and Ref[10-13,18]

表 1 本方案与文献[10-13,18]的比较

Scheme	Trusted Center	Discretionary Authorization	Different Documents Authorization	Authorization Independence of Index	Enc-Key Sharing
Ref[10]	Yes	No	No	Yes	Yes
Ref[11-12]	Yes	No	No	Yes	No
Ref[13]	Yes	No	No	Yes	Yes
Ref[18]	No	Yes	Yes	No	Yes
Ours	No	Yes	Yes	Yes	No

由表1可见,本方案的优势是:

1) 不需要一个可信的用户管理中心,而是由数据拥有者自主决定授权并可实现不同文档的不同授权;

2) 更新权限时不需要在user中再进行交互,也不需要更新文档的索引表,保证了授权操作与数据的独立性;

3) 实现让user利用自己的私钥解密文档的功能,不需要共享加密密钥.适合多用户文档共享的场景.

在效率方面,主要考虑查询阶段.作为user的用户在生成陷门时只需要一次指数运算,所以计算复杂性为$O(1)$.CSP拿到user的搜索陷门$Tr(w_s)$后,若此user只搜索一个owner的文档,并且他只拥有这个owner给他的一个权限,则CSP只需要做一次双线性对运算和k次Hash运算(k为这个级别的文档数目),返回目标文档时还需进行生成解密陷门的双线性对运算,若返回d篇文档,需要d次运算,因为目标文档一般较少,所以这个效率是可以接受的.

6 结　论

本文提出的多用户可搜索加密方案没有使用可信的用户管理中心,而是由owner自主按文档进行授权,满足当前的一些应用背景.在搜索时一方面可由user指定搜索对象,另一方面CSP也只在授予此user权限的owner文档中进行搜索,避免了遍历所有文档.又由于利用双线性对的性质减少了用户与服务器、用户与用户之间的交互,从而降低了系统的通信复杂度.并且本文在Random Oracle模型下形式化地证明了此方案的安全性.下一步我们将研究在此方案基础上支持多关键词查询以及模糊查询的高效方案.

参　考　文　献

[1] Li Hui, Sun Wenhai, Li Fenghua, et al. Secure and privacy-preserving data storage service in public cloud [J]. Journal of Computer Research and Development, 2014, 51(7): 1397-1409 (in Chinese)
(李晖, 孙文海, 李凤华, 等. 公共云存储服务数据安全及隐私保护技术综述[J]. 计算机研究与发展, 2014, 51(7): 1397-1409)

[2] Yao A C. Protocols for secure computations [C] //Proc of the 23rd Annual Symp on Foundations of Computer Science. Los Alamitos, CA: IEEE Computer Society, 1982: 160-164

[3] Gentry C. Fully homomorphic encryption using ideal lattices [C] //Proc of the 41st Annual ACM Symp on Theory of Computing(STOC 2009). New York: ACM, 2009: 169-178

[4] Song D X, Wagner D, Perrig A. Practical techniques for searches on encrypted data [C] //Proc of the 21st IEEE Symp on Security and Privacy. Piscataway, NJ: IEEE, 2000: 44-55

[5] Bellare M, Boldyreva A, O'Neill A. Deterministic and efficiently searchable encryption [G] //LNCS 4622: Advances in Cryptology (CRYPTO 2007). Berlin: Springer, 2007: 535-552

[6] Li Jingwei, Jia Chunfu, Liu Zheli, et al. Survey on the searchable encryption [J]. Journal of Software, 2015, 26(1): 109-128 (in Chinese)
(李经纬, 贾春福, 刘哲理, 等. 可搜索加密技术研究综述[J]. 软件学报, 2015, 26(1): 109-128)

[7] Boneh D, Di Crescenzo G, Ostrovsky R, et al. Public key encryption with keyword search [G] //LNCS 3027: Advances in Cryptology (Eurocrypt 2004). Berlin: Springer, 2004: 506-522

[8] Abdalla M, Bellare M, Catalano D, et al. Searchable encryption revisited: Consistency properties, relation to anonymous IBE, and extensions [G] //LNCS 3621: Advances in Cryptology (CRYPTO 2005). Berlin: Springer, 2005: 205-222

[9] Raykova M, Vo B, Bellovin S M, et al. Secure anonymous database search [C] //Proc of the 16th ACM Workshop on Cloud Computing Security. New York: ACM, 2009: 115-126

[10] Bao Feng, Deng R H, Ding Xuhua, et al. Private query on encrypted data in multi-user settings [G] //LNCS 4991: Information Security Practice and Experience. Berlin: Springer, 2008: 71-85

[11] Dong Changyu, Russello G, Dulay N. Shared and searchable encrypted data for untrusted servers [G] //LNSC 5094: Data and Applications Security 2008. Berlin: Springer, 2008: 127-143

[12] Dong Changyu, Russello G, Dulay N. Shared and searchable encrypted data for untrusted servers [J]. Journal of Computer Security, 2011, 19(3): 367-397

[13] Zhao Fangming, Nishide T, Sakurai K. Multi-user keyword search scheme for secure data sharing with fine-grained access control [G] //LNCS 7259: Information Security and Cryptology (ICISC 2011). Berlin: Springer, 2012: 406-418

[14] Li Jingwei, Li Jini, Chen Xiaofeng, et al. Efficient keyword search over encrypted data with fine-grained access control in hybrid cloud [G] //LNCS 7645: Network and System Security. Berlin: Springer, 2012: 490-502

[15] Zhao Fangming, Nishide T, Sakurai K. Realizing fine-grained and flexible access control to outsourced data with attribute-based cryptosystems [G] //LNCS 6672: Information Security Practice and Experience. Berlin: Springer, 2011: 83-97

[16] Liu Zheli, Wang Zhi, Cheng Xiaochun, et al. Multi-user searchable encryption with coarser-grained access control in hybrid cloud [C] //Proc of the 4th Int Conf on Emerging Intelligent Data and Web Technologies. Piscataway, NJ: IEEE, 2013: 249-255

[17] Popa R A, Zeldovich N. Multi-key searchable encryption [OL]. [2015-06-07]. http://eprint.iacr.org/2013/508.pdf

[18] Tang Q. Nothing is for free: Security in searching shared and encrypted data [J]. IEEE Trans on Information Forensics and Security, 2014, 9(11): 1943-1952

[19] Rosenthal A, Mork P, Li M H, et al. Cloud computing: A new business paradigm for biomedical information sharing [J]. Journal of Biomedical Informatics, 2010, 43(2): 342-353

[20] Li Ming, Yu Sucheng, Zheng Yao, et al. Scalable and secure sharing of personal health records in cloud computing using attribute-based encryption [J]. IEEE Trans on Parallel and Distributed Systems, 2013, 24(1): 131-143

[21] Zhao Gansen, Rong Chunming, Li Jin, et al. Trusted data sharing over untrusted cloud storage providers [C] //Proc of the 2nd IEEE Int Conf on Cloud Computing Technology and Science. Piscataway, NJ: IEEE, 2010: 97-103

[22] Curtmola R, Garay J, Kamara S, et al. Searchable symmetric encryption: Improved definitions and efficient constructions [C] //Proc of the 13th ACM Conf on Computer and Communications Security. New York: ACM, 2006: 79-88

[23] Blaze M, Bleumer G, Strauss M. Divertible protocols and atomic proxy cryptography [G] //LNCS 1403: Advances in Cryptology (EUROCRYPT'98). Berlin: Springer, 1998: 127-144

[24] Boneh D, Franklin M. Identity-based encryption from the Weil pairing [G] //LNCS 2139: Advances in Cryptology (CRYPTO 2001). Berlin: Springer, 2001: 213-229

[25] Popa R A. Building practical systems that compute on encrypted data [D]. Cambridge, MA: Massachusetts Institute of Technology, 2014

Li Zhen, born in 1979. PhD candidate and lecturer. Received her master and bachelor degree from Shandong University. Her research interests include secure multi-party computation and searchable encryption.

Jiang Han, born in 1974. PhD and lecturer. His main research interests include cryptography and information security, especially secure multi-party computation.

Zhao Minghao, born in 1991. Master candidate. His main research interests include secure computation, cloud computing, cloud security and searchable encryption (zhaominghao@hrbeu.edu.cn).