对称可搜索加密技术研究进展

王赟玲 陈晓峰*

(西安电子科技大学网络与信息安全学院 西安 710071)

摘 要：云计算作为一种新型计算模式，具有海量资源、动态扩展、按需分配等特点。资源受限的用户可以将计算任务外包给云服务器，在享受高质量数据服务的同时大大降低了本地管理开销。然而，数据外包导致数据所有权与管理权分离，如何保证数据的安全性成为云计算中亟待解决的关键问题。传统的加密技术虽然可以保证数据的机密性，但是在密文中如何执行有意义的检索操作成为一个巨大的挑战。为了保证数据机密性的同时实现密文数据的高效检索，可搜索加密技术应运而生。近年来，可搜索加密方案的设计日趋多样化，旨在提高方案的实用性。该文主要围绕目前可搜索加密方案的研究热点，从4个方面展开阐述，具体包括：单关键词检索、多模式检索、前/后向安全检索和可验证检索。该文主要介绍和分析具有代表性的研究成果，总结最新研究进展及提炼关键技术难点，最后对未来的研究方向进行展望。

关键词：密文检索；单关键词检索；多模式检索；前/后向安全检索；可验证检索

中图分类号：TN918 **文献标识码**：A **文章编号**：1009-5896(2020)10-2374-12

DOI: 10.11999/JEIT190890

Research on Searchable Symmetric Encryption

WANG Yunling CHEN Xiaofeng

(*School of Cyber Engineering*, *Xidian University*, *Xi'an* 710071, *China*)

Abstract: Cloud computing, as a new computing paradigm, offers dynamically scalable and seemly unbounded storage and computation resources in a pay-as-you-go manner. In order to enjoy superior data services and reduce the local maintenance cost, more and more resource-constrained users prefer to outsource their data to the cloud server. However, outsourcing data to the remote server suffers from data security concerns, because the server may try to learn the information of the outsourced data as much as possible for commercial purpose. The traditional encryption technique can protect the confidentiality of users' data, however, it leads to the loss of search ability over the encrypted data. Fortunately, searchable encryption, as a promising solution, enables the server to perform keyword-based search over encrypted data. Recently, the design of searchable encryption scheme is becoming more and more diversified, aiming to improve the practicability of the scheme. This paper focuses on the current research for searchable encryption scheme in four aspects, including single keyword search, multi-modal search, forward/ backward secure search and verifiable search. This paper mainly introduces and analyzes the representative research results, summarizes the latest research progress and key technical difficulties, and finally prospects the future research direction.

Key words: Search over encrypted data; Single keyword search; Multi-modal search; Forward/backward secure search; Verifiable search

1 引言

云计算是分布式计算、效用计算、并行计算、虚拟化等计算机技术的演进和跃升，实现了人们长期以来“把计算作为一种基础设施”的梦想。通过云计算技术，厂商建立网络服务器集群，向不同类型客户提供在线软件服务、硬件租借、数据存储、计算分析等服务。如云网络存储工具Dropbox、亚马逊简易储存服务(Amazon simple storage service)和微软的云计算平台Windows Azure等。资源受限的用户通过按需付费的方式获得云服务商提供的近似无尽的存储和计算资源，大大降低了用户在本地的数据管理开销。因此，云环境中数据外包技术成为研究热点。

收稿日期：2019-11-07；改回日期：2020-05-02；网络出版：2020-05-13
*通信作者： 陈晓峰 xfchen@xidian.edu.cn
基金项目：国家密码发展基金(MMJJ20180110)
Foundation Item: The National Cryptography Development Fund (MMJJ20180110)

然而，数据外包在为人们带来诸多益处的同时，不可避免地面临着一些安全挑战和问题。其中最主要的是用户外包数据中往往包含敏感信息，这些敏感信息存在不可控存储环境被泄漏的安全风险。如电子病历存储、个人邮件处理、财政数据审计等。因此，如何保证用户数据的安全性成为云计算中的关键问题。传统的数据加密技术能有效保证外包数据的机密性，但是，如何在密文中执行有意义的操作成为困难，特别是检索操作。一种直接的方法为：用户将数据加密后外包给云服务器，需要对某个关键词进行检索时，下载密文数据到本地，解密后，在明文中执行检索操作。然而，该方法会带来大量的通信代价和计算开销。因此，如何在密文中执行高效安全的检索成为具有挑战性的问题。

一种可行的解决方法是采用可搜索加密(Searchable Encryption, SE)技术，它能在密文中高效地执行检索操作。可搜索加密技术分为两大类：对称可搜索加密技术(Searchable Symmetric Encryption, SSE)[1]和公钥可搜索加密技术(Public key Encryption with Keyword Search, PEKS)[2,3]。两类技术在构造方法和使用场景方面都有不同，具体来讲：(1) 两类技术的构造方法不同。其中，SSE方案的构造基于对称密码原语来设计。一般有两方参与者：数据拥有者和云服务器。数据拥有者用私钥加密数据后外包给云服务器，之后数据拥有者使用私钥检索云服务器中的数据。与之对应的PEKS方案的构造基于公钥密码原语。一般有三方参与者：数据拥有者、云服务器和用户。数据拥有者用公钥加密数据后外包给云服务器，之后用户使用私钥检索云服务器中的数据。由于PEKS方案基于公钥密码构造，因此方案的构造效率低下。(2) 两类方法的应用场景不同。SSE的应用场景多样，如医疗数据、财务数据、政府数据等私有数据库的外包与检索，与之对应的PEKS主要应用于加密邮件系统。基于上述两个原因，SSE技术成为近年来学术界的研究热点，大量SSE方案被提出。此外，工业界微软公司提出的加密云存储平台(cryptographic cloud storage)实现了密文数据的存储与检索。

目前，SSE方案的研究内容主要分为如下4个方面：(1) 单关键词检索：单关键词检索是SSE的基本检索方式，即返回包含单个检索关键词的所有文档。在单关键词检索方面，如何提高检索效率和安全性成为研究热点；(2) 多模式检索：为了构造更加实用的SSE方案，需要支持更加丰富检索的方式，如多关键词检索、模糊关键词检索、检索结果排序和范围查询等；(3) 前/后向安全检索：动态SSE方案需要支持文件的更新和删除，但是在动态更新中会泄漏重要信息，如何保证动态SSE方案的前/后向安全成为关键；(4) 可验证检索：在恶意服务器模型中，如何设计可验证的SSE方案，即验证服务器返回结果的正确性和完整性成为必要。本文主要围绕这4个方面对SSE的研究热点和难点展开阐述。

2　对称可搜索加密技术

2.1　系统模型

SSE由Song等人[1]首次于2000年提出。一般来讲，SSE由数据拥有者和云服务器组成。如图1所示。

(1) 数据拥有者：由于数据拥有者自身的存储和计算资源不足，需要将本地文档数据外包给云服务器存储。首先，数据拥有者将外包文档加密，并生成检索索引，将密文文档和检索索引一起外包给云服务器。其次，当数据拥有者想要检索某个特定关键词时，生成该关键词的检索陷门，并提交给云服务器进行检索。

(2) 云服务器：云服务器的主要任务是执行检索操作。当服务器收到用户的检索陷门时，利用该陷门在检索索引中检索，最终将满足检索条件的所有文档返回给数据拥有者。服务器无法从检索索引、加密文档和检索陷门中获取用户检索关键词和外包数据的内容。

2.2　形式化定义

假设数据库由$\mathrm{DB}=(\mathrm{id}_i,W_i)_{i=1}^{d}$文档地址/关键词集构成，其中$\mathrm{id}_i$为第$i$文档，$W_i$为文档$i$包含的关键词集合，$d$为数据库中文档的数量。一个SSE方案由4个算法组成，即$\Pi=(\mathrm{KeyGen},\mathrm{Setup},\mathrm{Trapdoor},\mathrm{Search})$，具体定义如下：

(1) $\mathrm{KeyGen}(1^k)$：概率性的密钥生成算法。用户输入安全参数k，输出密钥K；

(2) $\mathrm{Setup}(K,\mathrm{DB})$：用户使用此算法生成密文的检索索引。该算法输入密钥K和明文数据库DB，输出密文检索索引$\mathcal{I}$；

(3) 用户生成检索关键词的陷门算法。该算法输入密钥K和检索关键词w，输出检索陷门T_w；

(4) $\mathrm{Search}(\mathcal{I},T_w)$：服务器执行检索算法。该

图 1 SSE方案系统模型

算法输入检索索引$\mathcal{I}$和检索陷门T_w，输出包含检索关键词w的所有文档标识DB(w)。

2.3 安全性定义

Curtmola等人[4]首次给出SSE的正式安全性定义，通过使用泄漏函数$\mathcal{L}$刻画SSE方案的具体泄漏，由真实游戏和理想游戏证明SSE方案除了泄漏函数外不泄漏其它任何消息。具体来讲，真实游戏$\text{Real}_{\mathcal{A}}^{\Pi}$和理想游戏$\text{Real}_{\mathcal{A},\mathcal{S}}^{\Pi}$定义如下，其中$\mathcal{A}$为攻击者，$\mathcal{S}$为模拟器。

$\text{Real}_{\mathcal{A}}^{\Pi}(k)$：攻击者选择数据库DB，该游戏运行Setup(DB)算法生成检索索引$\mathcal{I}$并发送给$\mathcal{A}$。接着，敌手$\mathcal{A}$自适应的选择一系列查询q，该游戏运行Trapdoor(q)算法生成的T_q并发送给$\mathcal{A}$。然后，该游戏运行Search($\mathcal{I},T_q$)，将运行的所有结果发送给$\mathcal{A}$。最后，$\mathcal{A}$输出一个比特b。

$\text{Ideal}_{\mathcal{A},\mathcal{S}}^{\Pi}$：模拟器初始化查询数组$\boldsymbol{q}$。攻击者选择数据库DB，该游戏运行$\mathcal{S}(\mathcal{L}(\text{DB}))$算法生成检索索引$\mathcal{I}$并发送给$\mathcal{A}$。接着，敌手$\mathcal{A}$自适应的选择查询$q$。模拟器记录下查询$\boldsymbol{q}[i]$，运行$\mathcal{S}(\mathcal{L}(\boldsymbol{q},\text{DB}))$。最后，$\mathcal{A}$输出一个比特$b$。

一个SSE方案Π，如果对于所有的多项式概率时间敌手$\mathcal{A}$，存在一个高效的模拟器$\mathcal{S}$和一个可忽略函数negl，使得

$$Pr[\text{Real}_{\mathcal{A}}^{\Pi}(k)=1]-Pr[\text{Ideal}_{\mathcal{A},\mathcal{S}}^{\Pi}=1]\le \text{negl}(k) \quad (1)$$

则Π是$\mathcal{L}$-适应性安全的方案。

在该定义中，敌手是自适应性的。具体来讲，敌手可以自适应地选择查询关键词。因此，在自适应的定义中，模拟器$\mathcal{S}$只有利用DB的泄漏信息$\mathcal{L}$(DB)来模拟检索索引。该定义通过说明敌手无法判断和证实游戏还是模拟游戏交互，证明了自适应性敌手除了获得泄漏函数$\mathcal{L}$外，无法获得关于用户数据的其他任意信息。在SSE方案中，检索模式和访问模式是常见泄露：

(1) 检索模式：泄露用户的两次查询是否相同。检索模式被定义为一个$t\times t$的矩阵$\mathbf{SP}$，如果$Q_i=Q_j$，那么$\text{SP}[i,j]=1$；如果$Q_i\ne Q_j$,$\text{SP}[i,j]=0$，其中，Q_i表示第i次查询，t为总共查询次数。

(2) 访问模式：泄露用户的检索到的文档标识。访问模式定义为{DB(Q_1),DB(Q_2),$\cdots$,DB(Q_t)}。

2.4 正确性定义

SSE方案的正确性需要满足如式(2)所示的性质[5]：对于所有的DB和W，若任意的$w\in \text{W}$，满足

$$\text{Search}(\mathcal{I},T_w)=\text{DB}(w) \quad (2)$$

其中，$\mathcal{I}=\text{Setup}(K,\text{DB})$, $T_w=\text{Trapdoor}(K,w)$。

2.5 安全模型

在SSE方案中，云服务器一般分为2种模型：诚实且好奇(honest-but-curious)模型和恶意(malicious)模型[6–9]。

(1) 诚实且好奇服务器模型：服务器会忠实地执行协议，但会尽可能多地挖掘用户数据中的隐私信息。

(2) 恶意服务器模型：服务器会忠实地执行部分协议，但会返回部分或者不正确检索结果给用户。

通常，SSE在诚实且好奇的服务器模型下构造，在模型中，主要考虑如何增强方案的安全性和提高效率。在恶意服务器模型下，出于服务器自身的经济利益(节省通信代价或计算量)或软硬件故障，服务器会返回不正确或部分检索结果，用户需要验证检索结果的正确性和完整性。

3 单关键词SSE方案

3.1 顺序扫描SSE方案

Song等人[1]首次提出可搜索加密的概念，并构造了第1个SSE方案。该方案的主要思想是：数据拥有者对文件中的每个关键词进行特殊的加密，然后将加密的文档外包给服务器；数据拥有者需要对特定的关键词进行检索时，将该关键词陷门发送给服务器；服务器将检索陷门和密文文档中的每个密文关键词进行顺序扫描，若检索陷门与某个密文关键词匹配成功，则返回该密文文档。接下来，本文简单介绍如何加密文档中的每个关键词，示意图如图2所示。

(1) KeyGen(1^k)：产生两个密钥k'和k''。

(2) Setup(K,DB)：对文档中的关键词w_i，采用特殊的两层加密。首先，加密关键词w_i为$X_i=E_{k''}(w_i)$。然后，将X_i分成左右两部分$X_i=<L_i,R_i>$，其中L_i为前$n-m$ bit, R_i为后m bit；生成$k_i=f_{k'}(L_i)$。接着，使用流密码生成器生成$n-m$ bit S_i，并产生m bit $F_{k_i}(S_i)$，记为$T_i=<S_i,F_{k_i}(S_i)>$。最终，产生密文$C_i=X_i\oplus T_i$。

(3) Trapdoor(K,w)：用户生成检索关键词w_i的陷门。具体为，$X_i=E_{k''}(w_i)$和S_i。

(4) Search($\mathcal{I},T_w$)：服务器对密文文档中的每个密文关键词进行匹配。首先，服务器用C_i的前$n-m$ bit与S_i异或得到L_i；然后，利用L_i生成k_i。接着，利用k_i生成$F_{k_i}(S_i)$，并与C_i的后m bit异或，得到R_i。最后，如果$<L_i,R_i>$与陷门X_i相同，则返回该密文关键词的文档。

虽然该方案首次实现了密文中的检索，不过存在以下2个不足：(1) 检索效率低下。在检索时，服务器需要根据检索陷门进行逐文档逐关键词的检索，因此，检索效率与整个数据库的大小相关。(2) 没有给出严格的安全性定义。

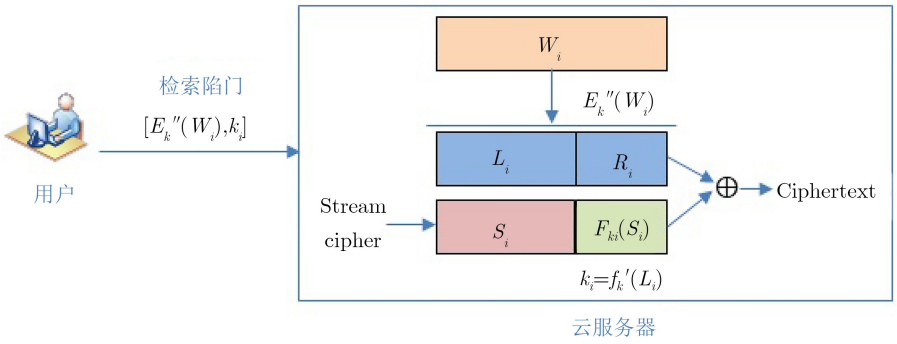


图 2 Song等人的方案

3.2 正排索引SSE方案

Goh[10]为了提高检索效率，构造了基于正排索引的SSE方案。正排索引主要是为每个外包文档建立一个索引，然后在该索引上执行高效检索，不需要对每个文档进行逐关键词匹配。该方案主要利用Bloom Filter[11]可以快速判断一个元素是否在一个集合中的思想来构造前向索引。具体构造思想为：对于每个文档，将该文档中的所有关键词插入到该文档的Bloom Filter中。检索时，服务器利用Bloom Filter来判断该文档是否包含查询关键词。该方案的主要思想如图3所示。相比于Song等人的方案，该方案提高了检索效率，该方案的检索复杂度与数据库中的文档数相关。

3.3 倒排索引SSE方案

Curtmola等人[4]为了进一步提高检索效率，构造了基于倒排索引的SSE方案。倒排索引主要是针对每个关键词对应的文档来构造检索索引。执行检索操作时，无需遍历所有文档来判断是否包含检索关键词，而是直接利用检索陷门找到对应的文档。该方案主要用地址链的思想来实现，以下本文用一个例子来简要说明。对于关键词w_1，有3个文档id_1, id_2, id_3包含该关键词，那么需要存储3个关键词-文档对(w_1,id_1), (w_1,id_2)和(w_1,id_3)。每个关键词-文档对用一个节点来存储，该节点有3部分组成，分别为文档、密钥和地址，记为$<\mathrm{id},k,\mathrm{addr}>$。具体来讲，$(w_1,\mathrm{id}_1)$对应的节点为$<\mathrm{id}_1,k_{12},\mathrm{addr}_{12}>$，该节点用$k_{11}$加密，存储在$\mathrm{addr}_{11}$的地址中；$(w_2,\mathrm{id}_2)$对应的节点为$<\mathrm{id}_2,k_{13},\mathrm{addr}_{13}>$，该节点用$k_{12}$加密，存储在$\mathrm{addr}_{12}$的地址中；$(w_3,\mathrm{id}_3)$对应的节点为$<\mathrm{id}_3,\perp,\perp>$，该节点用$k_{12}$加密，存储在$\mathrm{addr}_{12}$的地址中。当检索$w_1$时，检索陷门为$<k_{11},\mathrm{addr}_{11}>$。服务器利用$\mathrm{addr}_{11}$找到第1个节点的位置，然后利用$k_{11}$解密该节点，得到$<\mathrm{id}_1,k_{12},\mathrm{addr}_{12}>$，即得到一个结果$\mathrm{id}_1$；然后利用$\mathrm{addr}_{12}$找到第2个节点的位置，利用$k_{12}$解密，得到$<\mathrm{id}_2,k_{13},\mathrm{addr}_{13}>$，即得到结果$\mathrm{id}_2$；依照此方式，检索出所有结果$\mathrm{id}_1$, id_2和id_3。该方案的主要思想如图4所示，具体方案构造参考文献[4]。

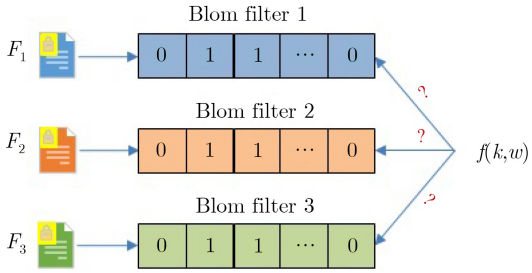


图 3 Goh方案

由此看出，该方案的检索复杂度为亚线性的，即检索复杂度只与检索关键词包含的文档数有关，进一步提高了基于正排索引的检索效率。此外，该方案给出了SSE正式的安全性定义(在本文第2.2节中描述)，被目前SSE方案广泛采用。本文的安全性定义指出SSE方案有两类典型的泄漏：访问模式泄漏和检索模式泄漏。其中，访问模式为从检索结果文档中获取的信息，检索模式为从两次查询是否对应相同关键词中获取的信息。目前，一些攻击表明[12–14]利用访问模式和检索模式可以恢复出用户的查询关键词。因此，访问模式和检索模式是两类重要的信息。

目前，如何保护访问模式和检索模式成为研究热点。(Oblivious Random Access Machine, ORAM)[15]是保护访问模式的有效方法，用户通过重构和重加密远程数据来保护访问模式。但是，如果要从n个文档中不经意的访问其中的一个文档，至少要访问$O(\log n)$文档。所以，通信代价和计算代价使得ORAM技术不实用。Chen等人[16]使用差分隐私的思想来隐藏访问模式。Mishra等人[17]利用ORAM, ODS (Oblivious Data Structure), ODM (Oblivious Sorted Multimap)和SGX (Software Guard Extensions)保护访问模式和检索长度。然而，保护访问模式的方法依旧面临效率不高的问题。同样，如何保护检索模式更是一个挑战。主要

原因是检索模式的泄漏不仅来自于确定性陷门生成函数，还可以由访问模式和检索长度推断。最近，Kamara等人[5]利用ORAM的思想保护检索模式，Wang等人[18]利用隐私集合求交集的思想保护检索模式。但是，保护检索模式的方法依然不高效。因此，如何高效地保护访问模式和检索模式仍然是一个挑战。

4 多模式SSE方案

单关键词检索使得服务器返回大量满足检索条件的文档，用户需进一步筛选出想要的文档，无法满足现实中高效检索的需求。因此，支持多模式的SSE方案引起人们的关注。主要包括：连接关键词检索、多用户群组检索、模糊关键词检索、排序检索和范围查询等。

4.1 连接关键词检索

连接关键词检索是多模式检索中最基本的模式，即服务器返回包含所有查询关键词的文档[19]。一种直接的方式为：服务器执行单关键词检索，然后将单关键词检索结果的交集返回给用户。然而，这种直接的方式会导致检索不高效(对每个关键词进行检索)和泄漏信息过多(每个查询关键词的结果会泄漏)。因此，如何构造高效安全的连接关键词检索方案成为研究热点。Golle等人[20]首次提出两个非直接的连接关键词检索方案。但是，两个方案中的检索复杂度都与文档数量线性相关，而且有大量的模指数运算和双线对运算。因此，该方案的检索效率不高。

Cash等人[21]提出首个亚线性的连接关键词查询方案(Oblivious Cross-Tags, OXT)，并可以将方案扩展为支持关键词的布尔查询。该方案的主要思想分为两步：首先，选取频率最小的关键词(在所有检索的关键词中，频率最小的关键词对应的文档数最少)来检索；然后，判断检索结果是否包含其它检索的关键词，若该结果包含其它所有的检索关键词，则该结果是最终的检索结果。具体来讲，OXT方案中的检索索引包含两个结构：TSet和XSet。本质上，TSet是倒排索引，用于检索单关键词。例如，用户查询$w_1 \wedge \cdots \wedge w_q$，其中$w_1$为频率最小的关键词。那么，用户使用$w_1$的检索陷门在TSet中检索出包含$w_1$的所有文档集$\mathrm{DB}(w_1)$。此外，XSet本质上存储数据库中所有的关键词-文档对的哈希值$h(w,\mathrm{id})$ (文档id包含关键词w)。用户生成w_1与其余查询关键词对(w_1,w_i)的“交叉陷门”$\mathrm{xtoken}(w_1,w_i)$, $2 \le i \le q$。服务器使用$\mathrm{xtoken}(w_1,w_i)$过滤出$\mathrm{DB}(w_1)$中包含w_i的文档$\mathrm{DB}(w_1) \cap \mathrm{DB}(w_i)$。从而，通过所有的$\mathrm{xtoken}(w_1,w_i)$，服务器能够检索出包含所有查询关键词的文档$\mathrm{DB}(w_1) \cap \cdots \cap \mathrm{DB}(w_q)$。更具体来讲，为了从$\mathrm{DB}(w_1)$中过滤出$\mathrm{DB}(w_1) \cap \mathrm{DB}(w_i)$，服务器需要遍历$\mathrm{DB}(w_1)$中的每个id，并且利用$\mathrm{xtoken}(w_1,w_i)$判断$h(w,\mathrm{id})$是否在XSet中。图5为该方案的一个简单示例，其中，用户想要检索同时包含w_1, w_2和w_3的文档。

4.2 泄漏优化的连接关键词检索

然而，Lai等人[22]指出该方案在泄漏信息方面仍然存在有待提高的地方。具体来讲，对于包含频率最小关键词w_1的所有文档，会泄漏它们是否分别包含其它的检索关键词，称为结果模式泄漏。如：id_1包含w_2，但不包含w_3; id_3同时包含w_2和w_3。然而，在该方案中，对于每个文档，知道该文档是否

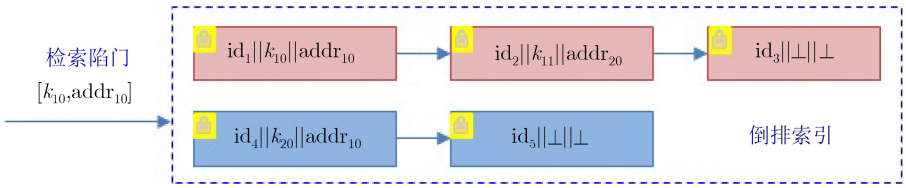


图 4 Curtmola等人方案

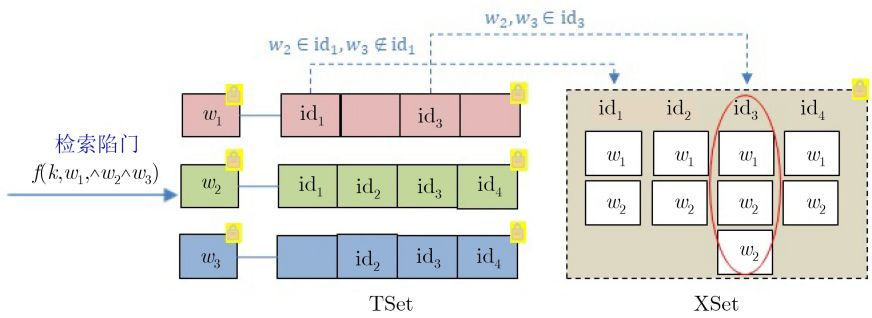


图 5 Cash等人方案

包含除了w_1以外的其它所有的关键词的信息。如：id_1是否同时包含w_2和w_3; id_3是否同时包含w_2和w_3。为了避免上述不必要的泄漏，Lai等人提出了对称的向量隐藏加密技术(Hidden Vector Encryption, HVE)，并基于该技术提出结果模式隐藏的连接关键词检索方案HXT。本质上讲，Cash等人方案的泄漏主要来自于XSet的结构，它需要对单个关键词-文档对进行判断，而HVE可以判断多个元素是否在一个集合中。因此，Lai等人的方案中服务器只能判断文件是否包含所有检索关键词，而不能单独去判断。该方案的主要思想如图6所示，具体构造参考文献[22]。

从Cash等人提出OXT方案后，研究者关注于对OXT在其他检索模式方面的扩展。Sun等人[23]将OXT扩展为多用户场景。具体来讲，传统的SSE方案只允许数据拥有者自身去执行检索，多用户场景的SSE允许除了数据拥有者外其他的用户执行检索。为此，Sun等人提出了无交互的细粒度多用户SSE方案。在该方案中，不同的授权用户拥有不同的检索和解密权限：检索权限为每个用户被授予不同的检索关键词集合，用户只能检索被授权的关键词；解密权限为文档标识由ABE加密，只有用户的属性满足加密策略时，才可以解密文档标识。然而，Wang等人[24]指出该多用户检索方案有严重的通信和计算代价。主要原因为：文档标识由ABE加密，服务器无法判断用户是否能解密某个加密文档。因此，服务器需要返回所有满足检索条件的结果，用户需自行判断能否解密，带来了额外的通信和解密代价。为此，Wang等人提出了服务器端匹配的匿名属性ABE技术，在不泄漏用户属性的情况下，服务器可以判断用户能否解密密文。用此技术加密文档标识，服务器可以仅仅返回满足检索条件且用户能够解密的结果，降低了通信代价并提高了解密效率。Kamara等人[25]指出OXT在执行析取关键字检索(返回包含任意查询关键词的文档)时效率低下的问题，并在OXT的基础上实现高效的析取关键字检索。

此外，多模式检索还包括以下几个方面。首先，为了在有少量拼写错误的情况下，依然可以检索出正确的结果，模糊关键词SSE方案被提出[26,27]。其次，当检索结果数量较大时，如何返回与检索条件最相关的k个检索结果，排序的SSE方案[28–30]被提出。而且，当外包数据是数字时，如何返回在某个数字区间的所有文档，支持范围查询的SSE方案被提出[31,32]。

5　前/后向安全的SSE方案

SSE方案需要支持动态的数据更新：文件的添加和删除。然而，数据更新中往往会泄漏重要的信息。因此，如何保证动态数据更新的安全性成为研究热点。

5.1　前向安全的SSE方案

前向安全性主要针对文档的添加。传统的SSE方案在添加文档时会泄漏如下信息：之前的检索关键词是否包含在添加的文档中。前向安全保证了无法判断新添加的文档是否包含之前检索的关键词，不满足前向安全的动态SSE方案会遭受灾难性的文件注入攻击[33]，该攻击会高效准确地恢复用户检索关键词的内容。

为了抵抗文件注入攻击，文件添加时需要满足前向安全性，即不泄漏添加的文档中是否包含之前检索过的关键词。Stefanov等人[34]首次给出前向安全SSE的定义，并给出了具体的构造。该构造主要采用ORAM的思想，如果数据库中有n个关键词-文档对，那么用户需要建立$d=\lceil \lg n \rceil$层不同密钥加密的数据集，即$(E_0, E_1, \cdots, E_{d-1})$，每一层存储$2^i$个实体，每一层的数据都用不同的密钥进行加密。具体更新过程如下：更新数据时，首先找数据结构中存有数据的下层空闲位置，若存在，则将数据存放在该位置上；若不存在，则用户下载所有层的数据到本地，解密并重新加密放到与数据个数相同的一层。该方案的主要思想如图7所示，以3层数据结构为例说明数据更新的过程。然而，该方案的更新需要较大的计算和存储开销，而且用户端也需要较大的存储空间。方案的具体构造可参考文献[34]。

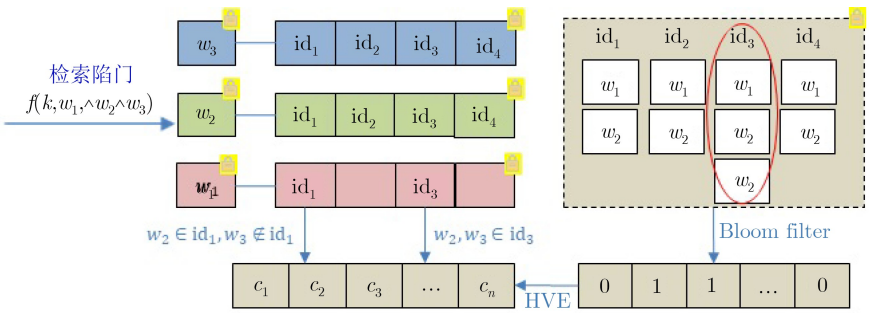
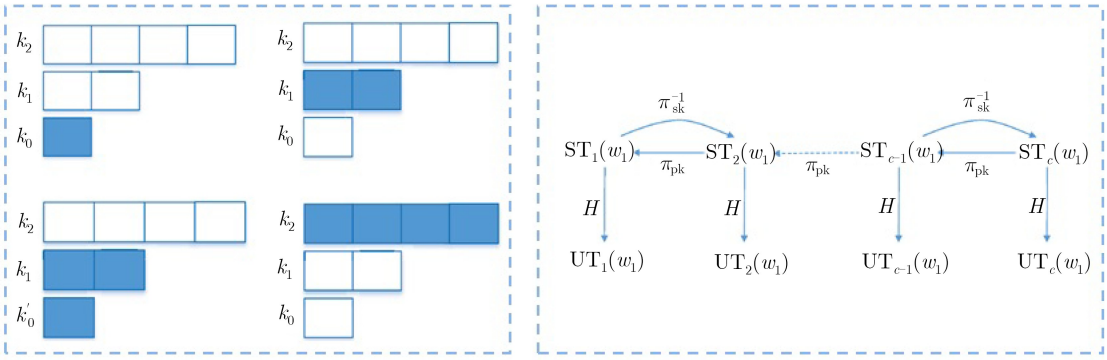


图 6 Lai等人方案



(1) Stefanov 等人方案 (2) Boct方案

图 7 前向安全方案

为了降低通信和计算开销，Bost[35]提出新的前向安全方案，更新数据时无需服务器与用户之间交互。主要思想依然采用倒排索引的结构。在更新关键词-文档对(w,id)时，利用单向陷门置换函数π、私钥和前一个状态ST_{c-1}，产生新的状态ST_c，然后利用新的状态生成新的位置UT_c，将id存放在该位置上。检索时，用户将最新的状态ST_c发送给服务器，服务器使用公钥生成所有之前的状态，$\mathrm{ST}_i, i<c$，然后产生对应的位置取出相应的文档。由于服务器没有私钥，无法使用当前的状态ST_c生成下个状态ST_{c+1}。因此，无法使用之前的检索陷门在新的文件中进行检索，保证了前向安全性。图7给出了该方案的主要思想，用户使用陷门置换函数π和私钥产生新的状态，服务器使用陷门置换函数π和公钥执行反向操作。通过此方式保证了SSE方案的前向安全性。方案的具体构造可参考文献[35]。

从Bost等人的方案提出后，后续前向安全SSE方案主要是对其的改进和扩展。Song等人[36]指出虽然Bost方案降低了服务器与用户端的通信量，但是，单向陷门置换函数π是公钥密码原语。因此，当频繁地更新时，公钥操作成为该方案主要的性能瓶颈。针对此问题，Song等人构造了利用对称密码原语的前向安全方案，进一步提高了更新和检索的效率。Kim等人[37]构造了支持高效删除的前向安全SSE方案。然而，以上方案主要针对前向安全SSE的单关键词检索，如何使前向安全SSE方案支持多模式的检索同样是研究热点。Zuo等人[38]利用Paillier加密构造了支持范围查询的前向安全方案，但是该方案受限于更新的文档数量。Wu和Li[39]基于树结构设计了支持连接关键词的前向SSE方案，然而，该方案会泄漏每个检索关键词的结果。Hu等人[40]利用内积加密构造了支持连接关键词的前向安全SSE方案，避免泄漏每个检索关键词的结果。但是，内积加密技术的引入使得方案效率不高。Wang等人[41]采用Bost单关键词前向安全SSE方案和OXT的思想，构造了高效的支持连接关键词的前向SSE方案，并不会存在泄漏每个检索关键词的结构。

5.2 后向安全的SSE方案

后向安全性主要针对文档的删除。传统的SSE方案在删除文档时存在删除的文档依然可以被检索出的问题。Stefanov等人[34]首次给出后向安全SSE方案的定义，但是没有给出具体的构造。Bost等人[42]首次给出后向安全SSE方案的具体构造。根据在删除过程中的泄漏信息的不同，后向安全有3个强度的定义：Ⅰ型、Ⅱ型和Ⅲ型。其中Ⅰ型的安全性最强，Ⅲ型的安全性最弱。下面简单介绍3种安全性的区别。

(1) Ⅰ型(带有插入模式的后向安全性)：该安全性会泄漏目前与关键词w匹配的文档，这些文档是何时插入到数据库的，以及关键词w总共更新(插入和删除)次数。

(2) Ⅱ型(带有更新模式的后向安全性)：该安全性会泄漏目前与关键词w匹配的文档，这些文档是何时插入到数据库的，以及关键词w的所有更新(插入和删除)发生的时间。

(3) Ⅲ型(弱后向安全性)：该安全性会泄漏目前与关键词w匹配的文档，这些文档是何时插入到数据库的，以及何时的删除操作是针何时的插入操作。

Bost等人首次构造了不同安全等级的后向安全方案，其中典型的为仅用一轮交互实现了弱后向安全的SSE方案Janus。该方案主要采用可穿刺加密(Puncturable Encryption, PE)原语[43]来实现，可穿刺加密技术通过更新密钥来删除消息中标签，那么，更新后的密钥无法解密包含删除标签的消息。利用该性质，用户通过更新密钥来删除数据，实现了后向安全性。然而，Sun等人[44]指出，Janus方案中的可穿刺加密是公钥原语，当删除频繁时，公

钥操作使得删除的效率低下。为此，Sun等人利用简单的密码学工具(伪随机函数)构造了对称的可穿刺加密原语，大大提高了更新和检索的效率。为了进一步优化检索效率，Chamani等人[45]利用Oblivious Map[46]和Path-ORAM[47]技术构造了新的不同安全等级的后向安全SSE方案。然而，对于后向安全的SSE方案，如何同时提高效率和安全等级依然有待进一步的研究。

6　可验证的SSE方案

需要指出的是，上述方案均考虑诚实且好奇的服务器模型，即服务器虽然会挖掘用户数据中的隐私信息，但会诚实地执行协议并返回正确完整的检索结果。然而，在实际中，由于服务器软硬件运行故障，或为了节省网络带宽和计算资源，它可能返回给用户不正确或不完整的检索结果。因此，如何实现检索结果的可验证性是一个挑战。具体来讲，可验证性主要包含如下两个方面：(1)正确性：检索结果是否满足检索条件；(2)完整性：检索结果是否包含全部满足查询条件的数据，在完整性方面，如何验证服务器返回结果为空集的情况是一个难点。

对于单关键词检索SSE方案来讲，使用布隆过滤器、哈希函数、聚合器、默克哈希树、哈希链等验证工具可以实现[48–52]。但是，对于多模式的检索情况来讲，如何实现检索验证性是一个挑战。比如：如何构造可验证的连接关键词检索方案和可验证的前向安全检索方案。

6.1　可验证的连接关键词SSE方案

Azraoui等人[53]利用Cuckoo哈希函数和聚合器提出可公开验证的连接关键词检索方案，Sun等人[54]利用聚合器和默克哈希树实现连接关键词的公开可验证。然而，这两种验证方式依赖于如下检索方式：服务器检索出每个关键词的结果，然后求交集。这种检索方式导致前面提到的检索效率低和泄漏多的问题。针对此问题，Wang等人[8]基于OXT的连接关键词检索方法，采用聚合器和采样检测技术提出了公开可验证的连接关键词检索方案。该方案的主要思想如图8所示。

在该方案中，设计了3类聚合器：第1类是对所有关键词陷门的聚合Acc_W，第2类是对每个关键词包含的所有文档的聚合Acc_{w_i}，第3类是对XSet的聚合$\mathrm{Acc}_{\mathrm{XSet}}$。利用这3类聚合器，对以下3种情况进行验证：(1)当频率最小关键词w_1的检索结果R_{w_1}为空时，服务器返回空集，这时使用第1类聚合器来验证；(2)当R_{w_1}不为空，但是检索结果R为空时，首先利用第2类聚合器给出R_{w_1}的正确性和完整性，然后利用第3类聚合器给出R_{w_1}中的文档都不是检索结果的证据。(3)当R_{w_1}和检索结果R都不为空时，首先利用第2类聚合器验证R_{w_1}的正确性和完整性，然后利用第3类聚合器验证R中所有文档的正确性和完整性。方案的具体构造可参考文献[8]。

6.2　可验证的前向安全SSE方案

目前的可验证方案的构造主要针对静态的数据集，原因为可验证结构的构造取决于数据集的内容，并与数据集一起上传到服务器上。然而，在数据集动态更新的场景中，相应的证据需要相应的更新。而且，在产生新的证据时，往往需要之前上传的数据。因此，用户产生新的证据时，需要下载在服务器上所有的数据，产生新的证据后，将数据和新证据上传到服务器，这将带来显著的通信代价。因此，如何构造动态场景下的SSE方案成为挑战。

Zhang等人[9]利用多重集哈希函数构造了高效的可验证的前向安全SSE方案，产生新的证据时无需下载已经更新的数据。主要原因为多重集哈希具有增量更新的特点。具体来讲，在已知一个集合的哈希值的情况下，若集合中增加一个元素形成新的集合，那么新集合的哈希值可以通过原来集合的哈希值和新的元素生成，无需利用新集合中所有的元素来生成哈希值。因此，该可验证的前向安全SSE方案解决了不断下载已有数据的问题，大大降低了服务器与用户之间的通信开销。

表1对一些典型的SSE方案进行对比：其中DB为所有的文档集合，d为所有的文档数。

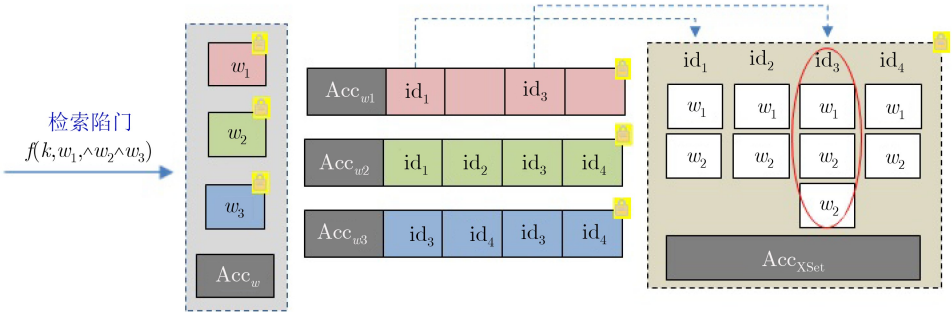


图 8 Wang等人方案

表 1 典型SSE方案的比较

方案	检索复杂度	检索类型	云计算模型	前/后向安全
Song等人方案[1]	$O(\mathrm{DB})$	单关键词	诚实且好奇	无
Goh方案[10]	$O(d)$	单关键词	诚实且好奇	无
Curtmola等人方案[14]	$O(\mathrm{DB}(w))$	单关键词	诚实且好奇	无
Cash等人方案[21]	$O(\mathrm{DB}(w_1))$	多关键词	诚实且好奇	无
Lai等人方案[22]	$O(\mathrm{DB}(w_1))$	多关键词	诚实且好奇	无
Bost方案[35]	$O(\mathrm{DB}(w))$	单关键词	诚实且好奇	前向安全
Bost等人方案[42]	$O(\mathrm{DB}(w))$	单关键词	诚实且好奇	前/后向安全
Wang等人方案[8]	$O(\mathrm{DB}(w_1))$	多关键词	恶意	无
Zhang等人方案[9]	$O(\mathrm{DB}(w))$	单关键词	恶意	前向安全

7 总结与展望

本文主要围绕云环境中密文数据的检索展开综述，分别介绍了单关键词密文检索、多模式密文检索、前/后向安全密文检索和可验证密文检索等方面的代表性研究成果。通过分析表明，现有的密文检索方案仍然存在一些不足，未来的研究工作可以关注以下几点：

(1) 对称可搜索加密方案的安全性主要靠泄露函数来刻画，然而，泄露函数往往会引发检索隐私泄露问题。如：访问模式和检索模式是密文检索中的典型泄漏，会被敌手用于恢复用户检索的关键词内容。目前，大量研究表明，检索结果数量同样是SSE方案中的重要信息，会被敌手用于恢复整个数据库中关键词对应的文档数量。然而，相比于访问模式和检索模式，检索结果数量更易于泄露和难以防范。一种直接的方法是将数据库中所有的关键词对应的文档数量填充为相同数量。但是，这种简单的填充会导致密文数量的大量扩张。最近，基于Cuckoo Hashing、伪随机函数等密码工具构造检索结果数量隐藏的密文检索方案相继提出。然而，这些方案仅支持单关键词检索，不支持多关键词查询、范围查询等多种检索模式。因此，如何设计支持多模式检索的检索结果数量隐藏的密文检索方案是未来的研究工作之一；

(2) 由于云服务器不完全可信，出于软硬件故障、经济利益考量，服务器可能会返回不正确或者不完整的检索结果，因此，可验证的密文检索方案成为研究热点。对于可验证的单关键词检索方案，可以使用消息鉴别码、默克哈希树等验证方法来构造。对于可验证多关键词检索，目前的方案主要使用聚合器和基于可证明数据拥有中随机抽取的思想来实现。然而，存在如下不足：(a) 聚合器的构造基于公钥密码原语，导致SSE的效率降低；(b) 使用随机抽取的思想导致验证为概率验证，存在误差。因此，如何设计高效数据认证结构，构造确定性可验证多关键词密文检索方案成为未来的研究工作之一。

参 考 文 献

[1] SONG D X, WAGNER D, and PERRIG A. Practical techniques for searches on encrypted data[C]. 2000 IEEE Symposium on Security and Privacy, Berkeley, USA, 2000: 44–55. doi: 10.1109/SECPRI.2000.848445.

[2] BONEH D, DI CRESCENZO G, OSTROVSKY R, *et al*. Public key encryption with keyword search[C]. International Conference on the Theory and Applications of Cryptographic Techniques, Interlaken, Switzerland, 2004: 506–522. doi: 10.1007/978-3-540-24676-3_30.

[3] 曹素珍, 郎晓丽, 刘祥震, 等. 抗关键词猜测的授权可搜索加密方案[J]. 电子与信息学报, 2019, 41(9): 2180–2186. doi: 10.11999/JEIT181103.

CAO Suzhen, LANG Xiaoli, LIU Xiangzhen, *et al*. Delegate searchable encryption scheme resisting keyword guess[J]. *Journal of Electronics & Information Technology*, 2019, 41(9): 2180–2186. doi: 10.11999/JEIT181103.

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

[5] KAMARA S, MOATAZ T, and OHRIMENKO O. Structured encryption and leakage suppression[C]. The 38th Annual International Cryptology Conference on Advances in Cryptology, Santa Barbara, USA, 2018: 339–370. doi: 10.1007/978-3-319-96884-1_12.

[6] WANG Jianfeng, MA Hua, TANG Qiang, *et al*. Efficient verifiable fuzzy keyword search over encrypted data in cloud computing[J]. *Computer Science and Information Systems*, 2013, 10(2): 667–684. doi: 10.2298/CSIS121104028W.

[7] WANG Jianfeng, CHEN Xiaofeng, HUANG Xinyi, *et al*.

Verifiable auditing for outsourced database in cloud computing[J]. *IEEE Transactions on Computers*, 2015, 64(11): 3293–3303. doi: 10.1109/TC.2015.2401036.

[8] WANG Jianfeng, CHEN Xiaofeng, SUN Shifeng, *et al*. Towards efficient verifiable conjunctive keyword search for large encrypted database[C]. The 23rd European Symposium on Research in Computer Security on Computer Security, Barcelona, Spain, 2018: 83–100. doi: 10.1007/978-3-319-98989-1_5.

[9] ZHANG Zhongjun, WANG Jianfeng, WANG Yunling, *et al*. Towards efficient verifiable forward secure searchable symmetric encryption[C]. The 24th European Symposium on Research in Computer Security on Computer Security, Luxembourg, 2019: 304–321. doi: 10.1007/978-3-030-29962-0_15.

[10] GOH E J. Secure indexes[J]. *IACR Cryptology ePrint Archive*, 2003, 2003: 216.

[11] BLOOM B H. Space/time trade-offs in hash coding with allowable errors[J]. *Communications of the ACM*, 1970, 13(7): 422–426. doi: 10.1145/362686.362692.

[12] ISLAM M S, KUZU M, and KANTARCIOGLU M. Access pattern disclosure on searchable encryption: Ramification, attack and mitigation[C]. Annual Network and Distributed System Security Symposium, San Diego, USA, 2012.

[13] CASH D, GRUBBS P, PERRY J, *et al*. Leakage-abuse attacks against searchable encryption[C]. The 22nd ACM SIGSAC Conference on Computer and Communications Security, Denver, USA, 2015: 668–679. doi: 10.1145/2810103.2813700.

[14] LIU Chang, ZHU Liehuang, WANG Mingzhong, *et al*. Search pattern leakage in searchable encryption: Attacks and new construction[J]. *Information Sciences*, 2014, 265: 176–188. doi: 10.1016/j.ins.2013.11.021.

[15] GOLDREICH O and OSTROVSKY R. Software protection and simulation on oblivious RAMs[J]. *Journal of the ACM*, 1996, 43(3): 431–473. doi: 10.1145/233551.233553.

[16] CHEN Guoxing, LAI T H, REITER M K, *et al*. Differentially private access patterns for searchable symmetric encryption[C]. IEEE Conference on Computer Communications, Honolulu, USA, 2018: 810–818. doi: 10.1109/INFOCOM.2018.8486381.

[17] MISHRA P, PODDAR R, CHEN J, *et al*. Oblix: An efficient oblivious search index[C]. 2018 IEEE Symposium on Security and Privacy, San Francisco, USA, 2018: 279–296. doi: 10.1109/SP.2018.00045.

[18] WANG Yunling, SUN Shifeng, WANG Jianfeng, *et al*. Achieving searchable encryption scheme with search pattern hidden[J]. *IEEE Transactions on Services Computing*, To be published. doi: 10.1109/TSC.2020.2973139.

[19] 孙瑾, 王小静, 王尚平, 等. 支持属性撤销的可验证多关键词搜索加密方案[J]. 电子与信息学报, 2019, 41(1): 53–60. doi: 10.11999/JEIT180237.

SUN jin, WANG Xiaojing, WANG Shangping, *et al*. Verifiable multi-keyword search encryption scheme with attribute revocation[J]. *Journal of Electronics & Information Technology*, 2019, 41(1): 53–60. doi: 10.11999/JEIT180237.

[20] GOLLE P, STADDON J, and WATERS B. Secure conjunctive keyword search over encrypted data[C]. The 2nd International Conference on Applied Cryptography and Network Security, Yellow Mountain, China, 2004: 31–45. doi: 10.1007/978-3-540-24852-1_3.

[21] CASH D, JARECKI S, JUTLA C, *et al*. Highly-scalable searchable symmetric encryption with support for Boolean queries[C]. The 33rd Annual Cryptology Conference on Advances in Cryptology, Santa Barbara, USA, 2013: 353–373. doi: 10.1007/978-3-642-40041-4_20.

[22] LAI Shangqi, PATRANABIS S, SAKZAD A, *et al*. Result pattern hiding searchable encryption for conjunctive queries[C]. 2018 ACM SIGSAC Conference on Computer and Communications Security, Toronto, Canada, 2018: 745–762. doi: 10.1145/3243734.3243753.

[23] SUN Shifeng, LIU J K, SAKZAD A, *et al*. An efficient non-interactive multi-client searchable encryption with support for boolean queries[C]. The 21th European Symposium on Computer Security, Heraklion, Greece, 2016: 154–172. doi: 10.1007/978-3-319-45744-4_8.

[24] WANG Yunling, WANG Jianfeng, SUN Shifeng, *et al*. Towards multi-user searchable encryption supporting Boolean query and fast decryption[J]. *The Journal of Universal Computer Science*, 2019, 25(3): 222–244.

[25] KAMARA S and MOATAZ T. Boolean searchable symmetric encryption with worst-case sub-linear complexity[C]. The 36th Annual International Conference on the Theory and Applications of Cryptographic Techniques on Advances in Cryptology, Paris, France, 2017: 94–124. doi: 10.1007/978-3-319-56617-7_4.

[26] LI Jin, WANG Qian, WANG Cong, *et al*. Fuzzy keyword search over encrypted data in cloud computing[C]. 2010 Proceedings IEEE INFOCOM, San Diego, USA, 2010: 441–445. doi: 10.1109/INFCOM.2010.5462196.

[27] KUZU M, ISLAM M S, and KANTARCIOGLU M. Efficient similarity search over encrypted data[C]. The 28th IEEE International Conference on Data Engineering, Washington, USA, 2012: 1156–1167. doi: 10.1109/ICDE.2012.23.

[28] WANG Cong, CAO Ning, REN Kui, *et al*. Enabling secure and efficient ranked keyword search over outsourced cloud

data[J]. *IEEE Transactions on Parallel and Distributed Systems*, 2012, 23(8): 1467–1479. doi: 10.1109/TPDS.2011.282.

[29] CAO Ning, WANG Cong, LI Ming, *et al*. Privacy-preserving multi-keyword ranked search over encrypted cloud[C]. 2011 Proceedings IEEE INFOCOM, Shanghai, China, 2011: 829–837. doi: 10.1109/INFCOM.2011.5935306.

[30] SUN Wenhai, WANG Bing, CAO Ning, *et al*. Verifiable privacy-preserving multi-keyword text search in the cloud supporting similarity-based ranking[J]. *IEEE Transactions on Parallel and Distributed Systems*, 2014, 25(11): 3025–3035. doi: 10.1109/TPDS.2013.282.

[31] FABER S, JARECKI S, KRAWCZYK H, *et al*. Rich queries on encrypted data: Beyond exact matches[C]. The 20th European Symposium on Research in Computer Security on Computer Security, Vienna, Austria, 2015: 123–145. doi: 10.1007/978-3-319-24177-7_7.

[32] POPA R A, REDFIELD C M S, ZELDOVICH N, *et al*. CryptDB: Protecting confidentiality with encrypted query processing[C]. The 23rd ACM Symposium on Operating Systems Principles, Cascais, Portugal, 2011: 85–100. doi: 10.1145/2043556.2043566.

[33] ZHANG Yupeng, KATZ J, and PAPAMANTHOU C. All your queries are belong to us: The power of file-injection attacks on searchable Encryption[C]. USENIX Security Symposium, Austin, USA, 2016: 707–720.

[34] STEFANOV E, PAPAMANTHOU C, and SHI E. Practical dynamic searchable encryption with small leakage[C]. Annual Network and Distributed System Security Symposium, NDSS, San Diego, USA, 2014.

[35] BOST R. $\sum o\phi o\varsigma$: Forward secure searchable encryption[C]. The 2016 ACM SIGSAC Conference on Computer and Communications Security, Vienna, Austria, 2016: 1143–1154. doi: 10.1145/2976749.2978303.

[36] SONG Xiangfu, DONG Changyu, YUAN Dandan, *et al*. Forward private searchable symmetric encryption with optimized I/O efficiency[J]. *IEEE Transactions on Dependable and Secure Computing*, 2020, 17(5): 912–927. doi: 10.1109/TDSC.2018.2822294.

[37] KIM K S, KIM M, LEE D, *et al*. Forward secure dynamic searchable symmetric encryption with efficient updates[C]. The 2017 ACM SIGSAC Conference on Computer and Communications Security, Dallas, USA, 2017: 1449–1463. doi: 10.1145/3133956.3133970.

[38] ZUO Cong, SUN Shifeng, LIU J K, *et al*. Dynamic searchable symmetric encryption schemes supporting range queries with forward (and backward) security[C]. 23rd European Symposium on Research in Computer Security on Computer Security, Barcelona, Spain, 2018: 228–246. doi: 10.1007/978-3-319-98989-1_12.

[39] WU Zhiqiang and LI Kenli. VBTree: Forward secure conjunctive queries over encrypted data for cloud computing[J]. *The VLDB Journal*, 2019, 28(1): 25–46. doi: 10.1007/s00778-018-0517-6.

[40] HU Chengyu, SONG Xiangfu, LIU Pengtao, *et al*. Forward secure conjunctive-keyword searchable encryption[J]. *IEEE Access*, 2019, 7: 35035–35048. doi: 10.1109/ACCESS.2019.2902855.

[41] WANG Yunling, WANG Jianfeng, SUN Shifeng, *et al*. Toward forward secure SSE supporting conjunctive keyword search[J]. *IEEE Access*, 2019, 7: 142762–142772. doi: 10.1109/ACCESS.2019.2944246.

[42] BOST R, MINAUD B, and OHRIMENKO O. Forward and backward private searchable encryption from constrained cryptographic primitives[C]. 2017 ACM SIGSAC Conference on Computer and Communications Security, Dallas, USA, 2017: 1465–1482. doi: 10.1145/3133956.3133980.

[43] GREEN M D and MIERS I. Forward secure asynchronous messaging from puncturable encryption[C]. 2015 IEEE Symposium on Security and Privacy, San Jose, USA, 2015: 305–320. doi: 10.1109/SP.2015.26.

[44] SUN Shifeng, YUAN Xingliang, LIU J K, *et al*. Practical backward-secure searchable encryption from symmetric puncturable encryption[C]. 2018 ACM SIGSAC Conference on Computer and Communications Security, Toronto, Canada, 2018: 763–780. doi: 10.1145/3243734.3243782.

[45] CHAMANI J G, PAPADOPOULOS D, PAPAMANTHOU C, *et al*. New constructions for forward and backward private symmetric searchable encryption[C]. 2018 ACM SIGSAC Conference on Computer and Communications Security, Toronto, Canada, 2018: 1038–1055. doi: 10.1145/3243734.3243833.

[46] WANG X S, NAYAK K, LIU Chang, *et al*. Oblivious data structures[C]. 2014 ACM SIGSAC Conference on Computer and Communications Security, Scottsdale, USA, 2014: 215–226. doi: 10.1145/2660267.2660314.

[47] STEFANOV E, VAN DIJK M, SHI E, *et al*. Path ORAM: An extremely simple oblivious RAM protocol[J]. *Journal of the ACM*, 2018, 65(4): 18. doi: 10.1145/3177872.

[48] DEVANBU P, GERTZ M, MARTEL C, *et al*. Authentic data publication over the internet[J]. *Journal of Computer Security*, 2003, 11(3): 291–314. doi: 10.3233/JCS-2003-11302.

[49] MYKLETUN E, NARASIMHA M, and TSUDIK G. Authentication and integrity in outsourced databases[C]. The Network and Distributed System Security Symposium, San Diego, USA, 2004.

[50] PANG H, JAIN A, RAMAMRITHAM K, *et al*. Verifying

completeness of relational query results in data publishing[C]. The 2005 ACM SIGMOD International Conference on Management of Data, Baltimore, USA, 2005: 407–418. doi: 10.1145/1066157.1066204.

[51] PANG H, ZHANG Jilian, and MOURATIDIS K. Scalable verification for outsourced dynamic databases[J]. *The VLDB Endowment*, 2019, 2(1): 802–813. doi: 10.14778/1687627.1687718.

[52] YUAN Jiawei and YU Shucheng. Flexible and publicly verifiable aggregation query for outsourced databases in cloud[C]. 2013 IEEE Conference on Communications and Network Security, National Harbor, USA, 2013: 520–524. doi: 10.1109/CNS.2013.6682770.

[53] AZRAOUI M, ELKHIYAOUI K, ÖNEN M, *et al*. Publicly verifiable conjunctive keyword search in outsourced databases[C]. 2015 IEEE Conference on Communications and Network Security, Florence, Italy, 2015: 619–627. doi: 10.1109/CNS.2015.7346876.

[54] SUN Wenhai, LIU Xuefeng, LOU Wenjing, *et al*. Catch you if you lie to me: Efficient verifiable conjunctive keyword search over large dynamic encrypted cloud data[C]. 2015 IEEE Conference on Computer Communications, Hong Kong, China, 2015: 2110–2118. doi: 10.1109/INFOCOM.2015.7218596.

王赟玲：女，1990年生，博士生，研究方向为密文检索和数据安全.

陈晓峰：男，1976年生，教授，研究方向为公钥密码学、云计算安全和数据安全.

责任编辑：余　蓉