

可验证对称可搜索加密技术研究综述

宿雅萍, 向宴颀, 蔡联芳

(中国民用航空飞行学院 计算机学院, 四川 广汉 618307)

摘要: 云计算通过其弹性可扩展的资源池、按需付费的服务模式以及近乎无限的存储能力, 为现代企业和个人用户提供了高效的数据外包解决方案。为了保护云端外包数据的机密性和完整性, 可验证对称可搜索加密技术已成为当前云计算安全领域的研究热点。针对关键词检索完整性验证问题, 首先, 阐述了现有方案主要采用的系统模型、安全模型和形式化定义; 其次, 围绕单用户和多用户2种应用场景, 基于不同技术路线介绍现有代表性研究成果; 再次, 分析方案所采用关键技术的优缺点; 最后, 展望未来的研究方向。研究结果表明, 构建兼顾安全性、动态适应性与高效的验证对称可搜索加密技术, 仍是当前验证对称可搜索加密技术的研究重点。

关键词: 云计算; 完整性验证; 可验证搜索

DOI: 10.11907/rjdk.251248

中图分类号: TP391

文献标识码: A

开放科学(资源服务)标识码(OSID):

文章编号: 1672-7800(2025)007-0022-07



Research Overview on Verifiable Symmetric Searchable Encryption Technology

SU Yaping, XIANG Yanjie, CAI Lianfang

(School of Computer Science, Civil Aviation Flight University of China, Guanghan 618307, China)

Abstract: Cloud computing provides efficient data outsourcing solutions for modern enterprises and individual users through its elastic and scalable resource pool, pay as you go service model, and almost unlimited storage capacity. In order to protect the confidentiality and integrity of cloud outsourced data, verifiable symmetric searchable encryption technology has become a research hotspot in the current field of cloud computing security. Regarding the issue of keyword search integrity verification, the existing solutions mainly adopt system models, security models, and formal definitions; Secondly, introduce existing representative research results based on different technological routes around two application scenarios: single user and multi-user; Then, analyze the advantages and disadvantages of the key technologies used in the proposed solution; Finally, looking forward to future research directions. The research results indicate that the construction of verifiable symmetric searchable encryption technology that balances security, dynamic adaptability, and efficiency remains the current research focus of verifiable symmetric searchable encryption technology.

Key Words: cloud computing; integrity verification; verifiable search

0 引言

随着云计算技术的发展, 大数据越来越受到关注。由于计算机本地设备存储空间和计算性能的限制, 用户将大量数据外包给云服务器, 是目前最受欢迎的数据存储解决方案之一^[1-5]。然而, 由于数据的所有权与使用权分离, 服务器发生故障或遭受恶意攻击, 云端外包数据的隐私与安全仍然面临着巨大的挑战^[6-9]。外包数据中往往包含重要

的敏感信息, 如电子邮件、医疗数据、政务数据等, 这些敏感信息在不可控存储环境中存在被泄露的安全风险。为了保证外包数据的机密性, 传统加密技术是一种最直接有效的手段。但是, 这使得云服务器无法向数据用户提供数据检索服务。因此, 如何实现高效的密文检索是一个极具挑战性的问题。

可搜索加密技术(Searchable Encryption, SE)应运而生, 它允许云服务器在密文中高效地执行检索操作。根据方案构造时所采用的不同密码原语, 可搜索加密技术分为

收稿日期: 2025-04-08

扫描二维码阅读全文:



基金项目: 中央高校基本科研业务费专项资金项目(24CAFUC03044、QJ2023-025)

作者简介: 宿雅萍(1993-), 女, 硕士, 中国民用航空飞行学院计算机学院讲师, 研究方向为数据安全和密文检索; 向宴颀(1996-), 女, 硕士, 中国民用航空飞行学院计算机学院助教, 研究方向为区块链安全; 蔡联芳(2002-), 女, 中国民用航空飞行学院计算机学院学生, 研究方向为网络空间安全。本文通讯作者: 宿雅萍。

对称可搜索加密技术(Symmetric Searchable Encryption, SSE)和公钥可搜索加密技术(Public key Encryption with Keyword Search, PEKS)。其中,SSE方案的构造基于对称密码原语进行设计,PEKS方案的构造基于公钥密码原语进行设计。相比SSE方案,PEKS方案的效率低下。因此,SSE技术逐渐成为学术界关注和研究的热点。

SSE方案可以为云服务器提供高效的密文检索能力,但其无法确保云服务器检索结果的正确性与完整性^[10-13]。如在对数据准确性要求非常高的医疗机构、金融行业,错误的数据或不完整的数据可能会对医疗诊断、金融行情分析造成严重的后果。基于此,如何实现对云端密文数据检索结果完整性的有效验证是设计SSE方案的另一个关注点。因此,可验证对称可搜索加密技术(Verifiable Symmetric Searchable Encryption, VSSE)近年来越来越受到学者们的重视,大量的VSSE方案相继被提出,确保了数据的可验证性和正确性。一方面,可验证性是指能够验证搜索结果的正确性和完整性;另一方面,正确性是指搜索结果中的密文与搜索关键字匹配,而完整性是指这些搜索结果包含与搜索关键字匹配的所有密文。VSSE方案的研究重点是如何设计和构造适用于特定场景的密文索引结构、密文检索算法和支持对检索结果完整性进行验证的验证算法。目前,VSSE方案的适用场景主要包括单用户场景和多用户场景。

本文首先系统梳理了可验证对称可搜索加密方案主要采用的系统模型、安全模型和形式化定义;其次,围绕单用户场景与多用户场景的双重视角,对国内外代表性工作展开分类综述,分析现有方案所采用的核心技术原理、技术优缺点及性能表现;最后,通过综合分析和对比现有研究成果的研究重点及其所使用的关键技术,对现有工作进行总结。基于现有方案的技术难点,本文对未来可能的研究方向进行展望。研究结果表明,构建兼顾安全性、动态适应性与高效的可验证对称可搜索加密技术,仍是当前可验证对称可搜索加密技术研究中亟待解决的问题。

1 模型与形式化定义

1.1 系统模型

(1)单用户场景。单用户场景下的VSSE系统模型如图1所示^[14]。



Fig. 1 System model of VSSE scheme in single user scenario

图1 单用户场景下VSSE方案系统模型

由图1可知,该系统模型包括2个不同的实体:数据拥有者(Data Owner, DO)和云服务器(Cloud Server, CS)。
①DO负责将需要外包的文档加密,并生成密文的检索索引和用于验证检索结果完整性的证据信息,将密文文档、密文索引和证据信息一起外包给CS。此外,当DO想要检索某个特定的关键词时,首先,生成该关键词的检索陷门;然后,提交给CS用于密文关键词检索;最后,当CS返回检索结果和用于验证结果完整性的证明时,DO利用该证明验证检索结果的完整性和正确性。若验证通过,则对检索结果进行解密;否则,拒绝接收。
②CS负责存储密文文档、密文检索索引和证据信息。根据DO所提交的检索陷门执行密文检索,并将检索结果以及对应的证明发送给DO。

(2)多用户场景。多用户场景下的VSSE系统模型如图2所示^[15]。

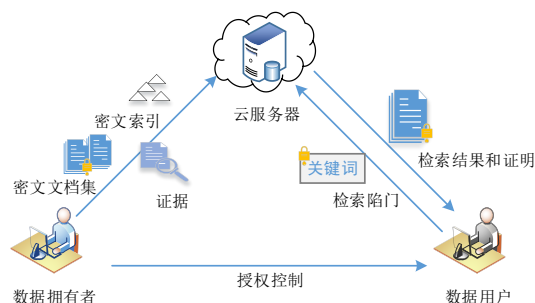


Fig. 2 System model of VSSE scheme in multi user scenario

图2 多用户场景下VSSE方案系统模型

由图2可知,该系统模型包括3个不同的实体:DO、CS和数据使用者(Data User, DU)。
①DO主要负责生成密文文档、密文检索索引和证据信息,并全部外包给CS。此外,DO还需要对DU进行授权,只有授权的DU才允许在CS上执行关键词检索。
②CS主要负责存储密文文档、检索索引和证据信息。当收到DU的检索陷门时,CS根据陷门信息在密文上执行检索,并将检索结果以及对应证明返回DU。
③DU主要负责在想要检索关键词时,生成该关键词的检索陷门并向CS发送检索请求。此外,当DU收到检索结果和证明时,DU需对检索结果的完整性和正确性进行验证。若验证通过,则对密文文档解密;否则,拒绝接收。

1.2 安全模型

在VSSE方案中,DO和DU一般都视为诚实的。CS可分为2种模型:半可信服务器模型和恶意服务器模型^[14-15]。

(1)半可信服务器模型。在半可信服务器模型中,云服务器会诚实地严格执行协议,但可能通过统计分析、窥视等方法获取外包数据的隐私信息。

(2)恶意服务器模型。在恶意服务器模型中,云服务器会诚实地执行部分协议,其可能为了自身利益(如节省计算资源或通信代价)而向用户返回不完整或不正确的检索结果。

1.3 形式化定义

(1) 单用户场景。假设明文数据库为 $DB = (id_i, W_i)_{i=1}^n$ 。其中, id_i 为第 i 个文档的标识符, W_i 为文档 id_i 包含的关键词集合, n 为数据库中文档的总数。单用户场景下的一个 VSSE 方案主要由 4 个算法组成^[14], 即 $\Pi = (\text{Setup}, \text{Trapdoor}, \text{Search}, \text{Verify})$, 具体定义如下: ① $\text{Setup}(1^\lambda, DB)$: 初始化系统。该算法由 DO 执行。DO 输入安全参数 λ 和明文数据库 DB , 该算法输出密钥 k 、密文数据库 EDB 、密文检索索引 I 和证据信息 $proof$ 。② $\text{Trapdoor}(k, w)$: 陷门生成算法。该算法由 DO 执行。算法输入密钥 k 和待查询的关键词 w , 输出关键词所对应的陷门 t_w 。③ $\text{Search}(t_w, I, EDB)$: 检索算法。该算法由 CS 执行。该算法输入检索陷门 t_w 、密文检索索引 I 和密文数据库 EDB , 输出文档标识 DB_w 和证据信息 $proof$ 。具体而言, CS 根据 DO 提供的 t_w 和存储在服务器上的 I 在 EDB 上执行密文检索, 并将包含关键词 w 的所有文档标识 DB_w 、证据信息 $proof$ 返回给 DO。④ $\text{Verify}(DB_w, proof)$: 验证算法。该算法由 DO 执行。算法输入由 CS 返回的文档集合 DB_w 和证据信息 $proof$, 若 DO 验证通过, 算法输出 1, 否则输出 0。

(2) 多用户场景。多用户场景下的一个 VSSE 方案主要由 4 个算法组成, 即 $\Pi = (\text{Setup}, \text{Trapdoor}, \text{Search}, \text{Verify})$ ^[15]。与单用户场景不同的是, 只有授权的 DU 才具有对外包数据的访问权限, 且部分算法的执行方已由 DO 变为 DU。具体定义如下: ① $\text{Setup}(1^\lambda, DB)$: 初始化系统。该算法由 DO 执行。DO 输入安全参数 λ 和明文数据库 DB , 该算法输出密钥 k 、辅助信息 σ 、密文数据库 EDB 、密文检索索引 I 和证据信息 $proof$ 。② $\text{Trapdoor}(k, w)$: 陷门生成算法。该算法由 DU 执行。算法输入密钥 k 和待查询的关键词 w , 输出关键词所对应的陷门 t_w 。③ $\text{Search}(t_w, I, EDB)$: 检索算法。该算法由 CS 执行。算法输入检索陷门 t_w 、密文检索索引 I 和密文数据库 EDB , 输出文档标识 DB_w 和证据信息 $proof$ 。具体而言, CS 根据 DU 提供的检索陷门 t_w 和存储在服务器上的密文检索索引 I , 在 EDB 上执行密文检索, 并将包含关键词 w 的所有文档标识 DB_w 、证据信息 $proof$ 返回给 DU。④ $\text{Verify}(DB_w, proof)$: 验证算法。该算法由 DU 执行。算法输入由 CS 返回的文档集合 DB_w 和证据信息 $proof$, 若 DU 验证通过, 算法输出 1, 否则输出 0。

综上, 无论是在单用户场景还是多用户场景下, VSSE 方案普遍采用以上 4 个算法。还有部分 VSSE 方案支持文件动态更新、用户动态更新或其他个性化搜索等特定功能, 即存在实现对应功能的算法。

2 单用户场景下的 VSSE 方案

在可验证对称可搜索加密研究中, 针对单用户场景下的 VSSE 方案最早被提出。这类方案主要关注单用户场景下的关键词检索, 并着重构建能够实现搜索结果完整性验

证的索引结构和验证策略。这些核心要素直接决定了用户验证所需的时间开销、空间开销以及安全性。在现有研究中, 常用的验证技术包括消息认证码、聚合器和多重集合哈希函数等。

2.1 基于消息认证码的 VSSE 方案

Kurosawa 等^[16]在恶意服务器模型下基于消息认证码 (Message Authentication Code, MAC) 提出第一个能够抵抗主动攻击敌手的 VSSE 方案。该方案的主要思想可以概括如下: ① 在密文索引生成阶段, DO 针对每个关键词, 通过一个伪随机函数对该关键词所属明文文档的标识符进行计算, 生成一个地址值; 随后, DO 利用消息认证码 (MAC) 对该关键词的地址值及其所属密文文档的标签进行计算。② 在验证阶段, DO 会利用 MAC 对本地存储的陷门信息以及 CS 返回的密文文档重新计算出一个新的标签, 并将其与 CS 返回的标签进行对比。然而, 研究表明, 通过该方案返回的结果可能存在不完整或不正确的情况。该方案虽然增强了安全模型, 能够抵抗主动敌手的攻击, 但仍然存在 2 个方面的不足: 一是无法验证空集。当 CS 返回的密文为空集时, DU 无法在本地重新计算出新的标签, 即无法跟 CS 返回的标签进行比较。二是搜索阶段的通信开销与文件数量呈线性增长关系。

Wang 等^[17]利用 MAC、聚合 MAC 和默克哈希技术构造了一个 VSSE 技术。该方法首先利用 MAC 分别为密文关键词、包含该关键词的文档加密标识符及该标识符对应的密文文档的关联值生成认证码; 其次, 利用聚合 MAC 的异或运算, 将多个认证码聚合为一个认证标签; 最后, 将所有关键词对应的认证标签存储在默克哈希树中, DU 在本地保留默克树根的哈希值。在验证阶段, DU 根据返回的搜索结果生成认证标签, 利用该认证标签重构默克树根的哈希值, 再与本地存储的根哈希值作比较, 即可验证搜索结果的完整性与正确性。研究表明, MAC、聚合 MAC 这 2 项技术的计算效率高, 但默克哈希树的存储空间随着数据集的增大而急剧增大, 故在大型文档数据库中实用性不强。

Yoneyama 等^[18]基于伪随机函数的 MAC 机制构造了一个满足前向安全的动态 VSSE 方案。该方案中, 用户在设置阶段为 PRF 生成一个密钥, 同时, 在更新阶段计算并发送一个标签, 将文档索引和关键词与 PRF 绑定至 CS。用户可以在更新阶段后擦除标签。在验证阶段, 客户端可通过验证标签检查搜索结果的有效性。由于 PRF 的密钥只有用户拥有, 因此, CS 很难根据 PRF 的属性伪造标签。然而, 在这种方法中, 用户的通信成本和计算成本都很高。

Wan 等^[19]融合随机挑战和同态 MAC 机制设计了一个支持多关键词搜索的 VSSE 技术。该方法将文档索引分为 2 个向量, 借助矩阵加密确保索引的机密性, 同时为索引、陷门中的每个元素计算一个添加了随机数和假关键词的认证标签。验证的证据信息是索引的认证标签与陷门的认证标签的乘积, 用户将 CS 返回的搜索结果中的文档相

关度得分与证据信息中的相关度得分作比较,再重构搜索函数,即可实现对返回结果完整性的验证。此外,该方案采用随机挑战技术对搜索结果根据相关度得分进行排序,提高解密效率。类似的,Li等^[20]采用同态MAC、随机挑战和局部敏感哈希函数技术设计了一个可排序的模糊关键词搜索方案。但是这2个方案在CS存在惰性计算的情况下,仍然存在验证失败的可能性。Ramasamy等^[21]利用同态MAC和位图技术构造了一个动态的VSSE方案。该方案在索引建立阶段为关键词以及包含该关键词的文档生成标签,其生成过程涉及模幂运算,导致索引生成效率降低。同时,在用户利用验证标签验证搜索结果完整性时,验证时间随着搜索结果规模的增加而呈线性增长。因此,以上2个方案在大型文档库中不具有实用性。

2.2 基于聚合器的VSSE方案

Sun等^[22]基于聚合器和默克哈希树构造了一个连接关键词查询方案(Oblivious Cross-Tags, OXT)。该方案支持连接关键词的搜索,并允许对检索结果的完整性和正确性进行公开验证。然而,该验证算法存在2个局限性:①无法对空集结果进行验证;②在搜索阶段,CS需要先检索每个关键词的独立结果,再对所有结果求交集,这种操作方式导致搜索效率较低。

为了解决以上2个问题,Wang等^[23]在恶意服务器模型下基于双线性对的聚合器和采样检测技术,采用OXT的连接关键词搜索方法,提出了公开可验证的连接关键词查询的VSSE方案。该方案的验证思路如下:首先,设计了3种不同的聚合器。第1种是对所有关键词陷门的聚合,第2种是为每个关键词所属的所有文档生成聚合值,第3种是为每个文档-关键词对的哈希值生成聚合值。其次,假设DO当前检索包含3个连接关键词的文档,在得到CS返回的搜索结果后,DO将利用这3种聚合器对以下3种搜索结果进行验证:①当频率最小关键词的搜索结果为空且CS返回空集时,DO利用第1种聚合器进行验证;②当频率最小关键词的搜索结果不为空,但CS返回的搜索结果为空时,DO先利用第2种聚合器验证频率最小关键词搜索结果的完整性和正确性,再利用第3种聚合器提供证据,证明频率最小关键词搜索结果中的文档不在CS返回的搜索结果中;③当频率最小关键词的搜索结果和CS返回的搜索结果都不为空集时,DO则先利用第2种聚合器验证频率最小关键词搜索结果的正确性与完整性,然后利用第3种聚合器验证CS返回的搜索结果中所有文档的完整性与正确性。然而,由于该聚合器的构造依赖于双线性对运算,使得验证效率较低。

针对此问题,GAN等^[24]设计了一种基于哈希的聚合器,并将其应用于OXT协议,实现了一个支持连接关键词搜索的VSSE方案。相比文献[23],该方案中哈希聚合器的构造仅涉及哈希运算,使得验证效率有了显著的提升。Zhou等^[25]基于RSA聚合器构造了一个VSSE方案。其核心

技术基于RSA累加器构造帕特里夏树(Patricia Tree)索引,即RAP树索引。该索引能够实现密文搜索与结果完整性验证。在该方案中,每个关键词和包含该关键词的密文文档通过一个素数实现关联,再利用模幂运算将该素数集成到RSA累加器中,从而生成对应的局部证据。在完成所有关键词的证明计算后,构建RAP树索引。在验证阶段,由DU重构RSA累加器,以此验证搜索结果的完整性。该技术的优点是将RSA累加器应用于压缩前缀树结构,可以同时实现高效搜索与结果验证。然而,RSA聚合器机制需要进行大量的大数模幂运算,初始化阶段耗时较长。此外,该方案只针对静态数据集处理,在数据动态更新的场景下不适用。

2.3 基于多重集合哈希函数的VSSE方案

Bost等^[26]基于可验证哈希表和多重集合哈希函数(MultiSet Hash Function)技术构造了一个动态VSSE方案。Zhang等^[27]基于多重集合哈希函数(Multiset Hash Functions)技术提出一个高效的动态VSSE方案。由于多重集合哈希函数具有增量更新的特性,在产生新的证据时,DO无需下载CS上的数据就可以高效地完成证据的更新。该方案采用多重集合哈希函数的一种最高效的构造,即仅包括异或操作的多重集-异或-哈希运算。当给定集合的哈希值已生成时,若向该集合添加新元素形成更新后的集合,此时无需基于整个新集合重新计算哈希值,而只需将原集合哈希值与新元素哈希值进行异或运算即可获得更新后的哈希值。此VSSE方案解决了从CS上不断下载数据的问题,降低了DO与CS之间的通信开销。

3 多用户场景下的VSSE方案

在多用户场景下,如企业协作、医疗机构共享数据库和多租户云服务,可验证可搜索加密方案的研究需要解决共享数据的安全性、细粒度访问控制以及对搜索结果完整性的验证。现有方案中,主要使用的验证技术包括前缀树、布隆过滤器和默克哈希树等。

3.1 基于前缀树的VSSE方案

文献[14]首次在半可信服务器模型下提出基于前缀树结构的VSSE方案。在索引生成阶段,DO利用关键词构建字典树作为密文索引。其中,树中的每个节点具有3个属性:①明文关键词的一个字符;②使用带密钥的哈希函数生成该节点的前缀签名;③使用位图技术计算出的该节点的位图数据。

在验证阶段,DO通过利用每个节点的位图数据,重构了所查询的关键词及其包含的文档信息,用于验证搜索结果的完整性和正确性。该方案首次在单用户场景下实现了对搜索结果完整性与正确性的验证,但其技术实现存在局限性:①索引构建效率低下。密文检索索引的生成时间复杂度与关键词集合规模呈线性正相关,当处理大规模数据

集时,索引构建耗时随着数据量增长显著增加,难以满足实时性需求。其根本原因在于字典树结构的逐节点插入机制,缺乏批量操作优化。②存储空间利用率不足。索引结构采用传统字典树,虽然支持前缀搜索,但由于共享前缀的关键词需重复存储相同路径节点,导致节点存储冗余问题突出。Kissel等^[28]提出了一种支持对用户组进行分层访问控制,同时支持组成员删除和添加的VSSE方案。其索引结构通过强制执行分层访问策略来保护特征树的根到叶子路径,通过搜索路径中每个节点的位图数据可以重构关键词与文档集信息,从而实现对搜索结果完整性的验证。但是,该方案仅实现了对静态数据集的搜索与验证。Wang等^[29]采用基于符号集的前缀树实现了对搜索结果完整性的验证,创新点在于构建多向树索引结构,并在路径节点中嵌入可验证的关键词陷门信息,同时,利用编辑距离实现了对模糊关键字的搜索。然而,文献[29]中的方法存在两方面局限:一是仅适用于静态数据集的搜索与验证;二是在处理大规模数据集时,前缀树会占用过多存储空间。

3.2 基于布隆过滤器VSSE方案

Wang等^[30]基于布隆过滤器(Bloom Filter, BF)和默克哈希树(Merkle Hash Tree, MHT)技术提出了一种多用户场景下的VSSE方案,其具备空集验证能力,即使恶意的CS故意返回空集,也能验证搜索结果的完整性。该方案首先为数据集中的所有元素生成一个BF、一个BFT以及为每一个元组生成一个MHT。MHT的叶子节点是该元组中的属性值和元组序号的哈希值。DU验证时,首先重构MHT生成根签名,再和DO生成的该元组的MHT根签名作比较,根据比较结果即可确认结果的正确性。针对结果完整性的验证,DU只需要统计CS返回的元组数,并与MHT根中包含的元组数作比较。由于BF不支持数据的动态删除与添加,故该方案只适用于处理静态文档库。

Su等^[31]基于改进的混淆布隆过滤器(Gabled Bloom Filter, GBF)技术构造了一个多用户场景下的VSSE方案。在该方案中,DO为每位授权的DU生成一个可验证的数据结构GBF,该结构存储与其共享的所有文档的标识符。具体而言,DO为每个文档分配一个随机值作为其标识符,利用GBF的异或运算特性,通过对所有文档标识符进行异或运算,生成一个针对包含特定关键字的所有文档的聚合值。借助GBF的自校验特性,授权的DU可以根据返回的结果重构GBF,从而实现对搜索结果完整性和正确性的验证。由于GBF的生成只涉及异或运算和哈希运算,故证据的生成效率与验证效率高。然而,该方案中GBF的生成过程仅考虑了静态的数据集,无法满足动态数据的可验证性需求。当需要删除或添加文档时,GBF只需通过异或该文档的标识符即可轻松实现动态更新,这使得GBF能够很好地扩展到动态数据集的场景中。

Wang等^[32]在用户动态共享文档的场景下利用由文献[31]改进的GBF技术构造了一个动态VSSE方案。在

GBF机制中,搜索关键词的存储位置通过双重加密映射生成,既包含了关键词本身的哈希特征,又深度融合了用户专属密钥的加密参数。这种动态位置生成算法确保了位置信息的不可预测性,即使服务器具备存储访问权限,也无法逆向推导出特定查询对应的具体存储坐标。由于篡改操作必须精准定位到原始数据位置,才能伪造有效验证结果,而GBF的位置遮蔽机制有效阻断了这种可能性,进而从底层架构上保障了检索结果的原始真实性。

Shi等^[33]基于计数布隆过滤器(Counting Bloom Filter, GBF)和B+树构造了一个支持多用户查询和动态更新的VSSE方案。CBF是基于BF改进的,能够支持数据的动态删除和添加,适用于需要频繁更新数据的场景。将标准BF的二进制位数组升级为计数器数组,每个位置记录被不同元素标记的次数。当删除或添加元素时,通过递减或递增这些计数器的值,从而支持删除或添加操作。相比BF,CBF占用更多的空间。在该方案中,DO首先基于B+树为所有的关键词-密文对生成密文索引;其次,将每一个关键词的查询标签映射到CBF中从而生成CBF;最后,同时生成认证器 π 和CBF的认证器 π_{bf} 。其中, π_{bf} 包含DO的签名。当用户查询的关键字不存在时, π_{bf} 可以提供证明。用户通过CBF检查搜索陷门是否存在。当CS搜索到的关键词存在时,DU需要根据CS返回的结果构建B+树的根,并通过将其与 π 的根进行比较来验证返回结果的完整性。当CS返回空集时,用户只需要利用 π_{bf} ,就可以验证所查询的关键词是否真的不存在,以此验证CS的行为是否为恶意。由于哈希存在冲突,BF、GBF和CBF都属于概率性验证数据结构。因此,由这3种技术所设计的VSSE方案,在验证结果方面存在一定的误判率。

3.3 基于默克哈希树的VSSE方案

MHT技术在验证外包数据完整性方面得到了大量的应用^[34]。通过根哈希唯一标识数据集,任何数据篡改都会导致根哈希变化,实现高效完整性检查。如文献[30]中DU只需重构MHT的根即可完成对搜索结果完整性的高效验证,但MHT的空间开销大。基于此,Wood等^[35]结合帕特里克夏树(Patricia Tree)和MHT构造了默克·帕特里克夏树(Merkle Patricia Tree, MPT)。与MHT相比,MPT技术的特点在于:通过压缩共享前缀的节点来减少存储空间,同时采用十六进制编码来缩小树的宽度,从而提升搜索效率。

Zhu等^[36]提出了一种基于MPT和增量哈希函数的动态VSSE方案。该方案的索引不仅支持数据更新,还能实现数据完整性验证,这主要得益于MHT的双重功能:既能作为搜索索引,又能提供数据完整性证明。MPT中存储的键是所有不同关键词的陷门,值是包含某一关键词的所有文档的增量哈希。其中,增量哈希是一种通过仅对新数据或变更部分进行计算,进而高效生成整体哈希值的技术。因此,当有新数据被删除或添加时,不是重复计算已有数据的哈希值,而是基于历史哈希值推导出新的结果。显

然,通过采用增量哈希技术可以有效提高数据更新的效率;MHT 技术则通过减少树的深度来降低存储空间的需求,然而,其存储开销仍与数据集的大小保持线性增长的关系。类似的,Li 等^[37]也基于 MPT 和增量哈希技术设计了一个动态 VSSE 方案。该方案基于权限切换和广播加密技术的思想,实现了针对合谋攻击的多用户搜索访问控制。此外,该方案中的 DO 不需要实时在线,更符合实际应用的需求。闫润雨等^[38]利用带 Rank 值的 MHT(Rank Value MHT, RMHT)和 CBF 技术构造了一个支持用户对搜索结果细粒度验证的动态 VSSE 方案。该方案不仅能够验证搜索结果的完整性和正确性,还可以细粒度筛检测出正确结果并检测出未返回的文件个数。与传统的 MHT 不同, RMHT 除了存储文件的哈希值外,还额外存储了对应文件细粒度的 Rank 值,即节点包含的子树中所有叶子节点的个数。针对每一个关键词,DO 将包含该关键词的密文文档的伪随机函数值插入到 CBF 中,CBF 作为验证集使用。在验证阶段,DO 首先根据 CS 返回的结果集重构 RMHT,并对比根节点哈希值,随后,将密文文件映射到 CBF 中并检查对应的计数器。若该密文文件对应的计数器值不全大于 0,则说明该文件是错误结果,将其从结果集中删除。这样就能获得正确的文件结果。随后,对相应位的计数器减 1,并将所有计数器累加后除以所用的哈希个数,即可得出未返回的文件数量。然而,该方案的验证时间与文件数量呈线性增长关系,且 RMHT 的空间消耗大,在海量数据共享场景中的实用性不高。

4 结语

本文重点围绕云计算安全领域可验证对称可搜索加密技术(VSSE)展开综述,分别介绍了在单用户和多用户场景下,VSSE 方案所采用的核心验证技术及其性能表现,并通过表 1 对部分经典的 VSSE 方案进行了对比。通过对

现有典型 VSSE 方案的综合分析和对比,本文发现该技术仍存在一些不足之处。未来的研究应当围绕以下几个方面进行深入探索和研究。

(1)支持海量数据外包场景中的 VSSE 方案。现有方案中,大多数 VSSE 方案未能充分考虑海量数据的应用场景,其主要原因在于现有的 VSSE 方案多数主要通过重构证据信息的方式来实现搜索结果的完整性验证。然而,重构证据信息通常会随着数据集规模的增大导致时间开销和空间开销的增加,从而降低了可验证搜索过程的执行效率。因此,在单用户和多用户场景中,研究能够在海量数据外包场景中兼顾时间和空间效率的 VSSE 方案具有重要研究价值和意义。

(2)研究基于新技术构建兼顾安全性、动态适应性和扩展性的高效 VSSE 方案。首先,在现有方案中,大多数 VSSE 方案不能够很好地平衡安全性、动态适应性、扩展性及执行效率。例如,在实际应用场景中,如医疗机构、金融机构、企业以及国家相关安全部门等,结合等级保护要求,各个机构对安全等级的需求各不相同。因此,研究实现不同安全等级且能够覆盖所有机构对安全性最低要求的 VSSE 方案是有必要的。其次,现有部分方案仅能针对静态数据集进行处理,难以延伸到需要实时更新数据的动态场景中。其主要原因在于现有方案中,用户生成的证据信息是静态的,当数据有更新时,需要使用更新后的全部数据重新生成证据信息,显然这对存储开销和时间开销并不是理想的选择。因此,研究具有良好扩展性、支持数据实时更新的 VSSE 方案具有重要意义。最后,现有的 VSSE 方案中,大多数方案的索引生成时间、搜索时间、验证时间与数据规模正相关。因此,探索新型的索引构建方法与验证技术仍是一个研究热点。

综上,研究基于新技术构建兼顾安全性、动态适应性和扩展性的高效 VSSE 方案,是目前验证对称可搜索加密领域的一个重要研究方向。

Table 1 Comparison of typical VSSE schemes

表 1 典型 VSSE 方案对比

场景	方案	安全模型	关键技术	检索类型	动态更新	完整性验证	正确性验证	空集验证
单用户	Chai 等人方案	半可信	字典树	单关键词	×	√	√	×
	Kurosawa 等人方案	恶意	消息认证码	单关键词	×	√	√	×
	Bost 等人方案	恶意	可验证哈希表	单关键词	√	√	√	√
	Wang 等人方案	恶意	聚合器	连接关键词	×	√	√	√
	Zhang 等人方案	恶意	多重集哈希	单关键词	√	√	√	√
多用户	Wang 等人方案	恶意	布隆过滤器	单关键词	×	√	√	√
			默克哈希树					
	Su 等人方案	恶意	混淆布隆过滤器	单关键词	×	√	√	√
	Shi 等人方案	恶意	计数布隆过滤器	单关键词	√	√	√	√
			B+树					

参考文献:

[1] KOLODNER E K, TAL S, KYRIAZIS D, et al. A cloud environment for data-intensive storage services[C]//2011 IEEE Third International Conference on Cloud Computing Technology and Science, 2011: 357-366.

[2] ALBA A, ALATORRE G, BOLIK C, et al. Efficient and agile storage management in software defined environments [J]. IBM Journal of Research and Development, 2014, 58(2/3):1-12.

[3] WU J Y, PING L D, GE X P, et al. Cloud storage as the infrastructure of cloud computing [C]//International Conference on Intelligent Computing

- and Cognitive Informatics, 2010: 380–383.
- [4] CHANG V, WILLS G. A model to compare cloud and non-cloud storage of Big Data[J]. *Future Generation Computer Systems*, 2016, 57: 56–76.
 - [5] ZHANG C, XU Y, HU Y, et al. A blockchain-based multi-cloud storage data auditing scheme to locate faults [J]. *IEEE Transactions on Cloud Computing*, 2021, 10(4): 2252–2263.
 - [6] YANG P, XIONG N X, REN J L. Data security and privacy protection for cloud storage: a survey[J]. *IEEE Access*, 2020, 8: 131723–131740.
 - [7] NAIDU P R, GURUPRASAD N, GOWDA V D. A high-availability and integrity layer for cloud storage, cloud computing security: from single to multi-clouds [C]//First International Conference on Advances in Smart Sensor, Signal Processing and Communication Technology, 2021: 19–20.
 - [8] PRAJAPATI P, SHAH P. A review on secure data deduplication: cloud storage security issue[J]. *Journal of King Saud University-Computer and Information Sciences*, 2022, 34(7): 3996–4007.
 - [9] SUN J, XU G, ZHANG T, et al. Verifiable, fair and privacy-preserving broadcast authorization for flexible data sharing in clouds[J]. *IEEE Transactions on Information Forensics and Security*, 2022, 18: 683–698.
 - [10] SONG D X, WAGNER D, PERRIG A. Practical techniques for searches on encrypted data [C]//2000 IEEE Symposium on Security and Privacy, 2000: 44–55.
 - [11] CASH D, JARECKI S, JUTLA C, et al. Highly-scalable searchable symmetric encryption with support for boolean queries [C]//2013 International Association for Cryptologic Research, 2013: 353–373.
 - [12] CAO S Z, LANG X L, LIU X Z, et al. Delegate searchable encryption scheme resisting keyword guess [J]. *Journal of Electronics & Information Technology*, 2019, 41(9): 2180–2186.
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.
 - [13] LI L, TAN Z W, ZHU J W, et al. Dynamic searchable encryption scheme supporting fuzzy multiple keywords [J]. *Journal of Information Security Research*, 2024, 10(11): 1064–1073.
李莉, 谭智伟, 朱江文, 等. 支持模糊多关键字的动态可搜索加密方案 [J]. *信息安全研究*, 2024, 10(11): 1064–1073.
 - [14] CHAI Q, GONG G. Verifiable symmetric searchable encryption for semi-honest-but-curious cloud servers [C]//2012 IEEE International Conference on Communications, 2012: 917–922.
 - [15] LIU X, YANG G, MU Y, et al. Multi-user verifiable searchable symmetric encryption for cloud storage [J]. *IEEE Transactions on Dependable and Secure Computing*, 2018, 17(6): 1322–1332.
 - [16] KUROSAWA K, OHTAKI Y. UC-secure searchable symmetric encryption [C]//16th International Conference on Financial Cryptography and Data Security, 2012: 285–298.
 - [17] WANG B, FAN X. Lightweight verification for searchable encryption [C]//17th IEEE International Conference on Trust, Security And Privacy in Computing and Communications, 2018: 932–937.
 - [18] YONEYAMA K, KIMURA S. Verifiable and forward secure dynamic searchable symmetric encryption with storage efficiency [C]//19th International Conference on Information and Communications Security, 2018: 489–501.
 - [19] WAN Z, DENG R H. VPSearch: achieving verifiability for privacy-preserving multi-keyword search over encrypted cloud data [J]. *IEEE Transactions on Dependable and Secure Computing*, 2016, 15(6): 1083–1095.
 - [20] LI X, TONG Q, ZHAO J, et al. VRFMS: verifiable ranked fuzzy multi-keyword search over encrypted data [J]. *IEEE Transactions on Services Computing*, 2022, 16(1): 698–710.
 - [21] RAMASAMY R, VIVEK S S, GEORGE P, et al. Dynamic verifiable encrypted keyword search using bitmap index and homomorphic MAC [C]//2017 IEEE 4th International Conference on Cyber Security and Cloud Computing, 2017: 357–362.
 - [22] 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]//2015 IEEE Conference on Computer Communications, 2015: 2110–2118.
 - [23] WANG J F, CHEN X F, SUN S F, et al. Towards efficient verifiable conjunctive keyword search for large encrypted database [C]//23rd European Symposium on Research in Computer Security, 2018: 83–100.
 - [24] GAN Q, LIU J K, WANG X, et al. Verifiable searchable symmetric encryption for conjunctive keyword queries in cloud storage [J]. *Frontiers of Computer Science*, 2022, 16(6): 166820.
 - [25] ZHOU Q, DAI H, SHENG W, et al. EVSS: an efficient verifiable search scheme over encrypted cloud data [J]. *World Wide Web*, 2023, 26(4): 1459–1479.
 - [26] BOST R, FOUQUE P-A, POINTCHEVAL D. Verifiable dynamic symmetric searchable encryption: optimality and forward security [DB/OL]. <https://eprint.iacr.org/2016/062>.
 - [27] ZHANG Z J, WANG J F, WANG Y L, et al. Towards efficient verifiable forward secure searchable symmetric encryption [C]//European Symposium on Research in Computer Security, 2019: 304–321.
 - [28] KISSEL Z A, WANG J. Verifiable symmetric searchable encryption for multiple groups of users [C]//Proceedings of the International Conference on Security and Management, 2013: 1–7.
 - [29] WANG J, XIAO Y, ZHAO M. Fault-tolerant verifiable keyword symmetric searchable encryption in hybrid cloud [J]. *International Journal of Network Security*, 2015, 17(4): 471–483.
 - [30] WANG J, CHEN X, HUANG X, et al. Verifiable auditing for outsourced database in cloud computing [J]. *IEEE Transactions on Computers*, 2015, 64(11): 3293–3303.
 - [31] SU Y, WANG J, WANG Y, et al. Efficient verifiable multi-key searchable encryption in cloud computing [J]. *IEEE Access*, 2019, 7: 141352–141362.
 - [32] WANG X, WU J, WANG J. One for all: efficient verifiable dynamic multi-user searchable encryption in the presence of corrupted users [J]. *Journal of King Saud University-Computer and Information Sciences*, 2023, 35(8): 101703.
 - [33] SHI Z, FU X, LI X, et al. ESVSSE: enabling efficient, secure, verifiable searchable symmetric encryption [J]. *IEEE Transactions on Knowledge and Data Engineering*, 2020, 34(7): 3241–3254.
 - [34] NIAZ M S, SAAKE G. Merkle hash tree based techniques for data integrity of outsourced data [J]. *GvD*, 2015, 1366: 66–71.
 - [35] WOOD G. A secure decentralised generalised transaction ledger [J]. *Ethereum Project Yellow Paper*, 2014, 151: 1–32.
 - [36] ZHU J, LI Q, WANG C, et al. Enabling generic, verifiable, and secure data search in cloud services [J]. *IEEE Transactions on Parallel and Distributed Systems*, 2018, 29(8): 1721–1735.
 - [37] LI J, CAO Z, SHEN J, et al. MCVDSSE: secure multi-client verifiable dynamic symmetric searchable encryption [C]//International Conference on Science of Cyber Security, 2023: 85–96.
 - [38] YAN R Y, GUO R, YAN Y B, et al. Fine-grained result verifiable searchable encryption scheme with designated tester in cloud [J/OL]. *Journal of Computer Applications*, 2024–11–22. <https://link.cnki.net/urlid/51.1307.TP.20241122.0944.002>.
闫润雨, 郭瑞, 闫永勃, 等. 云中指定测试者的细粒度结果可验证搜索加密方案 [J/OL]. *计算机应用*, 2024–11–22. <https://link.cnki.net/urlid/51.1307.TP.20241122.0944.002>.