Modeling and Simulation 建模与仿真, 2023, 12(3), 2597-2607
Published Online May 2023 in Hans. https://www.hanspub.org/journal/mos
https://doi.org/10.12677/mos.2023.123239

Hans汉斯

分布式环境下基于倒排索引的可搜索加密研究

曹　伟，佟国香，伍建平，梁哲华

上海理工大学光电信息与计算机工程学院，上海

收稿日期：2023年3月24日；录用日期：2023年5月19日；发布日期：2023年5月26日

摘　要

目的：大数据、云计算技术和分布式技术给人们带来便捷，但是将个人数据加密并上传到云端服务器存储带来了个人隐私泄漏的问题。可搜索加密技术实现了安全高效地检索密文。方法：本文基于Spark计算框架，提出了分布式可搜索加密方案。使用高效的伪随机标签和文档标签通过Spark集群的RDD操作分布式生成倒排索引结构，利用Spark集群的性能优势进行检索，并结合提出的验证算法，进一步提高了分布式可搜索加密方案在半诚实且好奇的威胁模型下的安全性。结果：我们在不同节点数量下考察集群大小对存储性能和对计算效率影响，实验表明集群环境可以有效缓解单机存储的压力，并提升加解密、索引构造和验证的效率。结论：验证了分布式环境下基于倒排索引的可搜索加密的优越性。

关键词

Spark，可搜索加密，倒排索引，分布式

Research on Searchable Encryption Based on Inverted Index in Distributed Environment

Wei Cao, Guoxiang Tong, Jianping Wu, Zhehua Liang

School of Optical-Electrical and Computer Engineering, University of Shanghai for Science and Technology, Shanghai

Received: Mar. 24th, 2023; accepted: May 19th, 2023; published: May 26th, 2023

Abstract

Purposes: Big data, cloud computing technology and distributed technology bring convenience to people, but encrypting and uploading personal data to the cloud server storage brings about the

文章引用: 曹伟, 佟国香, 伍建平, 梁哲华. 分布式环境下基于倒排索引的可搜索加密研究[J]. 建模与仿真, 2023, 12(3): 2597-2607. DOI: 10.12677/mos.2023.123239

problem of personal privacy leakage. Searchable encryption technology enables secure and efficient retrieval of ciphertext. Methods: Based on Spark computing framework, this paper proposes a distributed searchable encryption scheme. The efficient pseudo-random tags and document tags are used to generate the inverted index structure through the RDD operation of the Spark cluster, and the performance advantages of the Spark cluster are used for retrieval. Combined with the proposed verification algorithm, the security of the distributed searchable encryption scheme under the semi honest and curious threat model is further improved. Finding: We investigate the impact of cluster size on storage performance and computing efficiency under different node numbers. Experiments show that the cluster environment can effectively relieve the pressure of stand-alone storage, and improve the efficiency of encryption and decryption, index construction and verification. Conclusions: The advantages of searchable encryption based on inverted index in distributed environment are verified.

Keywords

Spark, Searchable Encryption, Inverted Indexing, Distributed

1. 引言

随着信息技术蓬勃发展，海量数据催生了云计算和云存储业务的发展。为了对云端数据进行高效地传输和使用，可搜索加密这一研究方向应运而生。Song 等人最早于 2000 年提出可搜索加密问题，并提出了解决方案[1]，但是该方案陷门的生成过程是固定的，因此又易于遭受统计攻击。基于其提出的方案，后来的研究也确定了在可搜索加密方向的研究需求，即数据、搜索算法、查询算法的效率以及安全性。基于 Song 等人的研究，Curtmola 等给出更高要求的安全定义，以更高效的算法实现可搜索加密[2]，提出了规范化的对称可搜索加密的安全目标。文章提出采用 Hash 结构存储关键字和文件标志符之间的映射关系，在这种结构基础之上提出了 SSE-1 密文检索方案，提升了文件的检索效率，但是新增及删除文件时重新构造索引时间开销较大。Wang [3]采用倒排索引结构提出了隐藏密文检索的概率陷门生成算法，提升了搜索次数限制，但存在对手发起关键字猜测攻击的风险。为了管理云储存的加密数据，并高效地检索。钟晗等人通过建立高维关键词的词嵌入，增加语义距离扩展关键词集的方式建立安全索引，并用伪随机函数对私钥和关键词进行安全保护[4]。Varri [5]等人提出了一种在云存储中基于密文策略属性的可搜索加密，是无量子攻击的。Fan [6]等人改进了数据挖掘算法 Apriori，提出了一种支持多关键字子集检索的可搜索加密方案。Varadharajan 等人[7]通过并行处理大量数据来最大程度地减少加密所需的时间。Gu 等人[8]扩展了数据存储方式，通过属性加密为分布式数据存储提供精确的访问控制。Zheng 等人[9]建立了一个基于 Lucene 架构与 Hadoop 架构相结合的配电网大数据中心全文检索系统，是 Hadoop 在检索系统应用的重要体现。Duan 等人[10]结合互联网搜索中涉及的海量数据的并行化方法和可行性，提出了利用 MapReduce 进行网页倒排索引并行处理，但是没有将算法应用到密文搜索且 MapReduce 存在 IO 开销和资源消耗较大的问题。吴志强等人[11]给出了一种并行可拆分式倒排索引结构，可适用于分布式环境。Liang [12]等人设计了一种基于加密云文件的动态多关键字搜索加密方案，支持关键词的更新，提高了方案的实用性。Govindharaj 等人[13]提出一种预先分类的数据检索框架，使用 MapReduce 对数据集并行处理，从而方便检索。MapReduce 操作之间必须落盘，导致网络 IO 开销和资源消耗较大。随着技术发

展，基于内存的 Spark 技术出现并广泛应用，越来越多的国内外企业开始转向 Spark 进行数据分析处理工作，目前 Spark 在可搜索加密研究方面得到部分学者使用。罗王平等人[14]针对传统属性加密算法效率低的问题，提出了基于 Spark 构造快速加密和共享算法，降低了加密计算的时间成本。

针对云环境下的可搜索加密，本文从陷门安全性、检索结果验证、分布式计算方面研究并设计了安全可靠的可搜索加密方案。对于文献[2]的索引构造开销问题，我们在可信服务器端构造索引，索引文件生成过程中使用伪随机标签和文档标识符等结构实现相关信息的加密操作，并利用 Spark 集群加速构造索引解决了文献[10]中 MapReduce 存在的问题，然后对索引文件和明文文档加密处理后上传云服务器，同时可以将检索操作交给集群处理，提升检索效率。加密算法选择 AES，可以避免文献[3]存在的一些安全问题。本文的主要贡献包括：1) 提出了一种基于 Spark 集群构造的倒排索引结构，以支持高效的分布式索引构造和密文检索功能；2) 基于消息验证码设计一种验证算法；3) 从索引构造时间、空间效率和检索时间等角度，与文献[2]和[3]在不同集群节点数量下对比实验，验证了本文提出的可搜索加密方案的优越性。

2. 可搜索加密

可搜索加密可描述为以下过程：将数据文档集处理上传到云端私有 Spark 集群中，其中每个文档都被抽取为含有 n 个关键词的集合 $W=\{w_1,w_2,w_3,\cdots,w_n\}$，将文档集加密并构造可靠的安全索引块。构造过程中，利用倒排索引结构，使用伪随机标签与每个关键词文档对关联，并将伪随机标签添加到文档集中，构成可供检索的索引结构。私有集群完成任务后，将安全索引和密文文档上传到公有 Spark 服务器集群，服务器提供给授权的数据使用者文档检索服务。当数据使用者查询某一关键词时，查询服务器根据陷门函数构造查询令牌，通过安全索引查询密文文档，验证成功后用户可查看明文文档集。整体方案的结构如图 1 所示。

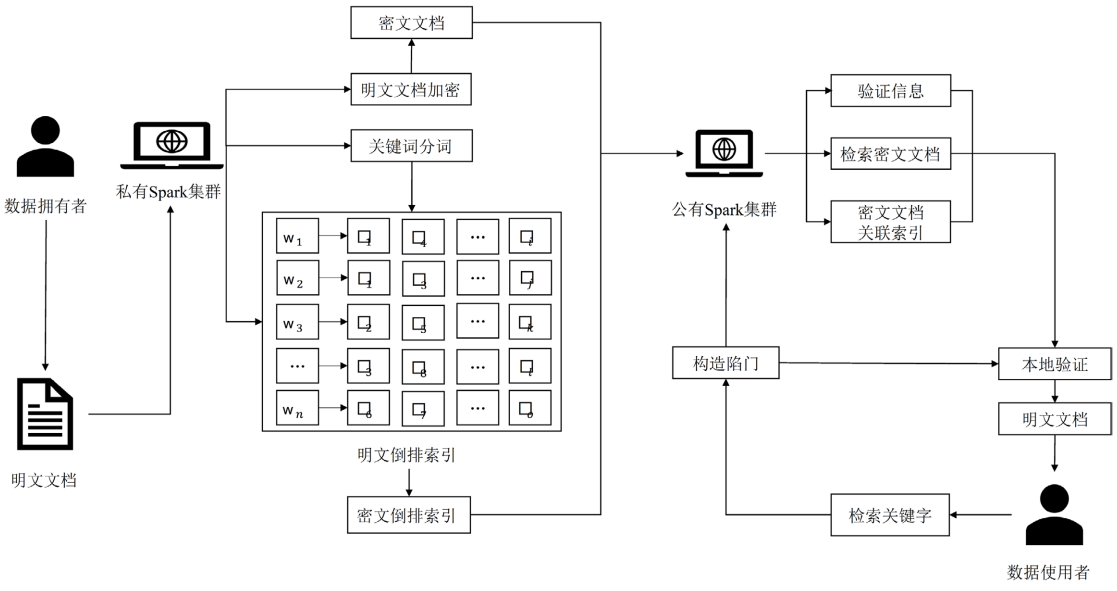

Figure 1. Solution architecture
图 1. 方案结构

2.1. 数据预处理及加解密模块

数据预处理模块用来完成关键词词典的构建。该模块首先读取原始文档内容，然后将读取的内容分

词并过滤停用词，保留其他关键词，以此获得所有关键词集合。大数据场景下的可搜索加密，我们选择AES 算法，该算法在保证数据安全的同时还能够顾及加解密操作的效率。此外，加解密部分将可变输入长度的伪随机函数和对称加密方案作为组件。定义函数可变长度伪随机函数 F，以密钥 $K\in\{0,1\}^{\lambda}$，$x\in\{0,1\}^{*}$ 为输入，输出 $\{0,1\}^{\lambda}$ 字符串。在敌手攻击游戏中，敌手选择索引 j 并输入 $x\in\{0,1\}^{*}$，对于真随机函数 *PRFReal*，存储在数组 $T[j]$处的密钥产生输出。在伪随机函数 *PRFRand* 中，敌手的响应从关键字字典的条目中返回，每个关键字在被使用时会初始化为一个随机值。

定理 1. 对于所有有效敌手 A，若以下表达式

$$Adv_{F,A}^{prf}(\lambda)=\Pr\left[PRFReal_{F}^{A}(\lambda)=1\right]-\Pr\left[PRFRand_{F}^{A}(\lambda)=1\right] \tag{1}$$

是可忽略的，则称函数 F 为可变输入长度的伪随机函数。

2.2. 分布式倒排索引生成模块

分布式倒排索引技术可以满足海量文档的检索效率需求。我们提取文档中的关键词，即每个文档会对应多个文档 ID，而这些文档中必定都包含所需的关键词，由此得到的倒排索引结构如图 2 所示。

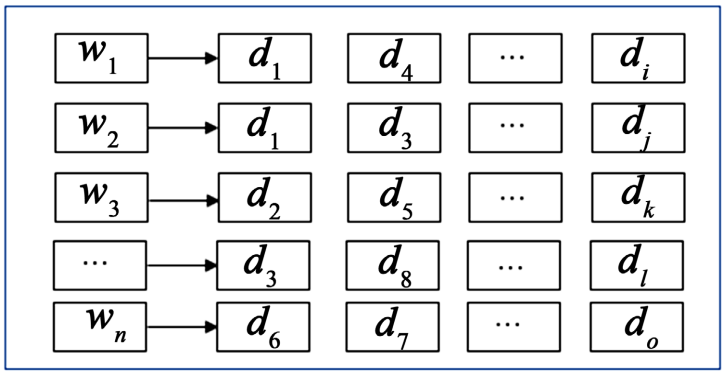

Figure 2. Inverted index structure
图 2. 倒排索引结构

对于明文倒排索引结构的安全缺陷和文献[2]频繁加解密操作，我们在索引文件生成过程中使用伪随机标签和文档标识符等结构实现相关信息的加密操作，利用 Spark 集群的计算能力提高了索引构造和加解密操作效率。

首先数据拥有者获取密钥 $K\xleftarrow{R}\{0,1\}^{\lambda}$，针对每个关键词，使用伪随机函数 F 将字符附加到 K 上，生成两个新的密钥 K_1 和 K_2，确保使用不同的密钥加密单词和文档记录。遍历整个文档数据集时，使用伪随机函数和密钥 K_1 生成单词标签 $l\leftarrow F(K_1,c)$，使用密钥 K_2 生成密文文档 $d\leftarrow Enc(K_2,id)$，将每个单独的标签 l 与文档 id 对 (l,d) 添加到索引集合L中，此集合作为索引用于密文检索。索引可以通过 Spark 使用 RDD 操作构造。

2.3. 检索模块

索引构建时，每个文档和关键词组合使用伪随机函数 F 和私钥 K 产生的标签是对应确定的。因此，在搜索阶段若给定相同的私钥 K 和伪随机函数 F，则可以重新产生相同的标签。根据这一特点，检索集群使用相同的私钥，按产生索引相同的方式构造陷门，就能够通过密文索引查找到相对应的文档标签。首先，输入密钥 K 和查询关键词 w；然后使用伪随机函数 F，根据 K 分别生成用于文档解密密钥 K_1、关键词解密密钥 K_2 和验证消息密钥 K_M。搜索服务器计算匹配加密伪随机标签，检索完成后验证结果。如

果验证通过，则使用 K_2 解密随机标签得明文文档集。检索具体算法过程如算法 1 所示。密文数据库 EDB 为一字典，其中包含有 $N=\sum_{w\in W}\left|DB(w)\right|$ 个文档标识符/密文关键词对。由于每个处理器都可以独立运行计算 $F(K_1,c)$，所以应用到 Spark 中可以分布式检索计算相应的密文。检索完成后，公有 Spark 服务器将检索结果密文文档集 EDB_Q，检索密文文档关联加密索引 $EIndex_Q$，文档集验证消息 Mac_{EQ}，检索密文文档关联加密索引验证消息 Mac_{EIQ} 发送给数据使用者。将接受到的内容在本地进行验证消息 Mac'_{EQ}、Mac'_{EIQ} 计算比对。若验证通过则将检索结果解密，否则结束。

2.4. 验证模块

为了保证数据使用者获得的数据完整性，我们设计了验证模块，即检索结果文档集存在于明文文档数据集中，并且返回的密文文档集无遗漏或改变。输入待验证消息 M 和消息验证密钥 MK，跟据公式 $HMAC(MK,M)=H(MK\oplus OP\,|\,H(MK\oplus IP\,|\,M))$ 计算消息验证值 $mac_M=Mac(M)$，根据不同情景分别在服务器和客户端进行，令第二次计算出的消息摘要为 $mac'_M=Mac'(M')$，若前后消息一致，则输出 1，否则输出 0。在索引构造阶段，当数据拥有者 OD 构造完成索引后，使用验证算法对安全索引 I 计算消息验证值 $mac_I=Mac(I)$。为了验证密文文档 $C=\{c_1,c_2,\cdots,c_3,\cdots,c_n\}$ 的完整性，在完成对文档加密后，还要根据文档加密标识符 $SID=\{sid_1,sid_2,\cdots,sid_i,\cdots,sid_n\}$ 对文档进行计算消息验证值 $mac_{ci}=Mac(c_i\,\|\,sid_i)$。在搜索阶段，不仅返回密文文档 C_Q，基于验证需求，还要返回所有感兴趣的检索关键词 W 对应的密文索引 I_Q，以及 mac_{I_Q} 和 mac_{c_i}。验证时，数据使用者使用验证密钥 MK 计算返回待验证的密文索引 I'_Q 的消息验证值 $mac_{I'_Q}=Mac(I'_Q)$，若 $mac_{I'_Q}=mac_{I_Q}$，则证明返回的密文索引没有被篡改。随后数据使用者使用验证无误的安全索引，根据陷门 TD 在本地进行检索，若对于每个查询关键词，都可以在本地查找到密文文档集，且文档集和服务器返回文档集相同，则证明检索结果无误，使用验证密钥计算密文文档的验证消息 $mac_{c'_i}=Mac(c'_i\,\|\,sid'_i)$，若有 $mac_{c'_i}=mac_{ci}$ 分别成立，则证明返回的密文文档集无遗漏或改变。

算法 1. $Search(K,w,EDB,EIndex)$

输入：密钥 K，搜索关键词 w，密文数据库 EDB，加密索引 $EIndex$；

输出：检索结果密文文档集 EDB_Q，检索密文文档关联加密索引 $EIndex_Q$，文档集验证消息 Mac_{EQ}，检索密文文档关联加密索引验证消息 Mac_{EIQ}.

1) $K_1\xleftarrow{R}F(K,Rand(x)\,\|\,w)$；
2) $K_2\xleftarrow{R}F(K,Rand(y)\,\|\,w)$；
3) 搜索集群分发搜索任务到各个节点，各个节点分布式计算 $d\leftarrow Get(\gamma,F(K_1,c))$；
4) 聚合检索结果，得检索结果密文文档集 EDB_Q，检索密文文档关联加密索引 $EIndex_Q$；
5) 分别使用 K_M 计算文档集验证消息 Mac_{EQ}，检索密文文档关联加密索引验证消息 Mac_{EIQ}.

3. 实验设计

3.1. 实验环境

硬件环境：本文仿真实验部署在 3 节点的 Spark 集群上，其中包含一个主节点，2 个从节点。服务器的配置参数如下所示：2.3 GHz 主频 2 核 CPU，8G 内存，512 G 硬盘空间。软件环境：本文实验全部在 Linux 环境下运行，源代码采用 IDEA 开发，编程语言为 Java、Scala。运行中的每个节点均安装部署 JDK1.8。实验集群配置如表 1 所示。

Table 1. Software configuration of Spark cluster
表 1. Spark 集群软件配置

软件名称	版本
操作系统	CentOS7
JDK	1.8.0_211
Hadoop	3.0.0
Spark	3.0.0
Zookeeper	3.4.6

3.2. 性能分析

为了验证提出的分布式可搜索加密算法，我们对文档数量，检索关键词个数，分区数量，节点个数和返回文档数量等角度进行分析。实验采用的数据集来自 Wikipedia，共有约 20,000 个文档。

明文到密文空间大小变化是体现索引构建性能的重要指标。从图 3 可知，密文文档大小会随着明文文档大小变化成比例增大，并且随着节点个数的变化几乎在各节点均匀分布。因此，分布式存储形式能够有效缓解单机情况下节点存储能力不足的问题。

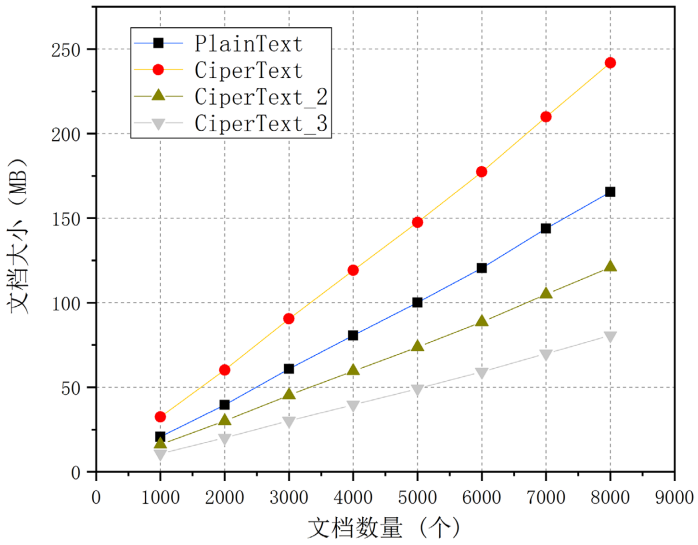

Figure 3. Specifies the relationship between ciphertext document size and data set size
图 3. 明密文文档大小随数据集规模变化关系

为了验证算法的索引构造空间效率，在文档大小在 20~180 MB 范围内时将该算法与文献[2]中的密文索引大小情况进行了比较，实验结果如图 4 所示。文献[2]方案占用空间更多，这主要是因为文献[2]在构造密文索引时，会首先生成明文索引，然后对明文索引中的数据和指针进行加密。而我们通过构造伪随机标签的形式存储索引，未使用指针加密，因此空间性能较好。此外，在多节点情况下索引文件分布式存储在集群上，单机情况下相较于文献[2]更加节约空间。因此基于 Spark 实现的密文倒排索引结构具有良好的空间性能。

密文索引构造所需的时间是对算法进行衡量的标准之一，从图 5 可知，文献[3]包含公钥配对和求幂以及乘法计算等操作，所以索引构造效率较低。在初始文档数量较少时，文献[2]方案索引构造效率较高。

在文档数量到达 6000 左右时，表现出分布式计算的优势，所以随着文档数据集的不断增大，本方案索引构造的时间开销表现优于文献[2]方案。资源相同资源下，基于 Spark 的分布式可搜索加密能够有效的提高索引构造效率。

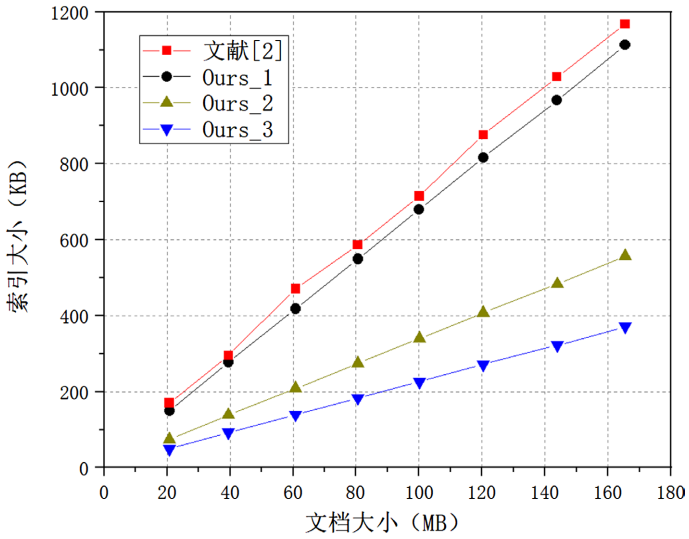

Figure 4. The size of an index relative to document size
图 4. 索引大小随文档大小的变化关系

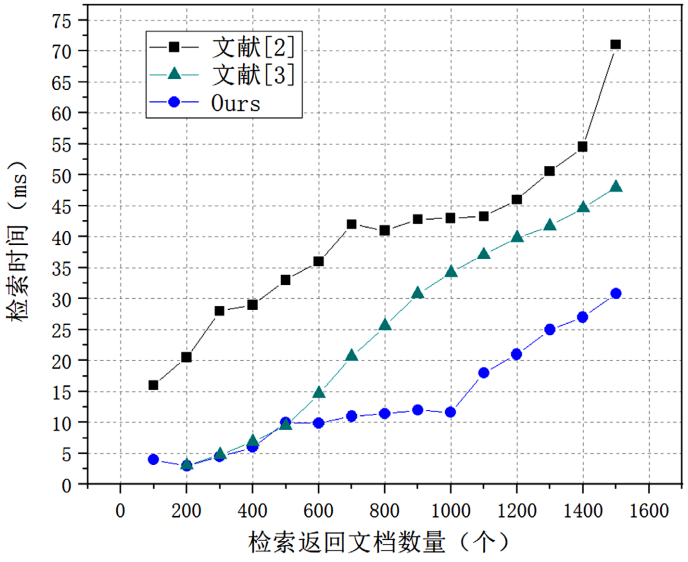

Figure 5. The construction time of ciphertext indexvaries with the number of documents
图 5. 密文索引构造时间随文档数量的变化关系

实验检索时间随检索返回文档数量的变化情况如图 6 所示，当返回相同数量文档时，分布式方案的检索时间远小于文献[2]。因为文献[2]方案中构造的倒排索引指针要经过加密处理，进行相关数据检索时需要先进行指针解密。随着文档数量逐渐增加，分布式方案所需的检索时间增长更为平缓。在检索返回文档数量小于 500 时，文献[3]方案和分布式可搜索加密方案检索耗时接近，但随着检索规模变大，分布式计算优势会逐渐显现出来。此外，由于采用本地验证方式，故实际通信开销较小，检索效率优于文献[3]。综合以上实验结果，分布式可搜索方案更加适用于大数据环境。

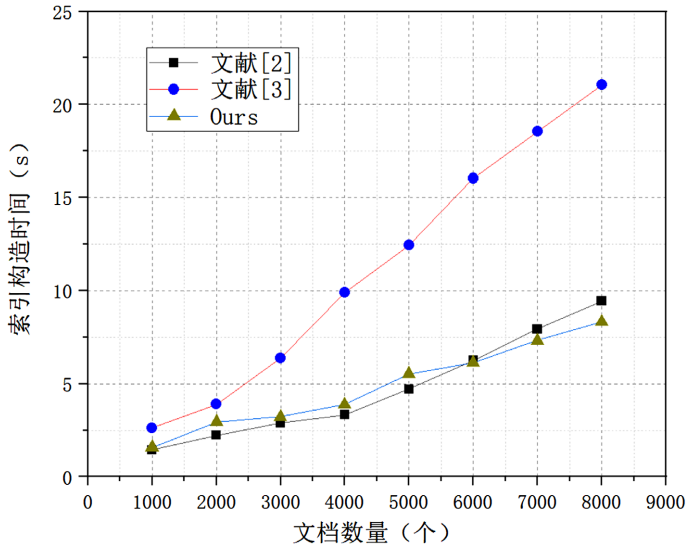

Figure 6. The relationship between retrieval time and the number of documents returned from retrieval
图 6. 密检索时间随检索返回文档数量的变化关系

为了保证数据的完整性，我们根据验证模块分别计算查询关键词返回的文档与验证文档的 MAC 值。在固定文档数量为 5000，抽取关键词个数为 500~3000 并且搜索关键词个数 $t=5$ 时，对公有 Spark 服务器返回得的 EDB_Q (检索结果密文文档集)，$EIndex_Q$ (检索密文文档关联加密索引)，Mac_{EQ} (文档集验证消息)，Mac_{EIQ} (检索密文文档关联加密索引验证消息)在本地进行验证，并计算比产生的 Mac'_{EQ}、Mac'_{EIQ}。结果显示检索返回和本地验证的 MAC 完全相同，即返回结果正确且无遗漏。

3.3. 安全性分析

考虑到要保证信息在传输过程中的有效性和完整性，本文提出的方案主要建立在“半诚实且好奇”的公有云服务器威胁模型基础上，并将服务器与潜在攻击对象视为威胁。这一威胁模型不同于以往的可搜索加密方案，其通常采用“诚实且好奇”的服务器威胁模型。在这一威胁模型下，检索过程可能会产生信息失真，就要求可搜索加密方案能够验证返回的检索结果，另一方面对于密文文档和陷门等信息也要验证其完整性、真实性。下面我们首先给出正确性和可靠性的定理。

定理 **2.** 正确性：对于一个可验证可搜索加密方案，若方案满足 $\forall(SK,MK,DK)\leftarrow KeyGen(1^{\lambda})$，对于 $\forall D_i\in D(1\le i\le n),\forall w\in W$，有：

$$\left(Search(TD,I)\rightarrow\left(C_Q,I_Q,mac_{I_Q},mac_{ci}\right)\right)\wedge\left(Verify(M,MK)\rightarrow(0,1)\wedge DecFile(C_Q,SK)=D_Q\right)=1 \quad (2)$$

则称该验证方案为正确的。其中 i 为常数，n 为文档个数。

定理 **3.** 可靠性：对于一个可验证的可搜索加密方案，面对一个多项时间内的敌手 $\mathbb{A}$，当用户提交关键词检索陷门 TD' 后，敌手 $\mathbb{A}$ 产生一个伪造检索结果 C'_Q，并生成对应的 I'_Q，$mac_{I'_Q}$，$mac_{c'_i}$ 返还给用户，而服务器运行产生的正确结果为 $\left(C_Q,I_Q,mac_{I_Q},mac_{c_i}\right)\leftarrow Search(TD,I)$，若

$$\Pr\left[Adv_{\mathbb{A}}(\lambda)=1\right]\le negl(\lambda) \quad (3)$$

即使用敌手伪造信息来进行验证，获得结果 $1\leftarrow Verify(M,MK)$ 成立的概率可以忽略，称该验证方案可靠。

下面我们采取选择关键字攻击(indistinguishability of SCF-PEKS against chosen keyword attack,

IND-SCF-CKA)游戏来证明关键字的不可区分性，在抗选择关键字攻击等安全性依赖于判定性子群假设和DBDH 假设前提下[15]，假设当前有敌手 A 和模拟者 B，我们建立两者相互攻击模型，安全参数设为 λ 。

Game1：假设 A 为内部攻击者(服务器)。

1) Setup：运行初始化算法 $GlobalSetup(\lambda)$ 生成系统参数 GP，三个密钥生成算法得到用户密钥中心、服务器、接受者的公私钥对：(pk_K, sk_K)，(pk_S, sk_S) 和 sk_R，随后模拟者 B 将 (pk_S, sk_S) 和 pk_K 发送给敌手 A。

2) Queryphase1：对任意关键词 w，敌手 A 向 B 询问关键字陷门，B 运行陷门生成算法 $T_w = Trapdoor(GP, sk_R, w)$，并将结果返回给敌手 A。

3) Challenge：A 向 B 发送挑战关键词对 (w_0, w_1)，并且 w_0, w_1 都不能出现在 Queryphase1 中，B 随机选取 $b \in (0,1)$，计算 $C^* = SCF - PEKS(GP, pk_S, pk_K, w_b)$ 作为挑战密文发给 A。

4) Queryphase2：敌手 A 在不选择 w_0 和 w_1 的前提下，继续对关键词进行陷门询问。

5) Guess：A 输出猜测 b'，若 $b' = b$，则敌手 A 获得胜利，否则 A 失败。

A 攻破 Game1 的优势为：

$$Adv_A^{Game1}(\lambda) = \left| \Pr\left[A(b' = b)\right] - \frac{1}{2} \right|. \tag{4}$$

Game2：假设 A 为外部攻击者，攻击游戏过程如下。

1) Setup：和 Game1 类似，通过 $GlobalSetup(\lambda)$ 算法获得系统参数 gp，并通过密钥生成算法得到三个公私钥对 (pk_K, sk_K)，(pk_S, sk_S) 和 sk_R，最后模拟者 B 将 pk_S 和 pk_K 发送给敌手 A。

2) Queryphase1：A 自适应的选择接收者的密钥 $x \in Z_N^*$，在密钥生成中心生成用户密钥，得到 sk_R。

3) Challenge：敌手 A 发送关键词对 (w_0, w_1)，B 随机选取 $b \in (0,1)$，计算 $C^* = SCF - PEKS(GP, pk_S, pk_K, w_b)$ 作为挑战密文发给 A。

4) Guess：A 输出猜测 b'，若等于 b 则敌手 A 获得胜利，否则 A 失败。

A 攻破 Game2 的优势为：

$$Adv_A^{Game2}(\lambda) = \left| \Pr\left[A(b' = b)\right] - \frac{1}{2} \right|. \tag{5}$$

上述两个攻击游戏中，如果 Adv_A^{Game1} 和 Adv_A^{Game2} 之于安全参数 λ，则称该方案是 IND-SCF-CKA 安全的。

假设 1 敌手的随机询问 $w_i (i = 1,2,\cdots,q)$ 每次都是不同的，且给定的 R_3 可以完美的随机化陷门。

假设 2 密钥注册中心不是敌手，那么其肯定忠诚且安全。

引理 1 若 DBDH 假设为困难的，那么可以忽略对任意多项式时间算法的敌手 A 能够区分关键字陷门的优势。

证明：假设存在一个多项式时间算法内的外部敌手对本文方案进行攻击，和游戏挑战者 B。攻击游戏过程为：挑战者 B 设置一个阶为 p' 的群 G', G_T' 和双线性映射 $e: G' \times G' \to G_T'$，其中 G', G_T' 分别为群 G, G_T 的素数阶子群。向挑战者输出参数 $(g_1, g_1^a, g_1^b, g_1^c, T)$，挑战者的目标为区分 T 是否等于 $e(g_1, g_1)^{abc}$。

敌手和挑战者的攻击游戏如下：

1) Setup：输入安全参数 λ，选择一个抗碰撞单向 hash 函数 $H: \{0,1\}^* \to Z_N$，双线性映射的参数有 (N, G, G_T, e)，关键字空间为 $KS_w = \{0,1\}^*$，全局参数为 $GP = \{N, G, G_T, e, H, KS_w\}$，运行密钥生成算法可得 $pk_s = \left\{ u^{\frac{\beta}{\alpha}} = Q \right\}$，$sk_s = \alpha$，$sk_R = \{x, R\}$，$sk_K = \beta$，$pk_K = \{g_1, u, X, Y\}$，挑战者 B 将 pk_S，sk_S，pk_K，sk_R 发送给敌手 A。

2) Queryphase：敌手 A 向随机预言机发送关键词 w_i 并询问其陷门，预言机获得 pk_K， sk_R， sk_K，然后随机选择 $R_3 \in G_{p_3}$，计算 $T_2 = T_1^n Y^{H(w')} R_3$，将计算出的陷门 $T_{w_i} = [T_1, T_2]$ 返回到敌手 A。

3) Challenge：敌手 A 多次询问预言机后，随机选取未经过查询的关键词 (w_0, w_1) 发送给挑战者 B，挑战者将随机选取 $b \in \{0,1\}$ 并设置挑战关键词 $w^* = w_b$，计算 $T_1 = R^{\frac{1}{x}}$，$n = H\left(e(T_1, u)^{\beta}\right)$，随机选择 $R_3 \in G_{p_3}$，计算 $T_2 = T_1^n Y^{H(w^*)} R_3$，将关键词陷门 $T_w^* = [T_1, T_2]$ 发送给敌手 A。

4) Guess：敌手 A 在获取陷门后进行猜测，输出猜测 b'。若 $b' = b$，则有 $t = H\left(e(T_1, Q)^{\alpha}\right) = H\left(e(g_1, g_1)^{abc}\right)$，$T = e(g_1, g_1)^{abc}$，则 T_w^* 为合法的输出，返回“Yes”；否则 T 为 G_T 中的随机元素，返回“No”。

如果敌手 A 可以区分关键词陷门，那么挑战者 B 就可以攻破 DBDH 假设。综上所述本方案中的关键词陷门具有不可区分性，可抵抗外部关键字猜测攻击，也能够抵抗选择关键字攻击，满足 IND-SCF-CKA 安全。

上文证明了加密方案可以抵抗外部关键字猜测攻击和选择关键字攻击，我们接着通过分析不同情况下敌手攻击来验证本文可搜索加密的安全性。假设敌手的攻击方式主要为唯密文攻击，则敌手能够获得本文方案下的加密索引文件和密文文档集合，我们使用的加密算法 AES 是完全公开的，在以上条件下，本方案的加密索引和密文文档的安全性通过密钥保护。由于 AES 加密算法为成熟安全的对称加密算法，因此，敌手在不知道密钥的情况下难以破解密文文档和密文索引的内容。在仅有加密文档和加密索引的情况下，敌手可以通过分析密文词频获取密文与明文的对应关系从而进行推测。但是本文使用伪随机标签将关键词和文档标识符的信息进一步隐藏，因此密文索引中只包含伪随机标签相关信息。伪随机标签的使用在一定程度上避免了文档词频统计攻击。在对倒排索引加密时，每个倒排索引关键词对应的密钥都随机生成，各个倒排索引之间关联性较弱，很大程度上可以避免敌手攻击。

在敌手已获得部分明文信息条件下，敌手可以对获得的信息构造加密索引并和本文方案中的密文索引进行比较。但是在密文倒排索引中，敌手无法获取各文档在索引中所处位置，难以通过对安全索引进行对比获取明文信息。综合以上考虑分析，基于 Spark 的可搜索加密方案是安全可靠的。

4. 结语

本文提出一种基于 Spark 的分布式可搜索加密方案，提高了大数据场景下单机存储性能和计算性能。另外通过伪随机标签和文档与关键词标签等技术，构造出轻量化方案。结合提出的验证算法，增强检索结果的正确性和完整性。实验对比本文方案与文献[2]和文献[3]方案不同分布式节点个数下的密文索引空间性能，结果显示本文方案随着分布式节点的不断增多，单机空间消耗减少，具有良好的空间性能。此外，通过索引构造的不同时间开销对比，表明该方案在索引构造过程消耗的时间远小于其余两种方案。而且在关键词检索方面，本文方案的时间效率仍优于文献[2]和文献[3]方案。验证结果证明检索能够正确完整的返回目标文档，查全率与查准率满足方案目标。综上，设计的基于 Spark 的分布式可搜索加密方案更适用于大数据环境下的可搜索加密需求。

在当下可搜索加密研究场景下，一对多、多对多模式已十分常见，分布式集群的计算能力可以处理海量数据，如何让这些数据同时给多名授权用户提供支持，充分发挥集群力量，是云环境下可搜索加密应用场景的探索方向。

参考文献

[1] Song, D.X., Wagner, D. and Perrig, A. (2000) Practical Techniques for Searches on Encrypted Data. *Proceedings of IEEE Symposium on Security and Privacy*, Berkeley, 14-17 May 2000, 44-55.

[2] Curtmola, R., Garay, J., Kamara, S., *et al*. (2006) Searchable Symmetric Encryption: Improved Definitions and Efficient Constructions. *Proceedings of the* 13*th ACM Conference on Computer and Communications Security*, Alexandria, 30 October-3 November 2006, 79-88. https://doi.org/10.1145/1180405.1180417

[3] Wang, B., Song, W., Lou, W., *et al*. (2015) Inverted Index Based Multi-Keyword Public-Key Searchable Encryption with Strong Privacy Guarantee. 2015 *IEEE Conference on Computer Communications* (*INFOCOM*), Kowloon, 26 April-1 May 2015, 2092-2100. https://doi.org/10.1109/INFOCOM.2015.7218594

[4] 钟晗, 郭飞. 基于词嵌入的云存储可搜索加密方案[J]. 重庆师范大学学报(自然科学版), 2017, 34(4): 70. https://doi.org/10.11721/cqnuj20170454

[5] Varri, U.S., Pasupuleti, S.K. and Kadambari, K.V. (2021) CP-ABSEL: Ciphertext-Policy Attribute-Based Searchable Encryption from Lattice in Cloud Storage. *Peer-to-Peer Networking and Applications*, **14**, 1290-1302. https://doi.org/10.1007/s12083-020-01057-3

[6] Fan, K., Chen, Q., Su, R., *et al*. (2021) MSIAP: A Dynamic Searchable Encryption for Privacy-Protection on Smart Grid with Cloud-Edge-End. *IEEE Transactions on Cloud Computing*.

[7] Varadharajan, V., Kuppusamys and Krishnamoorthy, K. (2016) MapReduce Based Framework for Searchable Encryption. 2016 15*th International Symposium on Parallel and Distributed Computing* (*ISPDC*), Fuzhou, 8-10 July 2016, 184-189. https://doi.org/10.1109/ISPDC.2016.32

[8] Gu, K., Zhang, W.B., Li, X., *et al*. (2022) Self-Verifiable Attribute-Based Keyword Search Scheme for Distributed Data Storage in Fog Computing with Fast Decryption. *IEEE Transactions on Network and Service Management*, **19**, 271-288. https://doi.org/10.1109/TNSM.2021.3123475

[9] Zheng, Y.Z., Fu, Y., Zhang, R.F., *et al*. (2020) Research on Lucene Based Full-Text Query Search Service for Smart Distribution System. 2020 3*rd International Conference Onartificial Intelligence and Big Data* (*ICAIBD*), Chengdu, 28-31 May 2020, 338-341.

[10] Duan, A.L. (2012) Research and Application of Distributed Parallel Search Hadoop Algorithm. 2012 *International Conference on Systems and Informatics* (*ICSAI*2012), Yantai, 19-20 May 2012, 2462-2465.

[11] 吴志强, 李肯立, 郑蕙. 高效可扩展的对称密文检索架构[J]. 通信学报, 2017, 38(8): 79-93.

[12] Liang, Y., Li, Y., Zhang, K., *et al*. (2021) DMSE: Dynamic Multi-Keyword Search Encryption Based on Inverted Index. *Journal of Systems Architecture*, **119**, Article ID: 102255. https://doi.org/10.1016/j.sysarc.2021.102255

[13] Govindharaj, I., Saravanan, D., Lavanya, R.V., *et al*. (2015) Effective Information Retrieval Approach Based on Parallel Matrix Method and Mapreduce Framework. *Proceedings of the* 2015 *International Conference on Advanced Research in Computer Science Engineering & Technology* (*ICARCSET* 2015), Eluru, 6-7 March 2015, 1-5. https://doi.org/10.1145/2743065.2743096

[14] 罗王平, 冯朝胜, 邹莉萍, 等. 一种支持快速加密的基于属性加密方案[J]. 软件学报, 2020, 31(12): 3923-3936. http://www.jos.org.cn/1000-9825/5856.html

[15] Deng, Y.J., Du, H.Z. and Dou, X.X. (2020) Registered Users Public Key Searchable Encryption Scheme with Secure-Channel Free. *Computer and Modernization*, No. 9, 43-48.