

支持联合查询和交叉模式隐藏的结构加密

杨紫依, 马昌社, 谢南江

(华南师范大学计算机学院, 广东 广州 510631)

摘要: 结构加密属于可搜索加密中在性能、安全性和查询表达能力之间权衡得较好的方案, 不仅保护了云存储数据的隐私, 且支持对密文数据库进行多种类型的查询。2013年提出的不经意交叉索引 (oblivious cross-tag, OXT) 方案支持多关键字联合查询, 但令牌的计算与通信开销均为线性复杂度, 近年来提出的联合过滤 (ConjFilter) 方案降低了令牌开销, 减少了支持布尔查询的结构加密方案的隐私泄露, 但导致不同查询之间的交叉泄露。为了防止出现联合查询泄露问题, 提出了基于随机置换和计数器加密模式的转换密钥加密 (transformable key encryption, TKE) 方案, 该方案可以在服务端不解密密文的条件下, 把一个密钥加密的密文转换为另一个密钥加密的密文, 且不泄露除转换模式之外的其他任何信息。基于该方案设计了支持联合查询的加密多映射方案可转换密钥过滤器 (transformable key filter, TK-Filter), 其与 ConjFilter 方案具有几乎相同的通信开销和存储开销, 且能有效防止不同查询之间的交叉泄露。实验分析结果表明, TK-Filter 在各种规模的数据集上相较 ConjFilter 减少了 58% 到 86% 的初始化时间, 提高了 34% 到 41% 的搜索效率。

关键词: 结构加密; 联合查询; 交叉泄露; 计数器加密模式

中图分类号: TP309

文献标志码: A

DOI: 10.11959/j.issn.2096-109x.2024048

Cross pattern hiding structured encryption for conjunctive queries

YANG Ziyi, MA Changshe, XIE Nanjiang

School of Computer Science, South China Normal University, Guangzhou 510631, China

Abstract: Structured encryption, a specialized form of searchable encryption, has been recognized for offering improved tradeoffs between performance, security, and expressiveness. It is designed to protect the privacy of cloud storage data and supports a variety of queries on the ciphertext database. The structured encryption scheme OXT, proposed in 2013, supports conjunctive queries of multi-keywords but was found to incur linear complexity in both

收稿日期: 2023-11-27; 修回日期: 2024-04-10

通信作者: 马昌社, chsma@163.com

基金项目: 国家自然科学基金项目 (62072192, U2336209)

Foundation Items: The National Natural Science Foundation of China (62072192, U2336209)

引用格式: 杨紫依, 马昌社, 谢南江. 支持联合查询和交叉模式隐藏的结构加密[J]. 网络与信息安全学报, 2024, 10(3): 175-187.

Citation Format: YANG Z Y, MA C S, XIE N J. Cross pattern hiding structured encryption for conjunctive queries[J]. Chinese Journal of Network and Information Security, 2024, 10(3): 175-187.

token computation and search communication overheads. The ConjFilter scheme, more recently introduced, was aimed at decreasing the token overhead and reducing privacy leakage for structured encryption schemes that support Boolean queries. However, it was observed that this scheme resulted in cross-leakage between different queries. To address this issue, a transformed key encryption (TKE) scheme, based on random permutation and counter encryption modes was proposed. In essence, the TKE scheme was capable of converting ciphertext encrypted by one key into ciphertext encrypted by another key, without the need for decrypting the ciphertext at the serving side and without disclosing any information beyond the transformation modes. Utilizing this scheme, an encrypted multi-map scheme, transformable key filter (TK-Filter), was designed to support conjunctive queries. This scheme was reported to have nearly identical communication and storage overhead as the ConjFilter scheme but effectively prevented cross-leakage between different queries. Experimental analysis results show that TK-Filter reduces the initialization time by 58% to 86% and improves the search efficiency by 34% to 41% compared to ConjFilter across datasets of various sizes.

Keywords: structured encryption, conjunctive query, cross leakage, counter encryption mode

0 引言

为了减少用户本地存储的资源开销,常用的技术是将数据库外包给云服务器,用户可以通过云服务器来获取数据。但云服务器一般不具备足够的可信性,因此用户需要将数据进行加密后再上传,但经标准加密(如高级加密标准(advanced encryption standard, AES))后的数据不支持查询操作,因此出现了可搜索加密(searchable encryption, SE)技术。经过多年的研究发展,现有的可搜索加密技术在性能、安全性和查询表达能力之间有着不同的权衡,如基于全同态加密^[1](fully homomorphic encryption, FHE)、功能加密^[2](functional encryption, FE)和 ORAM^[3](oblivious random access machine)等技术构建的可搜索加密方案具有高安全性和高查询表达能力,但由于查询开销太大,不适合实际部署。而基于确定性加密^[4](deterministic encryption, DTE)和保序加密^[5](order-preserving encryption, OPE)等技术构建的可搜索加密方案,虽然提供了较高的查询性能和查询表达能力,但相比上述方案存在更多信息泄露的问题,因此更容易遭受攻击。目前,在不同的可搜索加密方案中,结构加密^[6-8](structured encryption, STE)在性能、安全性和查询表达能力之间权衡得较好。STE 方案在允许泄露某些明文信息的前提下支持针对密文的高效搜索,但同时其允许的泄露被利用到各种隐私安全攻击中,对数据构成了隐私泄露威胁。因此,如何减少 STE 方案的泄露成为 STE 研

究的热点方向。近年提出的联合过滤^[9](ConjFilter)方案虽然减少了支持布尔查询的 STE 方案的隐私泄露,但存在不同查询之间的交叉泄露。

本文的主要焦点是 ConjFilter 方案所引发的交叉泄露问题,该泄露源于计算 dtag 时, tag 与第一个标签 a 有关, 密钥 K^x 与两个标签 a, b 有关,这意味着任何拥有标签 a 的 tag 都可以与 K^x 进行运算,返回额外的结果。有一种简单的解决方法是建立 n^2 张关系表,让密钥 K^x 与 3 个标签 a, b 和 c 相关联,但是这种方法会显著增加服务器的存储开销。为了实现解决交叉泄露且不增加任何存储开销的目的,本文提出了新的密码学原语转换密钥加密(transformable key encryption, TKE)方案,并基于该原语设计了支持联合查询的高效 STE 方案,称为可转换密钥过滤器(transformable key filter, TK-Filter)。本文证明了该方案在诚实且好奇的攻击模型下具有语义安全性,最后通过实验对 TK-Filter 方案与 ConjFilter 方案的性能进行了比较,结果显示 TK-Filter 方案的性能更优。

1 相关工作

STE 在保证数据隐私的同时,可以对“结构化数据”进行高效地查询。STE 最著名的子类是可搜索对称加密^[10-14](searchable symmetric encryption, SSE)。本文主要研究 STE 的重要类型—加密多映射(encrypted multi-map, EMM)。

早期支持多关键词查询的方案具有线性搜索时间。2004 年, Golle 等^[15]首先提出支持多关键

词搜索的SSE方案。然而,该方案的通信和存储成本与数据呈线性相关(当文件数量为 d ,通信开销的复杂度为 $O(d)$),导致效率不佳。2005年,Ballard等^[16]提出基于双线性对构建的方案,相较文献[15]通信开销更为高效,其具有常数级别的通信复杂度。但是该方案的服务端计算开销大量增加(当文件数量 d ,双线性对运算的复杂度为 $O(d)$)。2013年,Cash等^[17]提出具有次线性搜索复杂度的OXT方案,它将支持联合查询扩展到支持一般的布尔查询。2017年,Kamara等^[18]提出了非交互式的SSE方案IEX,它执行任意的布尔查询均能达到亚线性搜索效率,但其针对联合查询的表现差于OXT。而ConjFilter^[9]方案是改进OXT方案的过滤算法。与OXT方案相比,ConjFilter方案在传输过程中针对布尔公式中的每个关键词只需传输一个伪随机函数(pseudo-random function, PRF)密钥,而不是一组交叉令牌。

2 先验知识

先验知识中相关符号及含义如表1所示。

2.1 IND-CPA安全的对称加密方案

表1 符号及含义
Table 1 Symbol and meaning

符号	含义
λ	安全参数
$F_K(\cdot)$	用加密密钥 K 加密消息得到的输出
$\mathcal{CQ}$	联合查询序列
Φ^i	第 i 个联合查询
Π	对称加密方案
$p \leftarrow P(\cdot)$	概率多项式时间算法的输出值
$a // b$	两个字符串的拼接
$\text{negl}(\lambda)$	λ 的可忽略函数
$\mathcal{V}$	所有值的集合
$\mathcal{U}$	所有标签的集合
ℓ_i	标签
$\vec{v}_i$	值组成的元组
$\{(\ell_i, \vec{v}_i)\}_{i \in [n]}$	规模为 n 的多映射
$\{0, 1\}^\lambda$	长度为 λ 的由0和1组成的比特串
$\text{MM}[\ell]$	获取与标签 ℓ 相关联的元组
$s[i]$	查询中出现频率最低的标签对(s-term)
$\vec{x}[i]$	查询中剩余的标签(x-term向量)

对称加密方案^[19-20] (symmetric encryption scheme, SKE) $\Pi = (\text{Gen}, \text{Enc}, \text{Dec})$, 由密钥生成算法Gen、加密算法Enc和解密算法Dec 3个部分组成。给定安全参数 λ , 密钥生成算法Gen输入 λ , 输出密钥 $K \leftarrow \{0, 1\}^\lambda$ 。给定消息空间 $\{0, 1\}^*$, 从中选择任意明文 M , 加密算法Enc输入密钥 K 和明文 M , 输出密文 C , 密文空间为 $\{0, 1\}^*$ 。解密算法Dec输入密钥 K 和密文 C , 输出明文 M 。

给定一个任意敌手 $\mathcal{A}$, 对于安全参数 λ 和挑战者 $\mathcal{C}$, 定义方案 Π 的IND-CPA安全性实验。

实验 $\text{Exp}_{\Pi, \mathcal{A}}^{\text{IND-CPA}}(\lambda)$

1) $\mathcal{C}$: $K \xleftarrow{R} \text{Gen}(1^\lambda)$

2) $\mathcal{A}$: $m_0, m_1 \leftarrow \{0, 1\}^*$, 发送 m_0, m_1 给 $\mathcal{C}$

3) $\mathcal{C}$: $b \xleftarrow{\$} \{0, 1\}$, 计算 $c \leftarrow \text{Enc}(K, m_b)$, 发送 c 给 $\mathcal{A}$

4) $\mathcal{A}(c) = b'$

定义1 称对称加密方案 Π 是IND-CPA安全的, 如果对于任意多项式时间敌手 $\mathcal{A}$, 存在一个可忽略函数 $\text{negl}(\lambda)$, 有以下不等式成立:

$$|\Pr[b' = b] - \frac{1}{2}| \leq \text{negl}(\lambda) \quad (1)$$

2.2 伪随机函数

定义2 $F: \{0, 1\}^\lambda \times \{0, 1\}^l \rightarrow \{0, 1\}^l$ 是函数族, 其中 $\{0, 1\}^\lambda$ 表示密钥空间。 F 是安全的伪随机函数^[21], 如果对于任意多项式时间区分器 $\mathcal{D}$, 存在一个可忽略函数 $\text{negl}(\lambda)$, 有以下不等式成立:

$$|\Pr[\mathcal{D}^{F_K(\cdot)}(1^\lambda) = 1] - \Pr[\mathcal{D}^{f(\cdot)}(1^\lambda) = 1]| \leq \text{negl}(\lambda) \quad (2)$$

其中, $f: \{0, 1\}^l \rightarrow \{0, 1\}^l$ 是随机函数, K 表示随机密钥。

2.3 加密多映射

定义3 (布尔加密多映射方案) 给定安全参数 λ , 加密多映射方案 $\Sigma_{\text{MM}} = (\text{Setup}, \text{Token}, \text{Search})$ 包含3个算法, 其具体描述如下。

1) $(K_c, \text{EMM}) \leftarrow \Sigma_{\text{MM}}.\text{Setup}(1^\lambda, \text{MM})$: 系统初始化算法主要在客户端执行。输入安全参数 λ 和明文多映射MM, 输出密钥 K_c 和加密的数据结构EMM, 其中 K_c 由客户端保存, EMM由服务器保存。

2) $\text{tok}_\phi \leftarrow \Sigma_{\text{MM}}.\text{Token}(K_C, \Phi)$: 令牌生成算法主要在客户端执行。输入客户端密钥 K_C 和布尔公式 Φ , 输出令牌 tok_ϕ 。

3) $\text{res} \leftarrow \Sigma_{\text{MM}}.\text{Search}(\text{EMM}, \text{tok}_\phi)$: 搜索协议在服务器和客户端之间交互进行。客户端输入令牌 tok_ϕ , 服务器输入加密多映射 EMM, 输出搜索结果集合 res 。

一般加密方案有两个重要的属性: 计算正确性和语义安全性, 下面给出两个属性的定义。

定义 4 如果对于任意安全参数 λ 、任意敌手 $\mathcal{A}$ 和游戏 $\text{Cor}_A^\Sigma(\lambda)$, 均有 $\Pr[\text{Cor}_A^\Sigma(\lambda) = 1] \leq \text{negl}(\lambda)$, 则称 Σ_{MM} 具有计算正确性。

$\text{Cor}_A^\Sigma(\lambda)$: 对于明文数据集 MM , 游戏运行 $(K_C, \text{EMM}) \leftarrow \Sigma_{\text{MM}}.\text{Setup}(1^\lambda, \text{MM})$, 并提供 EMM 给敌手 $\mathcal{A}$ 。然后, 敌手 $\mathcal{A}$ 在 EMM 上进行自适应查询。对于查询 $\Phi^i = \ell_1 \wedge \dots \wedge \ell_n$ ($\Phi^i \in \mathcal{CQ}$) 和令牌 tok_{Φ^i} , 运行 $\text{res} \leftarrow \Sigma_{\text{MM}}.\text{Search}(\text{EMM}, \text{tok}_{\Phi^i})$ 。在此过程中, 如果搜索结果 $\text{res} \neq \text{MM}(\Phi^i)$, 则游戏获胜并输出 1, 否则游戏失败并输出 0。

对于加密多映射方案 Σ_{MM} , 利用泄露函数 $\mathcal{L} = (\mathcal{L}_{\text{Setup}}, \mathcal{L}_{\text{Query}})$ 描述其安全性。 $\mathcal{L}_{\text{Setup}}$ 描述的是敌手收到加密数据时可以掌握的信息, $\mathcal{L}_{\text{Query}}$ 由 $\mathcal{L}_{\text{Token}}$ 和 $\mathcal{L}_{\text{Search}}$ 两个元素组成, 其中 $\mathcal{L}_{\text{Token}}$ 描述有关查询令牌的信息, $\mathcal{L}_{\text{Search}}$ 描述有关查询结果的信息。

给定方案 $\Sigma_{\text{MM}} = (\text{Setup}, \text{Token}, \text{Search})$, 其泄露为 $\mathcal{L}_\Sigma = (\mathcal{L}_{\text{Setup}}, \mathcal{L}_{\text{Query}})$, 对于仿真器 $\mathcal{S}$ 和任意敌手 $\mathcal{A}$, 定义游戏 $\text{Real}_{\mathcal{A}}^\Sigma(\lambda)$ 和 $\text{Ideal}_{\mathcal{A}, \mathcal{S}}^\Sigma(\lambda)$ 。

$\text{Real}_{\mathcal{A}}^\Sigma(\lambda)$: 敌手 $\mathcal{A}(1^\lambda)$ 选择明文多映射 MM , 并提供给挑战者 $\mathcal{C}$ 。挑战者 $\mathcal{C}$ 对明文多映射 MM 执行 $(K_C, \text{EMM}) \leftarrow \Sigma_{\text{MM}}.\text{Setup}(1^\lambda, \text{MM})$, 并提供 EMM 给敌手 $\mathcal{A}$ 。对于 $i = 1, \dots, \text{poly}(\lambda)$, $\mathcal{A}$ 自适应选择 $\Phi^i \in \mathcal{CQ}$ 并提供给挑战者 $\mathcal{C}$ 。对于由 Φ^i 生成的 tok_{Φ^i} , 挑战者 $\mathcal{C}$ 执行 $\text{res} \leftarrow \Sigma_{\text{MM}}.\text{Search}(\text{EMM}, \text{tok}_{\Phi^i})$, 并将协议执行的消息副本提供给敌手 $\mathcal{A}$ 。敌手 $\mathcal{A}$ 返回一个判断值 $b \in \{0, 1\}$ 。

$\text{Ideal}_{\mathcal{A}, \mathcal{S}}^\Sigma(\lambda)$: 敌手 $\mathcal{A}(1^\lambda)$ 选择明文多映射 MM , 并提供给挑战者 $\mathcal{C}$ 。挑战者 $\mathcal{C}$ 执行 $\text{EMM} \leftarrow \mathcal{S}(\mathcal{L}_{\text{Setup}}(\text{MM}))$, 并提供 EMM 给敌手 $\mathcal{A}$ 。对于 $i = 1, \dots, \text{poly}(\lambda)$, 敌手 $\mathcal{A}$ 自适应选择 $\Phi^i \in \mathcal{CQ}$, 并提供给挑战者 $\mathcal{C}$ 。仿真器取得 $\mathcal{L}_{\text{Query}}(\text{MM},$

$\Phi^1, \dots, \Phi^{i-1})$, 执行 $\text{res} \leftarrow \mathcal{S}(\mathcal{L}_{\text{Query}}(\Phi^i))$, 并提供其输出给敌手 $\mathcal{A}$ 。敌手 $\mathcal{A}$ 返回一个判断值 $b \in \{0, 1\}$ 。

定义 5 (EMM 方案的语义安全性) 如果对于任意多项式时间敌手 $\mathcal{A}$, 都存在多项式时间仿真器 $\mathcal{S}$ 和一个可忽略函数 $\text{negl}(\lambda)$, 使 $|\Pr[\text{Real}_{\mathcal{A}}^\Sigma(\lambda) = 1] - \Pr[\text{Ideal}_{\mathcal{A}, \mathcal{S}}^\Sigma(\lambda) = 1]| \leq \text{negl}(\lambda)$, 则称 EMM 方案 Σ_{MM} 在自适应攻击下具有 $\mathcal{L}_\Sigma$ 语义安全。

2.5 交叉泄露

数据发送者与云服务器的交互过程中, 可能发生的信息泄露^[22-25]一般分为 3 种: 存储泄露、结果模式泄露和查询模式泄露。其中结果模式泄露一般来源于 3 个方面: 单关键词结果模式 (single keyword result pattern, SP) 泄露、关键词对结果模式 (keyword-pair result pattern, KPRP)^[12,24,26] 泄露和多关键词交叉查询结果模式 (multiple keyword cross-query intersection result pattern, MIP) 泄露。ConjFilter 方案的交叉 (cross-query result pattern, CP) 模式泄露属于特殊形式的 MIP。

发起一个联合查询序列, 序列由通用形式为 $\Phi = \langle s, \vec{x} \rangle$ 的查询组成, 其中 Φ^i 可表示为 $s[i]$ 和 $\vec{x}[i]$ 的联合。

定义 6 交叉模式泄露 CP 形式化定义为 $\text{CP}[i, j] = \text{MM}(s[i]) \cap \text{MM}(x_\beta[j]) \cup \text{MM}(s[j]) \cap \text{MM}(x_\alpha[i])$, 其中 α 和 β 代表不同的标签, i 和 j 代表不同的查询, $\cap$ 代表合取, $\cup$ 代表析取。

ConjFilter 方案支持联合查询的核心在于计算 dtag, 以验证标签对 (a, b) 与值 v 之间的关系。 v 仅受第一个标签的限制, 因此它可以与第一个标签相同的密钥 K^x 进行计算, 这导致了第一个标签相同的查询之间的交叉泄露问题。以检索关键词“他、喜欢、春节”和“他、吃、米饭”对应的文档为例, 服务器通过交叉计算可以推断出“他、喜欢、米饭”和“他、吃、春节”对应的文档。假设多映射的标签全集包含 $l_1, \dots, l_n$ 等 n 个标签, 发起两个查询 $\ell_1 \wedge \dots \wedge \ell_n$ 和 $\ell'_1 \wedge \dots \wedge \ell'_n$ (其中 $\ell_1 = \ell'_1$)。除了原本的查询结果, 它还会向服务器透露更多关于查询的信息, 包括 $\{\ell_1 \wedge \ell_2 \wedge \ell'_i\}_{3 \leq i \leq n}$ 和 $\{\ell'_1 \wedge \ell'_2 \wedge \ell_j\}_{3 \leq j \leq n}$ 。

3 TKE 方案

为了解决交叉泄漏问题，本文设计了转换密钥加密（transformable key encryption, TKE）方案，算法应用随机置换，达到无须解密密文的同时转换加密密钥的效果。

3.1 随机置换

随机置换是按照置换密钥 Kep 改变输入序列的字符顺序的运算。此外，需要介绍算法 1 帮助构建 TKE 方案。

置换密钥生成算法 $PGen^{[27-29]}$: $\{0, 1\}^{l \log l} \times Z^* \rightarrow \{0, 1\}^{Z' \log Z'}$ 是一个函数，它将密钥处理成所需的长度 Z^* 。算法输入置换密钥 Kep 和长度 n ，输出新置换密钥 Kep' 。其中 Kep' 是由 0 到 $n-1$ 的数字组成，且每个元素不重复。这个新的置换密钥将用于执行置换操作。

算法 1 PGen

输入 Kep ，序列大小 n

输出 Kep'

准备一个长度为 $n \log n$ 的数组 Kep' ，按照索引给数组的每一位赋值

$k \leftarrow \{0, 1\}^l$

For $i = 0$ to $n - 1$ do

$t = F(k, i)$

交换 $Kep'[i]$ 与 $Kep'[t]$

End for

转换密钥生成算法 $FindCK^{[27-29]}$: $Kep_A^n \times Kep_B^n \rightarrow CK^n$ 是一个双射函数，它接收两个置换密钥 Kep_A 和 Kep_B ，并计算出 CK 。对于任意输入序列 S ，利用 CK 可直接将 Kep_A 随机置换 S 的输出转换为 Kep_B 随机置换 S 的输出。

算法 2 FindCK

输入 Kep_A , Kep_B

输出 CK

For $i = 0$ to $n - 1$ do

For $i = 0$ to $n - 1$ do

If $Kep_A[i] = Kep_B[j]$ then

$CK[j] = i$

break

End if

End for

End for

3.2 TKE 及其安全性定义

基于转换密钥的设计，本文应用随机置换关联数据与两个标签，每个 CK 将限制转换前的原始标签与待转换的目标标签，即运行 $FindCK(Kep_{a,b}, Kep_{a,c})$ 得到的 CK 只能将 $\{E_{a,b}^1, E_{a,b}^2, \dots, E_{a,b}^s\}$ 转换为 $\{E_{a,c}^1, E_{a,c}^2, \dots, E_{a,c}^s\}$ ，而在任何原始标签或目标标签不同的情况下，转换将会失败，有效防止了不同查询之间对密钥的滥用。

定义 7 TKE 方案由 4 个部分组成，分别为加密算法 Enc 、解密算法 Dec 、密钥重新生成算法 $ReGen$ 、重加密算法 $ReEnc$ 。

1) $(c||ctr) \leftarrow TKE.Enc(K^t, K^p, a, b, v)$: 对称加密算法，算法输入主密钥 K^t 和 K^p 、标签对 (a, b) 和值 v ，输出密文 c 和加密计数器 ctr 。 K^t 和 K^p 将缓存在本地，密文 $(c||ctr)$ 将发送给云服务器。

2) $v \leftarrow TKE.Dec(K^t, K^p, a, b, (c||ctr))$: 对称解密算法，算法输入主密钥 K^t 和 K^p 、标签对 (a, b) 及密文 $(c||ctr)$ ，输出明文 v 。

3) $CK_\omega \leftarrow TKE.ReGen(K^t, a, b, \omega, l)$: 输入主密钥 K^t 、原始标签对 (a, b) 、目标标签对 (a, ω) ，输出重加密密钥 CK_ω 。 CK_ω 将发送给云服务器以支持转换密钥的操作。

4) $c' \leftarrow TKE.ReEnc(CK_\omega, c)$: 输入 $TKE.ReGen$ 算法生成的密钥 CK_ω 和密文 c ，输出新密文 c' 。该算法通过随机置换的操作更新密文的密钥，且新密文 c' 由 (a, ω) 加密而得。

为了分析 TKE 方案的安全性，对于方案 $TKE = (Enc, Dec, ReGen, ReEnc)$ ，用泄露函数 $\mathcal{L}_{TKE}$ 表示敌手 $\mathcal{A}$ 能从密文中获取到的信息。

定义 8 给定可转换密钥加密方案 $TKE = (Enc, Dec, ReGen, ReEnc)$ 和安全参数 λ ，对于任意多项式时间敌手 $\mathcal{A}$ ，定义概率性实验 $Real_{\mathcal{A}}^{TKE}(\lambda)$ 和 $Ideal_{\mathcal{A}}^{TKE}(\lambda)$ ，具体实验如下。

(1) $Real_{\mathcal{A}}^{TKE}(\lambda)$: 实验设置随机预言机 $\mathcal{O}_1$ 。 $\mathcal{O}_1$ 执行 $(c||ctr) \leftarrow TKE.Enc(K^t, K^p, a, b, v)$ ，并提供密文 $(c||ctr)$ 给 $\mathcal{A}$ 。对于 $\omega = 1, \dots, \text{poly}(\lambda)$ ，敌手 $\mathcal{A}$ 自适应地选择密钥 CK_ω 。然后 $\mathcal{O}_1$ 执行

TKE.ReEnc(CK_ω, c), 并将协议执行的消息副本提供给 $\mathcal{A}$, 最后输出 $\mathcal{A}$ 返回的判断值比特 b 。

(2) $\text{Ideal}_A^{\text{TKE}}(\lambda)$: 实验设置随机预言机 $\mathcal{O}_2$ 。 $\mathcal{O}_2$ 执行 $(c \parallel \text{ctr}) \leftarrow \mathcal{O}_2(\mathcal{L}_{\text{TKE}}(v))$, 并提供密文 $(c \parallel \text{ctr})$ 给敌手。对于 $\omega = 1, \dots, \text{poly}(\lambda)$, 敌手 $\mathcal{A}$ 自适应地选择 CK_ω 。然后 $\mathcal{O}_2$ 执行 $c' \leftarrow \mathcal{O}_2(\mathcal{L}_{\text{TKE}}(v, CK_\omega))$, 并将协议执行的消息副本提供给 $\mathcal{A}$, 最后输出 $\mathcal{A}$ 返回的判断值比特 b' 。

如果对于任意多项式时间算法 $\mathcal{A}$, 存在 $|\Pr[\text{Real}_A^{\text{TKE}}(\lambda) = 1] - \Pr[\text{Ideal}_A^{\text{TKE}}(\lambda) = 1]| \leq \text{negl}(\lambda)$, 则称 TKE 方案在自适应攻击下具有 IND-CPA (indistinguishability under chosen plaintext attack) 安全。

3.3 TKE 方案

TKE 方案 4 个算法的详细流程如下。

算法 TKE.Enc 输入主密钥 K^t 和 K^p 、标签对 (a, b) 和值 v 。流程主要分为两层加密运算：第一是运行哈希函数 H 输入 a 与值 v , 输出密钥 ctr , 对 ctr 使用 IND-CPA 对称加密算法获得流密码 rnd , 用 rnd 异或值 v 得到第一层密文 v' ; 第二是用伪随机函数修饰 K^t 以获得与标签 (a, b) 相关联的密钥 $K_{a,b}^t$, 并使用置换密钥生成算法 PGen, 将 $K_{a,b}^t$ 的长度调整为与 $|v|$ 一致, 表示为 $P_{a,b}$ 。使用密钥 $P_{a,b}$ 随机置换 v' , 输出密文 c ; 同时, 使用 IND-CPA 对称加密算法加密 ctr 为 ectr , 密钥为 K^p 。最终输出 $(c \parallel \text{ectr})$, 且密钥 K^t 和 K^p 缓存在客户端。

算法 3 TKE.Enc

输入 K^t, K^p, a, b, v

输出 $(c \parallel \text{ectr})$

$l = |v|$

$K_{a,b}^t \leftarrow F(K^t, a \parallel b)$

$P_{a,b} \leftarrow \text{PGen}(K_{a,b}^t, l)$

$\text{ctr} = H(a, v)$

$\text{rnd} = \text{Enc}(K^t, \text{ctr})$

$v' = \text{rnd} \oplus v$

For $i = 0$ to $l - 1$ do

$c[i] \leftarrow v'[P_{a,b}[i]]$

End for

$\text{ectr} = \text{Enc}(K^p, \text{ctr})$

Return $(c \parallel \text{ectr})$

算法 TKE.Dec 输入主密钥 K^t 和 K^p 、标签对 (a, b) 及密文 $(c \parallel \text{ectr})$, 该算法按照与 TKE.Enc 相逆的过程进行解密。首先, 用伪随机函数修饰 K^t 获得与标签对 (a, b) 相关联的密钥 $K_{a,b}^t$ 。然后, 使用 PGen 算法将密钥 $K_{a,b}^t$ 的长度处理成与 $|c|$ 一致, 即 $P_{a,b}$ 。接下来, 使用 $P_{a,b}$ 对密文 c 进行随机置换, 还原出 v' 。其次, 使用主密钥 K^p 解密 ectr 获得 ctr , 再通过 IND-CPA 对称加密算法加密 ctr 为 rnd , 用 rnd 异或 v' , 最终解密出明文 v 。

算法 4 TKE.Dec

输入 $K^t, K^p, a, b, (c \parallel \text{ectr})$

输出 v

$l = |c|$

$K_{a,b}^t \leftarrow F(K^t, a \parallel b)$

$P_{a,b} \leftarrow \text{PGen}(K_{a,b}^t, l)$

$\text{ctr} = \text{Dec}(K^p, \text{ectr})$

$\text{rnd} = \text{Enc}(K^t, \text{ctr})$

For $i = 0$ to $l - 1$ do

$v'[P_{a,b}[i]] \leftarrow c[i]$

End for

$v = v' \oplus \text{rnd}$

Return v

算法 TKE.ReGen 输入主密钥 K^t 、标签对 (a, b) 和待转换的标签对 (a, ω) , 输出重加密密钥 CK_ω 。首先, 该算法使用伪随机函数修饰主密钥 K^t , 分别获取与标签对 (a, b) 相关联的密钥 $K_{a,b}^t$, 以及与标签对 (a, ω) 相关联的密钥 $K_{a,\omega}^t$ 。然后, 使用 PGen 算法将密钥 $K_{a,b}^t$ 和 $K_{a,\omega}^t$ 处理成长度为 l 的置换密钥。最后调用 FindCK 算法计算两者的转换密钥 CK_ω 。

算法 5 TKE.ReGen

输入 K^t, a, b, ω, l

输出 CK_ω

$K_{a,b}^t \leftarrow F(K^t, a \parallel b)$

$K_{a,\omega}^t \leftarrow F(K^t, a \parallel \omega)$

$CK_\omega = \text{FindCK}(\text{PGen}(K_{a,b}^t, l),$

$\text{PGen}(K_{a,\omega}^t, l))$

Return CK_ω

算法 TKE.ReEnc 涉及客户端和服务端之间的交互。客户端输入 TKE.ReGen 生成的转换密钥

CK_ω ，云服务器输入需要进行转换密钥的密文 c ，输出更新密钥后的密文 c' 。其中主要根据 CK_ω 执行随机置换，便可获得置换密钥为 $K_{a,\omega}^t$ 的密文 c' 。

算法6 TKE.ReEnc

输入 CK_ω, c

输出 c'

For $i = 0$ to $l - 1$ do

$c'[i] \leftarrow c[CK_\omega[i]]$

End for

Return c'

对称 TKE 方案具有计算正确性，如果对于安全参数 λ ，对于任意标签对 (ℓ_1, ℓ_a) 和任意待转换的标签 ℓ_β ，以及由 $TKE.Enc(K_a, M)$ 算法生成的密文 C 。执行重加密算法 $TKE.ReEnc(K_{a \Rightarrow \beta}, C)$ 得到 C_1 ，与执行加密算法 $TKE.Enc(K_\beta, M)$ 得到 C_2 ，有

$$\Pr[C_1 = C_2] \geq 1 - \text{negl}(\lambda) \quad (3)$$

下面分析 TKE 方案的安全性，对于方案 $TKE = (Enc, Dec, ReGen, ReEnc)$ ，对于转换密钥集合 CK ，泄露函数为 $\mathcal{L}_{TKE}(\vec{v}) = N$ ， $\mathcal{L}_{TKE}(\vec{v}, CK) = (N, TP)$ 。其中 N 为密文数量， TP 为集合 CK 的转换模式， $TP[i, \omega] = ReEnc_{CK_\omega}(i)$ ， $ReEnc_{CK_\omega}(i)$ 表示使用转换密钥 CK_ω 转换元组中第 i 个密文得到的元素。

定理1 令 $\mathcal{L}_{TKE}$ 如上所述，假设对称加密算法 Enc 满足 IND-CPA 安全， F 是安全的伪随机函数， H 是哈希函数，那么 TKE 方案在自适应攻击下具有 $\mathcal{L}_{TKE}$ -语义安全。

证明 在游戏 $Ideal$ 中定义一个仿真器 $\mathcal{S}$ ，使用泄露函数 $\mathcal{L}_{TKE}(\vec{v}, CK) = (N, TP)$ 作为输入，仿真器 $\mathcal{S}$ 只能访问 $\mathcal{L}_{TKE}$ 返回的信息集合，不能获得任何关于明文的其他信息。 $\mathcal{S}$ 的操作分为两部分：首先，敌手 $\mathcal{A}$ 选择集合 $\vec{v}$ ，仿真器输入 $\mathcal{L}_{TKE}(\vec{v}) = N$ ，随机生成 $c \leftarrow \{0, 1\}^l$ ，以此类推，得到密文集合，并将其发送给敌手 $\mathcal{A}$ ；其次，敌手 $\mathcal{A}$ 选择密钥集合 CK ，仿真器输入 $\mathcal{L}_{TKE}(\vec{v}, CK) = (N, TP)$ 。对于转换密钥 CK_ω ，存在 $TP[i, \omega] = ReEnc_{CK_\omega}(i)$ ，并发送给敌手 $\mathcal{A}$ 。

对于任意敌手 $\mathcal{A}$ ，哈希 H 函数和 F 函数的输出不可区分，对称加密方案具有 IND-CPA 安全。因此， $Real_{\mathcal{A}}^{TKE}(\lambda)$ 生成的密文集合与 $Ideal_{\mathcal{A}, \mathcal{S}}^{TKE}(\lambda)$

生成的密文集合不可区分，有以下结论成立：

$$\left| \Pr[Real_{\mathcal{A}}^{TKE}(\lambda) = 1] - \Pr[Ideal_{\mathcal{A}, \mathcal{S}}^{TKE}(\lambda) = 1] \right| \leq \text{negl}(\lambda) \quad (4)$$

则 TKE 方案在自适应攻击下具有 $\mathcal{L}_{TKE}$ -语义安全。

4 TK-Filter

ConjFilter 方案改变了 OXT 方案中 DH (Diffie-Hellman) [30] 类型的指数操作，采用更简洁的加密方式和搜索过程，根据常数级别的交叉令牌即可完成联合查询。但其导致了 OXT 方案中不存在的查询间的交叉模式泄露问题。

基于 TKE 方案，本文提出了支持联合查询的高效 STE 方案 TK-Filter，该方案解决了 2.5 节中提出的交叉泄露问题。TK-Filter 方案的设计思想如下：ReGen 算法生成的转换密钥是严格确定原始密钥和目标密钥的，查询时通过不同的转换密钥，构建每个标签对与固定的值集合的关系向量。具体地，将值设计成双层加密的构造，第一层加密由第一个标签 a 和值 v 生成密钥，第二层加密再关联起标签对 a 和 b 。由于第二层加密的可转换性，可通过 TKE.ReEnc 算法转换原始标签对 (a, b) 为目标标签对 $(a, \cdot)$ 。通过 $n - 1$ 次置换与关系计算，可构建出每个标签对与固定的值集合的关系矩阵，以此实现过滤。

TK-Filter 方案有初始化算法 Setup、令牌算法 Token、搜索协议 Search 和解密算法 Resolve4 部分。给定 TKE 方案 $(Enc, Dec, ReGen, ReEnc)$ 、EMM 方案 $(\Sigma_{MM}, Setup, \Sigma_{MM}, Token, \Sigma_{MM}, Search)$ 、SKE 方案 (Enc, Dec) 、伪随机函数 F 和哈希函数 H 。下面将介绍方案的详细过程。

4.1 方案描述

TK-Filter.Setup 如算法 7 所示，TK-Filter 预先计算好所有可能的标签对对应的元组。对于每个标签 $\ell \in \mathcal{U}$ ，它映射标签 $\ell \in co(\ell)$ 为 $MM(\ell) \cap MM(\ell)$ 对应的元组，其中 $co(\ell)$ 代表 MM 中与 ℓ 共现的标签集合。

算法7 TK-Filter.Setup

输入 $\lambda, MM = \{(\ell_i, \vec{v}_i)\}_{i \in [m]}$

输出 mk, EMM

$K^t, K^p \leftarrow \{0, 1\}^\lambda$

```

 $\cdots \text{MM}^t \leftarrow \{ \}$ 
 $X \leftarrow \emptyset$ 
 $L = \{\ell_1, \ell_2, \dots, \ell_m\}$ 
For  $(a, b) \in L \times L$  do
   $\text{MM}^t[(a, b)] \leftarrow \emptyset$ 
  For all  $v \in \text{MM}[a] \cap \text{MM}[b]$  do
     $\text{ctr} = H(a, v)$ 
     $(c \| \text{ctr}) \leftarrow \text{TKE.Enc}(K^t, K^p, a, b, v)$ 
    add  $(c \| \text{ctr})$  into  $\text{MM}^t[(a, b)]$ 
   $X = \{c\} \cup X$ 
End for
End for
 $(\text{msk}^t, \text{EMM}^t) \leftarrow \Sigma_{\text{MM}}.\text{Setup}(1^t, \text{MM}^t)$ 
Return  $\text{mk} = (K^p, K^t, \text{msk}^t), \text{EMM} = (\text{EMM}^t, X)$ 

```

系统初始化算法由客户端执行，如算法 1 所示。客户端输入明文多映射 MM 和安全参数 λ ，输出加密多映射 EMM^t 、集合 X 和密钥 mk 从密钥空间 $\{0, 1\}^t$ 中随机抽样生成主密钥 K^t, K^p ，以及初始化 MM^t 为空映射。将 MM 中的标签 $a \in \mathcal{U}$ 与其共现标签 $b \in \text{co}(a)$ 作为多映射的键，并将 $\text{MM}[a] \cap \text{MM}[b]$ 对应的元组作为 MM^t 的值。对于元组中的每个值，按照如下步骤进行加密。

(1) 运行哈希函数 H 输入第一个标签 a 与值 v ，输出密钥 ctr ；对每个值 v ，调用加密算法 TKE.Enc ，生成密文 $(c \| \text{ctr})$ ；并将键值对 $\{(a, b), (c \| \text{ctr})\}$ 添加到 MM^t 中，同时将密文 c 添加到关系表 X 中。

(2) 对 MM^t 调用多映射的初始化算法，即 $(\text{msk}^t, \text{EMM}^t) \leftarrow \Sigma_{\text{MM}}.\text{Setup}(1^t, \text{MM}^t)$ ，得到 EMM^t ，并将其存储在服务器上。

初始化算法运行结束后，客户端存储主密钥 K^t, K^p 和多映射密钥 msk^t ，服务器存储加密多映射 EMM^t 和关系表 X 。

令牌算法由客户端执行， TK-Filter.Token 如算法 8 所示。客户端输入密钥 mk 和查询 Φ ，计算令牌 tok_Φ 。首先，展开联合范式 $\Phi = (\ell_1 \wedge \ell_2 \wedge \cdots \wedge \ell_n)$ ，对于第一个标签对 (ℓ_1, ℓ_2) ，客户端调用 $\text{tok}^t \leftarrow \Sigma_{\text{MM}}.\text{Token}(\text{msk}^t, (\ell_1, \ell_2))$ 计算得到令牌。其次，客户端根据标签 $\ell_3, \ell_4, \dots, \ell_n$ ，调用

TKE.ReGen 算法，计算转换密钥集合 $\{\text{CK}_3, \text{CK}_4, \dots, \text{CK}_n\}$ 。

算法 8 TK-Filter.Token

输入 $\text{mk} = (K^p, K^t, \text{msk}^t)$, $\Phi = (\ell_1 \wedge \cdots \wedge \ell_n)$

输出 tok_Φ

$\text{tok}^t \leftarrow \Sigma_{\text{MM}}.\text{Token}(\text{msk}^t, (\ell_1, \ell_2))$

For $d = 3, \dots, n$ do

$\text{CK}_d \leftarrow \text{TKE.ReGen}(K^t, \ell_1, \ell_2, \ell_d, \ell)$

End for

Return $\text{tok}_\Phi = (\text{tok}^t, \text{CK}_3, \text{CK}_4, \dots, \text{CK}_n)$

搜索算法由服务器执行，如算法 9 所示。服务器输入多映射 EMM^t 和关系表 X ，客户端输入令牌 tok^t 和密钥集合 $\{\text{CK}_3, \text{CK}_4, \dots, \text{CK}_n\}$ 。首先，服务器使用 tok^t 调用 $\Sigma_{\text{MM}}.\text{Search}$ 算法来检索加密多映射 EMM^t ，得到 $\text{MM}(\ell_1) \cap \text{MM}(\ell_2)$ 对应的元组 $S_2 = \{(c_{1,2} \| \text{ctr}_1), (c_{2,2} \| \text{ctr}_2), \dots\}$ 。然后，使用 CK_3 重加密 S_2 的密文，相当于用标签对 (ℓ_1, ℓ_3) 生成的置换密钥 P_{ℓ_1, ℓ_3} 加密 S_2 中的值得到新密文，即 $S_3 = \{(c_{1,3} \| \text{ctr}_1), (c_{2,3} \| \text{ctr}_2), \dots\}$ 。以此类推，对于每个转换密钥，服务器做以下工作：① 对每个 $c_{i,2} \in S_2$ ，使用转换密钥 CK_d 将 $c_{i,2}$ 转换为 $c_{i,d}$ ；② 对每个 $c_{i,d} \in S_d$ ，检查它是否存在于关系表 X 中，可获得结果向量。通过以上运算，可以把 $\text{MM}[\ell_1 \wedge \ell_2]$ 对应的 S_2 过滤为同时包含 ℓ_1, ℓ_2 和 ℓ_d 的集合 S_d ，且每个 c_i 由 (ℓ_1, ℓ_d) 进行加密。以此类推，直到完成最后一个标签对 (ℓ_1, ℓ_n) 的转换与过滤。整合以上 $n - 2$ 个结果向量获得一个结果矩阵，矩阵中一列均为 1 的密文为符合查询的结果，并将密文 $\{c_{i,n} \| \text{ctr}_{i,n}\}$ 添加到集合 ct 中。

算法 9 TK-Filter.Search

输入 $\text{tok}_\Phi, \text{EMM} = (\text{EMM}^t, X)$

输出 ct

$(c \| \text{ctr}_i)_{i \in [L]} \leftarrow \Sigma_{\text{MM}}.\text{Search}(\text{tok}^t, \text{EMM}^t)$

For $d = 3, \dots, n$ do

For $l = 1, \dots, L$ do

$c_{l,d} \leftarrow \text{TKE.ReEnc}(\text{CK}_d, c_{l,2})$

End for

End for

$\text{ct} \leftarrow \emptyset$

If $c_{l,3}, \dots, c_{l,n} \in X$ then

$$ct = \{c_{l,n} \parallel \text{ectr}_{l,n}\} \cup ct$$

End if

Return ct

解密算法由客户端和服务端共同执行，如算法10所示。服务器提供密文结果集ct给客户端，客户端输入本地密钥 K^p 和 K^t ，并调用TKE.Dec解密ct的每个密文，输出明文的搜索结果。

算法10 TK-Filter.Resolve

输入 ct, mk = (K^p , K^t)

输出 res

res ← $\emptyset$

For all ($c \parallel \text{ectr}$) ∈ ct do

$v \leftarrow \text{TKE.Dec}(K^t, K^p, \ell_1, \ell_n, (c \parallel \text{ectr}))$

res = res $\cup \{v\}$

End for

Return res

定理2 假设 F 是安全的伪随机函数，($\text{Gen}, \text{Enc}, \text{Dec}$)是具有计算正确性的SKE方案，($\text{TKE.Enc}, \text{TKE.Dec}, \text{TKE.ReGen}, \text{TKE.ReEnc}$)是具有计算正确性的TKE方案。那么对于任意多项式时间敌手 $\mathcal{A}$ 和任意安全参数 λ ，定义正确性游戏 $\text{Cor}_{\mathcal{A}}^{\text{TK-Filter}}(\lambda)$ 。该游戏的最大获胜概率是可忽略的，则称TK-Filter具有计算正确性。

$\text{Cor}_{\mathcal{A}}^{\text{TK-Filter}}(\lambda)$: 对于任何大小为 $\text{poly}(\lambda)$ 的MM，运行(mk, EMM) $\leftarrow \text{TK-Filter.Setup}(\lambda, \text{MM})$ ，并提供EMM给 $\mathcal{A}$ 。然后，敌手 $\mathcal{A}$ 在EMM上执行自适应查询，对于任意查询 $\Phi^i = \ell_1 \wedge \dots \wedge \ell_n$ ，游戏运行 $ct \leftarrow \text{TK-Filter.Search}(\text{EMM}, \text{tok}_{\Phi^i})$ 。在这个过程中，如果搜索结果 $ct \neq \text{MM}(\Phi)$ ，则游戏获胜并返回1，否则返回0。

4.2 安全性分析

在生成令牌或者服务器处理时会产生一些重复的对象，为了描述方案的泄露函数，本文根据对象出现的顺序编码其重复模式 (repetition pattern, RP)，即赋值一个正整数表示该对象第一次出现时的位置。下面会详细讨论MM和CQ透露给敌手的内容，从初始化泄露开始。

$$\mathcal{L}_{\text{Setup}}(\text{MM}) = N \quad (5)$$

其中， N 表示由初始化算法生成的加密数据库EMM包含的键值对的数量。

$$\mathcal{L}_{\text{Token}}(\text{MM}, \text{CQ}) = (\#\text{labels}, \text{tokRP}, \text{CKRP}) \quad (6)$$

其中， $\#\text{labels}$ 是单个联合查询中包含的标签数。tokRP是令牌重复模式，任意两个共享第一个标签对的查询，对应相同的搜索令牌。(convert key RP)是转换密钥重复模式，第一个标签对相同的两个查询 Φ^i 和 Φ^j ，其中 $j \leq i$ 是最小的索引使 Φ^i 的第 s 个标签等于 Φ^j 的第 t 个标签。正式定义如下：

$$\text{CKRP}[i, s] = \begin{cases} (j, t), & j \leq i, \Phi^i[s] = \Phi^j[t] \\ (i, s), & \text{其他} \end{cases}$$

$$\mathcal{L}_{\text{Search}}(\text{MM}) = (\text{ctRP}, \{\text{MV}_i\}_{i \in [\text{CQ}]}) \quad (7)$$

其中，ctRP是密文重复模式，可由tokRP和CKRP进行推导， $\{\text{MV}_i\}_{i \in [\text{CQ}]}$ 是每个 Φ^i 对应的结果矩阵。

定理3 令 F 是安全的伪随机函数， H 是哈希函数， $\text{Gen}, \text{Enc}, \text{Dec}$ 是具有IND-CPA安全的SKE方案，TKE具有计算正确性并在自适应攻击下具有IND-CPA安全， Σ_{MM} 是具有计算正确性和 $\mathcal{L}_{\Sigma}$ 安全的EMM方案。令 $\mathcal{L}_{\text{TK-Filter}}(\text{MM}, \text{CQ}) = (\mathcal{L}_{\text{Setup}}, \mathcal{L}_{\text{Token}}, \mathcal{L}_{\text{Search}})$ ，如果只泄露如下所示的信息：

$$\mathcal{L}_{\text{Setup}}(\text{MM}) = N$$

$$\mathcal{L}_{\text{Token}}(\text{MM}, \text{CQ}) = (\#\text{labels}, \text{tokRP}, \text{CKRP})$$

$$\mathcal{L}_{\text{Search}}(\text{MM}) = (\text{ctRP}, \{\text{MV}_i\}_{i \in [\text{CQ}]})$$

那么称TK-Filter方案是自适应的 $\mathcal{L}_{\text{TK-Filter}}$ 安全的加密多映射方案。

证明 使用混合论证的方法对TK-Filter方案进行安全性证明。

游戏 $G_0 = \text{Real}_{\mathcal{A}}(\lambda)$ 执行TK-Filter方案，随后构造一系列游戏 $G_1, G_2, \dots$ 来证明定理3，每个相邻的游戏之间仅有微小的区别。并定义TK-Filter的仿真器为SIM。游戏 G_0 的输出和 $\text{Real}_{\mathcal{A}}(\lambda)$ 的输出具有相同的分布，游戏 G_3 的输出和仿真器SIM的输出具有相同的分布。根据不可区分性的传递性，可以证明 $\text{Real}_{\mathcal{A}}$ 与 $\text{Ideal}_{\mathcal{A}, \text{SIM}}$ 之间是不可区分的。

定义游戏 G_1 。游戏 G_1 与 $\text{Real}_{\mathcal{A}}(\lambda)$ 基本一致，除了 G_1 将 G_0 中使用的伪随机函数 F 替换成真正的随机函数，并维护映射表Key存储($a \parallel b, K_{a,b}^t$)对。在初始化算法中，当需要 $K_{a,b}^t$ 时，游戏首先检查Key中是否存在标签对 $a \parallel b$ 。如果存在则返回其对应的条目，否则从 $\{0, 1\}^t$ 中随机选择一个

字符串, 并将映射关系 $(a||b, K_{a,b}^i)$ 存入 Key 中。敌手无法区分随机函数 f 与伪随机函数 F , 即游戏 G_0 与 G_1 是不可区分的。对于敌手 $\mathcal{B}_1$, 以下不等式成立:

$$|\Pr[G_1 = 1] - \Pr[G_0 = 1]| \leq \text{Adv}_{\mathcal{B}_1, F}^{\text{prf}}(\lambda) \quad (8)$$

其中, $\text{Adv}_{\mathcal{B}_1, F}^{\text{prf}}(\lambda)$ 是关于 λ 的可忽略函数。

定义游戏 G_2 。游戏 G_2 与 G_1 基本一致, 除了替换 G_1 的 $\text{rnd} = \text{Enc}(K^i, \text{ctr})$ 为 $e^i \leftarrow \text{Enc}(K^i, 0^i)$, 替换 $\text{ectr} = \text{Enc}(K^p, \text{ctr})$ 为 $e^p \leftarrow \text{Enc}(K^p, 0^i)$ 。由于加密方案 Enc 具有 IND-CPA 安全, 且 0^i 的加密次数是关于 λ 的多项式次, 比如 $\text{poly}(\lambda)$ 次。则存在敌手 $\mathcal{B}_2$, 以下不等式成立:

$$|\Pr[G_2 = 1] - \Pr[G_1 = 1]| \leq \text{poly}(\lambda) \cdot$$

$$\text{Adv}_{\mathcal{B}_2, \text{SKE}}^{\text{IND-CPA}}(\lambda) \quad (9)$$

其中, $\text{Adv}_{\mathcal{B}_2, \text{SKE}}^{\text{IND-CPA}}(\lambda)$ 是关于 λ 的可忽略函数。

定义游戏 $G_3 = \text{Ideal}_{\mathcal{A}, \text{SIM}}(\lambda)$ 。游戏 G_3 中设定一个仿真器 SIM, 其输入为 $\mathcal{L}_{\text{TK-Filter}}(\Phi^i)$, 输出是一个仿真视图。如算法 11 所示, 仿真器维护 Token, CK 和 $\mathbf{C}$ 等多个映射来模拟随机预言机查询。对于每个查询 Φ^i , 仿真器根据 $\#labels[i]$ 和 CK 决定转换密钥, 根据 $|\text{ctRP}[i]|$ 决定向量 $\vec{c}$ 的大小, 并通过映射 $\mathbf{C}$ 获得密文序列。最终预言机得到结果, 由仿真器 SIM 和 G_3 生成的视图是完全无法区分的。以下等式成立:

$$\Pr[\text{Ideal}_{\mathcal{A}, \text{SIM}}(\lambda) = 1] = \Pr[G_3 = 1] \quad (10)$$

算法 11 仿真器 SIM

SIM.Setup

Input: $\mathcal{L}_{\text{Setup}}$

初始化 Token, CK 为空映射

SIM.Token

Input: $\mathcal{L}_{\text{Token}}$

If $|\text{tokRP}[i]| < i$ then

$\text{tok}_{a,b} \leftarrow \text{Token}[(a, b)]$

End if

$\text{Token}[(a, b)] \xleftarrow{\$} \{0, 1\}^\lambda$

For $s = 3 \rightarrow \#labels[i]$ do

If $\text{CKRP}[i, s] = [j, t]$ then

$\text{ck}[l_s] \leftarrow \text{ck}[a, b, \ell_t]$

Else $\text{ck}[(a, b, \ell_s)] \xleftarrow{\$} \{0, 1\}^\lambda$

End for

发送 $\{\text{tok}_{a,b}, \text{ck}[\ell_3], \dots, \text{ck}[\ell_{\#labels}]\}$ 到服务器

SIM.Search

Input: L_{Search}

If $\mathbf{C}[\text{tok}_{a,b}] = \perp$ then

$l = |\text{ctRP}[i]|$

If $l == 0$

return $\emptyset$

End if

For $j = 1 \rightarrow l$ do

$\mathbf{C}[\text{tok}_{a,b}, j] \xleftarrow{\$} \{0, 1\}^\lambda$

End for

End if

$\vec{c} \leftarrow \mathbf{C}[\text{tok}_{a,b}]$

For $j = 1 \rightarrow l$ do

$ct \leftarrow \text{MV}_i(c_j, \text{CK}[\ell_3], \dots, \text{CK}[\ell_{\#labels}])$

End for

因此, 结合以上游戏, 有

$$|\Pr[\text{Real}_{\mathcal{A}}(\lambda) = 1] - \Pr[\text{Ideal}_{\mathcal{A}, \text{SIM}}(\lambda) = 1]| \leq \text{negl}(\lambda) \quad (11)$$

5 性能分析

本节将描述实验的构建, 并从存储开销、初始化时间和搜索效率等方面对比 TK-Filter 方案和 ConjFilter 方案。为此, 实验使用 C++ 语言重现 ConjFilter 方案和实现 TK-Filter 方案的源代码, 并对客户端和服务端都使用以下机器: 配置 12 核 CPU (Intel CITM) i7-8790K, 3.70 GHz 和 64 GB 内存的计算机, 搭载 Ubuntu 20.04 操作系统。采用谷歌远程过程调用 (Google remote procedure call, gRPC) 框架实现客户端和服务端之间的网络数据传输, 同时利用 CryptoPP 开源库中的算法接口, 采用基于哈希函数的消息认证码算法 (hash-based message authentication code with secure hash algorithm 256, HMAC-SHA256) 作为 PRF, 选取 AES 作为对称加密方案。

本实验从维基百科下载数据集, 对数据集进行处理, 构建出由键值对组成的多映射结构。并编写了 Python 脚本取 10^4 到 10^6 范围的标签、值对, 以覆盖不同的数据规模。

5.1 初始化开销

为了比较两个方案的初始化开销，实验选取 10^4 、 10^5 和 10^6 级别的键值对数据集。它们分别包含547个键、4 190个键和30 654个键。实验测量针对不同规模的数据集，方案构建加密数据的初始化效率。主要衡量初始化时间和存储开销两方面，其中初始化时间表示执行键值对加密操作的时间（不含构建2-conjunction的时间），存储开销表示数据集加密后所占的存储空间。从表1可以看到，随着键值对数量的增长，初始化时间呈线性相关增长。且在 10^4 、 10^5 和 10^6 级别的数据库中，TK-Filter方案的初始化时间相较ConjFilter方案分别减少了58.4%、85.4%和86.4%。效率优化源于编程实现过程中，重加密方案用随机置换和异或操作替换了大部分的伪随机运算，并且集合 X 的初始化不需要额外的加密操作。

表1 ConjFilter和TK-Filter方案的初始化时间对比

Table 1 Setup efficiency among ConjFilter and TK-Filter schemes

方案	初始化时间/ms		
	10^4	10^5	10^6
ConjFilter	6.51	501.12	40 055
TK-Filter	2.71	73.11	5 451

在存储开销方面，实验测量的是加密多映射和集合的存储规模，统计中第一个数字代表加密多映射 EMM¹，第二个数字代表集合 X 。对于ConjFilter方案与TK-Filter方案，它们均存储键为2-conjunction形式的加密多映射结构。如表2所示，TK-Filter方案的总存储量与ConjFilter方案基本一致。在代码编程中，ConjFilter存储etag需要16个字节，ev由AES加密生成，长度

相较加密前保持不变。同样地，TK-Filter方案存储ctr需16个字节， c 由随机置换 v 生成，长度相较加密前保持不变。因此，实验数据符合理论推论。

表2 两种STE方案的实际存储对比

Table 2 Realistic storage cost among two STE schemes

方案	服务端存储/MB		
	10^4	10^5	10^6
ConjFilter	0.044 2+0.032	3.351+2.472	248.499+186.875
TK-Filter	0.044 3+0.032	3.351+2.472	248.499+186.875

5.2 搜索效率

使用数量级分别为 10^4 、 10^5 和 10^6 的数据集进行实验，实验测量服务端运行联合查询的耗时（不包括网络延时和令牌计算时间）。本节通过调整查询匹配到的值的规模来评估不同方案的搜索性能，调整匹配结果的范围为 $10 \sim 10^5$ 。从图1~图3可以看出，TK-Filter的搜索效率整体优于ConjFilter，其中横坐标代表最终匹配查询的文档数量，纵坐标代表完成搜索的时间，以毫秒为单位。在 10^4 、 10^5 和 10^6 级别的数据集中，搜索匹配文档为不同数量级时，TK-Filter方案的搜索效率相较ConjFilter方案分别提高至少41.43%、41.44%和34.78%。两个方案的搜索效率差是由于TK-Filter方案无须进行多轮AES解密操作，且用随机置换运算代替了伪随机加密操作。

6 结束语

本文提出了新的密码学原语TKE方案，并基于该密码学原语及构造提出了支持联合查询的高效STE方案TK-Filter。与ConjFilter^[9]方案相比，TK-Filter方案在完全消除交叉泄露的情况下，对于ConjFilter方案将初始化时间减少了

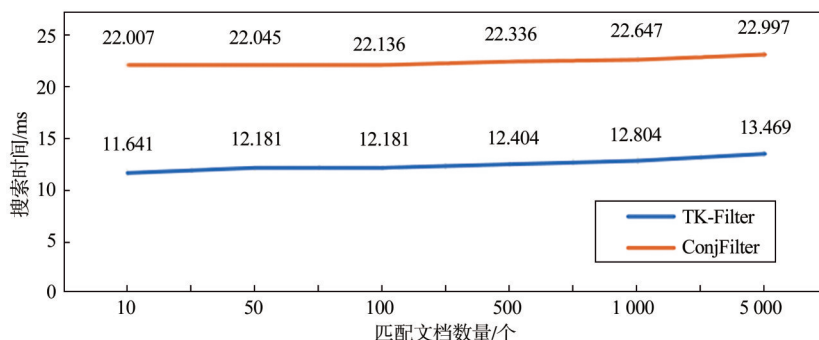


图1 10^4 级别的数据集的方案搜索效率对比

Figure 1 Search efficiency comparison in 10^4 size dataset

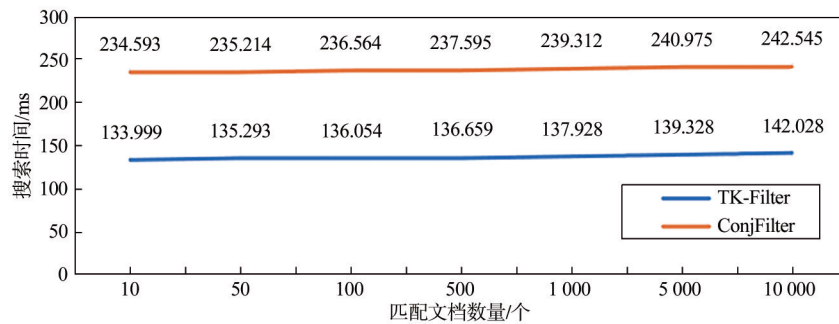


图2 10^5 级别的数据集的方案搜索效率对比
Figure 2 Search efficiency comparison in 10^5 size dataset

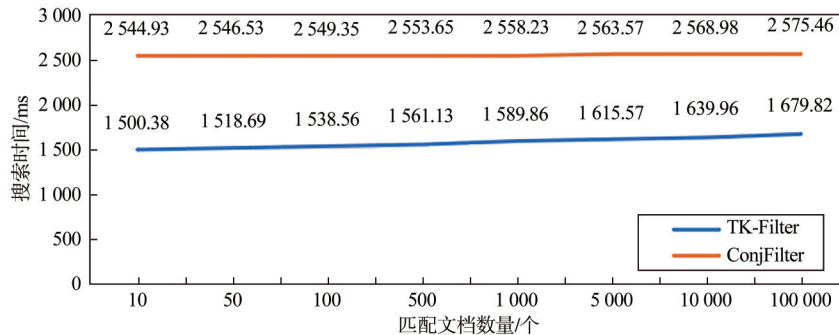


图3 10^6 级别的数据集的方案搜索效率对比
Figure 3 Search efficiency comparison in 10^6 size dataset

58%到86%，搜索效率提高了34%到41%。

消除交叉泄露的STE方案在执行联合查询时更加安全，但是此方案仍然不足以满足实际场景中添加和删除数据库条目的要求。因此，为了应对这种挑战，如何实现满足前向安全和后向安全的动态STE方案^[23,25]可作为进一步研究的方向。

参考文献:

- [1] GENTRY C. Fully homomorphic encryption using ideal lattices [C]//Proceedings of the Forty-First Annual ACM Symposium on Theory of Computing. New York: ACM, 2009: 169-178.
- [2] BONEH D, SAHAI A, WATERS B. Functional encryption: definitions and challenges[M]. ISHAI Y, ed. Theory of Cryptography. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011: 253-273.
- [3] GOLDBREICH O. Towards a theory of software protection and simulation by oblivious RAMs[C]//Proceedings of the 19th Annual ACM Symposium on Theory of Computing ' STOC '87. New York: ACM, 1987: 182-194.
- [4] BELLARE M, BOLDYREVA A, O'NEILL A. Deterministic and efficiently searchable encryption[C]//Advances in Cryptology-CRYPTO 2007: 27th Annual International Cryptology Conference, Berlin, Heidelberg: Springer Berlin Heidelberg, 2007: 535-552.
- [5] BOLDYREVA A, CHENETTE N, O'NEILL A. Order-preserving encryption revisited: improved security analysis and alternative solutions[J]. IACR Cryptol EPrint Arch, 2011, 2012: 625.
- [6] CHASE M, KAMARA S. Structured encryption and controlled disclosure[C]//Proceedings of Advances in Cryptology-ASIACRYPT 2010: 16th International Conference on the Theory and Application of Cryptology and Information Security, Berlin, Heidelberg: Springer Berlin Heidelberg, 2010: 577-594.
- [7] LI T C, WU J J, WANG J F. Efficient volume-hiding encrypted multi-maps with support for conjunctive queries[M]//ICT Systems Security and Privacy Protection. Cham: Springer International Publishing, 2022: 467-485.
- [8] GUO S X, GENG H Z, SU L, et al. A rankable Boolean searchable encryption scheme supporting dynamic updates in a cloud environment[J]. IEEE Access, 2023, 11: 63475-63486.
- [9] PATEL S, PERSIANO G, SEO J Y, et al. Efficient Boolean search over encrypted data with reduced leakage[C]//Proceedings of Advances in Cryptology-ASIACRYPT 2021: 27th International Conference on the Theory and Application of Cryptology and Information Security, Cham: Springer International Publishing, 2021: 577-607.
- [10] SONG D X, WAGNER D, PERRIG A. Practical techniques for searches on encrypted data[C]//Proceeding of 2000 IEEE Symposium on Security and Privacy. Piscataway: IEEE Press, 2000: 44-55.
- [11] JUTLA C, PATRANABIS S. Efficient searchable symmetric encryption for join queries[C]//Proceedings of International Conference on the Theory and Application of Cryptology and Information Security. Cham: Springer Nature Switzerland, 2022:

- 304-333.
- [12] MA C S, GU Y P, LI H F. Practical searchable symmetric encryption supporting conjunctive queries without keyword pair result pattern leakage[J]. IEEE Access, 2020, 8: 107510-107526.
- [13] BOST R. Σ οφος: forward secure searchable encryption[C]//Proceedings of the Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security. New York: ACM, 2016: 1143 - 1154.
- [14] ZHENG J B, ZHANG J Y, ZHANG X M, et al. Symmetric searchable encryption scheme that supports phrase search[J]. Microsystem Technologies, 2021, 27(4): 1721-1727.
- [15] GOLLE P, STADDON J, WATERS B. Secure conjunctive keyword search over encrypted data[M]. JAKOBSSON M, YUNG M, ZHOU J Y, eds. Applied Cryptography and Network Security. Berlin, Heidelberg: Springer Berlin Heidelberg, 2004: 31-45.
- [16] BALLARD L, KAMARA S, MONROSE F. Achieving efficient conjunctive keyword searches over encrypted data[M]. QINGS H, MAOW B, LÓPEZ J, alet, eds. Information and Communications Security. Berlin, Heidelberg: Springer Berlin Heidelberg, 2005: 414-426.
- [17] CASH D, JARECKI S, JUTLA C, et al. Highly-scalable searchable symmetric encryption with suppo iffie boolean queries[C]//Proceedings of Advances in Cryptology-CRYPTO 2013: 33rd Annual Cryptology Conference, Berlin, Heidelberg: Springer Berlin Heidelberg, 2013: 353-373.
- [18] KAMARA S, MOATAZ T. Boolean searchable symmetric encryption with worst-case sub-linear complexity[C]//Proceedings of Advances in Cryptology-EUROCRYPT 2017: 36th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Cham: Springer International Publishing, 2017: 94-124.
- [19] BELLARE M, DESAI A, JOKIPII E, et al. A concrete security treatment of symmetric encryption[C]//Proceedings of 38th Annual Symposium on Foundations of Computer Science. Piscataway: IEEE Press, 2002: 394-403.
- [20] JARECKI S, JUTLA C, KRAWCZYK H, et al. Outsourced symmetric private information retrieval[C]//Proceedings of the 2013 ACM SIGSAC Conference on Computer & Communications Security. 2013: 875-888.
- [21] BELLARE M, CANETTI R, KRAWCZYK H. Pseudorandom functions revisited: the cascade construction and its concrete security[C]//Proceedings of 37th Conference on Foundations of Computer Science. Piscataway: IEEE Press, 1996: 514-523.
- [22] KAMARA S, MOATAZ T, OHRIMENKO O. Structured encryption and leakage suppression[M]. SHACHAM H, BOLDYREVA A, eds. Lecture Notes in Computer Science. Cham: Springer International Publishing, 2018: 339-370.
- [23] YUAN D D, ZUO C, CUI S J, et al. Result-pattern-hiding conjunctive searchable symmetric encryption with forward and backward privacy[J]. Proceedings on Privacy Enhancing Technologies, 2023, 2023(2): 40-58.
- [24] ZHU X L, FU S J, HU H P, et al. HCV: practical multi-keyword conjunctive query with little result pattern leakage[J]. Security and Communication Networks, 2023, 2023: 8351353.
- [25] GUO C, LI W F, TANG X Y, et al. Forward private verifiable dynamic searchable symmetric encryption with efficient conjunctive query[J]. IEEE Transactions on Dependable and Secure Computing, 2024, 21(2): 746-763.
- [26] LAI S Q, PATRANABIS S, SAKZAD A, et al. Result pattern hiding searchable encryption for conjunctive queries[C]//Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security. New York: ACM, 2018: 745-762.
- [27] SYALIM A, NISHIDE T, SAKURAI K. Realizing proxy re-encryption in the symmetric world[C]//Proceedings of Informatics Engineering and Information Science: International Conference, ICIEIS 2011, Berlin, Heidelberg: Springer Berlin Heidelberg, 2011: 259-274.
- [28] SAKURAI K, NISHIDE T, SYALIM A. Improved proxy re-encryption scheme for symmetric key cryptography[C]//Proceedings of the 2017 International Workshop on Big Data and Information Security (IWBIS). Piscataway: IEEE Press, 2017: 105-111.
- [29] COOK D L, KEROMYTIS A D. Conversion and proxy functions for symmetric key ciphers[C]//Proceedings of International Symposium on Information Technology: Coding and Computing(IT-C'05)- Volume II. Piscataway: IEEE Press, 2005: 662-667.
- [30] CHEON J H. Security analysis of the iffie Diffie-Hellman problem[C]//Proceedings of the 24th Annual International Conference on the Theory and Applications of Cryptographic Techniques. New York: ACM, 2006: 1-11.

[作者简介]



杨紫依 (1998-), 女, 湖南岳阳人, 华南师范大学硕士生, 主要研究方向为可搜索加密。



马昌社 (1974-), 男, 湖南邵阳人, 华南师范大学教授, 主要研究方向为加密数据隐私计算、应用密码技术。



谢南江 (1998-), 男, 广东梅州人, 华南师范大学硕士生, 主要研究方向为密码学应用。