基于 SM9 的抗内部关键字猜测攻击的可搜索加密方案

徐嘉旺, 王化群

南京邮电大学 计算机学院 南京 中国 210023

摘要 随着云存储技术和 5G 通信的广泛应用, 云服务器成为用户节省本地存储空间和管理开销的重要手段。传统的加密技术能够有效保护云端的私密数据免受恶意敌手的攻击, 却也造成了数据检索和使用的不便。公钥可搜索加密技术允许用户在不解密密文数据的情况下进行数据检索, 既保护了数据的机密性, 又提供了高效的数据检索功能。然而, 目前大多数的公钥可搜索加密方案的设计都以国外密码体制为基础, 对以国家商用密码算法为基础的可搜索加密方案研究较少。且现有许多的公钥可搜索加密方案中, 云服务器的内部攻击者可以通过关键字猜测攻击的方式从给定的陷门中恢复出搜索关键字, 进而破坏数据机密性。为了拓展国产密码算法在公钥可搜索加密领域的应用, 以满足国产密码核心技术的自主性和安全可控性的需求。本文以国产商用密码算法 SM9 为基础, 构建了一种可认证公钥可搜索加密方案。相比于传统的公钥可搜索加密方案, 数据发送方需要对生成的关键字密文进行认证, 从而验证方确信该关键字密文只能由发送方生成。这样的设计防止了云服务器的内部攻击者通过关键字猜测攻击的方式从给定的陷门中恢复出搜索关键字, 进而破坏数据机密性。在随机预言模型下, 基于困难问题假设分别证明了本方案满足陷门不可区分性和密文不可区分性, 进而得出本方案具备抵御内部关键字猜测攻击的能力。理论分析与实验结果表明, 与经典的公钥可搜索加密方案相比, 本方案在具备较高的安全性的同时在关键字密文生成阶段也具有较高的效率。最后给出该领域的未来研究方向。

关键词 公钥可搜索加密; SM9 密码算法; 内部关键字猜测攻击; 不可区分性

中图法分类号 TP309.7 **DOI 号** 10.19363/J.cnki.cn10-1380/tn.2025.11.05

A Searchable Encryption Scheme based on SM9 for Resisting Internal Keyword Guessing Attacks

XU Jiawang, WANG Huaqun

School of Computer Science, Nanjing University of Posts and Telecommunications, Nanjing 210023, China

Abstract With the widespread application of cloud storage technology and 5G communication, cloud servers have become an important means for users to save local storage space and manage overhead costs. Traditional encryption techniques effectively protect sensitive data stored in the cloud from malicious attacks, but they also result in inconvenience in data retrieval and utilization. Public key searchable encryption technology allows users to perform data retrieval without decrypting ciphertext data, thus preserving data confidentiality while providing efficient data retrieval functionality. However, most existing public key searchable encryption schemes are based on foreign cryptographic primitives, with limited research on schemes based on domestic commercial cryptographic algorithms. In many existing public key searchable encryption schemes, internal attackers on cloud servers can recover search keywords from given trapdoors through keyword guessing attacks, thereby compromising data confidentiality. To expand the application of domestic cryptographic algorithms in the field of public key searchable encryption and meet the demand for the autonomy and security controllability of domestic cryptographic core technologies, this paper proposes a searchable encryption scheme based on the domestic commercial cryptographic algorithm SM9. In this scheme, termed as authenticated public key searchable encryption, the data sender needs to authenticate the generated keyword ciphertext, ensuring that the receiver can verify that the keyword ciphertext can only be generated by the sender. This design prevents internal attackers of cloud servers from guessing keywords through attacks, thereby compromising data confidentiality. Under the random oracle model, based on the hardness problem assumption, the scheme is proven to satisfy indistinguishability of trapdoors and ciphertexts, demonstrating its capability to resist internal keyword guessing attacks. Theoretical analysis and experimental results demonstrate that compared to classical public key searchable encryption schemes, this scheme maintains high security while exhibiting high efficiency in the generation of keyword ciphertexts. Finally, future research directions in this field are proposed.

通讯作者: 王化群, 博士, 教授, Email: whq@njupt.edu.cn。

本课题得到国家自然科学基金项目(No. U23B2002)资助。

收稿日期: 2023-12-18; 修改日期: 2024-04-26; 定稿日期: 2025-10-14

Key words public key searchable encryption; SM9 algorithm; inside keyword guessing attack; indistinguishability

1 引言

近年来, 随着云计算技术和 5G 通信的迅猛发展以及广泛应用, 云用户数量急剧增加。因此, 云存储和数据分析服务越来越向公众敞开大门, 例如亚马逊的 AWS 和谷歌的 Drive 等[1]。这些云服务平台都具备先进的云计算技术能力, 其优势包括存储空间大、计算速度快、服务可用性高、成本低等。云计算允许用户将数据托管和程序执行外包给具有更大存储、计算和网络容量的第三方, 即云服务提供商。这种模式使用户能够避免在资源受限的终端设备上进行烦琐的数据管理和存储操作, 同时提供了便捷的服务, 减少了对终端设备的需求。因此, 对于资源有限的移动云用户而言, 这提供了巨大的便利。

然而, 随之而来的是云存储面临的各种安全威胁。攻击者可以轻松窃取个人隐私数据并进行非法使用, 因此云数据的隐私保护问题引起了广泛关注。传统的加密方法可以保护数据的隐私不受恶意云服务器提供商的攻击, 但也阻止了云服务器提供商代表用户操作数据。

可搜索加密[2]是解决云存储隐私问题的有效方法。公钥可搜索加密(Public-key Encryption with Keyword Search, PEKS)[2-3]是可搜索加密的方法之一。2004 年, Boneh 等人[3]引入了 PEKS 的概念, 将使用关键字搜索密文文件的功能集成到了公钥加密体系中。他们提出了第一个基于匿名 IBE 的 PEKS 方案。对于 PEKS, 有三个参与方:一个名为 Alice 的数据拥有者、一个名为 Bob 的数据用户和一个云服务器提供商。首先, Alice 准备了一个要与 Bob 共享的文件, 并为该文件设置了关键字“encryption”。然后, Alice 将带有关键字密文的加密文件上传到云服务器提供商。为了搜索加密文件, Bob 可以使用他的私密密钥生成与关键字“w = encryption”相对应的陷门, 并使云服务器提供商能够检索与关键字 w 相关的所有密文文件。搜索完成后, 云服务器提供商将搜索结果返回给 Bob。Bob 可以确定具有所需关键字的密文文件是否包含在云服务器返回的文件中。如果包含, Bob 就可以解密加密的文件。在搜索密文文件的过程中, 云服务器提供商不知道文件的细节, 也不知道搜索的关键字。

目前, 可搜索加密方案(PEKS)得到广泛关注。在搜索模式方面, 涵盖了模糊搜索[4]、多关键字搜索[5]、聚合关键字搜索[6]等多种应用场景。在安全性能方面, 安全模型不断增强, 具备抗关键词猜测攻击[7-11]以及抗文件注入攻击[9,12]等特性。这些成果可用于云加密存储[2]、智能邮件路由[2-3]、移动电子医疗信息系统[12-13]、智能电网系统[6]、物联网[14-15]等领域。

然而, 目前绝大多数的 PEKS 设计都以国外密码体制为基础, 对以国家商用密码算法(例如 SM2、SM9)为基础的公钥可搜索加密方案的研究甚少。本项研究的目的是将国家商用密码算法应用到实际的领域中。在保证数据机密性和高效检索的同时, 拓展国家商用密码算法的实际应用以满足国产密码核心技术的自主性和安全可控性的需求。

1.1 关键字猜测攻击

2006 年, Byun 等人[16]揭示了 PEKS 体系存在显著的安全隐患, 因为关键词空间明显受限于密钥空间, 从而引发离线关键词猜测攻击的问题。在理想情况下, 一般认为关键字的空间具有超多项式的大小。然而, 由于实际应用程序中的关键字空间往往并非如此庞大, 服务器将能够在相当短的时间内完成关键字猜测攻击。因此, 攻击者发起的关键字猜测攻击(Keyword Guess Attack, KGA)对于用户的隐私造成了严重威胁。具体而言, 攻击者在获取到一个关键字所对应的陷门后, 可以通过尝试所有可能匹配的关键字, 将这些关键字挨个进行加密, 然后使用关键字陷门进行测试。当关键字匹配时, 攻击者就能确定这个陷门中封装的是哪一个关键字, 从而获取密文文件信息以及关键字信息。

通常情况下, 关键字猜测攻击是由外部敌手发起的。外部敌手如果进行攻击首先需要从公开信道上获取关键字陷门。而内部敌手本身就能获取存储在云服务器上的关键字陷门, 因此内部敌手获取的信息量大于等于外部敌手。如果关键字猜测攻击由内部敌手发起, 即由云服务器或云服务器管理内部的其他成员发起, 显然这样的攻击方式对用户隐私造成的威胁更大。这种攻击也称为内部关键字猜测攻击 (Inside Keyword Guessing Attack, IKGA) 。

为了抵御外部关键字猜测攻击, 研究人员提出在接收方和服务器之间建立安全信道, 确保只有服务器能获得关键字陷门, 或者指定测试服务器进行关键字配对。2010 年, Rhee 等人[17]提出了陷门安全的指定测试服务器下的公钥可搜索加密方案(PEKS with designed server, dPEKS), 并引入了“陷门不可区分性”的概念, 从而抵御了外部关键字猜测攻击。后来, Wang 等人[18]指出 dPEKS 方案即使在满足陷门不

可区分性的条件下, 也无法对抗来自恶意服务器的离线关键词猜测攻击, 即内部关键字猜测攻击。

近年来, 为了抵御 IKGA, 许多研究人员提出了一些变体的 PEKS 方案。Tang 等人 [19]引入了关键字注册的概念, 要求发送方提前向接收方注册关键字, 提出了一种基于注册关键词的 PEKS 方案。Xu 等人[4]提出了一种带有模糊关键字的 PEKS 方案, 通过保证每个陷门对应多个关键字来降低内部 KGA 的安全性。Wang 等人[8]给出了一种多服务器的 PEKS 方案, 将秘密信息分成多份存放在不同的服务器上, 从而在各个服务器不串通的条件下实现抵御内部关键字猜测攻击。Shao 等人[11]在 dPEKS 方案的基础上做出了改进, 引入了发送方身份。在关键字密文生成的过程中添加确定性的 RSA 签名, 使得恶意服务器即使通过猜测关键字来生成相应的密文, 也无法进行匹配测试。Huang 等人[9]提出了可认证的公钥可搜索加密方案(Public Key Authenticated Encryption With Keywords Search, PAEKS), 来抵御内部关键字猜测攻击。在该方案中, 数据拥有者需要使用其私钥来认证关键字密文, 恶意服务器在没有数据拥有者的私钥的情况下无法生成用于测试的关键字密文。因此 IKGA 不能成功攻破他们的方案。之后, Huang 等的方案被扩展到无证书 PAEKS[15,20-21]和基于身份的 PAEKS[22]。在物联网领域, 已经提出了基于 PEAKS 变体的应用[15,20,23]。Wang 等人[24]对 Huang 等所提出的 PAEKS 进行了改进, 提出了一种陷门不确定性方案, 防止关键字统计信息泄露, 抵御内部攻击者的关键字猜测攻击。Chen 等人[25]提出了在双服务器的 PEKS 方案, 将单一服务器存储密文和陷门的功能分开, 并同时能进行测试功能, 从而抵御内部关键字猜测攻击。

1.2 相关工作

近年来, 我国积极推动相关商用密码的研究, 并颁布了 GM/T 0044—2016《SM9 标识密码算法》密码行业标准[26]。 2017 年, SM9 数字签名算法成功纳入 ISO/IEC 国际标准[27]。SM9 密码算法的密钥长度为 256 比特, 采用 R-ate 双线性对, 具备快速运算和高安全性能。该算法应用嵌入度为 12 的椭圆曲线, 旨在提升安全性和计算效率。近年来, SM9 作为一种基于标识的密码算法, 在学术界和业界逐渐受到更多关注。

在进行双线性配对时, SM9 算法采用了两个安全参数较小的非对称双线性对, 与一般 PEKS 方案使用安全参数较大的对称双线性对的常规方法有所不同。因此, 将 SM9 算法与可搜索加密技术相结合的方案比较少见。但是, 国内学者在 SM9 算法的扩展应用方面已取得一些研究进展, 包括标识广播加密[28]、标识签密 [29]、属性基加密[30]、可搜索加密[31-32]等, 这些已存在的研究成果为 SM9 在公钥可搜索加密领域的发展提供了有益的借鉴。

1.3 贡献及结构

基于 SM9 标识算法, 利用 Huang 等人[9]所提出的认证加密思想, 提出了一种 PEKS 方案的变体。不同于常规 PEKS 的基于对称群, 该方案基于非对称群, 且采用 SM9 密钥生成算法和认证加密思想生成关键字密文及陷门。在确保配对一致性的同时, 实现对内部关键字猜测攻击的抵御。在随机预言模型中, 我们证明了本方案具备抵御 IKGA 的安全性能。最后, 我们对该方案的算法性能进行了深入分析, 并通过仿真实验得出了具体的时间开销。本文第 2 节主要简单描述了一些初步内容和基础知识, 包括 SM9 算法的形式化定义、双线性对的基础知识、关键字猜测攻击以及 PEKS 方案的形式化定义等; 第 3 节详细阐述了本文算法的安全模型及形式化定义; 第 4 节给出了方案的构造细节以及算法的正确性分析; 第 5 节证明了该算法满足陷门不可区分性和密文不可区分性; 第 6 节对算法进行了性能评估和实验结果分析; 第 7 节对本文的研究工作进行了总结。

2 基础知识

2.1 双线性映射

定义 1.(双线性对)双线性配对在许多密码方案的构建中起着重要作用, 本文方案也不例外。设 $(G_1,+)$ 和 $(G_2,+)$ 是两个阶为 p 的加法循环群, (G_T,g) 是阶为 p 的乘法循环群, 且 p 为大素数。P_1 是 G_1 的生成元, P_2 是 G_2 的生成元, 并存在 G_2 到 G_1 的同态映射 ψ 使得 $\psi(P_2)=P_1$。双线性运算 $\hat{e}:(G_1\times G_2)\to G_T$ 具有下列性质。

(1) 双线性: 对于任意 $P_1\in G_1,P_2\in G_2$ 和 $a,b\in Z_p$, 等式 $\hat{e}\left(aP_1,bP_2\right)=\hat{e}\left(P_1,P_2\right)^{ab}$ 成立。

(2) 非退化性: 存在 $P_1\in G_1,P_2\in G_2$, 满足 $\hat{e}\left(P_1,P_2\right)\neq 1$。

(3) 可计算性: 对于 $\forall P_1\in G_1,P_2\in G_2$, 存在有效的计算方法能够计算 $\hat{e}\left(P_1,P_2\right)$。

首先给出 DBDH 问题(Decisional Bilinear Diffie-Hellman Problem, DBDH)的定义: 在一个双线性映射 $\hat{e}\left(G_1\times G_2\right)\to G_T$ 中, 若 $P_1\in G_1$, $P_2\in G_2$, $Z\in G_T$,

对于给定的元组 $Y=(P_1,[x]P_1,[y]P_1,P_2,[y]P_2,[z]P_2,Z)$, 其中 x, y, z 为未知的随机数, Z 为 G_T 中的随机点, 区分 $\hat{e}(P_1,P_2)^{xyz}\in G_T$ 与随机元素 Z。

定义 2.(DBDH 困难问题假设)

DBDH 困难问题假设指对于任意多项式时间敌手 A, 只能以可忽略的优势区分随机点 Z 与 $\hat{e}(P_1,P_2)^{xyz}$, 即

$$|\Pr[\mathrm{A}(P_1,[x]P_1,[y]P_1,P_2,[y]P_2,[z]P_2,\hat{e}(P_1,P_2)^{xyz})=1]-\Pr[\mathrm{A}(P_1,[x]P_1,[y]P_1,P_2,[y]P_2,[z]P_2,Z)=1]|\le \mathrm{negl}(\lambda)$$

定义 3.(mDLIN 困难问题假设)

mDLIN 困难问题假设指对于任意多项式时间敌手 A 给定元组:

$Y=(P_1,[x]P_1,[y]P_1,[r/x]P_1,[sy]P_1,[z]P_1)$, 其中 P_1 为群 G_1 的生成元, $x,y,r,s,z\in Z_p$ 均为随机数, 敌手 A 只能以可忽略的优势区分 $(r+s)P_1$ 与 $[z]P_1$, 即

$$|\Pr[\mathrm{A}(P_1,[x]P_1,[y]P_1,[r/x]P_1,[sy]P_1,[r+s]P_1)=1]-\Pr[\mathrm{A}(P_1,[x]P_1,[y]P_1,[r/x]P_1,[sy]P_1,[z]P_1)=1]|\le \mathrm{negl}(\lambda)$$

2.2 公钥可搜索加密

PEKS 算法一般由 5 个概率多项式时间算法组成。

(1) Setup(λ): 使用系统安全参数 λ 作为输入, 输出全局参数 Param。

(2) KeyGen(Param): 使用系统参数 Param 作为输入, 输出用户的密钥 $(\mathrm{PK},\mathrm{SK})$。

(3) PEKS(w,PK): 使用关键字 w 和数据接收方的公钥 PK 作为输入, 输出关键字密文 C_w。

(4) Trapdoor(w',SK): 使用关键字 w' 和数据接收方私钥 SK 作为输入, 输出一个对应的关键字陷门 $T_{w'}$。

(5) Test(PK,C_w,$T_{w'}$): 云服务器使用数据接收方的公钥 PK, 关键字密文 C_w, 关键字陷门 $T_{w'}$ 作为输入, 如果密文 C_w 与陷门 $T_{w'}$ 中包含相同的关键字则输出 1 并返回对应的密文文件, 否则输出 0。

2.3 SM9 密码算法

SM9 密码算法是我国自主研发的基于标识的密码算法。SM9 标识密码算法目前在安全电子邮件、物联网、医疗卫生中都有重要的应用。SM9 密码算法主要包含了数字签名、密钥交换协议、密钥封装协议以及加密算法四个部分[32]。这里主要详细介绍 SM9 加密算法的组成部分。

(1) Setup(λ): 输入安全参数 λ, 密钥生成中心 (Key Generation Center, KGC) 选取双线性对群 $\mathrm{BP}=(G_1,G_2,G_T,e,N)$, 其中 $N>2^{\lambda}$, e 为双线性对映射 $(G_1\times G_2)\to G_T$。选取 P_1 为 G_1 的随机生成元, P_2 是 G_2 的随机生成元, 哈希函数 $H_1:\{0,1\}^*\to Z_N^*$。密钥派生函数 KDF(Z,klen): 输入比特串 Z, 整数 klen, 输出长度为 klen 的密钥数据比特串 K。消息认证码 MAC(K_2,Z): 输入比特串 K_2, 消息比特串 Z, 输出消息认证码数据比特串 K_1。KGC 选择随机数 $\mathrm{ke}\in[1,N-1]$ 作为加密主私钥, 使用加密主私钥计算 $P_{\text{pub-e}}=[\mathrm{ke}]P_1$ 作为加密主公钥, 得到加密主密钥对为 $(\mathrm{ke},P_{\text{pub-e}})$。KGC 计算 $g=\hat{e}(P_{\text{pub-e}},P_2)$, 并生成加密私钥生成函数识别符 hid。最后, 输出公开参数:

$$\mathrm{pp}=(\mathrm{BP},g,P_1,P_2,H_1,P_{\text{pub-e}},\mathrm{hid},\mathrm{KDF},\mathrm{MAC})。$$

(2) KeyGen(pp,ID): 密钥生成函数。KGC 首先计算 $t_1=H_1(\mathrm{ID}\|\mathrm{hid},N)+\mathrm{ke}$, 如果 $t_1=0$ 则重新选取 ke, 使用重新选取的 ke 计算 $P_{\text{pub-e}}$, 并对系统中已有的用户进行密钥更新; 否则有 $t_2=\mathrm{ke}\cdot t_1^{-1}$, 然后计算 $d_{\mathrm{ID}}=[t_2]P_2$。

(3) Encrypt(ID,m,$P_{\text{pub-e}}$): 使用用户 ID, 明文 m, 系统主公钥 $P_{\text{pub-e}}$。计算 $Q_{\mathrm{ID}}=[H_1(\mathrm{ID}\|\mathrm{hid},N)]P_1+P_{\text{pub-e}}$, 选取随机数 $r\in[1,N-1]$ 并计算 $C_1=[r]Q_{\mathrm{ID}}$, $u=g^r$, $K=\mathrm{KDF}(C_1\|u\|\mathrm{ID},\mathrm{klen})=(K_1,K_2)$, $C_2=K_1\oplus m$, $C_3=\mathrm{MAC}(K_2,C_2)$, 输出 m 的密文 $C=(C_1,C_2,C_3)$。

(4) Decrypt(ID,C,d_{ID}): 使用密文 C 和用户私钥 d_{ID}, 计算 $u'=\hat{e}(C_1,d_{\mathrm{ID}}),K'=\mathrm{KDF}(C_1\|u'\|ID,\mathrm{klen})=(K_1',K_2')$, $C_3'=\mathrm{MAC}(K_2',C_2)$。若 $C_3'=C_3$, 则输出明文 m, 否则输出 $\perp$。

3 形式化定义和安全模型

本文的系统模型如图 1 所示, 其中数据拥有者(发送方)负责提取并加密关键字, 数据使用者(接收方)生成关键字陷门。云服务器负责存储加密后的密文文件和提取的关键字密文, 并对数据使用者提交的关键字陷门进行检索操作。

3.1 算法形式化定义

本节给出算法的形式化定义, 具体构造如下。

(1) Setup(λ): 以安全参数 λ 为输入, 输出系统全局参数 Params。

(2) KeyGen(Params): 输入系统全局参数 Params, 输出用户的公私钥对 (PK,SK)。

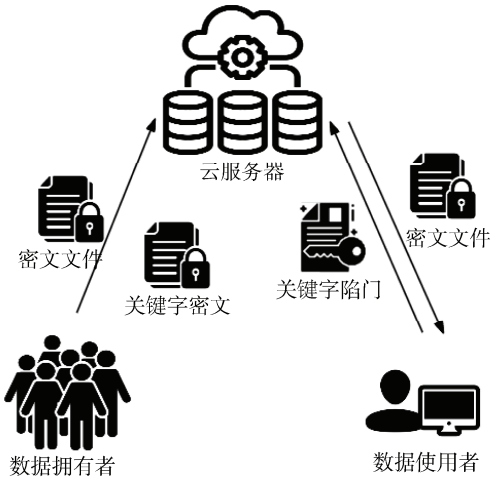

图 1　系统模型

Figure 1　System model

(3) Index(w,SK_S,PK_R,Params)：数据发送方输入其私钥SK_S，提取出的关键字w，数据接收方的公钥PK_R，输出关键字w对应的密文C_w。

(4) Trapdoor(w',PK_S,SK_R,Params)：数据接收方输入其私钥SK_R，数据发送方的公钥PK_S，以及关键字w'，输出w'对应的关键字陷门$T_{w'}$。

(5) Search($T_{w'}$,C_w)：以关键字w'的搜索陷门$T_{w'}$，关键字w对应的关键字密文C_w作为输入，如果C_w和$T_{w'}$含有相同的关键字则算法输出 1，否则输出 0。

系统建立算法以可信的方式运行，使得系统中的每个人都信任这些参数。数据发送方和数据接收方各自运行一次密钥生成算法，获取自己的公私钥对。对关键字w进行加密时，数据发送方通过 Index 算法生成对应的密文C,然后将包含关键字w的加密文档以及密文C一起上传到云服务器。当数据接收方需要对发送方共享的密文进行搜索操作时，数据接收方通过陷门生成算法 Trapdoor 为其中某一个关键字w生成关键字陷门，并将陷门通过安全信道传递给云服务器。云服务器对于给定的陷门，运行 Search 算法，搜索数据发送方和数据接收方之间共享的密文，并且将搜索结果返回给接收方。

3.2　安全模型

在 PEKS 方案中，存在外部敌手以及云服务器内部敌手的潜在威胁。外部敌手的攻击可以通过建立安全信道或者授权指定服务器进行测试来抵抗，而且内部敌手的威胁等级高于外部敌手。因此当本文方案能够抵御来自云服务器内部敌手的攻击时也就能抵御外部敌手的关键字猜测攻击。本文方案的安全性要求不存在概率多项式敌手可以正确区分陷门或密文。我们通过攻击者$\mathcal{A}$和挑战者$\mathcal{C}$之间的游戏来定义方案的密文不可区分性和陷门不可区分性。

游戏 1：陷门不可区分性

陷门不可区分性的目的在于防止半诚实的服务器敌手$\mathcal{A}$#从提供的陷门中获取关键字信息。这确保了服务器无法生成有效的密文索引，从而测试获取有用信息。

(1) 在给定安全参数的情况下，挑战者$\mathcal{C}$生成全局系统参数 Params、并准备挑战发送方的公钥PK_S和挑战接收方的公钥PK_R，并且将 Params、准备挑战的挑战发送方的公钥PK_S以及挑战接收方的公钥PK_R发送给敌手$\mathcal{A}$。

(2) 敌手$\mathcal{A}$可以自适应地对如下预言机发出询问，质询次数由多项式时间决定。陷门预言机$\mathcal{O}_T$：给定数据发送方的公钥PK_S和关键字w，$\mathcal{O}_T$预言机使用接收方的私钥计算出对应的陷门T_w，并且将陷门发送给敌手$\mathcal{A}$。关键字密文预言机$\mathcal{O}_C$：给定数据接收方的公钥PK_R和关键字w，$\mathcal{O}_C$预言机使用发送方的私钥计算出对应的密文C_w，并且将关键字密文发送给敌手$\mathcal{A}$。

(3) 敌手$\mathcal{A}$选择两个关键字(w_0^*, w_1^*)作为挑战关键字发送给挑战者$\mathcal{C}$，且这两个关键字在这之前未询问过$\mathcal{O}_C$和$\mathcal{O}_T$。挑战者$\mathcal{C}$#随机选择比特$b \in \{0,1\}$，计算关键字w_b^*的陷门$T_{w_b^*} \leftarrow$ Trapdoor(w',PK_S,SK_R, Params)并返回给敌手$\mathcal{A}$。

(4) 敌手$\mathcal{A}$可以继续访问$\mathcal{O}_C$和$\mathcal{O}_T$预言机，但要求不能就关键字(w_0^*, w_1^*)进行询问。

(5) 最终，敌手$\mathcal{A}$输出$b' \in \{0,1\}$，当且仅当$b'=b$时，敌手$\mathcal{A}$赢得游戏。定义敌手$\mathcal{A}$赢得陷门不可区分性游戏的优势为

$$\mathrm{Adv}_A^T = \left|\Pr[b'=b] - 0.5\right|$$

游戏 2：密文不可区分性

与游戏 1 相同，密文不可区分性旨在攻击者$\mathcal{A}$在不知道数据发送方和数据接收方的私钥的情况下，$\mathcal{A}$无法区分给定的密文是对两个关键字($\mathcal{A}$自己选择)中的哪一个进行加密的结果。

(1) 同游戏 1，在给定安全参数的情况下，挑战者$\mathcal{C}$生成全局系统参数 Params、并准备挑战发送方公钥PK_S和挑战接收方的公钥PK_R，并且将*Params*、准备挑战的挑战发送方的公钥PK_S以及挑战接收方的公钥PK_R发送给敌手$\mathcal{A}$。

(2) 同游戏 1, 敌手 $\mathcal{A}$ 可以就关键字对陷门预言机 $\mathcal{O}_T$ 和 $\mathcal{O}_C$ 预言机进行质询。

(3) 敌手 $\mathcal{A}$ 选择两个关键字 (w_0^*, w_1^*) 作为挑战关键字发送给挑战者 $\mathcal{C}$, 且这两个关键字在这之前未询问过 $\mathcal{O}_C$ 和 $\mathcal{O}_T$。挑战者 $\mathcal{C}$ 随机选择 $b \in \{0,1\}$, 计算关键字 w_b^* 的对应密文 $C_{w_b^*} \leftarrow \text{Index}(w, \text{SK}_S, \text{PK}_R, \text{Params})$ 并返回给敌手 $\mathcal{A}$。

(4) 敌手 $\mathcal{A}$ 可以继续访问 $\mathcal{O}_C$ 和 $\mathcal{O}_T$ 预言机, 但要求不能就关键字 (w_0^*, w_1^*) 进行询问。

(5) 最终, 敌手 $\mathcal{A}$ 输出 $b' \in \{0,1\}$, 如果 $b' = b$, 则敌手 $\mathcal{A}$ 赢得游戏。我们定义敌手 $\mathcal{A}$ 赢得密文不可区分性游戏的优势为

$$\text{Adv}_A^C = \left|\Pr[b' = b] - 0.5\right|$$

定义 4. 如果敌手 $\mathcal{A}$ 在多项式时间内 Adv_A^T 和 Adv_A^C 的优势是可以忽略的, 则称该方案是内部关键字猜测攻击语义安全的。

4 具体方案

本文提出一个基于 SM9 且能抵抗内部关键词猜测攻击的方案, 方案具体算法包括系统建立算法 Setup、用户密钥生成算法 KeyGen、关键字密文索引生成算法 Index、陷门生成算法 Trapdoor 和搜索算法 Search。

1) Setup(λ)

全局参数生成算法, 选择一个循环群上的双线性对映射 $\hat{e}(G_1, G_2) \to G_T$, G_1, G_2 和 G_T 的阶均为大素数 q。P_1 为循环群 G_1 的随机生成元, P_2 为循环群 G_2 的随机生成元。选取哈希函数 $H: \{0,1\}^* \to G_1$, $H_1: \{0,1\}^* \to Z_q^*$。系统的公共参数 $\text{Params} = (G_1, G_2, G_T, P_1, P_2, \hat{e}, N, H, H_1)$。

2) KeyGen(Params)

数据发送方随机选取 $z \leftarrow Z_q$, 设置公钥 $\text{PK}_S = zP_2$, 私钥 $\text{SK}_S = z$, 返回 $(\text{PK}_S, \text{SK}_S)$。

3) KeyGen(Params)

数据接收方随机选取 $y \leftarrow Z_q$, 设置公钥 $\text{PK}_R = yP_1$, 私钥 $\text{SK}_R = y$, 返回 $(\text{PK}_R, \text{SK}_R)$。

4) Index($w, \text{SK}_S, \text{PK}_R, \text{Params}$)

关键字密文索引生成算法, 数据发送方使用其私钥 SK_S, 数据接收方的公钥 PK_R, 生成 w 对应的密文搜索索引 C_w, 将密文信息发送给云服务器, 其中有 $C_1 = \text{SK}_S H(\text{PK}_S, \text{PK}_R, w) + r\text{SK}_S Q_w$, $C_2 = r\text{PK}_R$。其中 $Q_w = H_1(w)P_1 + PK_R$。

5) Trapdoor($w', \text{PK}_S, \text{SK}_R$)

关键字陷门生成算法, 数据接收方使用其私钥 SK_R、数据发送方公钥 PK_S, 生成关键字 w' 对应的陷门 $T_{w'}$。首先随机选择随机数 $\theta \in [1, q-1]$,然后计算陷门信息 $T_{w'} = (T_1, T_2, T_3)$, 并将陷门发送给云服务器:

$T_1 = \hat{e}\left(\text{SK}_R (H_1(w') + \text{SK}_R)^{-1} H(\text{PK}_S, \text{PK}_R, w'), \theta \text{PK}_S\right)$,

$T_2 = \theta d_{w'}$, $T_3 = \theta \text{PK}_S$。

其中 $d_{w'} = \text{SK}_R (H_1(w') + \text{SK}_R)^{-1} P_2$。

6) Search($T_{w'}, C_w$)

搜索算法, 服务器运行该算法来测试接收到的陷门信息与存储的关键字密文索引信息是否能正确匹配。计算 $\hat{e}(C_1, T_2)$, $\hat{e}(C_2, T_3)$, 验证 $T_1 \cdot \hat{e}(C_2, T_3) = \hat{e}(C_1, T_2)$, 若等式成立说明验证通过, 输出 1, 并且返回密文文件; 否则输出 0。正确性证明:

$$\hat{e}(C_1, T_2) = \hat{e}(\text{SK}_S H(\text{PK}_S, \text{PK}_R, w) + r\text{SK}_S Q_w, \theta d_{w'})$$

$$= \hat{e}(\text{SK}_S H(\text{PK}_S, \text{PK}_R, w), \theta d_{w'}) \cdot \hat{e}(r\text{SK}_S Q_w, \theta d_{w'})$$

如果有 $w = w'$ 则

$$= \hat{e}(\text{SK}_S H(\text{PK}_S, \text{PK}_R, w), \theta d_w) \cdot \hat{e}(P_1, P_2)^{\theta r \text{SK}_R \text{SK}_S}$$

$$= \hat{e}(\text{SK}_S H(\text{PK}_S, \text{PK}_R, w), \theta d_w) \cdot \hat{e}(P_1, P_2)^{\theta r \text{SK}_R \text{SK}_S}$$

$$= \hat{e}\left(\text{SK}_S H(\text{PK}_S, \text{PK}_R, w), \theta \text{SK}_R (H_1(w) + \text{SK}_R)^{-1} P_2\right)$$

$$\cdot \hat{e}(P_1, P_2)^{\theta r \text{SK}_R \text{SK}_S}$$

$$= \hat{e}\left(\text{SK}_R (H_1(w) + \text{SK}_R)^{-1} H(\text{PK}_S, \text{PK}_R, w), \theta \text{PK}_S\right)$$

$$\cdot \hat{e}(P_1, P_2)^{\theta r \text{SK}_R \text{SK}_S}$$

$$= T_1 \cdot \hat{e}(P_1, P_2)^{\theta r \text{SK}_R \text{SK}_S} = T_1 \cdot \hat{e}(C_2, T_3)$$

因此, 本文的方案满足正确性要求。

5 方案分析与证明

在本节中, 我们给出方案具体的安全性证明。

5.1 陷门不可区分性

定理 1: 对于任何多项式时间算法敌手 $\mathcal{A}$, 如果 DBDH 假设成立, 则其能正确区分关键字陷门的概率优势 Adv_A^T 是可以忽略不计的。

证明: 假设存在多项式时间敌手 $\mathcal{A}$ #, 他以不可忽略的优势 ϵ_T 区分了本文方案的陷门, 用他来构建一个概率多项式时间算法 $\mathcal{B}$ 来解决 DBDH 问题。即

给定实例元组: $\{G_1,G_2,G_T,\hat{e},P_1,[x]P_1,[y]P_1,P_2,[y]P_2,[z]P_2,Z\}$。算法 $\mathcal{B}$ 要判断 Z 是否等于 $\hat{e}(P_1,P_2)^{xyz}$。其中 Z 的值取决于挑战者选择的随机比特 b, 当 $b=0$ 时, $Z=\hat{e}(P_1,P_2)^{xyz}$。当 $b=1$ 时, Z 是群 G_T 中随机点。$\mathcal{B}$ 设置全局参数 $\text{Params}=(G_1,G_2,G_T,P_1,P_2,\hat{e},p,H,H_1)$。并且生成挑战发送者的公钥 $\text{PK}_S=zP_2$, 私钥 $\text{SK}_S=z$, 以及接收者的公钥 $\text{PK}_R=yP_1$, 私钥 $\text{SK}_R=y$。将 $(\text{Params},\text{PK}_S,\text{PK}_R)$ 发送给敌手 $\mathcal{A}$。然后算法 $\mathcal{B}$ 开始回答敌手 $\mathcal{A}$ 对下列预言机的质询。为了简单起见, 我们做出如下假设。

(1) 敌手 $\mathcal{A}$ 最多对哈希预言机 $\mathcal{O}_H$,陷门预言机 $\mathcal{O}_T$, 密文预言机 $\mathcal{O}_C$ 分别进行 q_H,q_T,q_C 次质询。

(2) 敌手 $\mathcal{A}$ 不能对某一个预言机进行相同的询问。

(3) 敌手 $\mathcal{A}$ 在向哈希预言机 $\mathcal{O}_H$ 发出问询之前不会询问预言机 $\mathcal{O}_T$ 和预言机 $\mathcal{O}_C$。

算法 $\mathcal{B}$ 模拟预言机的详情如下。

(1) 哈希预言机 $\mathcal{O}_H$: 算法 $\mathcal{B}$ 维护一个列表 L_H, 列表用于存放元组 $\langle(\widetilde{\text{PK}_S},\widetilde{\text{PK}_R},w_i),h_i,a_i,c_i\rangle$。给定一个元组: $(\widetilde{\text{PK}_S},\widetilde{\text{PK}_R},w_i)$, 算法 $\mathcal{B}$ 随机选择 $a_i\leftarrow Z_p$, 此外以概率 $\Pr[c_i=0]=\delta$ 选取 $c_i\in\{0,1\}$。如果 $c_i=0$, 算法 $\mathcal{B}$ 令 $h_i=xP_1+a_iP_1\in G_1$; 否则令 $h_i=a_iP_1\in G_1$。$\mathcal{B}$ 将所得元组 $\langle(\widetilde{\text{PK}_S},\widetilde{\text{PK}_R},w_i),h_i,a_i,c_i\rangle$ 加入到列表 L_H 中去, 并且将 $h_i=H(\widetilde{\text{PK}_S},\widetilde{\text{PK}_R},w_i)$ 作为 $(\widetilde{\text{PK}_S},\widetilde{\text{PK}_R},w_i)$ 的哈希结果返回给敌手 $\mathcal{A}$。

(2) 陷门预言机 $\mathcal{O}_T$: 给定 $(\widetilde{\text{PK}_S},w_i)$, 算法 $\mathcal{B}$ 首先选定 $\theta_i\leftarrow Z_p$, 并从列表 L_H 中查找元素 $\langle(\widetilde{\text{PK}_S},\widetilde{\text{PK}_R},w_i),h_i,a_i,c_i\rangle$。若 $c_i=0$, 算法终止并输出一个随机比特 b' 作为对 b 的猜测值。否则算法 $\mathcal{B}$ 计算 $T_{w_i}=(T_{w_{i,1}},T_{w_{i,2}},T_{w_{i,3}})$, $T_{w_{i,1}}=\hat{e}\left((H_1(w_i)+\text{SK}_R)^{-1}\text{PK}_R,\theta\widetilde{\text{PK}_S}\right)^{a_i}$, $T_{w_{i,2}}=\theta\text{SK}_R(H_1(w_i)+\text{SK}_R)^{-1}P_2$, $T_{w_{i,3}}=\theta\text{PK}_R$。并将 T_{w_i} 发送给敌手 $\mathcal{A}$。

(3) 密文预言机 $\mathcal{O}_C$: 给定 $(\widetilde{\text{PK}_R},w_i)$, 算法 $\mathcal{B}$ 首先选定 $r_i\leftarrow Z_p$, 并从列表 L_H 中查找元素 $\langle(\widetilde{\text{PK}_S},\widetilde{\text{PK}_R},w_i),h_i,a_i,c_i\rangle$。若 $c_i=0$, 算法终止并输出一个随机比特 b' 作为对 b 的猜测值。否则计算密文 $C_{w_i}=(C_{w_1},C_{w_2})=(a_i\text{SK}_SP_1+r_iQ_{w_i},r_i\widetilde{\text{PK}_R})$。其中 $Q_{w_i}=H_1(w_i)P_1+\widetilde{\text{PK}_R}$, 将 C_{w_i} 发送给敌手 $\mathcal{A}$。

在某一时刻, 敌手 $\mathcal{A}$ 选择两个未查询过 $\mathcal{O}_T$ 和 $\mathcal{O}_C$ 预言机的挑战关键词 w_0^*,w_1^* 发送给算法 $\mathcal{B}$。$\mathcal{B}$ 从 L_H 中取回关键字 w_0^*,w_1^* 对应的元组 $\langle(\widetilde{\text{PK}_S},\widetilde{\text{PK}_R},w_0^*),h_0^*,a_0^*,c_0^*\rangle$, $\langle(\widetilde{\text{PK}_S},\widetilde{\text{PK}_R},w_1^*),h_1^*,a_1^*,c_1^*\rangle$。并按照下述规则计算挑战陷门。

当 $c_0^*=c_1^*=1$ 时, 算法 $\mathcal{B}$ 终止并输出一个随机比特 $b'\in\{0,1\}$, 作为对 b 的猜测。当 c_0^* 或者 c_1^* 中有一个值为 0 时, 假设 $c_{b'}^*=0$。算法 $\mathcal{B}$ 计算挑战陷门 $T_1^*=Z\cdot\hat{e}(yP_1,\theta zP_2)^{a_{b'}^*(H_1(w_i)+Sk_R)^{-1}}$。如果有 $Z=\hat{e}(P_1,P_2)^{xyz}$, 则可以得到挑战陷门 $T_1^*=\hat{e}(P_1,P_2)^{(x+\theta a_{b'}^*(H_1(w_i)+\text{SK}_R)^{-1})yz}=\hat{e}\left((x+\theta a_{b'}^*(H_1(w_i)+\text{SK}_R)^{-1})P_1,yzP_2\right)$。如果 Z 为 G_T 中的随机元素, 那么 T_1^* 同样也是 G_T 中的随机元素。

算法 $\mathcal{B}$ 将挑战陷门 T^* 发送给敌手 $\mathcal{A}$, 敌手可以继续访问 $\mathcal{O}_T$ 和 $\mathcal{O}_C$, 但不能使用关键词 w_0^*,w_1^* 对 $\mathcal{O}_T$ 和 $\mathcal{O}_C$ 进行询问。最终, 敌手 $\mathcal{A}$ 输出猜测结果 $\hat{b}'$。如果 $\hat{b}'=b'$, 算法 $\mathcal{B}$ 输出 $b'=0$; 否则输出 $b'=1$。

使用 abt 来表示算法 $\mathcal{B}$ 终止游戏的两种情况。

(1) 算法 $\mathcal{B}$ 在模拟陷门预言机 $\mathcal{O}_T$ 和密文预言机 $\mathcal{O}_C$ 时。存在 $c_i=0$ 的情况, 此时算法 $\mathcal{B}$ 不终止游戏的概率为 $\Pr[\overline{\text{abt}_1}]=(1-\delta)^{q_T+q_C}$。

(2) 在生成挑战关键字陷门时, 有 $c_0^*=c_1^*=1$, 此时算法 $\mathcal{B}$ 不终止游戏的概率为 $\Pr[\overline{\text{abt}_2}]=1-(1-\delta)^2$。由此可得算法 $\mathcal{B}$ 不终止游戏的概率:

$$\Pr[\overline{\text{abt}}]=\Pr[\overline{\text{abt}_1}]\Pr[\overline{\text{abt}_2}]=(1-\delta)^{q_T+q_C}(1-(1-\delta)^2)$$

当 $\delta=1-\sqrt{\dfrac{q_T+q_C}{q_T+q_C+2}}$ 时, 算法 $\mathcal{B}$ 不终止游戏的概率为 $\dfrac{2}{e(q_T+q_C)}$。

如果游戏没有终止, 那么敌手 $\mathcal{A}$ 的意图可以视为真实攻击。在游戏不终止的前提下, 如果敌手 $\mathcal{A}$ 成功破解本方案的陷门不可区分性, 那么算法 $\mathcal{B}$ 成功猜对比特 b 的概率为

$$\begin{aligned}\Pr[b'=b]&=\Pr[b'=b\wedge\text{abt}]+\Pr[b'=b\wedge\overline{\text{abt}}]\\&=\Pr[b'=b|\text{abt}]\Pr[\text{abt}]+\Pr[b'=b\,|\,\overline{\text{abt}}]\Pr[\overline{\text{abt}}]\\&=\frac{1}{2}(1-\Pr[\overline{\text{abt}}])+\left(\varepsilon_T+\frac{1}{2}\right)\Pr[\overline{\text{abt}}]\\&=\frac{1}{2}+\varepsilon_T\Pr[\overline{\text{abt}}]\end{aligned}$$

显然 ε_T 和 $\Pr[\overline{\text{abt}}]$ 均是不可忽略的，那么 $|\Pr[b'=b]-1/2|$ 也是不可忽略的，即算法 $\mathcal{B}$ 能以不可忽略的概率优势解决 DBDH 问题。与定义 2 矛盾，从而本文方案满足陷门不可区分性。

5.2 密文不可区分性

定理 2: 对于任何多项式时间算法敌手 $\mathcal{A}$，如果 mDLIN 假设成立，则其能正确区分关键字密文的概率优势 Adv_A^C 是可以忽略不计的。

证明: 假设存在多项式时间敌手 $\mathcal{A}$，他以不可忽略的优势 ϵ_C 区分了本文方案的密文，用他来构建一个概率多项式时间算法 $\mathcal{B}$ 来解决 mDLIN 问题。即给定实例元组: $\{G_1,G_2,G_T,\hat{e},p,P_1,[x]P_1,[y]P_1,[rx]P_1,[s/y]P_1,Z\}$，其中 x,y,r,s 均为 Z_q 中的随机数。算法 $\mathcal{B}$ 要判断 Z 的值是否等于 $(r+s)P_1$ 还是群 G_1 中的随机点。其中 Z 的值取决于挑战者选择的随机数 b，当 $b=0$ 时，$Z=(r+s)P_1$。当 $b=1$ 时，Z 是群 G_1 中随机点。算法 $\mathcal{B}$ 首先要设置全局参数 $\text{Params}=(G_1,G_2,G_T,P_1,P_2,\hat{e},p,H,H_1)$。并且生成挑战发送者的公钥 $\text{PK}_S=yP_2$，私钥 $\text{SK}_S=y$，以及接收者的公钥 $\text{PK}_R=xP_1$，私钥 $\text{SK}_R=x$。将 $(\text{Params},\text{PK}_S,\text{PK}_R)$ 发送给敌手 $\mathcal{A}$。然后算法 $\mathcal{B}$ 开始回答敌手 $\mathcal{A}$ 对下列预言机的质询。为了简单起见，我们做出假设与定理 1 中相同。算法 $\mathcal{B}$ 模拟预言机的详情如下。

(1) 哈希预言机 $\mathcal{O}_H$: 算法 $\mathcal{B}$ 维护一个列表 L_H，列表用于存放元组 $\langle(\widetilde{\text{PK}_S},\widetilde{\text{PK}_R},w_i),h_i,a_i,c_i\rangle$。给定一个元组: $(\widetilde{\text{PK}_S},\widetilde{\text{PK}_R},w_i)$，算法 $\mathcal{B}$ 随机选择 $a_i\leftarrow Z_p$，此外以概率 $\Pr[c_i=0]=\delta$ 选取 $c_i\in\{0,1\}$。如果 $c_i=0$，算法 $\mathcal{B}$ 令 $h_i=(s/y)P_1+a_iP_1\in G_1$；否则令 $h_i=a_iP_1\in G_1$。算法 $\mathcal{B}$ 将所得元组 $\langle(\widetilde{\text{PK}_S},\widetilde{\text{PK}_R},w_i),h_i,a_i,c_i\rangle$ 加入到列表 L_H 中去，并且将 $h_i=H(\widetilde{\text{PK}_S},\widetilde{\text{PK}_R},w_i)$ 作为 $(\widetilde{\text{PK}_S},\widetilde{\text{PK}_R},w_i)$ 的哈希结果返回给敌手 $\mathcal{A}$。

(2) 陷门预言机 $\mathcal{O}_T$: 同定理 1 中的陷门预言机。

(3) 密文预言机 $\mathcal{O}_C$: 同定理 1 中的密文预言机。

在某一时刻，敌手 $\mathcal{A}$ 选择两个未查询过 $\mathcal{O}_T$ 和 $\mathcal{O}_C$ 预言机的挑战关键词 w_0^*,w_1^* 发送给算法 $\mathcal{B}$。$\mathcal{B}$ 从 L_H 中取回关键字 w_0^*,w_1^* 对应的元组 $\langle(\widetilde{\text{PK}_S},\widetilde{\text{PK}_R},w_0^*),h_0^*,a_0^*,c_0^*\rangle,\langle(\widetilde{\text{PK}_S},\widetilde{\text{PK}_R},w_1^*),h_1^*,a_1^*,c_1^*\rangle$。并按照下述规则计算挑战密文:

当 $c_0^*=c_1^*=1$ 时，算法 $\mathcal{B}$ 终止并输出一个随机比特 $b'\in\{0,1\}$ 作为对 b 的猜测。当 c_0^* 或者 c_1^* 中有一个值为 0 时，假设 $c_{b'}^*=0$。此时有 $h_{b'}^*=((s+ya_{b'}^*)/y)P_1$ 算法 $\mathcal{B}$ 计算密文 $C^*=(C_1^*,C_2^*),C_1^*=Z+(a_{b'}^*+ya_{b'}^*)P_1+(rxt_1)P_1$，$C_2^*=xrP_1+xa_{b'}^*P_1$。其中 $t_1=H_1(w)+x$ 若 $Z=(r+s)P_1$，则 $C_1^*=h_{b'}^{*\,y}+(r+a_{b'}^*+rxt_1)P_1$，$C_2^*=x(r+a_{b'}^*)P_1$。对于敌手而言 $r+a_{b'}^*$ 是随机值。当 Z 是随机元素时，C^* 对于敌手也是随机的。

算法 $\mathcal{B}$ 将挑战密文 C^* 发送给敌手 $\mathcal{A}$，敌手可以继续访问 $\mathcal{O}_T$ 和 $\mathcal{O}_C$，但不能使用关键词 w_0^*,w_1^* 对 $\mathcal{O}_T$ 和 $\mathcal{O}_C$ 进行询问。最终，敌手 $\mathcal{A}$ 输出猜测结果 $\overline{b'}$。如果 $\widehat{b'}=b'$，算法 $\mathcal{B}$ 输出 $b'=0$；否则输出 $b'=1$。

使用 abt 来表示算法 $\mathcal{B}$ 终止游戏的两种情况。算法 $\mathcal{B}$ 不终止游戏的概率和定理 1 中的证明相同。所以当 $\delta=1-\sqrt{\dfrac{q_T+q_C}{q_T+q_C+2}}$ 时，算法 $\mathcal{B}$ 不中止游戏的概率为 $2/e(q_T+q_C)$。

如果游戏没有终止，那么敌手 $\mathcal{A}$ 的意图可以视为真实攻击。在游戏不终止的前提下，如果敌手 $\mathcal{A}$ 成功破解本方案的密文不可区分性，那么算法 $\mathcal{B}$ 成功猜对比特 b 的概率为

$$\begin{aligned}\Pr[b'=b]&=\Pr[b'=b\wedge\text{abt}]+\Pr[b'=b\wedge\overline{\text{abt}}]\\&=\Pr[b'=b|\text{abt}]\Pr[\text{abt}]+\Pr[b'=b\,|\,\overline{\text{abt}}]\Pr[\overline{\text{abt}}]\\&=\frac{1}{2}\left(1-\Pr[\overline{\text{abt}}]\right)+\left(\varepsilon_C+\frac{1}{2}\right)\Pr[\overline{\text{abt}}]\\&=\frac{1}{2}+\varepsilon_C\Pr[\overline{\text{abt}}]\end{aligned}$$

显然 ε_C 和 $\Pr[\overline{\text{abt}}]$ 均是不可忽略的，那么 $|\Pr[b'=b]-1/2|$ 也是不可忽略的，即算法 $\mathcal{B}$ 能以不可忽略的概率优势解决 mDLIN 问题。与定义 3 矛盾，从而本文方案满足密文不可区分性。

6 效率分析

在本节中，我们将对本文提出的方案与其他一些相关的 PEKS 方案进行比较。表 1 对方案的计算效率和安全性能进行了详细比较。我们分别使用符号 P,E_1,E_t 和 M_1,M_2 来表示双线性配对运算、群 G_1,G_T 上的模幂运算和群 G_1，G_2 上的标量乘法。安全性比较则将详细说明方案是否具备抵御 IKGA 攻击的能力。

本文将在相同的标准下对 Huang 等人[9]的 PAEKS 方案、Shao 等人[11]的方案、Pu 等人[32]的方案以及 Zhang 等人[31]的方案进行比较。

表 1　方案性能及安全性比较

Table 1　Solution performance and security comparison

方案	PEKS()	Trapdoor()	Test()	抵御 IKGA 攻击
PAEKS	$3E_1$	E_1+P	$2P$	是
Shao 等人	$9E_1+3P$	$2E_1$	$5E_1+4P$	是
Pu 等人	$2M_1+E_t$	M_2	P	否
Zhang 等人	$3M_1+P$	M_1+M_2	$2P$	否
本文	$3M_1$	M_1+2M_2+P	$2P$	是

PAEKS 采用了认证加密的方式, 以确保服务器无法生成关键字密文, 实现对内部关键字猜测攻击的抵抗。Shao 等人的方案则通过指定服务器执行配对算法, 在关键字密文生成时添加与发送方身份有关的确定性 RSA 签名, 从而抵抗内部关键字猜测攻击。Pu 等人提出的方案运用了 SM9 算法, 使用 SM9 的密钥生成算法为每个关键字生成对应的密文及陷门, 只有当关键字的密文与陷门配对时才会返回对应信息。Zhang 等人提出的方案则是基于身份的 PEKS, 使用 SM9 算法为用户生成密钥对, 再使用该密钥生成关键字密文及关键字陷门。

从算法执行的性能角度来看, 本文方案使用了 4 个 G_1 群上的标量乘法, 2 个 G_2 群上的标量乘法, 3 个双线性配对运算。整体性能与使用了 4 个 G_1 群上的模幂运算, 3 个双线性配对运算的 PAEKS 方案相当。Pu 等人的方案使用了 2 个 G_1 群上的标量乘法, 1 个 G_2 群上的标量乘法, 1 个 G_T 群上的模幂运算, 1 个双线性配对运算, 显然在速率方面优于本文方案, 但是该方案不具备抵御 IKGA 攻击的安全性。并且该方案中数据接收方自身不能产生关键字陷门, 关键字陷门需由数据发送方生成再转发到数据接收方, 当系统中存在多个用户时, 通信成本会对应增加。Zhang 等人的方案使用了 4 个 G_1 群上的标量乘法, 1 个 G_2 上的标量乘法, 3 个双线性配对运算。从性能上看, 本文方案生成关键字密文速率更高, 而 Zhang 等人的方案生成关键字陷门速率更高, 算法整体的性能相似。但是该方案的关键字密文对于内部敌手仍然是可伪造的, 且一旦密文包含的关键字与服务器存储的陷门包含的关键字相同, 就会泄露关键字信息。因此该方案不能有效抵御 IKGA 攻击。而 Shao 等人的方案尽管具备抵御 IKGA 攻击的安全性, 但由于使用的模幂运算与双线性配对运算较多, 总体性能低于本方案。

从安全性角度来看, 能够抵御 IKGA 攻击的方案有 Huang 等人的 PAEKS 方案、Shao 等人的方案和本文的方案。PAEKS 方案、Shao 等人方案的运算使用的都是对称的双线性对群, 且使用的椭圆曲线方程为 $E(F_p): y^2=x^3+x$, 曲线的嵌入度为 2。

而本文方案的运算建立在非对称双线性对的基础上, 根据国家 SM9 密码算法参数规范, 使用的 BN 曲线方程为 $E: y^2=x^3+5$, 曲线的嵌入度 12。对本文的方案、PAEKS 及 Zhang 等人的方案进行了编程实现。实验采用联想笔记本电脑, 配置为: 16GB 运行内存、64 位 Windows 10 操作系统、AMD R9-7945HX, 2.5GHz 的 CPU, 使用 JPBC 密码库和 JAVA 编程语言。在编程实现过程中, 采用了 256bit 的 BN 曲线。实验结果如图 2～图 4 所示, 其中横坐标对应使用的关键字数目, 纵坐标表示算法的实际执行时间(单位为 ms)。图 2 表示了各个方案关键字密文生成算法的运行时间。从图 2 可以看出本文方案在关键字密文生成上占据优势, 耗时最低, 而 Zhang 等人方案的关键字密文生成耗时最高。图 3 表示各个方案关键字陷门算法的运行时间, 可以看出本文方案和 PAEKS 方案在陷门生成上用时相近, 但是都高于 Zhang 等人的方案。图 4 表示了各个方案配对算法的运行时间, 可以看出三个方案用时基本相同, 符合理论分析结果。因此整体看来, 三个方案的整体用时相似, Zhang 等人的方案整体用时略低。但是本文方案在具备抵御内部关键字猜测攻击的同时使用了 SM9 算法, 具备更高的安全性和实用性。

7　总结

可搜索加密技术为用户提供了在密文数据上进行关键字检索的功能。然而, 当前提出的 PEKS 方案往往缺乏对 IKGA 攻击的抵御能力, 易受到潜在的

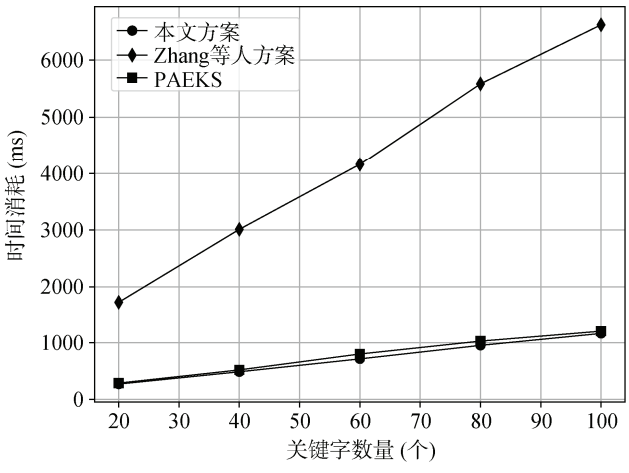

图 2　密文生成算法运行时间

Figure 2　Runtime of PEKS algorithm

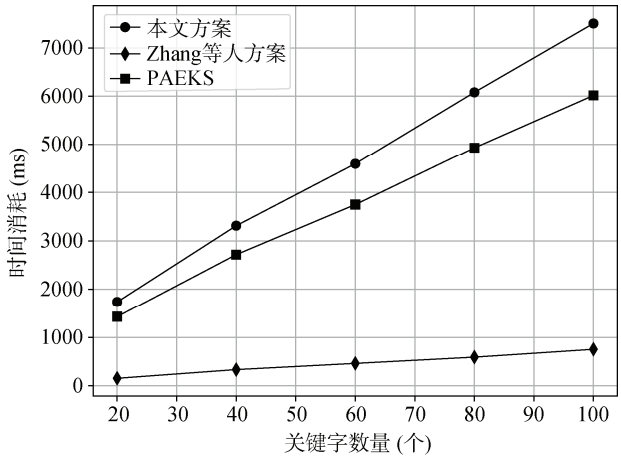

图 3　陷门生成算法运行时间

Figure 3　Runtime of Trapdoor algorithm

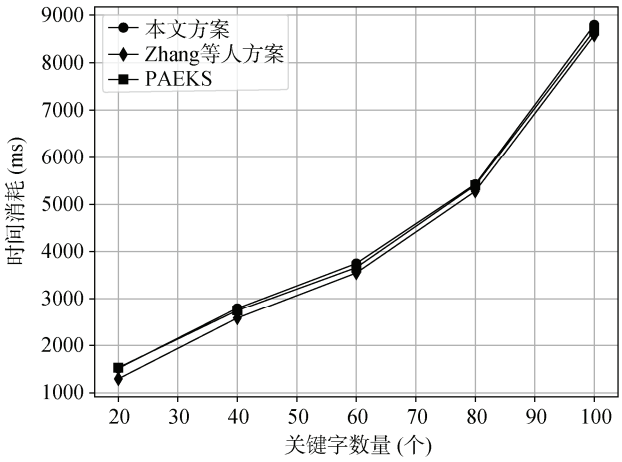

图 4　配对算法运行时间

Figure 4　Runtime of Search algorithm

恶意敌手的威胁。此外, 多数 PEKS 方案是基于国外密码体制实现的, 未能满足国内实际发展的需求。这使得在选择合适的可搜索加密方案时, 需要更为谨慎考虑安全性和国内实际应用需求的平衡。

本文基于 SM9 标识密码算法, 结合抗内部关键字猜测攻击的可搜索公钥加密算法结构, 设计了一种基于 SM9 的可搜索公钥加密方案。并在随机预言模型下证明了本方案满足抗 IKGA 语义安全的要求。最后, 通过性能评估和仿真实验分析验证了本文方案的实用性和效率。

未来的工作将围绕场景应用和效率提升展开。具体而言, 可以研究如何实现多关键字搜索的可搜索加密方案, 并结合联盟链、边缘计算等实际场景, 以增强本文方案在实际应用中的效率。

参考文献

[1] Li J W, Jia C F, Liu Z L, et al. Survey on the Searchable Encryption[J]. *Journal of Software*, 2015, 26(1): 109-128. (李经纬, 贾春福, 刘哲理, 等. 可搜索加密技术研究综述[J]. *软件学报*, 2015, 26(1): 109-128.)

[2] Song D X, Wagner D, Perrig A. Practical Techniques for Searches on Encrypted Data[C]. *Proceeding 2000 IEEE Symposium on Security and Privacy*. *S&P 2000*, 2002: 44-55.

[3] Boneh D, Di Crescenzo G, Ostrovsky R, et al. Public Key Encryption with Keyword Search[C]. *Advances in Cryptology - EUROCRYPT 2004*, 2004: 506-522.

[4] Xu P, Jin H, Wu Q H, et al. Public-Key Encryption with Fuzzy Keyword Search: A Provably Secure Scheme under Keyword Guessing Attack[J]. *IEEE Transactions on Computers*, 2013, 62(11): 2266-2277.

[5] Hamlin A, Shelat A, Weiss M, et al. Multi-Key Searchable Encryption, Revisited[C]. *Public-Key Cryptography - PKC 2018*, 2018: 95-124.

[6] Uwizeye E, Wang J Y, Cheng Z H, et al. Certificateless Public Key Encryption with Conjunctive Keyword Search and Its Application to Cloud-Based Reliable Smart Grid System[J]. *Annals of Telecommunications*, 2019, 74(7): 435-449.

[7] Yau W C, Phan R C W, Heng S H, et al. Keyword Guessing Attacks on Secure Searchable Public Key Encryption Schemes with a Designated Tester[J]. *International Journal of Computer Mathematics*, 2013, 90(12): 2581-2587.

[8] Wang C H, Tu T Y. Keyword Search Encryption Scheme Resistant Against Keyword-Guessing Attack by the Untrusted Server[J]. *Journal of Shanghai Jiaotong University* (*Science*), 2014, 19(4): 440-442.

[9] Huang Q, Li H B. An Efficient Public-Key Searchable Encryption Scheme Secure Against Inside Keyword Guessing Attacks[J]. *Information Sciences*, 2017, 403/404: 1-14.

[10] Islam S H, Obaidat M S, Rajeev V, et al. Design of a Certificateless Designated Server Based Searchable Public Key Encryption Scheme[C]. *Mathematics and Computing*, 2017: 3-15.

[11] Shao Z Y, Yang B. On Security Against the Server in Designated Tester Public Key Encryption with Keyword Search[J]. *Information Processing Letters*, 2015, 115(12): 957-961.

[12] Zhang Y P, Katz J, Papamanthou C. All Your Queries Are Belong to Us: The Power of File-Injection Attacks on Searchable Encryption[C]. *The 25th USENIX Conference on Security Symposium*, 2016: 707-720.

[13] Ma M M, He D B, Khan M K, et al. Certificateless Searchable Public Key Encryption Scheme for Mobile Healthcare System[J]. *Computers & Electrical Engineering*, 2018, 65: 413-424.

[14] Zhang X J, Xu C X, Wang H X, et al. FS-PEKS: Lattice-Based Forward Secure Public-Key Encryption with Keyword Search for Cloud-Assisted Industrial Internet of Things[J]. *IEEE Transactions on Dependable and Secure Computing*, 2021, 18(3): 1019-1032.

[15] Wu L B, Zhang Y B, Ma M M, et al. Certificateless Searchable Public Key Authenticated Encryption with Designated Tester for Cloud-Assisted Medical Internet of Things[J]. *Annals of Telecommunications*, 2019, 74(7): 423-434.

[16] Byun J W, Rhee H S, Park H A, et al. Off-Line Keyword Guessing Attacks on Recent Keyword Search Schemes over Encrypted Data[C]. *Secure Data Management*, 2006: 75-83.

[17] Rhee H S, Park J H, Susilo W, et al. Trapdoor Security in a Sear-

chable Public-Key Encryption Scheme with a Designated Tester[J]. *Journal of Systems and Software*, 2010, 83(5): 763-771.

[18] Wang B J, Chen T, Jeng F. Security Improvement Against Malicious Server's Attackfor a dPEKS Scheme[J]. *International Journal of Information and Education Technology*, 2011: 350-353.

[19] Tang Q, Chen L Q. Public-Key Encryption with Registered Keyword Search[C]. *Public Key Infrastructures, Services and Applications*, 2010: 163-178.

[20] He D B, Ma M M, Zeadally S, et al. Certificateless Public Key Authenticated Encryption with Keyword Search for Industrial Internet of Things[J]. *IEEE Transactions on Industrial Informatics*, 2018, 14(8): 3618-3627.

[21] Cheng L, Meng F. Public-key Authenticate Searchable Encryption with Probabilistic Trapdoor Generation[J]. *Cryptology ePrint Archive*, 2020.

[22] Li H B, Huang Q, Shen J, et al. Designated-Server Identity-Based Authenticated Encryption with Keyword Search for Encrypted Emails[J]. *Information Sciences*, 2019, 481: 330-343.

[23] Pakniat N, Shiraly D, Eslami Z. Certificateless Authenticated Encryption with Keyword Search: Enhanced Security Model and a Concrete Construction for Industrial IoT[J]. *Journal of Information Security and Applications*, 2020, 53: 102525.

[24] Wang S H, Zhang Y X, Wang H Q, et al. Efficient Public-Key Searchable Encryption Scheme Against Inside Keyword Guessing Attack[J]. *Computer Science*, 2019, 46(7): 91-95.
(王少辉, 张彦轩, 王化群, 等. 抗内部关键词猜测攻击的高效公钥可搜索加密方案[J]. *计算机科学*, 2019, 46(7): 91-95.)

[25] Chen B W, Wu L B, Zeadally S, et al. Dual-Server Public-Key Authenticated Encryption with Keyword Search[J]. *IEEE Transactions on Cloud Computing*, 2022, 10(1): 322-333.

[26] 国家密码管理局. 国家密码管理局公告(第 30 号)[EB/OL]. 2016: http://www.sca.gov.cn/sca/xxgk/2016-03/28/content_1002815.shtml.

[27] 密码行业标准化技术委员会. 我国 SM2 和 SM9 数字签名算法正式成为 ISO/IEC 国际标准[EB/OL]. 2017: http://www.gmbz.org.cn/main/postDetail.html id=20180118171408.

[28] Lai J C, Huang X Y, He D B. An Efficient Identity-Based Broadcast Encryption Scheme Based on SM9[J]. *Chinese Journal of Computers*, 2021, 44(5): 897-907.
(赖建昌, 黄欣沂, 何德彪. 一种基于商密 SM9 的高效标识广播加密方案[J]. *计算机学报*, 2021, 44(5): 897-907.)

[29] Lai J C, Huang X Y, He D B, et al. An Efficient Identity-Based Signcryption Scheme Based on SM9[J]. *Journal of Cryptologic Research*, 2021, 8(2): 314-329.
(赖建昌, 黄欣沂, 何德彪, 等. 基于商密 SM9 的高效标识签密[J]. *密码学报*, 2021, 8(2): 314-329.)

[30] Shi Y, Ma Z Y, Qin R F, et al. Implementation of an Attribute-Based Encryption Scheme Based on SM9[J]. *Applied Sciences*, 2019, 9(15): 3074.

[31] Zhang C, Peng C G, Ding H F, et al. Searchable Encryption Scheme Based on China State Cryptography Standard SM9[J]. *Computer Engineering*, 2022, 48(7): 159-167.
(张超, 彭长根, 丁红发, 等. 基于国密 SM9 的可搜索加密方案[J]. *计算机工程*, 2022, 48(7): 159-167.)

[32] Pu L, Lin C, Wu W, et al. A Public-Key Encryption with Keyword Search Scheme from SM9[J]. *Journal of Cyber Security*, 2023, 8(1): 108-118.
(蒲浪, 林超, 伍玮, 等. 基于 SM9 的公钥可搜索加密方案[J]. *信息安全学报*, 2023, 8(1): 108-118.)

徐嘉旺 于 2022 年在常州大学计算机与科学专业获得学士学位。现在南京邮电大学电子信息专业攻读硕士学位。研究领域为密码学与信息安全。研究兴趣包括密码学、信息安全、云计算。Email:xjw18206119963@163.com。

王化群 于 2006 年在南京邮电大学信号与信息处理专业获得博士学位。现任南京邮电大学计算机学院教授, 中国计算机学会区块链专委会委员。研究领域为应用密码学、区块链、云计算安全。研究兴趣包括密码学、区块链、云计算。Email: whq@njupt.edu.cn。