软件学报 ISSN 1000-9825, CODEN RUXUEW
Journal of Software,2015,26(1):109−128 [doi: 10.13328/j.cnki.jos.004700]

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel: +86-10-62562563

可搜索加密技术研究综述*

李经纬[1], 贾春福[1,3], 刘哲理[1], 李 进[2], 李 敏[1]

[1](南开大学 计算机与控制工程学院 计算机与信息安全系,天津 300071)

[2](广州大学 计算机科学与教育软件学院,广东 广州 510006)

[3](中国民航大学 信息安全评测中心,天津 300300)

通讯作者: 贾春福, E-mail: cfjia@nankai.edu.cn

摘 要: 从可搜索加密的两类基本问题出发,回顾了相关研究历史.介绍了可搜索加密的分类,包括其应用场景和应用模型,并探讨了相应的解决策略,从构造角度,将其分为对称可搜索加密和非对称可搜索加密.基于这种分类,围绕基本定义、典型构造和扩展研究,对可搜索加密相关工作进行了综述.最后,总结和展望了待解决的关键性问题和未来的研究方向.这些工作将对可搜索加密的进一步研究起到一定的促进作用.

关键词: 可搜索加密;对称可搜索加密;非对称可搜索加密;关键词猜测攻击;云安全

中图法分类号: TP309

中文引用格式: 李经纬,贾春福,刘哲理,李进,李敏.可搜索加密技术研究综述.软件学报,2015,26(1):109−128. http://www.jos.org.cn/1000-9825/4700.htm

英文引用格式: Li JW, Jia CF, Liu ZL, Li J, Li M. Survey on the searchable encryption. Ruan Jian Xue Bao/Journal of Software, 2015,26(1):109−128 (in Chinese). http://www.jos.org.cn/1000-9825/4700.htm

Survey on the Searchable Encryption

LI Jing-Wei[1], JIA Chun-Fu[1,3], LIU Zhe-Li[1], LI Jin[2], LI Min[1]

[1](Department of Computer & Information Security, College of Computer and Control Engineering, Nankai University, Tianjin 300071, China)

[2](School of Computer Science and Educational Software, Guangzhou University, Guangzhou 510006, China)

[3](Information Security Evaluation Cener of Civil Aviation, Civil Aviation University of China, Tianjin 300300, China)

Abstract: This paper reviews previous research on the two basic searchable encryption problems, and introduces the classification of searchable encryption (SE), including its application scenarios and usage models. After discussing the resolution strategies, it divides SE into two groups, that is symmetric searchable encryption and asymmetric searchable encryption. Based on this classification, the research advance is surveyed on basic definition, typical construction and extended research. Finally, the need-to-be-solved problems and main research directions are discussed. This study aims at promoting further research of searchable encryption.

Key words: searchable encryption; symmetric searchable encryption; asymmetric searchable encryption; keyword guessing attack; cloud security

可搜索加密问题源于文献[1]:假设用户 Alice 试图将个人文件存放在一个诚实但具有好奇心的外部服务器,以降低本地资源开销.为保护文件隐私,须采用某种加密方式将文件加密后存储.使用传统分组密码,只有密

* 基金项目: 国家重点基础研究发展计划(973)(2013CB834204); 国家自然科学基金(61272423, 61100224, 61472091); 高等学校博士学科点专项科研基金(20100031110030, 20120031120036); 天津市自然科学基金(14JCYBJC15300); 中国民航大学信息安全评测中心开放课题基金(CAAC-ISECCA-201403)

收稿时间: 2013-03-11; 定稿时间: 2014-07-09; jos 在线出版时间: 2014-08-19

CNKI 网络优先出版: 2014-08-19 14:17, http://www.cnki.net/kcms/doi/10.13328/j.cnki.jos.004700.html

钥拥有者才具备解密能力,意味着 Alice 在执行基于关键词的查询操作时,需要下载所有已上传的文件,完全解密后再检索,会带来两个问题:① 如果 Alice 在服务器上已存有大量文件,一一下载会占用大量网络带宽,可能造成服务器堵塞;② 对已下载的所有文件完全解密会占用大量本地计算资源,效率极低.

解决此类问题的加密技术称为可搜索加密(searchable encryption,简称 SE),该技术要求只有合法用户才具备基于关键词检索的能力.随着研究的推进,其应用并不仅限于此:2004 年,Boneh 提出使用非对称可搜索加密(asymmetric searchable encryption,简称 ASE)解决“不可信赖服务器路由”问题[2];最近兴起的云计算[3]将是 SE 的最佳应用平台,由于服务提供商的不可控性,用户必须应对存储到云端的个人数据可能泄密的威胁,SE 提供的加密和密文直接检索功能使服务器无法窃听用户个人数据,但可以根据查询请求返回目标密文文件,这样既保证了用户数据的安全和隐私,又不会过分降低查询效率.

本文关注近年来可搜索加密的研究进展,描述了可搜索加密基本问题的研究历史,并围绕定义、典型构造和扩展研究,分别对对称和非对称密码体制下的可搜索加密研究成果进行综述,最后展望了可搜索加密未来的研究方向,以期对其在国内的研究起到一定的推动作用.

1 可搜索加密

1.1 可搜索加密过程

如图 1 所示,可搜索加密可分为 4 个子过程:

Step 1. 加密过程.用户使用密钥在本地对明文文件进行加密,并将其上传至服务器.

Step 2. 陷门生成过程.具备检索能力的用户,使用密钥生成待查询关键词的陷门,要求陷门不能泄露关键词的任何信息.

Step 3. 检索过程.服务器以关键词陷门为输入,执行检索算法,返回所有包含该陷门对应关键词的密文文件,要求服务器除了能知道密文文件是否包含某个特定关键词外,无法获得更多信息.

Step 4. 解密过程.用户使用密钥解密服务器返回的密文文件,获得查询结果.

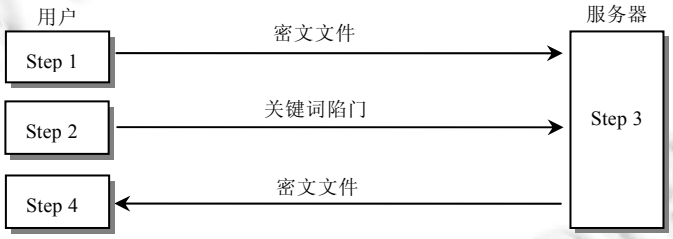


Fig.1 Steps in searchable encryption

图 1 可搜索加密过程

1.2 研究历史

可搜索加密问题的提出,源于解决两类可搜索加密的基本问题:① 不可信赖服务器的存储问题;② 不可信赖服务器的路由问题.

1.2.1 不可信赖服务器存储问题的相关研究

不可信赖服务器的存储问题最早提出于 2000 年[1],Song 等人[1]提出了基于密文扫描思想的 SWP 方案,将明文文件划分为“单词”并对其分别加密,通过对整个密文文件扫描和密文单词进行比对,就可确认关键词是否存在,甚至统计其出现的次数.Goh[4]提出了基于索引的 Z-IDX 方案,使用布隆过滤器(Bloom filter)作为单个文件的索引结构,将文件包含的关键词映射为码字存储于该文件的索引中,通过布隆过滤器的运算,就能判定密文文件是否包含某个特定关键词.Chang 和 Mitzenmacher[5]考虑了该可搜索加密基本问题的一个应用场景:用户通过个人电脑以密文形式存储文件至服务器,然后使用移动设备(例如手机等)检索服务器上的密文文件,并针对此应用提出 PPSED(privacy preserving keyword searches on remoted encrypted data)方案.Curtmola[6]规范化了对称可

搜索加密(symmetric searchable encryption,简称 SSE)及其安全目标,提出能够在非自适应和自适应攻击模型下达到不可区分性安全的 SSE-1 和 SSE-2 方案.这里,SSE-1 和 SSE-2 都基于“关键词-文件”索引构建思想,服务器只需 $O(1)$时间即可完成检索操作.然而,执行文件的添加或删除操作需要重新构建索引,时间开销较大.

近年来,围绕基本 SSE 方案中仍然存在的一些需要解决的问题,学者们进行了广泛的研究,包括:(1) 如何对服务器存放的密文文件进行动态添加、更新或删除[7−9];(2) 如何对基本方案中“单个关键词精确匹配”查询方式进行扩展,以适应更广泛的查询需求[10−15];(3) 如何对基本方案中“包含与不包含”查询模式进行优化,以进一步降低用户端筛选目标文件的计算量[16−18];(4) 如何应对在半可信且具有好奇心的威胁模型下,服务器并不总是诚实地计算并返回检索结果的情况[19,20].

1.2.2 不可信赖服务器路由问题的相关研究

不可信赖服务器的路由问题源于文献[2]:Bob 通过不可信赖邮件服务器向 Alice 发送包含某些关键词的邮件,要求服务器不能获取邮件内容和相关关键词信息,但需根据关键词将邮件路由至 Alice 的某个终端设备.例如,如果邮件的关键词为“urgent”,则服务器将邮件分配至 Alice 的手机,如果邮件的关键词为“lunch”,则服务器将邮件分配至 Alice 的电脑.Boneh 等人[2]最早提出 PEKS(public key encryption with keyword search)概念,并基于 BF-IBE[21]构造了第一个 PEKS 方案 BDOP-PEKS,安全性可归结为 BDH(bilinear Diffie-Hellman)数学假设.Khader[22]基于 K-resilient IBE 构造 KR-PEKS 方案,在标准模型下达到 IND-CKA 安全.Crescenzo 等人[23]提出基于二次剩余中二次不可区分性问题(quadratic indistinguishability problem,简称 QIP)的 PEKS 方案.Abdalla 等人[24]针对 PEKS 算法一致性定义缺陷,提出统计一致性(statistically consistency)和计算一致性(computationally consistency),并描述了从基于身份加密(identity-based encryption,简称 IBE)到 PEKS 的一般变换算法 IBE2PEKS.

文献[25−27]指出了当前 PEKS 的一个较为严重的安全隐患:由于关键词空间远小于密钥空间,而且用户通常仅检索一些常用关键词,攻击者可借此实施关键词猜测攻击(keyword guessing attack,简称 KGA),进而证明了不存在满足算法一致性并且在 KGA 下是安全的 PEKS 方案.因此,抵御 KGA 意味着需对 PEKS 机制本身加以修改.鉴于此,Tang 等人[28]提出 PERKS(public-key encryption with registered keyword search)方案,要求接收者在初始化阶段注册关键词,并将产生的预标签(pre-tag)通过安全信道传递给发送者;Xu 等人[29]提出 PEFKS(public key encryption with fuzzy keyword search)方案,向不可信赖服务器提供模糊陷门以进行初次检索,对返回结果再在本地进行基于精确陷门的二次检索.这些方案都能抵御 KGA.

近年来,关于 PEKS 的研究集中于:(1) 对基本 PEKS 方案的安全性加以完善,提高 PEKS 密文与邮件密文的耦合度[30−35];(2) 扩展查询方式,适应更广泛的查询需求[36−39];(3) 以实际背景为依托,探索满足高级应用需求的方案[40−43].

2 可搜索加密的分类

2.1 应用模型分类

如图 2 所示,从当前的应用角度可将可搜索加密问题模型分为 4 类.

1) 单用户(单服务器)模型.

用户加密个人文件并将其存储于不可信赖外部服务器,要求:① 只有该用户具备基于关键词检索的能力;② 服务器无法获取明文文件和待检索关键词的信息.文献[1]中的应用问题以及单用户模式的云存储服务都是单用户模型的实例.

2) 多对一(单服务器)模型.

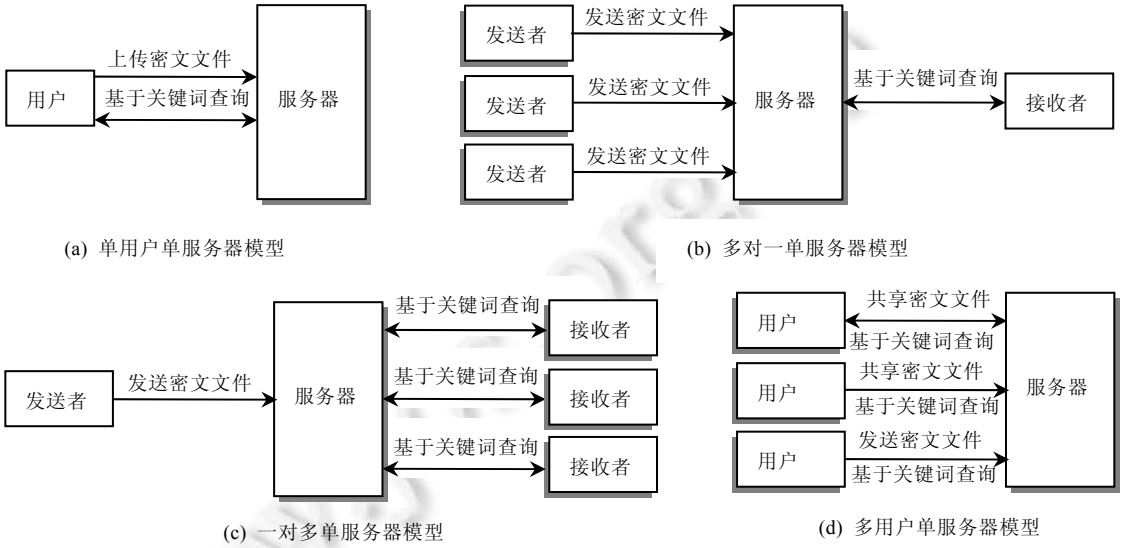
多个发送者加密文件后,将其上传至不可信赖外部服务器,以期达到与单个接收者传送数据的目的.要求:① 只有接收者具备基于关键词检索的能力;② 服务器无法获取明文文件信息.需要指出的是,不同于单用户模型,多对一模型要求发送者和接收者不能是同一用户.文献[2]中的应用问题和具备简单共享机制的云存储服务都是多对一模型的实例.

3) 一对多(单服务器)模型.

与多对一(单服务器)模型类似,但为单个发送者将加密文件上传至不可信赖外部服务器,借此与多个接收者共享数据.该模型遵循着一种广播共享的模式,文献[7]中的研究问题是一对多模型的实例.

4) 多对多(单服务器)模型.

在多对一模型的基础上,任意用户都可成为接收者,其通过访问控制和认证策略以后,具备基于关键词的密文检索方式提取共享文件的能力.要求:① 只有合法用户(例如能够满足发送者预先指定的属性或身份要求)具备基于关键词检索的能力;② 服务器无法获取明文文件信息.该模型既是多对一模型的扩展,同时也是云计算中复杂共享机制的抽象,具备广阔的应用前景.



(a) 单用户单服务器模型 (b) 多对一单服务器模型

(c) 一对多单服务器模型 (d) 多用户单服务器模型

Fig.2 Usage models classification in searchable encryption

图 2 可搜索加密应用模型分类

2.2 解决策略

从密码构造角度可将 SE 问题模型的解决策略分为 3 类.

1) 对称可搜索加密,适用于单用户模型.

对称可搜索加密的构造通常基于伪随机函数,具有计算开销小、算法简单、速度快的特点,除了加解密过程采用相同的密钥外,其陷门生成也需密钥的参与.单用户模型的单用户特点使得对称可搜索加密非常适用于该类问题的解决:用户使用密钥加密个人文件并上传至服务器,检索时,用户通过密钥生成待检索关键词陷门,服务器根据陷门执行检索过程后返回目标密文.

2) 非对称可搜索加密,适用于多对一模型.

非对称可搜索加密使用两种密钥:公钥用于明文信息的加密和目标密文的检索,私钥用于解密密文信息和生成关键词陷门.非对称可搜索加密算法通常较为复杂,加解密速度较慢,然而,其公私钥相互分离的特点,非常适用于多用户体制下可搜索加密问题的解决:发送者使用接收者的公钥加密文件和相关关键词,检索时,接收者使用私钥生成待检索关键词陷门,服务器根据陷门执行检索算法后返回目标密文.该处理过程避免了在发送者与接收者之间建立安全通道,具有较高的实用性.

3) 对称可搜索加密或非对称可搜索加密,可解决一对多和多对多模型中的可搜索加密问题.

非对称可搜索加密本身即能有效地支持最基本形式的隐私数据的共享,通过共享密钥,其可被拓展到多对多的应用场景.对称可搜索加密虽然通常适用于单用户模型,但其由于计算开销小、速度快,更适合于大型文件数据的加密和共享,通过混合加密与基于属性加密技术相结合,或与代理重加密结合,也可用于构造共享方案.

鉴于对称和非对称可搜索加密作为基本工具,在解决实际可搜索加密问题时的重要性,本文接下来将围绕定义、构造和扩展研究,分别对对称和非对称可搜索加密的研究成果进行综述.

3 对称可搜索加密

3.1 定 义

3.1.1 算法描述

定义 1(对称可搜索加密). 定义在字典$\Delta=\{W_1,W_2,\ldots,W_d\}$上的对称可搜索加密算法可描述为五元组:

$$\mathcal{SSE}=(KeyGen,Encrypt,Trapdoor,Search,Decrypt),$$

其中,

1) $K=KeyGen(\lambda)$:输入安全参数λ,输出随机产生的密钥K;
2) $(I,C)=Encrypt(K,D)$:输入对称密钥K和明文文件集$D=(D_1,D_2,\ldots,D_n)$,$D_i\in 2^{\Delta}$,输出索引I和密文文件集$C=(C_1,C_2,\ldots,C_n)$.对于无需构造索引的SSE方案(例如SWP方案[1]),$I=\varnothing$;
3) $T_W=Trapdoor(K,W)$:输入对称密钥K和关键词W,输出关键词陷门T_W;
4) $D(W)=Search(I,T_W)$:输入索引I和陷门T_W,输出包含W的文件的标识符构成的集合$D(W)$;
5) $D_i=Decrypt(K,C_i)$:输入对称密钥K和密文文件C_i,输出相应明文文件D_i.

如果对称可搜索加密方案$\mathcal{SSE}$是正确的,那么对于$\forall\lambda\in\mathbb{N},n\in\mathbb{Z},W\in\Delta,D=(D_1,D_2,\ldots,D_n)$以及$KeyGen(\lambda)$和$Encrypt(K,D)$输出的$K$和$(I,C)$,都有$Search(I,Trapdoor(K,W))=D(W)$和$Decrypt(K,C_i)=D_i$成立.

这里,$C_i\in C,i=1,2,\ldots,n$.

基于定义1,对称可搜索加密流程如下:加密过程中,用户执行$KeyGen$算法生成对称密钥K,使用K加密明文文件集D,并将加密结果上传至服务器.检索过程中,用户执行$Trapdoor$算法,生成待查询关键词W的陷门T_W;服务器使用T_W检索到文件标识符集合$D(W)$,并根据$D(W)$中文件标识符提取密文文件以返回用户;用户最终使用K解密所有返回文件,得到目标文件.

3.1.2 安全目标

在设计密码方案时,主要考虑可能面临攻击模型下需达到的安全目标,通常使用安全目标与攻击模型相结合的方式定义方案的安全性.早在2000年,Song等人[1]将可证安全理论的不可区分性安全目标引入可搜索加密机制,要求密文不会泄漏任何原始文件信息.然而,Song的原始定义并不足以描述攻击者在现实场景中所具备的可搜索攻击能力.针对此问题,Goh[4]提出了选择关键词攻击下的不可区分性安全目标IND-CKA,要求攻击者即使能够任意询问(或以黑盒方式产生)密文文件和关键词陷门,也无法获得比通过陷门检索方式更多的原始文件信息.进一步地,Chang等人[5]考虑攻击者在实施攻击时能够获得之前所有轮次服务器端的查询结果的情况,描述了可搜索加密机制基于模拟的安全性定义,以限制服务器除每一轮查询结果外,无法获得任何信息.

2006年,Curtmola等人[6]指出:① 文献[4]未明确考虑关键词陷门在可搜索加密机制中的安全性;② 文献[5]中的安全性定义无法描述具备自适应攻击能力的攻击者,且能够被任何可搜索加密方案平凡地(trivially)满足.Curtmola等人[6]进而在自适应(adaptive)和非自适应(nonadaptive)模型下形式化地定义了SSE的语义安全(semantic security,简称SS)和不可区分性安全(indistingsuishability,简称IND).描述安全目标之前,引入几个概念:

定义 2. 假设$\Delta=\{W_1,W_2,\ldots,W_d\}$表示关键词字典,$D=(D_1,D_2,\ldots,D_n)$表示明文文件集合,$W=(W^{(1)},W^{(2)},\ldots,W^{(q)})$表示一组已查询关键词,这里,$D_i\in 2^{\Delta},W_i\in\Delta$.可定义如下概念:

1) q-查询历史$H=(D,W)$,这里,$|W|=q$;
2) H的查询格式$\partial(H)=(D(W^{(1)}),D(W^{(2)}),\ldots,D(W^{(q)}))$;
3) H的检索格式$\sigma(H)$为$q\times q$矩阵,对于$1\leqslant i,j\leqslant q$,如果$W^{(i)}=W^{(j)}$,那么第$i$行$j$列元素$\sigma(H)_{ij}=1$;否则,$\sigma(H)_{ij}=0$;
4) 攻击者关于H的视图定义为$V_K(H)=(I,C,T_1,T_2,\ldots,T_q,id(D_1),id(D_2),\ldots,id(D_n))$,包括密钥$K$作用下产生

的密文文件及其索引、历史查询关键词的陷门和一些额外信息,例如各文件标识符等;

5) H 的轨迹 $\tau(H)=(|D_1|,|D_2|,\ldots,|D_n|,\partial(H),\sigma(H))$,包括 H 的查询格式、检索格式和 D 中各文件长度信息.

SSE 安全目标的定义源于攻击者和挑战者的博弈过程:挑战者首先执行 *KeyGen* 算法产生对称密钥 K,并按照如图 3 所示的某种方式(在图 3(b)和图 3(d)中,$\mathcal{S}(\cdot)$为模拟算法,可根据历史的轨迹模拟产生密文文件集及其索引),根据秘密产生的随机参数 b 响应攻击者的询问,最后,由攻击者通过计算输出一个判定值 b'作为对 b 的猜测:如果 $b'=b$,判定成功;否则失败.因此,可定义攻击者$\mathcal{A}$在相应安全目标下的攻击优势为 $Adv_{SSE}(\mathcal{A})=|2\cdot\Pr[b'=b]-1|$.如果对任意$\mathcal{A}$和$\varepsilon>0$,都有 $Adv_{SSE}(\mathcal{A})<\varepsilon$,那么对称可搜索加密算法 SSE 达到了相应的安全目标.

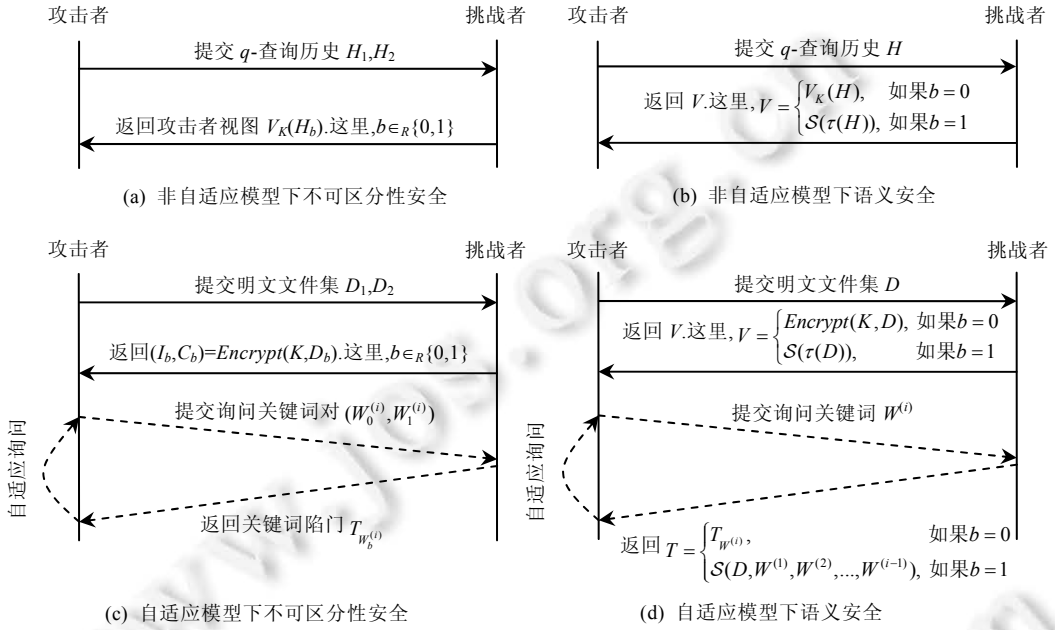


Fig.3 Games in SSE security notions

图 3 SSE 安全目标中的博弈过程

自适应和非自适应模型下的安全目标之间的关系如图 4 所示.

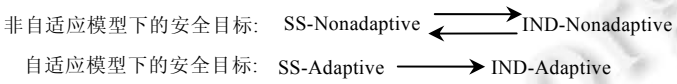


Fig.4 Relation of security goals under adaptive and nonadaptive attack model

图 4 自适应和非自适应模型下安全目标间的关系

图 4 中,箭头表示能推导出.通过图 4 可以看出:

- 非自适应攻击模型下,SS-Nonadaptive 和 IND-Nonadaptive 相互等价,即,达到 SS-Nonadaptive 安全的 SSE 同时也达到 IND-Nonadaptive 安全;反之亦然;
- 自适应攻击模型下,SS-Adaptive 安全能够推导出 IND-Adaptive 安全,因此,SS-Adpative 比 IND-Adaptive 具备更高的安全度.

3.2 典型构造

SSE 典型构造方式包括 SWP 方案[1]、Z-IDX 方案[4]和 SSE-1 方案[6].本节从构造角度对 3 种典型方案的加密过程进行综述.由于 Z-IDX 和 SSE-1 采用基于索引的加密,对数据文件本身采用传统分组密码直接加密即可,因此,这里只详细介绍这两种方法的索引构建过程.

3.2.1 SWP 方案

SWP 方案[1]在预处理过程中根据文件长度产生伪随机流 $S_1,S_2,\ldots,S_n$(n 为待加密文件中“单词”个数),然后采用两个层次加密:在第 1 层,使用分组密码 E 逐个加密明文文件单词;在第 2 层,对分组密码输出 $E(K',W_i)$进行处理:① 将密文等分为 L_i 和 R_i 两部分;② 基于 L_i 生成二进制字符串 $S_i||F(K_i,S_i)$,这里,$K_i=f(K'',L_i)$,||为符号串连接,F 和 f 为伪随机函数;③ 异或 $E(K',W_i)$和 $S_i||F(K_i,S_i)$以形成 W_i 的密文单词.

查询文件 D 中是否包含关键词 W,只需发送陷门 $T_W=(E(K',W),K=f(K'',L))$至服务器(L 为 $E(K',W)$的左部),服务器顺序遍历密文文件的所有单词 C,计算 C XOR $E(K',W)=S||T$,判断 $F(K,S)$是否等于 T:如果相等,C 即为 W 在 D 中的密文;否则,继续计算下一个密文单词.

SWP 方案[1]通过植入“单词”位置信息,能够支持受控检索(检索关键词的同时,识别其在文件中出现的位置).例如,将所有“单词”以 $W||\alpha$形式表示,α为 W 在文件中出现的位置,仍按图 5 所示加密,但查询时可增加对关键词出现位置的约束.SWP 方案[1]存在一些缺陷:① 效率较低,单个单词的查询需要扫描整个文件,占用大量服务器计算资源;② 在安全性方面存在统计攻击的威胁.例如,攻击者可通过统计关键词在文件中出现的次数来猜测该关键词是否为某些常用词汇.

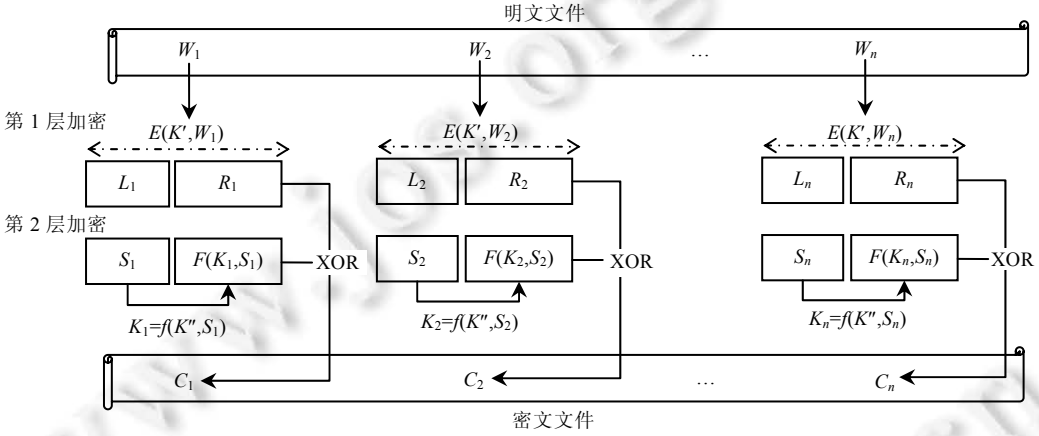


Fig.5 SWP scheme

图 5 SWP 方案

3.2.2 Z-IDX 方案

Z-IDX[4]方案使用布隆过滤器作为文件索引,以高效跟踪文件中的关键词.布隆过滤器由二进制向量 Mem(假设为 m 位)和哈希函数族$\{h_1(\cdot),h_2(\cdot),\ldots,h_r(\cdot)\}(h_i:\{0,1\}^*\rightarrow\{1,2,\ldots,m\},i=1,2,\ldots,r)$组成,用于判断某元素是否存在于某集合中.例如,对集合 S,初始时刻,Mem 所有比特位置 0.以后,对每个元素 $s\in S$,置 $Mem[h_1(s)]$,$Mem[h_2(s)]$,…,$Mem[h_r(s)]$为 1.因此,为确定待判断元素 a 是否属于 S,只需检查比特位 $Mem[h_1(a)]$,$Mem[h_2(a)]$,…,$Mem[h_r(a)]$,如果所有比特位都为 1,则 a 属于 S;否则 a 不属于 S.

Z-IDX[4]构建索引的过程如图 6 所示,关键词通过两次伪随机函数作用形成码字存储于索引中,第 1 次伪随机函数以关键词 W_i 为输入,分别在子密钥 $K_1,K_2,\ldots,K_r$ 作用下生成 $x_{i1},x_{i2},\ldots,x_{ir}$;第 2 次伪随机函数分别以 $x_{i1},x_{i2},\ldots,x_{ir}$ 为输入,在当前文件标识符 id 作用下生成码字 $y_{i1},y_{i2},\ldots,y_{ir}$,确保了相同关键词在不同文件中形成不同码字.另外,在布隆过滤器中加入混淆措施(随机添加若干个 1)预防了针对关键词数目的攻击.

判断文件 D_{id}(id 为该文件的标识符)中是否包含关键词 W_i:① 用户使用密钥 $K=(K_1,K_2,\ldots,K_r)$生成 W_i 的陷门 $T_i=(x_{i1},x_{i2},\ldots,x_{ir})$,这里,$x_{ij}=f(K_j,W_i)$,$j=1,2,\ldots,r$;② 服务器基于 T_i 生成 W_i 的码字$(y_{i1},y_{i2},\ldots,y_{ir})$,这里,$y_{ij}=f(id,x_{ij})$,$j=1,2,\ldots,r$;③ 服务器判断 D_{id} 的索引 Mem_{id} 的 $y_{i1},y_{i2},\ldots,y_{ir}$ 位是否全为 1:若是,则 $W_i\in D_{id}$;否则,D_{id} 不包含 W_i.

Z-IDX[4]存在一些不足:

(1) 空间代价上,服务器除存储密文文件本身外,还需记录文件索引,当文件较短时,其索引可能是文件长度

的数倍,空间利用率较低.文献[4]给出一个例子,只包含一个单词且长度为 9 字节的文件,加密后的索引却为 90 字节;

(2) 时间代价上,服务器检索需逐个文件地计算和判断,整个关键词查询操作时间消耗为 $O(n)$(n 为服务器上存储文件数目),效率较低.

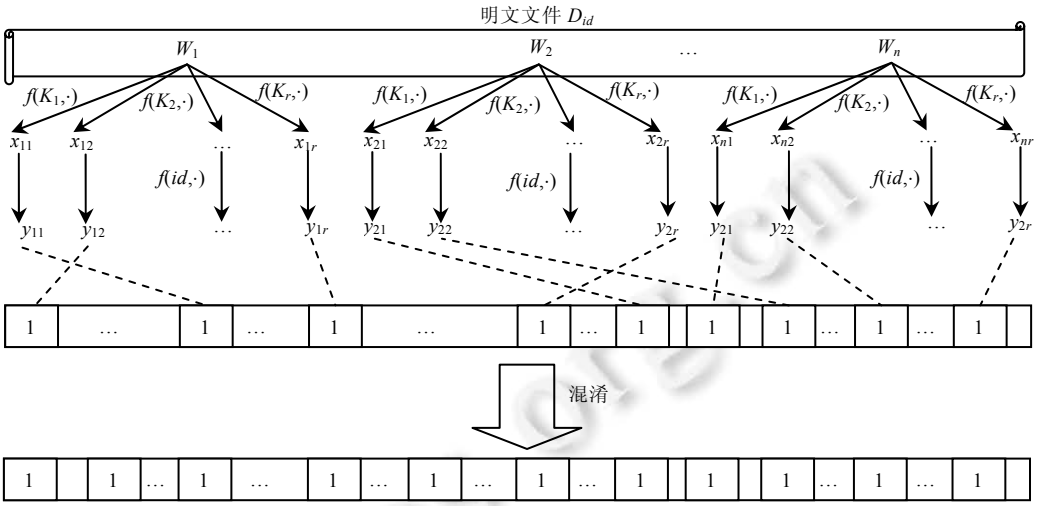


Fig.6 Z-IDX scheme

图 6 Z-IDX 方案

3.2.3 SSE-1 方案

SSE-1[6]为支持高效检索,引入额外数据结构:对任意关键词 $W\in\Delta$:① 数组 A 存储 $D(W)$的加密结果;② 速查表 T 存储 W 的相关信息,以高效定位相应关键词信息在 A 中的位置.

SSE-1[6]构建索引过程如下所示(图 7 描述了一个采用 SSE-1 方案构建仅包含一个关键词索引的实例,其中,SKE 为使用的底层对称加密算法):

1) 构建数组 A

初始化全局计数器 ctr=1,并扫描明文文件集 D,对于 $W_i\in\Delta$,生成文件标识符集合 $D(W_i)$,记 $id(D_{ij})$为 $D(W_i)$中字典序下第 j 个文件标识符,随机选取 SKE 的密钥 $K_{i0}\in\{0,1\}^{\lambda}$(这里,$\lambda$为安全参数),然后按照如下方式构建并加密由 $D(W_i)$中各文件标识符形成的链表: L_{W_i} :$1\leqslant j\leqslant|D(W_i)|-1$,随机选取 SKE 密钥 $K_{ij}\in\{0,1\}^{\lambda}$,并按照“文件标识符||下一个节点解密密钥||下一个节点在数组 A 的存放位置”这一形式创建链表 L_{W_i} 的第 j 个节点.

$$N_{ij}=id(D_{ij})\|K_{ij}\|\psi(K_1,ctr+1).$$

这里,K_1 为 SSE-1 的一个子密钥,$\psi(\cdot)$为伪随机函数.使用对称密钥 $K_{i(j-1)}$加密 N_{ij} 并存储至数组 A 的相应位置,即 $A[\psi(K_1,ctr)]=SKE.Encrypt(K_{i(j-1)},N_{ij})$;而对于 $j=|D(W_i)|$,创建其链表节点 $N_{i|D(W_i)|}=id(D_{i|D(W_i)|})\|0^{\lambda}\|NULL$ 并加密存储至数组 A, $A[\psi(K_1,ctr)]=SKE.Encrypt(K_{i(|D(W_i)|-1)},N_{i|D(W_i)|})$;最后,置 $ctr=ctr+1$.

2) 构建速查表 T

对于所有关键词 $W_i\in\Delta$,构建速查表 T 以加密存储关键词链表 L_{W_i} 的首节点的位置及密钥信息,即:

$$T[\pi(K_3,W_i)]=(addr_A(N_{i1})\|K_{i0})\ \mathrm{XOR}\ f(K_2,W_i).$$

这里,K_2 和 K_3 为 SSE-1 的子密钥,$f(\cdot)$为伪随机函数,$\pi(\cdot)$为伪随机置换,$addr_A(\cdot)$表示链表节点在数组 A 中的地址.

检索所有包含 W 的文件,只需提交陷门 $T_W=(\pi_{K_3}(W),f_{K_2}(W))$ 至服务器,服务器使用 $\pi_{K_3}(W)$ 在 T 中找到 W 相关链表首节点的间接地址 $\theta=T[\pi_{K_3}(W)]$,执行 $\theta\ \mathrm{XOR}\ f_{K_2}(W)=\alpha\|K'$,$\alpha$为 L_W 首节点在 A 中的地址,K'为首节点加密使用的对称密钥.由于在 L_W 中,除尾节点外所有节点都存储下一节点的对称密钥及其在 A 中的地址,服务器获得首节点的地址和密钥后,即可遍历链表所有节点,以获得包含 W 的文件的标识符.

SSE-1[6]避免了关键词查询过程中逐个文件进行检索的缺陷,具备较高的效率.然而,由于SSE-1需构建关键词相关链表,并将其节点加密后存储至数组 A,意味着现有文件的更新删除或新文件的添加需重新构建索引,造成较大开销.因此,SSE-1更适用于文件集合稳定,具有较少文件添加、更新和删除操作的情况.

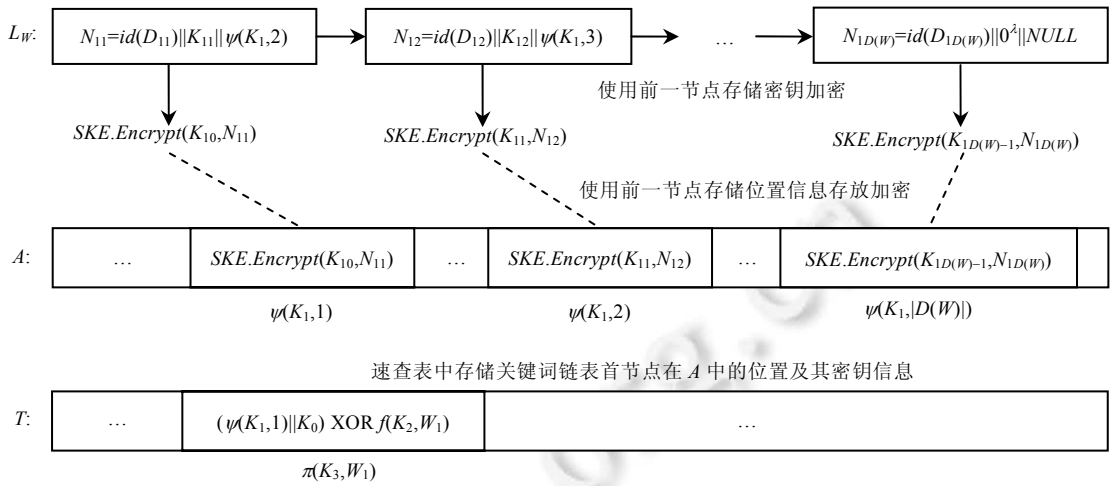


Fig.7 SSE-1 scheme

图 7 SSE-1 方案

3.3 构造特点

3.3.1 方案构建策略分析

以上3种典型构造代表两类SSE构建策略.

1) 基于顺序扫描的SSE构建策略

SWP 方案[1]采用顺序扫描的构建策略,其特点是将明文文件划分为若干个“单词”,对“单词”逐词加密,检索时顺序扫描整个密文文件,寻找与待检索单词匹配的密文单词.逐词加密和全文扫描的特点,使该策略具备如下优点:① 支持对文件中任意单词的检索;② 支持受控检索.然而,该策略要求服务器检索需遍历整个文件,时间复杂度为 $O(ns)$(n 为服务器上存储的文件数目,s 为文件平均长度),效率极低.目前,效率是阻碍顺序扫描构建策略在SSE方案中广泛应用的关键问题.

2) 基于索引的SSE构建策略

Z-IDX[4]和 SSE-1[6]采用基于索引的构建策略,其特点是将SSE方案的构造直接划分为两个子过程:构建索引和加密文件.加密文件用于保护不可信赖服务器上用户数据的隐私,通常采用AES直接加密即可;构建索引用于实现对密文文件的高效关键词查询.然而,由于索引存储在服务器端,可能作为服务器窃听用户数据的工具,传统的文件索引并不适用,这里的索引通常为加密索引.构建索引的特点,使该策略能够以较高效率支持关键词查询操作.目前,几乎所有SSE方案都采用基于索引的构建策略.

3.3.2 索引构建思想分析

设计基于索引的SSE方案的关键在于索引的构建,Z-IDX[4]和 SSE-1[6]采用了两种截然不同的索引构建思想:① Z-IDX[4]采取的是“文件-关键词”索引构建思想,构造以文件为基本单位的数据结构(Z-IDX[4]中为布隆过滤器),检索时,通过该数据结构的操作判断待检索关键词是否存在于某个文件中;② SSE-1[6]采取的是“关键词-文件”索引构建思想,构造以关键词为基本单位的数据结构(这里为链表),检索时,通过该数据结构找到包含待检索关键词的所有文件.

这两种构建思想已被广泛应用于SSE方案构造中,目前,几种主要基于索引的SSE方案及其采用的索引构建思想见表1.

由表1可知:在这两种索引构建思想下,操作处理基本单位的不同,决定了方案在完成操作时效率上的差异.

1) 采用“文件-关键词”构建思想的 SSE 方案以文件为基本单位,使其在文件更新(该操作以文件为基本单位)时,只需更新当前使用的数据结构,具有较高的效率.由于检索是以关键词为基本单位进行,要求服务器遍历所有文件的数据结构,花费 $O(n)$时间(事实上,这里的时间还包含查询每个文件的数据结构,例如 Z-IDX[4]对每个文件的判断还需 r 次伪随机函数作用),效率较低.

2) 采用“关键词-文件”构建思想的 SSE 方案处理以关键词为基本操作单位的检索操作时,只需 $O(1)$时间(若将所有文件读出,还需执行若干次索引解密操作,这里,次数与查找到的文件数目相关),具备较高的效率.由于文件更新的基本操作单位为文件,而该思想的基本构建单位为关键词,使得文件更新时,服务器需重新构建适应更新后状态以关键词为基本单位的索引,造成较大开销.

Table 1 Index-Based SSE scheme

表 1 基于索引的 SSE 方案

基于索引的 SSE 方案	服务器检索计算量	查询交互次数	陷门通信量	索引构建思想	文件更新
Z-IDX[5]	$O(n)$	1	$O(1)$	文件-关键词	更新布隆过滤器
PPSED-1[6]	$O(n)$	1	$O(1)$	文件-关键词	更新遮蔽索引
PPSED-2[6]	$O(n\|\Delta\|)$	2	$O(1)$		
SSE-1[7]	$O(1)$	1	$O(1)$	关键词-文件	重建索引
SSE-2[7]	$O(1)$	1	$O(n)$	关键词-文件	重建索引

注:n 表示文件集合中的文件数目,Δ表示关键词字典

3.4 扩展研究

近年来,SSE 的研究集中于对基本方案的功能扩展和安全性优化上,本节将总结 SSE 在这些方面的相关研究进展情况.

3.4.1 密文文件动态更新

基于“关键词-文件”思想的构建方案具有较高的查询效率,但在处理文件更新时,需重建索引;基于“文件-关键词”思想的构建方案具有较高的文件更新效率,但查询速度较慢.如何将这两者的优势结合起来,以同时支持高效的查询和文件更新,是研究者长期关注的一个问题.

最早关于密文文件动态更新的尝试始于文献[7],Van Liesdonk 等人[7]提出了两个新型 SSE 方案:方案 1 具有较低的计算复杂度,但关键词检索和文件更新过程要求服务器与客户端两次交互,占用了较大的网络带宽;方案 2 只需一次交互即可完成检索等操作,但由于使用了哈希函数链,文件更新时的计算复杂度与总更新次数呈线性关系.

直到 2012 年,Kamara 等人[8]才首次提出了支持子线性时间内的关键词检索和密文文件的高效动态更新的动态 SSE 方案.Kamara 等人[8]的方案基于了 SSE-1[7](重定义了 SSE-1 中的数组和速查表为检索数组 A_s 和检索速查表 T_s),并结合“文件-关键词”的索引构建思想,引入额外的删除数组 A_d 和相应的速查表 T_d 以记录和追踪每个文件中包含的关键词.当需要进行远端文件的添加或删除时,用户发送相应的标记(token)至服务器,服务器利用这些标记更新相关数据结构和密文文件.最近,Kamara 等人[9]引入红黑树作为索引结构,使动态 SSE 能够支持多处理器并行处理.

3.4.2 查询方式扩展

SSE 查询方式扩展包括对以逻辑连接词连接的多个关键词的检索和模糊关键词检索.

3.4.2.1 多关键词检索

如何对以逻辑连接词连接的多个关键词进行检索,是一个有趣的问题.早期的 SSE 方案[1]大多采用:① 分别检索关键词组中的每个关键词,再根据逻辑连接词对各自检索结果进行后处理;② 为所有可能的多关键词检索组合设置元关键词(meta-keyword),并与相关文件关联(例如对于关键词 Bob,urgent 和 finance,可设置元关键词 Bob:urgent:finance,并与所有包含 Bob,urgent 和 finace 的文件关联).然而,方法①使服务器获得比检索结果更多的额外信息,例如,服务器能够知道某些单个关键词包含于哪些密文文件;方法②造成服务器存储量的指数级增长,例如,具有 m 个不同关键词的文件集合 D,可能需要为其设置 2^m 个元关键词以适应不同查询组合.

因此,迫切需要探索一种多关键词检索的直接实现方案,文献[10]最早研究了此类问题.目前,关于多关键词检索的研究大多集中于对“逻辑与”连接词的支持上,表 2 列出了当前主要多关键词检索方案.

效率方面,GSW-1[10]和 SCKS-SS[11]具有明显优势,无需双线性对运算,只通过简单的指数运算和伪随机函数作用即可完成多关键词检索的过程.然而传送陷门时,这两种方案都需 $O(n)$通信量,造成网络负载过重;

安全性方面,这些多关键词检索方案的安全性通常依赖于某种数学难题,所依赖的数学难题的强弱也反映了方案安全性的高低.

Table 2 Multi-Keywords search scheme

表 2 多关键词检索方案

方案	通信量	服务器端存储量	检索中对运算次数	安全模型	安全假设
GSW-1[10]	$(n+1)\|q\|+\log\|W\|$	$\sum_{i=1}^{n}(l_i+1)\|q\|$	0	RO	DDH
GSW-2[10]	$3\|q\|+\log\|W\|$	$\sum_{i=1}^{n}(2l_i+1)\|q\|$	3	RO	非标准假设
SCKS-SS[11]	$n\|q\|+\log\|W\|$	$\sum_{i=1}^{n}2l_i\|q\|$	0	SM	伪随机函数存在
SCKS-XDH[11]	$2\|q\|+\log\|W\|$	$\sum_{i=1}^{n}(l_i+1)\|q\|$	2	SM	MXDH
ECKS-PS[12]	$2\|q\|+\log\|W\|$	$\sum_{i=1}^{n}(l_i+2)\|q\|$	2	RO	MDBDH

注:这里,n 为服务器端存储的文件数目,l_i 为文件 D_i 中的关键词数目,$|W|$为待查询关键词组中的关键词数目,$|q|$为使用到的群$\mathbb{G}$中元素所占存储空间,RO 和 SM 分别表示在随机预言机模型和标准模型下可证安全

3.4.2.2 模糊关键词检索

原有的 SSE 方案由于缺乏对细微文字以及格式错误的容忍,使其无法适用于云计算.鉴于此,文献[13]提出了模糊关键词检索的方案.该方案中,采用编辑距离来定义和度量关键词间的相似度,并使用了基于通配符和基于克(gram)的两种模糊关键词集构造方法.关键词查询时,用户计算待查询关键词在编辑距离门限下的模糊陷门集合并交给服务器,服务器使用陷门集与存储的模糊关键词集进行一一匹配,返回可能包含 W 的密文文件集合.2012 年,Wang 等人[14]进一步研究模糊关键词检索方案,并给出了形式化的安全性证明.

文献[15]提出包含通配符的关键词检索方法,类似于 Z-IDX[4],也将文件包含的关键词插入布隆过滤器.所不同的是,为避免关联攻击,该方法产生基于文件标识符的伪随机数,用以对布隆过滤器的二进制向量进行遮蔽.最后,在包含通配符的关键词检索功能的实现上,该方法通过预先为关键词生成所有通配检索形式(例如,关键词 flower 的所有通配检索形式包括 flower,*flower,flower*,*lower,…,flowe*,*ower,…,flow*,*wer,f*er,fl*r,flo*,*er,f*r,fl*,*r,f*等),插入索引,从而将包含通配符的关键词检索转化为精确匹配检索.

3.4.3 查询结果优化

原有的SSE方案仅支持布尔检索(即,判断某个关键词是否存在于文件中)而不能追踪关键词与文件的相关度.对于查询返回结果,用户必须进行后处理:解密所有返回文件,以获取目标文件.然而,相对于返回结果,目标文件可能只是其中单个或极少一部分,这就导致大量不必要的解密操作.因此,迫切需要优化查询结果,降低用户端后处理工作开销.

一种可能的优化办法是:在返回结果中,按照关键词与文件的相关度对加密文件进行排序,并将排序结果返回给用户.这样,用户只需根据相关度解密少量密文即可获得目标文件.目前,关于此方面的研究包括:

1) 文献[16]提出保护隐私的排序检索算法,采用保序加密算法加密文件中关键词词频.关键词查询时,首先检索出包含关键词的密文文件;然后,对用保序算法加密的词频对应的密文信息进行排序处理;最后,将相关度高的密文文件返回给用户.

2) 文献[17]提出云计算环境下的排序检索方案,对于每个关键词 W,计算其与被包含文件 D 的相关度,并使用一对多保序映射加密后,作为文件 D 节点的部分信息插入倒排链接索引表.关键词检索时,服务器通过陷门找到包含 W 的密文文件及其与 W 的相关度顺序,返回给用户.

3) 文献[18]提出支持多关键词查询的排序检索算法 MRSE.该方案为每个文件 D_i 分配$|\Delta|$二进制向量 A_i,并

以 $A_i[j]$描述关键词 W_j是否存在于文件 D_i中,这里,Δ为由所有关键词构成的字典;然后,采用增维、分割的方法生成$|\Delta|+1$ 维向量 A_i' 和 A_i'',最后形成 D_i 的索引 $I_i=(M_1^T A_i', M_2^T A_i'')$.这里,$M_1$,$M_2$ 均为$(|\Delta|+1)\times(|\Delta|+1)$维矩阵密钥.关键词查询时,生成$|\Delta|$维查询向量 Q,并类似地对 Q 进行增维、分割处理,形成陷门 $T=(M_1^{-1}Q', M_2^{-1}Q'')$.服务器遍历所有文件,计算 $I_i\cdot T$ 后对结果排序返回.

3.4.4 安全性优化

Chai 考虑半可信且好奇(semi-honest-but-curious)模型下,服务器可能为了节省其计算量或者下载带宽而返回错误的搜索结果或者部分搜索结果.为抵抗这种攻击,Chai[19]提出了可验证的对称可搜索加密方案(verifiable symmetric searchable encryption,简称 VSSE),其中引入了基于哈希的检索树,要求服务器将检索路径的哈希序列作为证据一并返还给用户,用户可根据证据对服务器的检索结果进行完整性和正确性验证.Kurosawa 等人[19]进一步形式化地定义了 SSE 的隐私性和可靠性,证明了其与 UC(universal composability)安全性的等价性,并提出了满足检索结果可验证性的 SSE 构造方案.

4 非对称可搜索加密

本节将围绕定义、典型构造思想和应用扩展,对非对称可搜索加密研究成果进行综述,并分析总结其特点.

4.1 定 义

4.1.1 算法描述

Boneh 等人[2]在非对称密码体制中引入可搜索加密,提出 PEKS(public key encryption with keyword search)概念,算法描述如下.

定义 3(PEKS). 非对称密码体制下可搜索加密算法可描述为$\mathcal{PEKS}$=(*KeyGen*,*Encrypt*,*Trapdoor*,*Test*):

1) (*pk*,*sk*)=*KeyGen*(λ):输入安全参数λ,输出公钥 *pk* 和私钥 *sk*;
2) C_W=*Encrypt*(*pk*,*W*):输入公钥 *pk* 和关键词 *W*,输出关键词密文 C_W;
3) T_W=*Trapdoor*(*sk*,*W*):输入私钥 *sk* 和关键词 *W*,输出陷门 T_W;
4) *b*=*Test*(*pk*,C_W,$T_{W'}$):输入公钥 *pk*、陷门 $T_{W'}$和关键词密文 C_W,根据 *W* 与 *W*′的匹配结果,输出判定值 $b\in\{0,1\}$.

基于定义 2,Boneh 等人[2]提出不可信赖服务器路由问题的解决思路:Bob 使用 Alice 的公钥 *pk* 加密邮件和相关关键词,并将形如($\mathcal{PKE}$.*Encrypt*(*pk*,*MSG*),$\mathcal{PEKS}$.*Encrypt*(*pk*,W_1),…,$\mathcal{PEKS}$.*Encrypt*(*pk*,W_n))的密文发送至邮件服务器.这里,$\mathcal{PKE}$.*Encrypt* 为公钥密码加密算法,*MSG* 为邮件内容,W_1,…,W_n 为与 *MSG* 关联的关键词.Alice 将 $T_{\text{“urgent”}}$或 $T_{\text{“lunch”}}$长驻服务器,新邮件到来时,服务器自动对其关联的关键词执行与 $T_{\text{“urgent”}}$或 $T_{\text{“lunch”}}$相关的 *Test* 算法,如果输出 1,便将该邮件转发至 Alice 的手机或个人电脑.

4.1.2 算法一致性

加密算法的一致性是指解密与加密互为逆过程,即,对任意明文 *M*,使用公钥 *pk* 加密后得到密文 *C*,如果再使用 *pk* 对应的私钥 *sk* 解密,必能得到 *M*.PEKS 的一致性应满足:① 对任意关键词 *W*,Pr[*Test*(*pk*,*PEKS*(*pk*,*W*),*Trapdoor*(*sk*,*W*))=1]=1;② 对任意关键词 W_1,W_2 且 $W_1\neq W_2$,Pr[*Test*(*pk*,*PEKS*(*pk*,W_1),*Trapdoor*(*sk*,W_2))]=1]=0.文献[2]中的方法无法满足上述要求[24],鉴于此,Abdalla 等人[24]对如上所述的完美一致性进行扩展,定义针对 PEKS 的计算一致性和统计一致性.

计算一致性和统计一致性的定义都基于实验 $Exp^{consist}$.攻击者$\mathcal{A}$已知公钥 *pk*,其目标是通过一定次数访问随机预言机$\mathcal{O}_H(\cdot)$后($\mathcal{O}_H(\cdot)$以 PEKS 中使用的哈希函数 $H(\cdot)$响应$\mathcal{A}$的查询),输出关键词对(W_1,W_2),满足 $W_1\neq W_2$ 且 *Test*(*pk*,*PEKS*(*pk*,W_1),*Trapdoor*(*sk*,W_2))=1.$\mathcal{A}$具有攻击优势 $Adv^{consist}_{PEKS,\mathcal{O}_H}(\mathcal{A})=\Pr[Exp^{consist}_{PEKS,\mathcal{O}_H}(\mathcal{A})\Rightarrow \text{true}]$:

(1) 如果$\mathcal{A}$为任意攻击者且 $Adv^{consist}_{PEKS,\mathcal{O}_H}<\varepsilon$, 则该 PEKS 方案达到统计一致性;

(2) 如果$\mathcal{A}$为任意多项式时间攻击者且 $Adv^{consist}_{PEKS,\mathcal{O}_H}<\varepsilon$,则该 PEKS 方案达到计算一致性.

4.1.3 安全目标

PEKS 需要满足:① 没有陷门的服务器除文件长度外,无法获取任何文件信息;② 拥有陷门 T_W 的服务器能够检索到所有包含 W 的密文文件.基于此,文献[2]定义了 PEKS 在选择关键词攻击下的不可区分性安全:IND-CKA(indistinguishability under chosen keyword attack).如图 8 所示,IND-CKA 的定义基于实验 $Exp^{\text{IND-CKA}}$,挑战者 $\mathcal{C}$执行 *KeyGen* 算法生成公钥 *pk* 和私钥 *sk*,将 *pk* 交给攻击者 $\mathcal{A}$.$\mathcal{A}$在自适应查询若干次陷门预言机 $\mathcal{O}_{Trapdoor}(\cdot)$后,输出关键词对$(W_0,W_1)$.$\mathcal{C}$随机选取 $b\in\{0,1\}$,生成挑战密文 $C^*=Encrypt(pk,W_b)$.$\mathcal{A}$的目标是:在再次查询若干次陷门预言机 $\mathcal{O}_{Trapdoor}(\cdot)$后,输出判定值 b',如果 $b'=b$ 攻击成功,否则失败.$\mathcal{A}$的攻击优势为

$$Adv^{\text{IND-CKA}}_{PEKS,\mathcal{O}_{Trapdoor}}(\mathcal{A})=|2\cdot\Pr[Exp^{\text{IND-CKA}}_{PEKS,\mathcal{O}_{Trapdoor}}\Rightarrow \text{true}]-1|.$$

如果 $Adv^{\text{IND-CKA}}_{PEKS,\mathcal{O}_{Trapdoor}}<\varepsilon$, 则该 PEKS 方案达到 IND-CKA 安全.

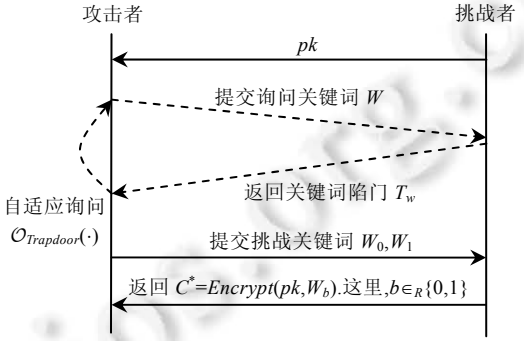


Fig.8 Games in PEKS security notions

图 8 PEKS 安全目标中的博弈过程

4.2 基本构建思想

4.2.1 PEKS 与 IBE 的相互变换

PEKS 和 IBE(完整定义可参考文献[21])都具有“判定”的共性:

- PEKS 使用 *Test* 算法判定陷门是否与某个关键词密文匹配;
- IBE 通过 *Decrypt* 算法进行判定:解密的私钥与加密的身份匹配时才能成功解密.

鉴于此种内在共性,探索 PEKS 与 IBE 的联系,并从中挖掘构建思想,成为众多研究者关注的热点.

Boneh 等人[2]使用 IND-CKA 安全的 PEKS 方案构造出明文空间$\mathcal{M}=\{0,1\}$,并达到 IND-ID-CCA (idistinguishability under identity-chosen ciphertext attack)安全的 IBE 方案.

假设采用的 PEKS 方案为$\mathcal{PEKS}=(KeyGen,Encrypt,Trapdoor,Test)$,则构造的 IBE 方案$\mathcal{IBE}=(Setup,Extract,Encrypt,Decrypt)$可描述如下:

1) $(param,mk)=\mathcal{IBE}.Setup(1^\lambda)$:输入安全参数 1^λ,输出系统参数和主密钥$(param,mk)=\mathcal{PEKS}.KeyGen(1^\lambda)$;
2) $sk_{id}=\mathcal{IBE}.Extract(param,mk,id)$:输入系统参数 *param*、主密钥 *mk* 和身份 $id\in\{0,1\}^*$,输出私钥:
$$sk_{id}=(T_{id||0},T_{id||1})=(\mathcal{PEKS}.Trapdoor(mk,id||0),\mathcal{PEKS}.Trapdoor(mk,id||1));$$
3) $Y=\mathcal{IBE}.Encrypt(param,id,X)$:输入系统参数 *param*、身份 $id\in\{0,1\}^*$和明文 $X\in\{0,1\}$,输出密文:
$$Y=\mathcal{PEKS}.Encrypt(param,id||X);$$
4) $X=\mathcal{IBE}.Decrypt(param,sk_{id},Y)$:输入系统参数 *param*、私钥 $sk_{id}=(T_{id||0},T_{id||1})$和密文 *Y*:
   - 如果 $Test(param,T_{id||0},Y)=1$,输出 0;
   - 如果 $Test(param,T_{id||1},Y)=1$,输出 1.

相反地,Abdalla 等人[24]描述了从 IBE 到 PEKS 的一般变换算法,能够将 IND-CPA(indistinguishability under chosen plain attack)和 ANO-CPA(anonymity under chosen plain attack)安全的 IBE 方案变换为 IND-CKA 安全,并满足计算一致性的 PEKS 方案.

假设采用的 IBE 方案为$\mathcal{IBE}$=(*Setup*,*Extract*,*Encrypt*,*Decrypt*),则构造的 PEKS 方案$\mathcal{PEKS}$=(*KeyGen*,*Encrypt*,*Trapdoor*,*Test*)可描述如下:

1) (*pk*,*sk*)=$\mathcal{PEKS}$.*KeyGen*(1^λ):输入安全参数 1^λ,输出公钥和私钥(*pk*,*sk*)=$\mathcal{IBE}$.*Setup*(1^λ);
2) C_W=$\mathcal{PEKS}$.*Encrypt*(*pk*,*W*):输入公钥 *pk* 和关键词 *W*,随机选择明文 $M\in\{0,1\}^\lambda$,执行 IBE 加密算法产生 *C*=$\mathcal{IBE}$.*Encrypt*(*pk*,*W*,*M*),输出关键词密文 C_W=(*C*,*M*);
3) T_W=*Trapdoor*(*sk*,*W*):输入私钥 *sk* 和关键词 *W*,输出陷门 T_W=$\mathcal{IBE}$.*Extract*(*pk*,*sk*,*W*);
4) *b*=$\mathcal{PEKS}$.*Test*(*pk*,$T_{W'}$,C_W):输入公钥 *pk*、陷门 $T_{W'}$和关键词密文 C_W=(*C*,*M*):
   - 如果$\mathcal{IBE}$.*Decrypt*(*param*,$T_{W'}$,*C*)=*M*,输出 1;
   - 否则,输出 0.

4.2.2 典型构造

目前的 PEKS 方案包括 BDOP-PEKS[2],KR-PEKS[22]和 DS-PEKS[23],都基于某种 IBE,它遵循一类最基本的 PEKS 构建策略,即:通过已有 IBE 方案由 IBE 到 PEKS 的变换算法,以构造安全的 PEKS.表 3 从通信量、运算效率以及所基于的数学假设等方面对 3 种典型构造进行了对比.

Table 3 Typical PEKS schemes comparison

表 3 PEKS 典型方案对比

方案	通信量	服务器端存储量	服务器检索效率	加密效率	数学假设
BDOP-PEKS[2]	$\|g\|$	$\sum_{i=1}^{n} l_i(\|g\|+\log p)$	1 次对运算	1 次对运算	BDH
KR-PEKS[22]	$6\|g\|$	$\sum_{i=1}^{n} l_i(3\|g\|+k)$	$O(1)$	$O(k)$	DDH
DS-PEKS[23]	$4\|g\|\log N$	$4\|g\|\log N\sum_{i=1}^{n} l_i$	$O(N)$	$O(N)$	QIP

注:*k* 为安全参数,$|g|$表示所用到的$\mathbb{Z}$或$\mathbb{G}$中元素所占用的存储空间,*p* 为$\mathbb{G}$的阶,*n* 为总文件数目,l_i为第 *i* 个文件中的关键词个数,*N* 为关键词字典中的关键词个数

由表 3 可知:① BDOP-PEKS[2]拥有较低用户服务器通信量,但是,加密和检索时都需一次对运算,效率较低;② KR-PEKS[22]拥有较高服务器检索效率,然而由于该方案只能抵御至多 *k* 次恶意陷门查询,通常需将 *k* 设置为较大数值,这会导致服务器端存储量增长;③ DS-PEKS[23]拥有较短关键词密文长度,检索和加密效率也较高,但要求服务器和用户间的交互需要占用较大带宽.

事实上,由于 PEKS 方案都是基于某种已有 IBE 方案构造而成,因此,PEKS 的性质也一定程度地反映了其源于 IBE 方案的性质.

4.3 关键词猜测攻击及其防御措施

文献[2]定义了关于 PEKS 的 IND-CKA 安全目标,并认为达到该目标的 PEKS 方案都是安全的.但是 PEKS 本身定义存在严重的安全隐患:文献[25,26]构造了针对 PEKS 方案[2,30,32,36]的关键词猜测攻击,关键词猜测攻击是由于关键词空间远小于密钥空间,而且用户通常使用常用关键词进行检索,这就给攻击者提供了只需采用字典攻击就能达到目的的“捷径”.

导致关键词猜测攻击的原因可归结为:

(1) 关键词空间较小,且用户集中于使用常用词汇,给攻击者提供了遍历关键词空间的可能;

(2) PEKS 算法一致性约束,使攻击者拥有对本次攻击是否成功的预先判定:执行 *Test* 算法,返回 1 说明本次攻击成功;否则,可以再继续猜测(该预先判定针对了实验 $Exp^{\text{IND-CKA}}$).

鉴于此:

1) 文献[28]提出了 PERKS 方案,要求接收者在预处理过程中执行关键词注册算法,将输出的预标签通过安全信道传送给发送者,发送者才能为注册后的关键词 *W* 生成密文 C_W.该方案实际上是对情形①的弥补,通过引入关键词注册过程,限制攻击者遍历关键词空间的能力.

2) 文献[29]提出了 PEFKS 方案,在服务器端进行模糊陷门测试,过滤大部分不相关邮件,最后在本地精确

匹配,得到检索结果.该方案通过引入模糊陷门,一定程度地降低了接收者外部 PEKS 算法的一致性,使其能够抵御关键词猜测攻击,但增加了客户服务器通信量和用户端计算量.

4.4 扩展研究

4.4.1 安全性完善

PEKS 中,发送者将邮件以($\mathcal{PKE}.Encrypt(pk,MSG)$,$\mathcal{PEKS}.Encrypt(pk,W)$)的形式通过服务器发送给接收者,安全要求仅考虑邮件密文中 PEKS 部分,而忽略整个系统(结合公钥密码和 PEKS)的隐私性,使恶意攻击者可利用$\mathcal{PKE}.Encrypt(pk,MSG)$与$\mathcal{PEKS}.Encrypt(pk,W)$的"间隙"实施某些攻击.例如:删除$\mathcal{PKE}.Encrypt(pk,MSG)$的部分内容,使接收者无法获得完整邮件;将服务器上的两封密文邮件的 PEKS 部分交换位置,使接收者无法检索到期望的邮件等.

针对此问题,文献[30]提出了 PKE/PEKS-1 和 PKE/PEKS-2 方案,在加密过程中除执行$\mathcal{PKE}.Encrypt(pk,MSG)$和$\mathcal{PEKS}.Encrypt(pk,W)$外,额外生成标签$\sigma$,用于邮件内容密文和关键词密文的相关性和完整性保护,只有在σ是($\mathcal{PKE}.Encrypt(pk,MSG)$,$\mathcal{PEKS}.Encrypt(pk,W)$)的合法标签时,才能进行测试.文献[31]在文献[30]的基础上提出了关于 PKE/PEKS 的新安全模型,增加对关键词隐私的要求,并描述了 PKE/PEKS 在标准模型下的一般构造框架.

另一方面,基本 PEKS 中,关键词陷门的传输需要在安全信道进行,否则,外部恶意攻击者能够通过公开信道截获和篡改陷门和查询结果.另外,某些恶意服务器也可以存储已查询的陷门与检索结果,以预测未来的查询结果.这就要求发送者预先指定服务器,在其上执行检索功能.

针对此问题,文献[32]提出了 dPEKS 方案,发送者在加密过程中使用接收者和服务器的公钥,使只有发送者指定的服务器才具备检索能力.该方案使用聚合技术,在随机预言机模型下可证安全.文献[33]定义了安全模型,描述了在具有恶意服务器和恶意接收者的内部攻击方式下 dPEKS 的安全目标.基于此,Rhee 等人[33]提出了 dPEKS 方案,在随机预言机模型下可证安全.另外,关于 dPEKS 的研究还包括文献[34,35],研究了能够抵御关键词猜测攻击的 dPEKS 方案.

4.4.2 查询方式扩展

4.4.2.1 多关键词检索

PECK(public key encryption with conjunctive field keyword search)的概念源于文献[36],其目的是在 PEKS 的基础上检索使用"逻辑与"连接的多个关键词的问题.典型方案见表 4.

Table 4 Comparison on PECK schemes

表 4 PECK 典型方案对比

方案	通信量	服务器端存储量	检索对运算次数	数学假设
PECK-1[36]	$2q+\log\|W\|$	$\sum_{i=1}^{n}(l_i+2)\|q\|$	1	DBDH
PECK-2[36]	$3q+\log\|W\|$	$\sum_{i=1}^{n}(2l_i+2)\|q\|$	2	DBDHI
HL-PECK[37]	$3q+\log\|W\|$	$\sum_{i=1}^{n}(l_i+2)\|q\|$	2	DLDH
HVE[38]	$3\|W\|+\log\|W\|$	$\sum_{i=1}^{n}(2l_i+2)\|N\|$	2	BDH

注:这里,n 为服务器端存储的文件数目;$|W|$为待查询关键词组中的关键词数目;l_i 为第 i 个文件中的关键词个数;$|q|$为素数 q 存储所占空间,这里假设使用到的群$\mathbb{G}$的阶为 q;$|N|$为合数 N 所占存储空间,这里,$N=pq$,p 和 q 均为素数

由表 4 可知:现有 PECK 方案几乎都基于双线性对运算;通信量、服务器端存储量和检索时双线性对运算次数方面,PECK-1[36]都具有较好的效果,然而该方案在加密关键词时,需执行 l_i 次对运算,效率很低;HVE[38]计算和存储耗费最大,却具有广泛适用性,可用于实现逻辑连接词连接的多关键词查询和数据库中的比较查询,成为后来查询方式扩展研究的主要参考.

4.4.2.2 模糊关键词检索

文献[39]使用 HVE[38]构造包含通配符的 PEKS 方案.HVE 中,每个密文 C 和密钥 K 都分别与一个二进制属性向量 $X=(X_1,X_2,\ldots,X_n)$和 $Y=(Y_1,Y_2,\ldots,Y_n)$相关联,这里,Y 中的某个分量不存在时,记为'*'.K 可以用来解密 C 当且

仅当 X 与 Y 在除‘*’外的所有分量都相同.包含通配符的 PEKS 中,将每个关键词视为属性向量:发送者使用邮件包含的关键词作为公钥,对该邮件执行 HVE 加密;接收者发送给服务器的陷门是待检索关键词的解密密钥,服务器试图执行 HVE 解密,如果解密成功,说明两个关键词除‘*’外所有分量都相同,从而达到包含通配符的关键词检索的目的.

5 工作展望

5.1 应用场景展望

由于可搜索加密技术能够提供安全的加密和密文直接检索功能,因此适合于外包敏感数据加密领域,除第 1.2 节中概述的个人数据外包加密和邮件信息加密外,还包括:

1) 外包数据库字段加密

在外包数据库中,由于服务提供商的不可信赖,对数据库中的敏感信息加密是非常必要的.然而,在数据库中加密数据一直是一个难题,因为加密数据库中的信息即意味着破坏该信息的本质特征,以致无法执行某些查询和检索操作.引入确定性的可搜索加密技术[42],使用数据拥有者的公钥分别对所有字段加密,可极大地提高数据库的安全性.同时,由于加密方式的确定性,也可使数据库系统结构仍然以标准数据结构组织和管理.目前,该领域已成为可搜索加密的一个新的研究热点.

2) 云计算中隐私数据的保护和共享

近年来,随着云计算的不断普及,面临的安全问题的重要性逐步上升,保护数据的隐私是云安全的一个重要课题[3].由于可搜索加密技术能够提供对密文进行基于关键词检索的功能,使其非常适用于云端隐私数据的保护,同时也不会牺牲对云端数据的提取和使用效率.在云端隐私数据的高效共享方面,可搜索加密也发挥巨大作用,其能有效地支持最基本形式的隐私数据共享,即,文件的发送和接收.结合基于属性加密和代理重加密技术,SE 能够适用于各种场景下高细粒度控制的云端数据共享.

3) 密文直接操作的相关应用

除了上述敏感信息的数据加密以外,SE 还非常适用于对密文数据进行直接操作的一类应用,这将是 SE 在未来的一个重要的应用领域.例如,文献[43]在邮件路由问题的基础上进一步考虑了另一种实际应用场景:假设发送者所发邮件中携带恶意程序,由于邮件已被加密,服务器无法检测其恶意内容,而直接交由接收者执行本地检查会增加终端计算负载.引入可搜索加密技术,通过接收者向服务器提供陷门,使服务器能够使用本地恶意代码特征库扫描匹配密文邮件恶意代码,无需解密查看邮件明文.

5.2 研究方向展望

作为基本工具,对称和非对称可搜索加密可与其他技术结合,应用于解决各种场景下的各种形式的关键词检索问题.表 5 根据当前和未来可能广泛应用的 SE 系统模型与关键词检索中查询策略和返回结果的灵活性,总结了当前的研究工作中已解决哪些问题,从而展望未来的研究方向.

Table 5 Research of SE

表 5 SE 研究情况总结

系统模型		单个关键词精确匹配查询	多关键词查询	模糊关键词查询	查询结果排序
单用户单服务器		√	√	√	√
多用户单服务器	多对一	√	√	√	×
	一对多	√	√	√	×
	多对多	√	√	√	×
单用户多服务器		×	×	×	×
多用户多服务器		×	×	×	×

注:多用户单服务器模型下,多对一为多个发送者一个接收者(与第 2.1 节中的多对一模型相同);一对多为一个发送者和多个接收者(与第 2.1 节中的一对多模型相同);多对多为多个发送者多个接收者的共享模式(与第 2.1 节中的多对多模型相同)

需要指出的是,除当前广泛研究和应用的4种系统模型外(详细介绍见第2.1节),我们认为,用户存储的多服务器模型将是未来发展的方向.简单地说,多服务器模型是对用户使用习惯和网络发展趋势的一种顺应.例如,随着如今云服务市场的兴起,针对同一类型的云服务,通常有多个服务提供商存在(从宏观角度来看,每个云服务提供商都可视为一个逻辑上的服务器).用户因此期望能够将个人数据分散存储到多个服务提供商处,然后通过多个提供商的协议合作,共同来为用户提供存储和关键词检索的服务.相比于单服务器在数据存储和处理方面的局限性,这种多个服务器存储文件的模式能够加强用户远程数据的可靠性,并可通过多个服务器协同处理提高对大型数据文件的处理能力.

通过对上述研究情况进行的总结和分析,我们认为,SE研究中仍存在着值得深入研究的问题,主要包括:

1) 多服务器系统模型下的关键词检索问题

相比于单服务器,多服务器的关键词检索涉及到了多个不可信赖实体,其难点在于如何设计一种执行于这些存储服务器的多方协议,使其能够协作地执行密文关键词检索的任务.传统的多方计算方法允许多个实体在分别保有自己秘密的前提下共同计算并得到基于所有实体持有秘密的结果,但难以适用于多服务器的关键词检索情景,这是因为用户并不希望最后协作计算检索到的文件(像多方计算的计算结果一样)泄漏给任何一个不可信赖服务器.因此,解决多服务器模型下的关键词检索问题成为应用这种模型前的一个迫切需求.

2) 多用户共享模式下潜在的密钥泄漏问题

虽然现有方案已能解决多用户共享模式下的关键词检索问题,但其基于较高的系统模型假设(例如,引入可信第三方)[44]或较低的安全目标(例如,允许服务器获得关键词信息)[45,46]或由于通过服务器重加密或其他复杂运算而导致效率较低[47−49].若要同时解决以上3个问题,需要所有用户共享同一私钥,这会导致密钥泄漏的潜在风险.因此,完善现有的多用户共享模式下的关键词检索机制,是未来需要研究的一个重要方向.

3) 多用户单服务器模型下的查询结果排序问题

文献[17,18]关于SE中查询结果排序的研究仅仅适用于广播模式,并要求广播者事先向授权用户发布必要的秘密参数以辅助查询.该技术难以推广到多个广播者的情况,且由于发送者与接收者之间的秘密沟通,限制了其应用范围.因此,未来需要探究一种更加完备、多用户单服务器模型下的查询结果排序机制.

4) PEKS的安全性问题

除以上问题之外,PEKS的安全性也需要受到关注.PEKS能够适用于最基本的共享模式,具有广阔的应用空间.然而,其安全性问题是阻碍PEKS应用的重要原因.如前所述,几乎所有PEKS都遭受猜测关键词攻击的潜在威胁.虽然PERKS[28]和PEFKS[29]能够抵御关键词猜测攻击,但牺牲了一定的性能:PERKS[28]要求构建安全通道完成关键词注册;PEFKS[29]的返回结果包含非目标文件,需进行本地二次精确陷门测试.因此,设计一种更加安全、高效的PEKS扩展方案,也是未来研究的方向之一.

6 结 论

可搜索加密为不可信服务器的密文检索提供了解决办法,已在云存储密文检索环境中得到了广泛应用.可搜索加密的提出,源于解决两类可搜索加密的基本问题:① 不可信赖服务器的存储问题;② 不可信赖服务器的路由问题.问题①推动了对称可搜索加密的研究进展,问题②推动了非对称可搜索加密的研究进展.两类可搜索加密均为不可信服务器的密文检索提供了解决办法,已在云存储密文检索环境中得到了广泛应用.本文围绕两类非对称可搜索加密进行了综述.

在对称可搜索加密方面,介绍了SSE定义、相关安全目标和典型构造,总结了两种基本SSE方案的构建策略:基于顺序扫描的构建策略和基于索引的构建策略.前者需要遍历整个文件,效率极低;后者通过构造索引,能够高效地支持关键词查询操作,已成为SSE方案构造的主要策略.分析了两种主流索引构建思想:“文件-关键词”和“关键词-文件”索引构建思想,指出操作处理的基本单位的不同决定了在完成这些操作时效率上的差异.介绍了SSE的相关扩展,主要表现在对查询方式的扩展、对查询结果的优化以及对方案本身安全机制的完善上.

在非对称可搜索加密方面,介绍了PEKS定义、算法一致性和相关安全目标,描述了PEKS的典型构造及其

与 IBE 的相互转换算法,指出现有 PEKS 的构造通常基于已提出的 IBE.安全性方面,对导致关键词猜测攻击的原因进行了分析,总结了针对关键词猜测攻击的预防措施.介绍了 PEKS 的相关扩展,主要表现在对安全性的改善和对查询方式的扩展上.

最后,总结和展望了可搜索加密的未来研究方向,以期对其在国内的研究起到一定的推动作用.

References:

[1] Song XD, Wagner D, Perrig A. Practical techniques for searches on encrypted data. In: Proc. of the IEEE Symp. on Security and Privacy. IEEE Press, 2000. 44−55. [doi: 10.1109/SECPRI.2000.848445]

[2] Boneh D, Di Crescenzo G, Ostrovsky R, Persiano G. Public key encryption with keyword search. In: Camenisch LJ, Cachin C, eds. Proc. of the Advances in Cryptology—EUROCRYPT 2004. LNCS 3027, Berlin: Springer-Verlag, 2004. 506−522. [doi: 10.1007/978-3-540-24676-3_30]

[3] Feng DG, Zhang M, Zhang Y, Xu Z. Study on cloud computing security. Ruan Jian Xue Bao/Journal of Softwase, 2011,22(1): 71−83 (in Chinese with English abstract). http://www.jos.org.cn/1000-9825/3958.htm [doi: 10.3724/SP.J.1001.2011.03958]

[4] Goh E. Secure indexes. Technical Report, 2003/216, IACR ePrint Cryptography Archive, 2003. http://eprint.iacr.org/2003/216

[5] Chang Y, Mitzenmacher M. Privacy preserving keyword searches on remote encrypted data. In: Ioannidis J, Keromytis A, Yung M, eds. Proc. of the Applied Cryptography and Network Security. LNCS 3531, Berlin: Springer-Verlag, 2004. 391−421. [doi: 10.1007/11496137_30]

[6] Curtmola R, Garay J, Kamara S, Ostrovsky R. Searchable symmetric encryption: Improved definitions and efficient constructions. In: Proc. of the 13th ACM Conf. on Computer and Communications Security (CCS 2006). New York: ACM Press, 2006. 79−88.

[7] Van Liesdonk P, Sedghi S, Doumen J, Hartel P, Jonker W. Computationally efficient searchable symmetric encryption. In: Jonker W, Petkovic M, eds. Proc. of the Secure Data Management. LNCS 6358, Berlin: Springer-Verlag, 2010. 87−100. [doi: 10.1007/978-3-642-15546-8_7]

[8] Kamara S, Papamanthou C, Roeder T. Dynamic searchable symmetric encryption. In: Proc. of the 19th ACM Conf. on Computer and Communications Security (CCS 2012). New York: ACM Press, 2012. 965−976. [doi: 10.1145/2382196.2382298]

[9] Kamara S, Papamanthou C. Parallel and dynamic searchable symmetric encryption. In: Sadeghi AR, ed. Proc. of the Financial Cryptography and Data Security. LNCS 7859, Berlin: Springer-Verlag, 2013. 258−274. [doi: 10.1007/978-3-642-39884-1_22]

[10] Golle P, Staddon J, Waters B. Secure conjunctive keyword search over encrypted data. In: Jakobsson M, Yung M, Zhou J, eds. Proc. of the Applied Cryptography and Network Security. LNCS 3089, Berlin: Springer-Verlag, 2004. 31−45. [doi: 10.1007/978-3-540-24852-1_3]

[11] Ballard L, Kamara S, Monrose F. Achieving efficient conjunctive keyword searches over encrypted data. In: Qing S, Mao W, Lopez J, Wang G, eds. Proc. of the Information and Communications Security. LNCS 3783, Berlin: Springer-Verlag, 2005. 414−426. [doi: 10.1007/11602897_35]

[12] Byun J, Lee D, Lim J. Efficient conjunctive keyword search on encrypted data storage system. In: Atzeni SA, Lioy A, eds. Proc. of the Public Key Infrastructure. LNCS 4043, Berlin: Springer-Verlag, 2006. 184−196. [doi: 10.1007/11774716_15]

[13] Li J, Wang Q, Wang C, Cao N, Ren K, Lou WJ. Fuzzy keyword search over encrypted data in cloud computing. In: Proc. of the 29th IEEE Int'l Conf. on Computer Communications (INFOCOM). IEEE Press, 2010. 1−5. [doi: 10.1109/INFCOM.2010.5462196]

[14] Wang C, Ren K, Yu SC, Urs KMR. Achieving usable and privacy-assured similarity search over outsourced cloud data. In: Proc. of the 31th IEEE Int'l Conf. on Computer Communications (INFOCOM). IEEE Press, 2012. 451−459. [doi: 10.1109/INFCOM.2012.6195784]

[15] Bösch C, Brinkman R, Hartel P, Jonker W. Conjunctive wildcard search over encrypted data. In: Jonker W, Petkovic, eds. Proc. of the Secure Data Management. LNCS 6933, Berlin: Springer-Verlag, 2011. 114−127. [doi: 10.1007/978-3-642-23556-6_8]

[16] Swaminathan A, Mao Y, Su GM, Gou H, Varna A, He S, Wu M, Oard D. Confidentiality-Preserving rank-ordered search. In: Proc. of the ACM Workshop on Storage Security and Survivability (StorageSS 2007). New York: ACM Press, 2007. 7−12. [doi: 10.1145/1314313.1314316]

[17] Wang C, Cao N, Li J, Ren K, Lou WJ. Secure ranked keyword search over encrypted cloud data. In: Proc. of the 30th Int'l Conf. on Distributed Computing Systems (ICDCS). IEEE Press, 2010. 253−262. [doi: 10.1109/ICDCS.2010.34]

[18] Cao N, Wang C, Li M, Ren K, Lou WJ. Privacy-Preserving multi-keyword ranked search over encrypted cloud data. In: Proc. of the 32th IEEE Int'l Conf. on Computer Communications (INFOCOM). IEEE Press, 2011. 829−837. [doi: 10.1109/INFCOM.2011.5935306]

[19] Chai Q, Gong G. Verifiable symmetric searchable encryption for semi-honest-but-curious cloud servers. In: Proc. of the IEEE Int'l

Conf. on Communications. IEEE Press, 2012. 917−922. [doi: 10.1109/ICC.2012.6364125]

[20] Kurosawa K, Ohtaki Y. UC-Secure searchable symmetric encryption. In: Keromytis A, ed. Proc. of the Financial Cryptography and Data Security. LNCS 7397, Berlin: Springer-Verlag, 2012. 285−298. [doi: 10.1007/978-3-642-32946-3_21]

[21] Boneh D, Franklin M. Identity-Based encryption from the Weil pairing. In: Kilian J, ed. Proc. of the Advances in Cryptology—CRYPTO 2001. LNCS 2139, Berlin: Springer-Verlag, 2001. 213−229. [doi: 10.1007/3-540-44647-8_13]

[22] Khader D. Public key encryption with keyword search based on *K*-resilient IBE. In: Gavrilova M, Gervasi O, Kumar V, *et al*., eds. Proc. of the Computational Science and Its Applications—ICCSA 2006. LNCS 3982, Berlin: Springer-Verlag, 2006. 298−308. [doi: 10.1007/978-3-540-74484-9_95]

[23] Crescenzo GD, Saraswat V. Public key encryption with searchable keywords based on Jacobi symbols. In: Srinathan K, Rangan PC, Moti Y, eds. Proc. of the Progress in Cryptology—INDOCRYPT 2007. LNCS 4859, Berlin: Springer-Verlag, 2007. 282−296. [doi: 10.1007/978-3-540-77026-8_21]

[24] Abdalla M, Bellare M, Catalano D, Kiltz E, Kohno T, Lange T, Malone Lee J, Neven G, Paillier P, Shi HX. Searchable encryption revisited: Consistency properties, relation to anonymous IBE, and extensions. In: Shoup V, ed. Proc. of the Advances in Cryptology—CRYPTO 2005. LNCS 3621, Berlin: Springer-Verlag, 2005. 205−222. [doi: 10.1007/s00145-007-9006-6]

[25] Byun J, Rhee H, Park HA, Lee DH. Off-Line keyword guessing attacks on recent keyword search schemes over encrypted data. In: Jonker W, Petkovic M, eds. Proc. of the Secure Data Management. LNCS 4165, Berlin: Springer-Verlag, 2006. 75−83. [doi: 10.1007/11844662_6]

[26] Yau WC, Heng SH, Goi BM. Off-Line keyword guessing attacks on recent public key encryption with keyword search schemes. In: Rong C, Jaatun MG, Sandnes FE, *et al*., eds. Proc. of the Autonomic and Trusted Computing. LNCS 5060, Berlin: Springer-Verlag, 2008. 101−105. [doi: 10.1007/978-3-540-69295-9_10]

[27] Jeong IR, Kwon JO, Hong D, Lee DH. Constructing PEKS schemes secure against keyword guessing attacks is possible. Computer Communications, 2009,32(2):394−396. [doi: 10.1016/j.comcom.2008.11.018]

[28] Tang Q, Chen L. Public-Key encryption with registered keyword search. In: Martinelli F, Preneel B, eds. Proc. of the Public Key Infrastructures, Services and Applications. LNCS 6391, Berlin: Springer-Verlag, 2010. 163−178. [doi: 10.1007/978-3-642-16441-5_11]

[29] Xu P, Jin H. Public-Key encryption with fuzzy keyword search: A provably secure scheme under keyword guessing attack. IEEE Trans. on Computers, 2012,62(11):2266−2277. [doi: 10.1109/TC.2012.215]

[30] Baek J, Safavi NR, Susilo W. On the integration of public key data encryption and public key encryption with keyword search. In: Katsikas KS, Lopez J, Backes M, *et al*., eds. Proc. of the Information Security. LNCS 4176, Berlin: Springer-Verlag, 2006. 217−232. [doi: 10.1007/11836810_16]

[31] Zhang R, Imai H. Generic combination of public key encryption with keyword search and public key encryption. In: Bao F, Ling S, Okamato T, *et al*., eds. Proc. of the Cryptology and Network Security. LNCS 4856, Berlin: Springer-Verlag, 2007. 159−174. [doi: 10.1007/978-3-540-76969-9_11]

[32] Baek J, Safavi NR, Susilo W. Public key encryption with keyword search revisited. In: Gervasi O, Murgante B, Lagana A, *et al*., eds. Proc. of the Computational Science and Its Applications—ICCSA 2008. LNCS 5072, Berlin: Springer-Verlag, 2008. 1249−1259. [doi: 10.1007/978-3-540-69839-5_96]

[33] Rhee HS, Park JH, Susilo W, Lee DH. Improved searchable public key encryption with designated tester. In: Proc. of the 4th Int'l Symp. on Information, Computer, and Communications Security (ASIACCS 2009). New York: ACM Press, 2009. 376−379. [doi: 10.1145/1533057.1533108]

[34] Rhee HS, Susilo W, Kim HJ. Secure searchable public key encryption scheme against keyword guessing attacks. IEICE Electronics Express, 2009,6(5):237−243. [doi: 10.1587/elex.6.237]

[35] Hu C, Liu P. A secure searchable public key encryption scheme with a designated tester against keyword guessing attacks and its extension. In: Lin S, Huang X, eds. Proc. of the Advances in Computer Science, Environment, Ecoinformatics, and Education. CCIS 215, Berlin: Springer-Verlag, 2011. 131−136. [doi: 10.1007/978-3-642-23324-1_23]

[36] Park D, Kim K, Lee P. Public key encryption with conjunctive field keyword search. In: Lim HC, Yung M, eds. Proc. of the Information Security Applications. LNCS 3325, Berlin: Springer-Verlag, 2005. 73−86. [doi: 10.1007/978-3-540-31815-6_7]

[37] Hwang Y, Lee P. Public key encryption with conjunctive keyword search and its extension to a multi-user system. In: Takagi T, Okamoto T, Okamoto E, *et al*., eds. Proc. of the Pairing-Based Cryptography—Pairing 2007. LNCS 4575, Berlin: Springer-Verlag, 2007. 2−22. [doi: 10.1007/978-3-540-73489-5_2]

[38] Boneh D, Waters B. Conjunctive, subset, and range queries on encrypted data. In: Vadhan PS, ed. Proc. of the Theory of

Cryptography. LNCS 4392, Berlin: Springer-Verlag, 2007. 535−554. [doi: 10.1007/978-3-540-70936-7_29]

[39] Sedghi S, Van Liesdonk P, Nikova S, Hartel P, Jonker W. Searching keywords with wildcards on encrypted data. In: Garay AJ, Prisco DR, eds. Proc. of the Security and Cryptography for Networks. LNCS 6280, Berlin: Springer-Verlag, 2010. 138−153. [doi: 10.1007/978-3-642-15317-4_10]

[40] Li M, Yu SC, Cao N, Lou WJ. Authorized private keyword search over encrypted data in cloud computing. In: Proc. of the 31st Int'l Conf. on Distributed Computing System (ICDCS). Minneapolis: IEEE Press, 2011. 383−392. [doi: 10.1109/ICDCS.2011.55]

[41] Waters B, Balfanz D, Durfee G, Smetters DK. Building an encrypted and searchable audit log. In: Proc. of the 11th Annual Network and Distributed System Security Symp. 2004. http://www.isoc.org/ndss04/proceedings/Papers/Waters.pdf

[42] Bellare M, Boldyreva A, O'Neill A. Deterministic and efficiently searchable encryption. In: Menezes A, ed. Proc. of the Advances in Cryptology—CRYPTO 2007. LNCS 4622, Berlin: Springer-Verlag, 2007. 535−552. [doi: 10.1007/978-3-540-74143-5_30]

[43] Ibraimi L, Nikova S, Hartel P, Jonker W. Public-Key encryption with delegated search. In: Lopez J, Tsudik G, eds. Proc. of the Applied Cryptography and Network Security. LNCS 6715, Berlin: Springer-Verlag, 2011. 532−549. [doi: 10.1007/978-3-642-21554-4_31]

[44] Li JW, Li J, Liu ZL, Jia CF. Enabling efficient and secure data sharing in cloud computing. Concurrency and Computation: Practice and Experience, 2014,26(5):1052−1066. [doi: 10.1002/cpe.3067]

[45] Canard S, Fuchsbauer G, Gouget A, Laguilaumie F. Plaintext-Checkable encryption. In: Dunkelman O, ed. Proc. of the Topics in Cryptology—CT-RSA 2012. LNCS 7178, Berlin: Springer-Verlag, 2012. 332−348. [doi: 10.1007/978-3-642-27954-6_21]

[46] Li JW, Li J, Chen XF, Liu ZL, Jia CF. Privacy-Preserving data utilization in hybrid clouds. Future Generation Computer Systems, 2014,30:98−106. [doi: 10.1016/j.future.2013.06.011]

[47] Dong C, Russello G, Dulay N. Shared and searchable encrypted data for untrusted servers. Journal of Computer Security, 2011, 19(3):367−397. [doi: 10.1007/978-3-540-70567-3_10]

[48] Popa RA, Zeldovich N. Multi-Key searchable encryption. Technical Report, 2013/508, IACR ePrint Cryptography Archive, 2013. https://eprint.iacr.org/2013/508

[49] Zheng Q, Xu S, Ateniese G. VABKS: Verifiable attribute-based keyword search over outsourced encrypted data. Technical Report, 2013/462, IACR ePring Cryptography Archive, 2013. https://eprint.iacr.org/2013/462

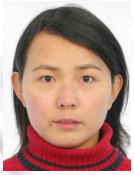
附中文参考文献:

[3] 冯登国,张敏,张妍,徐震.云计算安全研究.软件学报,2011,22(1):71−83. http://www.jos.org.cn/1000-9825/3958.htm [doi: 10.3724/SP.J.1001.2011.03958]

李经纬(1987－),男,四川成都人,博士生,主要研究领域为信息安全,应用密码学.

李进(1981－),男,博士,教授,主要研究领域为云计算中服务外包的安全和隐私问题.

贾春福(1967－),男,博士,教授,博士生导师,主要研究领域为信息安全与可信计算,恶意代码发现与分析.



李敏(1975－),女,博士生,主要研究领域为信息安全,应用密码学.

刘哲理(1978－),男,博士,主要研究领域为密码学及应用,智能卡操作系统.