(19)国家知识产权局

(12)发明专利申请

(10)申请公布号 CN 117951730 A
(43)申请公布日 2024.04.30

(21)申请号 202311732782.3
(22)申请日 2023.12.15
(71)申请人 南京航空航天大学深圳研究院
地址 518063 广东省深圳市南山区粤海街道高新南四道19号虚拟大学园R4栋A区218室
申请人 南京航空航天大学
(72)发明人 王昊 王慎卿 李明慧 殷常春 张佳乐
(74)专利代理机构 杭州大道知识产权代理有限公司 33525
专利代理师 张荣鑫
(51)Int.Cl.
*G06F 21/62*(2013.01)
*G06F 16/22*(2019.01)
*G06F 16/2458*(2019.01)

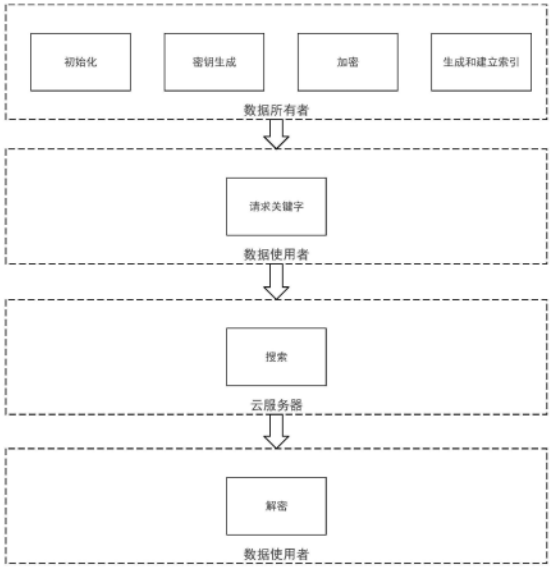
权利要求书2页 说明书9页 附图2页

(54)发明名称

一种基于哈希索引的云端安全可搜索加密方法

(57)摘要

本发明公开了一种基于哈希索引的云端安全可搜索加密方法,包括数据所有者、云服务器,其中,数据所有者持有文件集合,包括若干个文件、若干个关键词;包括如下子步骤:S1:设定有限域和椭圆曲线群,以此设定哈希函数一、二、三;S2:选择密钥,生成屏蔽密钥;S3:数据所有者选择一个关键词的权重,并获取得到加密权重;S4:获取关键词集合,建立身份信息集合,并获取身份信息随机数集合;S5:为身份信息计算获取其权重,加密得到加密权重,建立一个可搜索的索引;S6:云服务器为文件集合中的文件提取对应的加密权重,建立二叉查找树保存对应数据;S7:数据用户获取文件的数据,并获取权重,随后计算获取最大权重;S8:获取最大权重对应的文件。




CN 117951730 A

1.一种基于哈希索引的云端安全可搜索加密方法,其特征在于:包括数据所有者、云服务器,其中,数据所有者持有文件集合,所述文件集合包括若干个文件,包括若干个关键词;

包括如下子步骤:

S1:设定有限域和椭圆曲线群,以此设定哈希函数一、哈希函数二和哈希函数三;

S2:在有限域中选择密钥,通过密钥生成屏蔽密钥;

S3:数据所有者从文件集合随机选择一个关键词的权重,并通过哈希函数一和屏蔽密钥获取得到加密权重;

S4:数据所有者通过文件集合获取关键词集合,根据关键词集合建立身份信息集合,通过哈希函数三映射获取身份信息随机数集合;

S5:数据所有者为身份信息计算获取其权重,通过加密得到对应的加密权重,并建立一个可搜索的索引;

S6:云服务器通过一个查询向量为文件集合中的每个文件提取对应的加密权重,建立二叉查找树保存对应的文件的身份信息和加密权重;

S7:数据用户对数据所有者进行申请,获取文件的身份信息和加密权重,通过屏蔽密钥和哈希函数二对加密权重进行解密获取权重,并计算获取最大权重;

S8:通过最大权重找到最大权重对应的文件。

2.如权利要求1所述的一种基于哈希索引的云端安全可搜索加密方法,其特征在于:其中,哈希函数一通过有限域映射到椭圆曲线群中;哈希函数二通过椭圆曲线群映射到有限域中;哈希函数三通过有限域映射到有限域中。

3.如权利要求1所述的一种基于哈希索引的云端安全可搜索加密方法,其特征在于:所述S3包括如下子步骤:

S31:随机选择一个权重;

S32:通过哈希函数一将权重映射到椭圆曲线群中的一个椭圆曲线点;

S33:通过屏蔽密钥对椭圆曲线点进行加密,得到一个加密权重。

4.如权利要求1所述的一种基于哈希索引的云端安全可搜索加密方法,其特征在于:所述S4包括如下子步骤:

S41:数据所有者根据文件集合选择若干个关键词,并选择若干个伪关键词;

S42:数据所有者将伪关键词加入关键词中形成关键词集合;

S43:数据所有者为关键词集合中的每一个元素选择一个随机的身份信息,生成身份信息集合,其中身份信息数量与关键词、伪关键词数量的总和相同;

S44:对身份信息集合的每一个身份信息通过哈希函数三映射一个身份信息随机数,生成身份信息随机数集合。

5.如权利要求1所述的一种基于哈希索引的云端安全可搜索加密方法,其特征在于:所述S5包括如下子步骤:

S51:数据所有者通过身份信息集合计算每一个身份信息的身份信息权重,并对身份信息权重进行加密得到身份信息加密权重;

S52:数据所有者通过身份信息随机数集合、与之相关的身份信息的加密权重、对应的文件生成一个可搜索的索引。

6.如权利要求1所述的一种基于哈希索引的云端安全可搜索加密方法,其特征在于:所

述S6包括如下子步骤:

S61:云服务器获取一个查询向量,所述查询向量包括身份信息,产生关键词集合映射到身份信息随机数集合的哈希值;

S62:通过查询向量为每个文件提取对应的身份信息随机数的加密权重;

S63:构建一个二叉查找树保存对应的文件的身份信息和加密权重。

7.如权利要求6所述的一种基于哈希索引的云端安全可搜索加密方法,其特征在于:如果多个关键词出现在一个文件中,则通过同态加法来添加加密权重;其中矩体表现为:对于每个被查询向量请求的关键词的文件,在二叉查找树中均只创建一个节点,当文件有多个被请求的关键词,则对加密权重进行累加;最后映射所有请求的关键词和哈希表,并在二叉查找树中添加加密权重。

8.如权利要求1所述的一种基于哈希索引的云端安全可搜索加密方法,其特征在于:所述S7包括如下子步骤:

S71:数据用户通过查询向量生成对多个关键词的请求,获取对应的加密权重;

S72:通过屏蔽密钥对加密权重进行计算,获取获取椭圆曲线点;

S73:通过哈希函数二对椭圆曲线点进行计算获取其加密前的权重;

S74:计算获取最大权重。

9.如权利要求8所述的一种基于哈希索引的云端安全可搜索加密方法,其特征在于:所述S74包括如下子步骤:

S741:初始化最大变量,设置为0;

S742:遍历二叉查找树的每一个节点,对于每个节点,首先获取其加密前的权重;

S743:对权重进行判定,若权重大于1,则将该权重的值赋予最大变量;反之,则继续;

S744:在遍历完成后,最大变量的值即为最大权重。

# 一种基于哈希索引的云端安全可搜索加密方法

## 技术领域

[0001] 本发明涉及网络空间安全技术领域,特别涉及一种基于哈希索引的云端安全可搜索加密方法。

## 背景技术

[0002] 云计算提供了数据存储和计算作为一种服务。访问的灵活性为数据所有者(Do)提供了存储和访问数据的便利,而无需考虑存储位置、容量或维护。数据共享很重要,因为它通过更高的生产力和更好的决策来改善业务或数据质量。然而,云存储,也增加了敏感信息的脆弱性,如健康记录、财务信息、政府文件或敏感信息等。

[0003] 一个数据所有者通常需要授权多个用户安全地访问其数据文件的。然而,云服务器的好奇性质不可避免地将数据处于危险之中。因此,数据所有者和好奇的云服务器都属于不同的信任区。为了控制数据暴露,敏感数据在外包到云之前应该对其进行加密,云必须只向合法用户提供服务,而不侵犯数据隐私。然而,加密增加了有效的数据搜索和利用的挑战。为了提供数据的可访问性,基于关键字的搜索机制是一种一般方法,这种基于关键字的搜索机制允许有效地搜索用户所需的文件,而不需要检索所有文件。但是,关键字隐私成为了数据隐私的要求。

[0004] 云服务安全的重点是避免在搜索操作过程中或之后泄露信息。通过严格的隐私保护方法,如可搜索加密(SE),可以改善搜索过程和有效的数据利用。SE允许搜索加密的关键字,并向服务器透露尽可能少的信息,并保护用户数据的隐私。提出了许多基于可搜索对称加密(SSE)方案和关键字搜索(PEKS)方案的SE方案。SSE方案只允许一个用户执行关键字搜索和访问加密的数据。这意味着这些方案是固定在单个用户中的。随后,在多用户设置下,采用关键词搜索技术,开发了多用户可搜索加密方案。多用户设置中的关键字搜索允许数据所有者与一组授权用户共享他们的文档集合。正如预期的那样,PEKS提供了更好的安全级别,但具有较高的计算开销、不正确的安全定义,以及实际上不适合大型数据集。在大多数PEKS计划中,存在内部攻击的安全,如诚实但好奇的服务器作为内部人没有被考虑。

[0005] 尽管在可搜索加密领域取得了进展,但许多问题仍未得到解决。首先,诚实但好奇的云服务器或其他对手仍然可以设法破坏隐私,而这种方案对术语和文件的相似性相关性没有足够的稳健性。第二,传统的基于公钥的方案有很大的复杂性。

[0006] Wang等人提出了一种新颖的多关键字模糊搜索方案,可以直接在云端的加密数据上进行关键字搜索。但这导致了不准确的搜索结果,并且缺乏有效性和安全性之间的权衡,面临着与布尔表示法有关的问题。另一方面,Ziqing等人提出了一种方法,该方法使用平衡二叉树和带有文件排名的查询向量,在多所有者环境的加密云数据上进行多关键词排名搜索。对于二进制树,即使关键字不存在于特定文件中,也需要分配空间。这导致了高内存消耗。因此,这种如何在云环境下保证可搜索加密的性能、准确性和内存问题亟待进一步研究。具体地,该技术的技术方案包括以下步骤:

[0007] 1.多关键字模糊搜索方案:Wang等人为了不增加索引或搜索复杂性的情况下有效

支持多个关键字的模糊搜索,提出一种新颖的多关键字模糊搜索方案,可直接对云中的加密数据进行关键字搜索。通过算法设计实现模糊匹配,而不是扩展索引文件,无需预定义字典,即可对加密的云数据实现多关键字模糊搜索。引入一种全新的想法,即通过将每个关键字转换为其双元向量表示形式并利用欧几里得距离来捕获关键字的相似度,实现多关键字(连词关键字)模糊搜索。

[0008] 2.安全的多关键字排名搜索方案:Ziqing等人为了解决以前的安全可搜索方案仅支持搜索属于单一所有者的数据,不能搜索不同数据所有者外包的多个数据集的问题。提出了使用可信第三方进行密钥管理,使用向量空间模型来生成索引和查询,并使用新设计的KDO算法来提供关键字权重,该算法同时考虑了相关性和文档质量。非对称标量产品保留加密方法用于加密加权索引和查询以保护隐私。提出的分组平衡二叉树索引通过Greedy Depth-First搜索算法提高了搜索效率。

[0009] 现有技术的缺点:

[0010] 然而论文在实现中为了实现模糊匹配,而不是扩展索引文件,并且在不增加索引或搜索复杂性的情况下有效地支持多关键字模糊搜索,但这导致了不准确的搜索结果,并且缺乏有效性和安全性之间的权衡,面临着与布尔表示法有关的问题。论文中为了实现搜索多数据所有者数据集,假设数据所有者是诚实的,不相互串通。如果数据所有者串通,则拟议计划的安全性可能会受到损害。在试图实现预期目标的同时,该方案使用可信的代理来为数据用户创建加密的索引和陷阱门。这有被受信任的代理人破坏的漏洞。此外,对于二进制树,即使关键字不存在于特定文件中,也需要分配空间,导致高内存消耗。因此,这种如何在云环境下保证可搜索加密的性能、准确性和内存问题亟待进一步研究。

## 发明内容

[0011] 本发明的目的在于提供一种基于哈希索引的云端安全可搜索加密方法,以克服现有技术中的不足。

[0012] 为实现上述目的,本发明提供如下技术方案:

[0013] 本申请公开了一种基于哈希索引的云端安全可搜索加密方法,包括数据所有者、云服务器,其中,数据所有者持有文件集合,所述文件集合包括若干个文件,包括若干个关键词;

[0014] 包括如下子步骤:

[0015] S1:设定有限域和椭圆曲线群,以此设定哈希函数一、哈希函数二和哈希函数三;

[0016] S2:在有限域中选择密钥,通过密钥生成屏蔽密钥;

[0017] S3:数据所有者从文件集合随机选择一个关键词的权重,并通过哈希函数一和屏蔽密钥获取得到加密权重;

[0018] S4:数据所有者通过文件集合获取关键词集合,根据关键词集合建立身份信息集合,通过哈希函数三映射获取身份信息随机数集合;

[0019] S5:数据所有者为身份信息计算获取其权重,通过加密得到对应的加密权重,并建立一个可搜索的索引;

[0020] S6:云服务器通过一个查询向量为文件集合中的每个文件提取对应的加密权重,建立二叉查找树保存对应的文件的身份信息和加密权重;

[0021] S7:数据用户对数据所有者进行申请,获取文件的身份信息和加密权重,通过屏蔽密钥和哈希函数二对加密权重进行解密获取权重,并计算获取最大权重;

[0022] S8:通过最大权重找到最大权重对应的文件。

[0023] 作为优选,其中,哈希函数一通过有限域映射到椭圆曲线群中;哈希函数二通过椭圆曲线群映射到有限域中;哈希函数三通过有限域映射到有限域中。

[0024] 作为优选,所述S3包括如下子步骤:

[0025] S31:随机选择一个权重;

[0026] S32:通过哈希函数一将权重映射到椭圆曲线群中的一个椭圆曲线点;

[0027] S33:通过屏蔽密钥对椭圆曲线点进行加密,得到一个加密权重。

[0028] 作为优选,所述S4包括如下子步骤:

[0029] S41:数据所有者根据文件集合选择若干个关键词,并选择若干个伪关键词;

[0030] S42:数据所有者将伪关键词加入关键词中形成关键词集合;

[0031] S43:数据所有者为关键词集合中的每一个元素选择一个随机的身份信息,生成身份信息集合,其中身份信息数量与关键词、伪关键词数量的总和相同;

[0032] S44:对身份信息集合的每一个身份信息通过哈希函数三映射一个身份信息随机数,生成身份信息随机数集合。

[0033] 作为优选,所述S5包括如下子步骤:

[0034] S51:数据所有者通过身份信息集合计算每一个身份信息的身份信息权重,并对身份信息权重进行加密得到身份信息加密权重;

[0035] S52:数据所有者通过身份信息随机数集合、与之相关的身份信息的加密权重、对应的文件生成一个可搜索的索引。

[0036] 作为优选,所述S6包括如下子步骤:

[0037] S61:云服务器获取一个查询向量,所述查询向量包括身份信息,产生关键词集合映射到身份信息随机数集合的哈希值;

[0038] S62:通过查询向量为每个文件提取对应的身份信息随机数的加密权重;

[0039] S63:构建一个二叉查找树保存对应的文件的身份信息和加密权重。

[0040] 作为优选,如果多个关键词出现在一个文件中,则通过同态加法来添加加密权重;其中矩体表现为:对于每个被查询向量请求的关键词的文件,在二叉查找树中均只创建一个节点,当文件有多个被请求的关键词,则对加密权重进行累加;最后映射所有请求的关键词和哈希表,并在二叉查找树中添加加密权重。

[0041] 作为优选,所述S7包括如下子步骤:

[0042] S71:数据用户通过查询向量生成对多个关键词的请求,获取对应的加密权重;

[0043] S72:通过屏蔽密钥对加密权重进行计算,获取获取椭圆曲线点;

[0044] S73:通过哈希函数二对椭圆曲线点进行计算获取其加密前的权重;

[0045] S74:计算获取最大权重。

[0046] 作为优选,所述S74包括如下子步骤:

[0047] S741:初始化最大变量,设置为0;

[0048] S742:遍历二叉查找树的每一个节点,对于每个节点,首先获取其加密前的权重;

[0049] S743:对权重进行判定,若权重大于1,则将该权重的值赋予最大变量;反之,则继

续;

[0050] S744:在遍历完成后,最大变量的值即为最大权重。

[0051] 本申请公开了一种基于哈希索引的云端安全可搜索加密装置,包括存储器和一个或多个处理器,所述存储器中存储有可执行代码,所述一个或多个处理器执行所述可执行代码时,用于上述的一种基于哈希索引的云端安全可搜索加密方法。

[0052] 本申请公开了一种计算机可读存储介质,其上存储有程序,该程序被处理器执行时,实现上述的一种基于哈希索引的云端安全可搜索加密方法。

[0053] 本发明的有益效果:

[0054] (1)、在本发明针对搜索复杂度问题所提出的对称加密方法。多关键字模糊搜索方案,对文件的搜索应该以线性搜索的形式进行,因为存储格式和加密算法都是以这种方式设计。在本方法中,可以直接使用哈希函数获取各自的关键词,并遍历链接列表以提取所有具有该关键词的文件;

[0055] (2)、本发明针对高内存消耗的问题所提出的可叠加的TF-IDF权重方案比安全的多关键字排名搜索方案在消耗的存储量方面减少50%;

[0056] (3)、本发明所提出的方法的很重要的一个模块是搜索模块。这个模块是由数据使用者通过生成查询向量来完成的。实验是在三个数据集上进行的。结果表明,数据集的大小对搜索请求没有任何影响,因为分数是预先计算的。搜索操作不需要遍历整个文件系统,它只需要找到散列的索引条目并给出相应文件的总分。本方法不需要在每次产生搜索时扫描整个文件系统,搜索时间的要求只取决于找到哈希索引和分数的添加操作。为了更深入地分析提议的方案,形成了两个主要的关键词类别。一是最常用的关键词,二是很少使用的关键词。对2、4、8和10个常用的关键词以及2和4个很少使用的关键词进行了搜索请求。进行这一分析的动机是为了发现请求中的关键词数量如何导致更多的遍历和加法操作。结果表明,不同关键词的数据集大小不同,其结果与数据集大小无关。实验结果表明,搜索时间的消耗与关键词的通用性和要求的关键词数量直接相关。我们可以用结果来标明,更多的关键词会带来更多的附加结果。与具有稀有关键词的关键词集相比,常见的关键词会调用更多的数据匹配和添加操作。

[0057] 本发明的特征及优点将通过实施例结合附图进行详细说明。

## 附图说明

[0058] 图1是本发明一种基于哈希索引的云端安全可搜索加密方法的流程示意图;

[0059] 图2是本发明的主要内容步骤示意图;

[0060] 图3是本发明的装置示意图。

## 具体实施方式

[0061] 为使本发明的目的、技术方案和优点更加清楚明了,下面通过附图及实施例,对本发明进行进一步详细说明。但是应该理解,此处所描述的具体实施例仅仅用以解释本发明,并不用于限制本发明的范围。此外,在以下说明中,省略了对公知结构和技术的描述,以避免不必要地混淆本发明的概念。

[0062] 参阅图1～2,本发明实施例提供一种基于哈希索引的云端安全可搜索加密方法,

该方法中,通过对现有基于哈希索引和基于椭圆曲线的加法同态加密算法在云服务安全问题进行深入分析研究,研究基于哈希索引的方法在云服务中如何减轻计算负荷问题。本发明提出了一种基于哈希索引的云端安全可搜索加密方案,以保证云数据的安全性。首先,与传统基于公钥的方案相比,使用基于哈希的索引减少了云服务器和用户的计算负荷。其次,与使用平衡二叉树和带有文件排名的查询向量的方法相比,使用基于椭圆曲线的ElGamal加法同态加密,避免了对复杂的陷阱系统和二进制查询方案的生成和资源密集型处理的需要。本方法评估和比较了所提出的框架与其他最先进的可搜索加密方案在云服务器上的搜索时间、复杂性和索引存储复杂性。因此,本发明所提出的一种基于哈希索引的云端安全可搜索加密方法,使基于哈希索引的加密算法更适用于数据外包的云服务场景中的应用;

[0063] 一种基于哈希索引的云端安全可搜索加密方法,包括数据所有者、云服务器;

[0064] 其中:数据所有者是数据集的贡献者和管理者。将数据文件外包给云服务器,供授权的数据用户使用。

[0065] (1)首先,数据所有者使用对称加密方法对文件进行加密,以使数据保密。数据所有者还列出了一组关键字,并将这些关键字分发给经过身份验证的数据用户。对于搜索安排,数据所有者用关键字和相关的权重分数来构建索引。在该方案中,数据所有者选择关键字后,以随机方式给每个关键字一个唯一标识。数据所有者负责计算权重分数,并使用EC ElGamal方法加密这些值(指的是权重分数)。最后,数据所有者将加密的数据和加密的索引外包给云服务器,并与授权的数据用户共享密钥s、屏蔽密钥和ECC参数。在该方案中,数据所有者将一些虚拟关键字与实际的关键字集混合。

[0066] (2)云服务器提供了数据托管设施,并存储了由数据所有者外包的加密文件和索引,为数据用户提供了针对匹配的关键字请求的搜索设施。在所提出的方案中,我们选择在云服务器侧保持最大的潜在处理和计算,以保持数据用户的舒适,我们希望允许最小的计算负担。所提出的方案避免了云服务器的许多操作,因为我们认为云服务器是诚实但奇怪的。云服务器存储加密的数据和加密的索引,处理搜索查询并提供匹配加密数据发送给请求数据用户。这项工作认为,云服务器将不会对数据集或加密的索引执行任何其他操作;

[0067] 云服务器在这个过程中负责根据查询向量Q搜索哈希表,提取加密权重,并构建包含文件标识符和加密权重的二叉搜索树,最终形成用户所需的查询结果。

[0068] (3)数据用户接收到数据所有者共享的关键字列表及其唯一标识、对称密钥和其他ECC参数。数据用户以查询向量的形式请求查询并发送到云服务器,收到返回值后解码,然后请求所需文件,然后用相应的对称密钥解密。

[0069] 其中,数据所有者持有文件集合,所述文件集合包括若干个文件,包括若干个关键词;

[0070] 包括如下子步骤:

[0071] S1:设定有限域和椭圆曲线群,以此设定哈希函数一、哈希函数二和哈希函数三;

[0072] S2:在有限域中选择密钥,通过密钥生成屏蔽密钥;

[0073] S3:数据所有者从文件集合随机选择一个关键词的权重,并通过哈希函数一和屏蔽密钥获取得到加密权重;

[0074] S4:数据所有者通过文件集合获取关键词集合,根据关键词集合建立身份信息集合,通过哈希函数三映射获取身份信息随机数集合;

[0075] S5:数据所有者为身份信息计算获取其权重,通过加密得到对应的加密权重,并建立一个可搜索的索引;

[0076] S6:云服务器通过一个查询向量为文件集合中的每个文件提取对应的加密权重,建立二叉查找树保存对应的文件的身份信息和加密权重;

[0077] S7:数据用户对数据所有者进行申请,获取文件的身份信息和加密权重,通过屏蔽密钥和哈希函数二对加密权重进行解密获取权重,并计算获取最大权重;

[0078] S8:通过最大权重找到最大权重对应的文件。

[0079] 其中,哈希函数一通过有限域映射到椭圆曲线群中;哈希函数二通过椭圆曲线群映射到有限域中;哈希函数三通过有限域映射到有限域中。

[0080] 所述S3包括如下内容:

[0081] 首先进行初始化,其中输入是一个安全参数p,输出是在有限域$F_p$上的一个椭圆曲线群G,哈希函数$H_1:F_p\rightarrow G$,,哈希函数$H_2:G\rightarrow F_p$和哈希函数$H_3:F_p\rightarrow F_p$。数据所有者输入G,其中P是群G的生成点,数据所有者选择一个随机数$s\in F_p$,形成一个屏蔽密钥$mk=sP$。

[0082] 然后数据所有者随机在$w\in\{0,999\}$中选择一个权重,其中w表示一个关键字的tf-idf权重,使用哈希函数$H_1$映射到一个椭圆曲线点$P_M=H_1(M)$。使用EC-ElGamal加密算法加密$P_M$,得到加密后的tf-idf权重$w'=P_M+mk$。其中mk是是用来加密权重的掩蔽密钥。这里,权重表示范围在0-999之间的标准化tf-idf权重。

[0083] 包括如下子步骤:

[0084] S31:随机选择一个权重;

[0085] S32:通过哈希函数一将权重映射到椭圆曲线群中的一个椭圆曲线点;

[0086] S33:通过屏蔽密钥对椭圆曲线点进行加密,得到一个加密权重。

[0087] 所述S4包括如下内容:

[0088] 使用EC-ElGamal加密允许在加密的权值上进行同态加法。这使得云服务器可以添加多个加密的权重。此外,标准化的权值允许在二叉检索树上进行有效的搜索。数据所有者包含一个文件集合$F=f_1,f_2,...,f_n$。然后数据所有者从F中选择m个关键词$K=k_1,k_2,...,k_m$。并且选择d个伪关键字$K_d=k_{d1},k_{d2},...,k_{dd}$加到K中形成$K_t=K+K_d$。数据所有者为每个$k_j\in K_t$选择一个随机的身份$kid_j\in F_p$,其中$0\leqslant j\leqslant m+d$,因此$KID_t=\{kid_1,kid_2,...,kid_{m+d}\}$.并且为每个$kid\in KID_t$使用哈希函数$H_3$来映射一个随机数$kid_j\in F_p$。其中$HKID_t=\{kid_1,kid_2,...,kid_{m+d}\}$。

[0089] 包括如下子步骤:

[0090] S41:数据所有者根据文件集合选择若干个关键词,并选择若干个伪关键词;

[0091] S42:数据所有者将伪关键词加入关键词中形成关键词集合;

[0092] S43:数据所有者为关键词集合中的每一个元素选择一个随机的身份信息,生成身份信息集合,其中身份信息数量与关键词、伪关键词数量的总和相同;

[0093] S44:对身份信息集合的每一个身份信息通过哈希函数三映射一个身份信息随机数,生成身份信息随机数集合。

[0094] 所述S5包括如下内容:

[0095] 数据所有者为每个$kid_j$计算出tf-idf的权重$w_j$,然后加密$w_j$得到$w'_j$。

[0096] 最后,数据所有者使用$HKID_t$和他相关的加密权重$w'_j$和文件$f_i$生成一个可搜索的

索引$I=kid_j\leftarrow\{(F_1,w'_{j1}),(F_2,w'_{j2}),\ldots,(F_n,w'_{jn})\}$。其中$0\leqslant j\leqslant m+d$,在这里,每个关键字标识的加密tf-idf权重w对于每个文件都是不同的。

[0097] 包括如下子步骤:

[0098] S51:数据所有者通过身份信息集合计算每一个身份信息的身份信息权重,并对身份信息权重进行加密得到身份信息加密权重;

[0099] S52:数据所有者通过身份信息随机数集合、与之相关的身份信息的加密权重、对应的文件生成一个可搜索的索引。

[0100] 所述S6包括如下内容:

[0101] 云服务器取包含$kid_j$的查询向量$\vec{Q}$,产生哈希值$kid_j\leftarrow k_j$。它在哈希表中找到$kid_j$,并遍历链接列表,为每个文件$f_i$提取$kid_j$的加密权重$w'_{ji}$。它构造了一个二叉查找树来保存提取的文件$f_i$的id和它加密的权重$w'_{ji}$。

[0102] 如果多个关键字标识出现在一个文件中,那么加密的tf-idf权重w'将使用同态加法来添加。因此,对于每个有被请求的关键字的文件,在树中只创建一个节点,如果文件有多个被请求的关键字,则累加它的w'。最后映射所有请求的关键字与哈希表,并在二叉查找树中添加w',结果是为用户准备好的。

[0103] 包括如下子步骤:

[0104] S61:云服务器获取一个查询向量,所述查询向量包括身份信息,产生关键词集合映射到身份信息随机数集合的哈希值;

[0105] S62:通过查询向量为每个文件提取对应的身份信息随机数的加密权重;

[0106] S63:构建一个二叉查找树保存对应的文件的身份信息和加密权重。

[0107] 如果多个关键词出现在一个文件中,则通过同态加法来添加加密权重;其中矩体表现为:对于每个被查询向量请求的关键词的文件,在二叉查找树中均只创建一个节点,当文件有多个被请求的关键词,则对加密权重进行累加;最后映射所有请求的关键词和哈希表,并在二叉查找树中添加加密权重。

[0108] 所述S7包括如下内容:

[0109] 数据使用者通过生成查询向量$\vec{Q}=\{kid_j\}$,生成它对多个关键字的请求。其中$0\leqslant j\leqslant p$.在这里,p表示每个查询的关键字标识的总数可能有所不同。最后使用EC-ElGamal解密,首先计算$P_M=w'-mk$,其中是mk掩蔽密钥,计算$w=H_2(w')$,首先,我们选择一个变量Max并将其初始化为0。遍历BST中的每个节点,并为每个文件$f_i$获取$w'_i$。在mk的帮助下,从$w'_i$中恢复$P_M$。$w_i$最终使用$H_2$从$P_M$中恢复,如果假设$w_i$大于1,在第一次迭代时将Max设置为$w_i$。该过程重复n次,最后将值最大的$w_i$存储在Max中。最后,数据使用者将向服务器请求权重$w_i$等于Max的文件$f_i$进行进一步检索;

[0110] 解密和计算权重:对于每个存储在BST中的节点,包含了一个文件标识符fi和相应的加密权重w'为了解密这些权重,首先需要使用相应的屏蔽密钥mk。使用屏蔽密钥mk,从w'中恢复出加密前的椭圆曲线点PM,即PM=w'-mk。

[0111] 从椭圆曲线点恢复权重:使用哈希函数H2,将从步骤1中获得的椭圆曲线点P M映射回加密前的权重wi。具体来说,计算H2(PM)得到加密前的权重值。

[0112] 计算最大权重:初始化一个变量Max,将其初始值设置为0。然后遍历BST中的每个

节点,针对每个节点执行以下步骤:

[0113] a.从步骤2中获得的加密前的权重值wi。

[0114] b.如果wi大于1(这是因为只有权重大于1的才有意义),则将Max设置为wi。

[0115] c.持续比较当前节点的加密前权重值wi与Max,如果更大,则更新Max。

[0116] 找到最大权重文件:在遍历完成后,Max中存储了所有文件的加密前权重中的最大值。最终,将具有最大加密前权重的文件标识符fi找出来,这是需要由数据用户(DU)向云服务器(CS)进一步检索的文件

[0117] 包括如下子步骤:

[0118] S71:数据用户通过查询向量生成对多个关键词的请求,获取对应的加密权重;

[0119] S72:通过屏蔽密钥对加密权重进行计算,获取获取椭圆曲线点;

[0120] S73:通过哈希函数二对椭圆曲线点进行计算获取其加密前的权重;

[0121] S74:计算获取最大权重。

[0122] 所述S74包括如下子步骤:

[0123] S741:初始化最大变量,设置为0;

[0124] S742:遍历二叉查找树的每一个节点,对于每个节点,首先获取其加密前的权重;

[0125] S743:对权重进行判定,若权重大于1,则将该权重的值赋予最大变量;反之,则继续;

[0126] S744:在遍历完成后,最大变量的值即为最大权重。

[0127] 本发明中:针对搜索复杂度问题,本发明中,文件采用对称加密的方法进行加密。安全索引是关键字标识(ID)的散列,以便与用户更快、安全地匹配搜索请求。其中还包括等价的tf-idf。tf-idf采用基于EC ElGamal加密系统进行加密,该系统提供了加性同态,即支持多关键字请求。在接收到搜索请求后,服务器可以成功地将每个文件的所有请求关键字的加密tf-idf权重添加到一起,并分别存储总权重。这产生了一个隐私保护排名多关键字搜索加密数据。数据用户可以自由选择他所需要的任何等级的文件。许多方法声称要提供top-k文件,但在实际生活中,top-k文件并不总是数据用户所需要的。

[0128] 针对高内存消耗的问题,从$\vec{Q}$中获取一个关键字标识kid,并使用哈希函数$H_3$计算其哈希值。如果散列映射到动态数组A的索引,则遍历相应索引指向的链表。包含文件及其相关加密权重的每个节点被添加到BST。现在,来自$\vec{Q}$的下一个kid被到来,这导致了三个案例。在案例1中,hkid=$H_3$(kid)没有在A的索引中映射,这意味着相关的关键字在f中不存在。在情况2中,如果hkid指向的新节点包含一个在BST中不存在的文件f中,则将其添加到它中。在情况3中,如果hkid指向一个节点,其中包含一个在BST中已经存在的文件f,则添加这两个节点的加密权值,并更新BST中f的权值。因此,每个文件在BST中只添加一次,而映射到同一文件的关键字标识数量越多,将意味着该文件在BST中的组合权重越高。因此,这样一不会造成内存浪费。

[0129] 本发明一种基于哈希索引的云端安全可搜索加密装置的实施例可以应用在任意具备数据处理能力的设备上,该任意具备数据处理能力的设备可以为诸如计算机等设备或装置。装置实施例可以通过软件实现,也可以通过硬件或者软硬件结合的方式实现。以软件实现为例,作为一个逻辑意义上的装置,是通过其所在任意具备数据处理能力的设备的处

理器将非易失性存储器中对应的计算机程序指令读取到内存中运行形成的。从硬件层面而言,如图3所示,为本发明一种基于哈希索引的云端安全可搜索加密装置所在任意具备数据处理能力的设备的一种硬件结构图,除了图3所示的处理器、内存、网络接口、以及非易失性存储器之外,实施例中装置所在的任意具备数据处理能力的设备通常根据该任意具备数据处理能力的设备的实际功能,还可以包括其他硬件,对此不再赘述。上述装置中各个单元的功能和作用的实现过程具体详见上述方法中对应步骤的实现过程,在此不再赘述。

[0130] 对于装置实施例而言,由于其基本对应于方法实施例,所以相关之处参见方法实施例的部分说明即可。以上所描述的装置实施例仅仅是示意性的,其中所述作为分离部件说明的单元可以是或者也可以不是物理上分开的,作为单元显示的部件可以是或者也可以不是物理单元,即可以位于一个地方,或者也可以分布到多个网络单元上。可以根据实际的需要选择其中的部分或者全部模块来实现本发明方案的目的。本领域普通技术人员在不付出创造性劳动的情况下,即可以理解并实施。

[0131] 本发明实施例还提供一种计算机可读存储介质,其上存储有程序,该程序被处理器执行时,实现上述实施例中的一种基于哈希索引的云端安全可搜索加密装置。

[0132] 所述计算机可读存储介质可以是前述任一实施例所述的任意具备数据处理能力的设备的内部存储单元,例如硬盘或内存。所述计算机可读存储介质也可以是任意具备数据处理能力的设备的外部存储设备,例如所述设备上配备的插接式硬盘、智能存储卡(Smart Media Card,SMC)、SD卡、闪存卡(Flash Card)等。进一步的,所述计算机可读存储介质还可以既包括任意具备数据处理能力的设备的内部存储单元也包括外部存储设备。所述计算机可读存储介质用于存储所述计算机程序以及所述任意具备数据处理能力的设备所需的其他程序和数据,还可以用于暂时地存储已经输出或者将要输出的数据。

[0133] 以上所述仅为本发明的较佳实施例而已,并不用以限制本发明,凡在本发明的精神和原则之内所作的任何修改、等同替换或改进等,均应包含在本发明的保护范围之内。

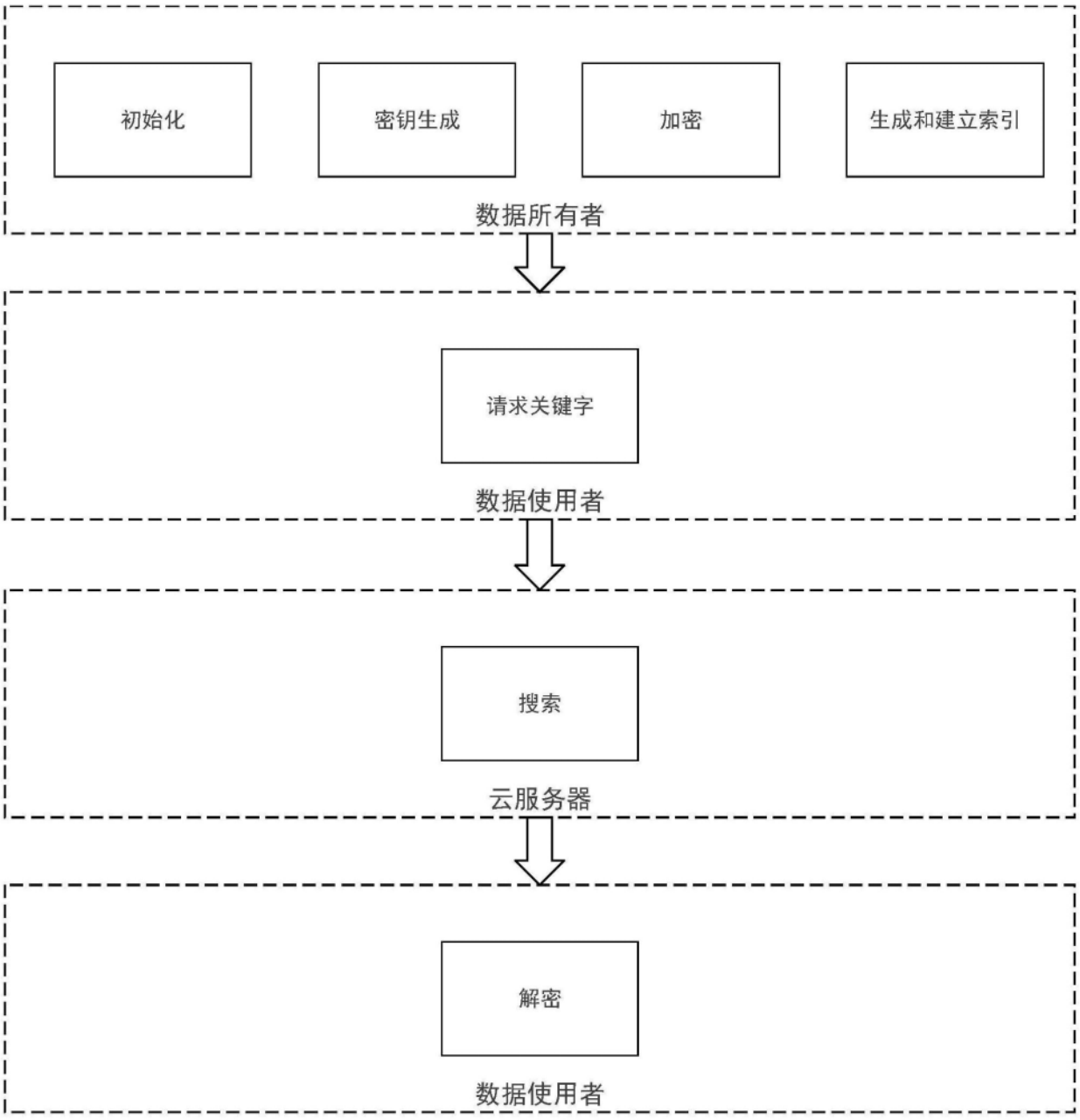



图1

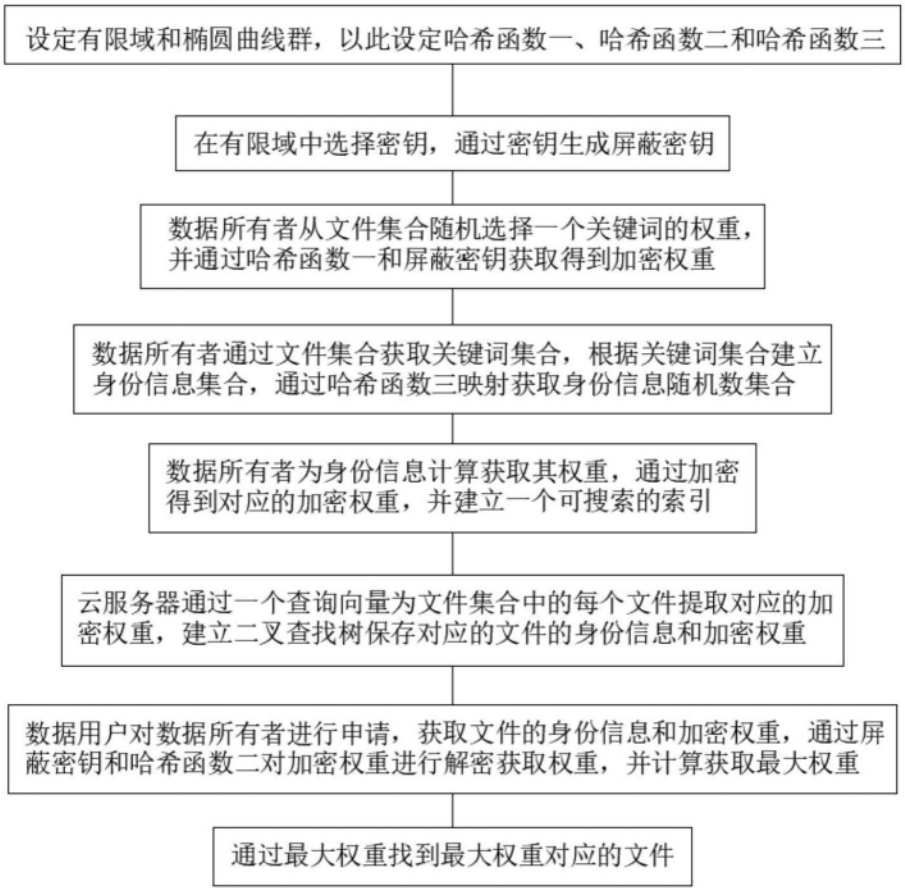

设定有限域和椭圆曲线群，以此设定哈希函数一、哈希函数二和哈希函数三
在有限域中选择密钥，通过密钥生成屏蔽密钥
数据所有者从文件集合随机选择一个关键词的权重，并通过哈希函数一和屏蔽密钥获取得到加密权重
数据所有者通过文件集合获取关键词集合，根据关键词集合建立身份信息集合，通过哈希函数三映射获取身份信息随机数集合
数据所有者为身份信息计算获取其权重，通过加密得到对应的加密权重，并建立一个可搜索的索引
云服务器通过一个查询向量为文件集合中的每个文件提取对应的加密权重，建立二叉查找树保存对应的文件的身份信息和加密权重
数据用户对数据所有者进行申请，获取文件的身份信息和加密权重，通过屏蔽密钥和哈希函数二对加密权重进行解密获取权重，并计算获取最大权重
通过最大权重找到最大权重对应的文件

图2

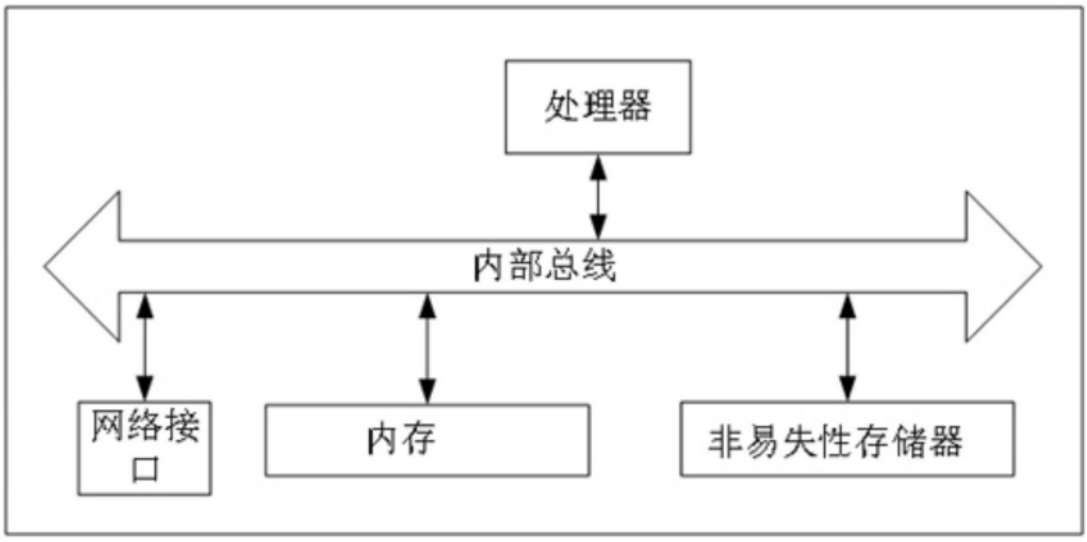

处理器
内部总线
网络接口
内存
非易失性存储器

图3