面向物联网搜索技术的高效访问控制方案

章园园 秦 岭
(南京工业大学计算机科学与技术学院 南京 211816)

摘 要 物联网搜索技术在日常生活中有着广泛应用,但由于物联网搜索引擎的开放性和搜索后台的不完全可信性,存储于搜索后台的信息存在严重的安全问题。针对该问题,提出一种安全、高效的支持密文搜索的属性基访问控制方案。在数据保护方面,为了确保用户属性信息和数据的安全,使用了访问策略部分隐藏和属性授权机构去中心化等方法,并且使用密文定长的方式提高算法效率和节约存储空间。同时,提出一种支持策略对比的属性撤销方案,降低了传统撤销方案中的计算复杂度,提高了重加密效率。在密文搜索方面,引入超级节点并使用混合索引的方式提高了检索效率。实验分析表明,该方案高效地解决了物联网搜索技术中的安全问题。

关键词 物联网搜索技术,访问策略部分隐藏,属性撤销,属性授权机构,密文搜索

中图法分类号 TP393 **文献标识码** A **DOI** 10.11896/j.issn.1002-137X.2019.08.032

Efficient Access Control Scheme for Internet of Things Search Technology

ZHANG Yuan-yuan QIN Ling
(School of Computer Science and Technology, Nanjing Tech University, Nanjing 211816, China)

Abstract Internet of Things search technology is widely used in daily life, however, due to the openness of the Internet of Things search engine and the incomplete credibility of the search center, information stored in the search background has serious security issues. This paper proposed a secure and efficient attribute-based access control scheme for supporting ciphertext search to solve this problem. In terms of data protection, in order to ensure the security of user attribute information and data, access policy partial hiding and attribute authority decentralization are used. Besides, ciphertext fixed length is used to improve algorithm efficiency and save storage space. At the same time, this paper proposed an attribute revocation scheme that supports policy comparison, which can reduce the computational complexity in the traditional revocation scheme and improve the efficiency of re-encryption. In the ciphertext search, the super peer is introduced and the hybrid index is used to improve the retrieval efficiency. The analysis results show that the solution effectively solves the security problem in the Internet of Things search technology.

Keywords Internet of Things search technology, Access strategy partially hidden, Property revocation, Attribute authority, Ciphertext search

1 引言

物体与网络通过物联网相连接,并利用传感器等技术随时采集实物的相关信息,从而对物体进行跟踪、定位和管理等[1-3]。随着云计算、大数据等技术的广泛运用,在生产生活中,快速、实时、有效地搜索到现实世界中物体的有关信息,并对这些信息进行高效的组织和管理已经变得越来越重要。比如搜索当前快递的位置信息,搜索从学校到健身房的最佳路线等,因此,物联网搜索技术应运而生。瑞士苏黎世联邦理工大学、德国吕贝克大学和德国都科摩通信实验室研发的Dyser搜索引擎支持静态和动态物体信息的搜索;Shodan搜索引擎提供在线设备,输入关键字即可搜索到与互联网相连的相关设备[4-8]。

物联网搜索技术已经渗透到人们生活的各个方面,如仓储和物流、健康医疗、环境监测等。物联网搜索技术在带来便利的同时,也存在着严重的数据安全隐私问题。物联网设备通常会收集日常生活中物体的信息,并将其交由智能对象进行存储、分析和管理,以便为用户提供各种搜索服务并返回满足搜索请求的信息。但是,如果物联网设备被攻击者恶意攻击和利用,则很有可能导致隐私数据的泄露[9]。

为了实现搜索过程中的数据安全保护,加密算法被广泛应用于物联网搜索技术中。基于属性的加密方案[10](ABE)由Sahai和Waters首次提出。现有的ABE方案包括密文与访问结构关联、密钥与用户属性关联的基于密文策略的属性加密方案(CP-ABE),以及密钥与访问结构关联、密文与用户属性关联的基于密钥策略的属性加密方案(KP-ABE)。由于

到稿日期:2018-07-17 返修日期:2018-11-01

章园园(1994-),女,硕士,主要研究方向为信息安全,E-mail:1427369987@qq.com;秦 岭(1980-),男,硕士,讲师,主要研究方向为工业信息化、工业系统集成,E-mail:ql@njtech.edu.cn(通信作者)。

CP-ABE 本身的特性[11-14]，即 CP-ABE 不需要完全可信赖的数据存储系统，因此，其更适合应用于物联网搜索技术。

Boneh 等首次提出一种基本的支持关键字搜索的公钥加密方案，但该方案的算法效率和安全性能都较低。文献[15]提出一种支持属性撤销的可检索的加密方案，但该方案中密钥密文的更新由数据拥有者、属性授权机构和系统共同完成，增加了通信代价。文献[16]提出一种支持多关键字搜索的加密方案，提高了搜索的准确性，但数据拥有者需要将加密后的关键字上传至系统，降低了加密效率。

针对以上问题，本文提出一种安全、高效的支持密文搜索的属性基访问控制方案。通过安全性证明和实验分析，证明了该方案是安全可行的。本文方案使用密文定长、属性授权机构去中心化和隐藏属性值等方法提高了加解密效率和数据安全性。此外，提出支持策略对比的属性撤销方案，提高了重加密效率。另外，密文更新由搜索系统完成，减轻了数据拥有者的计算压力。在上述属性基访问控制方案的基础上，本文还引入超级节点，使用混合索引和二次属性验证等方法，从而提高了密文搜索的效率并进一步保证了数据的安全性。

2　预备知识

2.1　双线性对

G 和 G_T 分别指阶为 p 和 q 的乘法循环群，p 和 q 为素数，则通常称映射 $e:G\times G\rightarrow G_T$ 为一个双线性对，且 e 满足下列性质[17]。

1）双线性：对于任意的 $a\in Z$，$b\in Z$ 和 $R\in G$，$S\in G$，均存在 $e(R^a,S^b)=e(R,S)^{ab}$。

2）非退化性：存在 $R\in G$，$S\in G$，有 $e(R,S)\neq 1$。

3）可计算性：对于任意 $R\in G$，$S\in G$，存在有效算法计算 $e(R,S)$。

4）对于任意 R_1，R_2，$S\in G_1$，都有 $e(R_1\cdot R_2,S)=e(R_1,S)e(R_2,S)$。

5）对于任意 R，S_1，$S_2\in G_1$，都有 $e(R,S_1\cdot S_2)=e(R,S_1)e(R,S_2)$。

2.2　访问结构

与密文相关的多个属性组成的属性集 A 构成本文的访问结构[18]。假设一个实体集合 $S=\{s_1,\cdots,s_n\}$，集合 A 构成的访问结构须满足下列条件：

1）集合 A 是单调的，即对于 $\forall C,D$，若当 $C\in A$ 且 $C\subseteq D$ 时，有 $D\in A$，则称 $A\subseteq 2^S$ 是单调的。

2）集合 A 是实体集合 $S=\{s_1,\cdots,s_n\}$ 的一个非空子集，即 $A\subseteq 2^S\backslash\{\phi\}$。

2.3　线性秘密共享方案(LSSS)

基于成员集 P 的线性秘密共享方案 Π 在 Z_p 上是线性的，并且其要满足下条件[19]：

1）每个成员所分配到的秘密构成一个 Z_p 上的矩阵。

2）Π 中存在一个 $l\times n$ 的秘密共享矩阵 $\boldsymbol{M}$，对于 $\forall i\in 1,\cdots,l$，矩阵 $\boldsymbol{M}$ 的第 i 行表示第 i 个成员，设列向量 $\boldsymbol{v}=(s,r_1,\cdots,r_n)$，其中 $s\in Z_p$ 是待分享的秘密，$r_1,\cdots,r_n\in Z_p$ 是随机的，$\boldsymbol{M}\cdot\boldsymbol{v}$ 则将秘密 s 根据 Π 分为 l 个部分。

2.4　安全模型

本文使用选择访问结构和选择明文攻击下的不可区分性(IND-SAS-CPA)的安全模型[20]。攻击游戏的具体过程如下：

初始化：攻击者创建其要挑战的访问结构 Γ^*，挑战者运行初始化算法并将公共参数和公钥分发给攻击者。

查询阶段 1：攻击者不断地提交不属于访问结构 Γ^* 的属性，并发出私钥请求，挑战者收到查询请求后，计算生成相应的私钥组件并发送给攻击者。

挑战：攻击者提交两个相同长度的信息 M_0 和 M_1。挑战者随机得到 $b\in\{0,1\}$，并在 Γ^* 下加密 M_b 获得密文 C^*，并将 C^* 发送给攻击者。

查询阶段 2：重复查询阶段 1 的过程，攻击者继续询问其他的属性私钥组件。

猜想：攻击者猜测哪条消息被加密，并输出猜想 b^*，如果 $b^*=b$，则表明攻击者挑战成功，在此过程中，攻击者的挑战优势为 $|\Pr(b^*=b)-\frac{1}{2}|$。

3　系统方案

3.1　系统模型

本文的系统模型如图 1 所示，包括 5 个参与者：数据拥有者(Data Owner，DO)、数据搜索者(Data Searcher，DS)、物联网搜索服务提供者(Search Service Provider，SSP)、多个属性授权机构(Attribute Authority，AA)和超级节点(Super Peers，SP)。

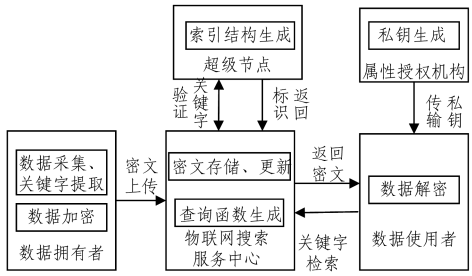

图 1　系统模型

Fig. 1　System model

3.1.1　数据拥有者

数据拥有者表示上传信息的个体或机构(传感器等)。其主要进行关键字提取操作和运行加密算法，并将加密后的数据上传到物联网搜索服务中心。若发生属性撤销，则数据拥有者动态更新密钥，并与物联网搜索服务提供者建立安全连接，由物联网搜索服务提供者完成密文更新。

3.1.2　数据搜索者

数据搜索者表示使用物联网搜索服务的个体或机构。个体或机构通过密文搜索技术从后台中心获取密文文件，向属性授权机构声明属性并请求属性私钥，当其属性满足访问结构时，则可获得用户私钥并解密密文。

3.1.3　物联网搜索服务提供者

物联网搜索服务提供者表示提供搜索服务的后台系统，属于不可完全信任的存储介质，容易受到攻击，并具有一定的计算能力和存储功能。本模型中，加密后的信息保存在物联

网搜索服务中心中，一切在服务中心进行操作的数据都是加密过的。

3.1.4 属性授权机构

属性授权机构主要负责用户属性的管理和用户私钥的生成和分发。每个授权机构根据数据搜索者提交的属性序列独立地计算用户私钥，并将生成的用户私钥分配给相应的数据搜索者。

3.1.5 超级节点

超级节点主要负责索引结构的建立和存储。超级节点将索引结构中的关键字和单射查询函数中的关键字进行比较，将关键字对应的 FID 和 UID 返回给物联网搜索服务提供者，物联网搜索服务提供者根据 FID 和 UID 将密文返回给数据使用者。

3.2 方案设计

本文提出一种支持密文搜索的属性基访问控制方案，该方案主要包括数据加密、属性撤销和密文检索 3 个方面，具体过程如下所述。

3.2.1 加密方案

与对称加密方案相比，ABE 加密方案的复杂度较高，因此本方案采用混合加密。首先使用对称加密（AES128）对数据进行加密，获得对称密钥 K，再使用本方案的加密方案对 K 进行加密，获得密文 C。

本文使用单调张成方案构造 LSSS，将访问结构转换为线性秘密共享矩阵，从而实现属性加密。属性由属性名和属性值两部分组成，设属性授权机构 $AA=(AA_1,\cdots,AA_n)$ 管理 N 个属性名，记为 $N=(a_1,\cdots,a_n)$，任意 a_i 拥有 t_i 个不同属性值，记为 $b_i=(a_{i,1},a_{i,2},\cdots,a_{i,t_i})$，则 AA 管理的属性集为 $S=(a_1:b_1,a_2:b_2,\cdots,a_n:b_n)$。$\Gamma(\boldsymbol{M},\rho,Z)$ 为访问结构，其中，$\boldsymbol{M}$ 为 $l\times n$ 的矩阵，$\rho(i)$ 将矩阵的第 i 行与第 i 个属性名相映射，其中 $z_{\rho(i)}$ 表示属性名 $\rho(i)$ 的值。

1）系统初始化 $GlobalSetup(1^\lambda)\rightarrow PP$

该过程由系统执行。G 和 G_T 指阶为素数 p 的乘法循环群，g 和 g_1 是 G 的生成元，并有 $e:G\times G\rightarrow G_T$，$\lambda$ 为随机参数。具体执行过程如下：

输入 λ，定义哈希函数：$H_0:\{0,1\}^{2\lambda}\rightarrow Z_p$，$H_1:Z_p^*\rightarrow G$ 和 $H_2:\{0,1\}^*\rightarrow G$。输出公共安全参数：

$$pp=(e,p,g,g_1,G,G_T,H_0,H_1,H_2) \tag{1}$$

2）属性授权机构初始化 $AuthoritySetup(pp)\rightarrow(PK_i,MSK_i)$

该过程由 AA 执行，输入 pp，AA_i 随机选取 $\alpha_i,y\in Z_p$，计算 $A_i=e(g,g)^{\alpha_i}$，$Y=e(g,g_1)^y$。对于由 AA_i 管理的属性名 a_i，随机选取 $\gamma_i\in Z_p$，计算 $R_i=g^{\gamma_i}$；对于属性名 a_i 下的任意属性值 a_{i,t_i}，随机选取 β_{i,t_i}，计算 $T_{i,t_i}=g^{\beta_{i,t_i}}$。属性授权机构 AA_i 的公钥为：

$$PK_i=\{A_i,Y,R_i,T_{i,t_i}\} \tag{2}$$

主密钥为：

$$MSK_i=\{\alpha_i,y,\gamma_i,\beta_{i,t_i}\} \tag{3}$$

3）用户私钥生成 $KeyGen(PP,MSK_i,U)\rightarrow SK_i$

该过程由 AA 执行，DO 向 AA 提交含有唯一身份标识 σ 的用户属性序列 $U=(u_1,\cdots,u_n)$，属性授权机构 AA_i 首先确认提交的属性值 u_i 是否是其授权管理，若不是，输出 $\perp$；若是，AA_i 随机选取 $sk\in Z_p$ 并计算 $IK_\sigma=(\sigma,H_1(sk))$，$D_i=g^{\alpha_i}h^{\gamma_i}$，$R_{i,t_i}=(T_{i,t_i}^{a_i})^{\gamma_i}$。由 AA_i 生成的私钥组件为 $SK_i=\{D_i,R_{i,t_i}\}$。AA_i 随机选择 $d_i\in Z_p$，计算 $g^{d_i}\cdot D_i$，并将其发送给其他属性授权机构，任意 AA 计算 $D=\prod_{i\in\{1,\cdots,l\}}g^{d_i}\cdot D_i=g^{\sum d_i\alpha_i}h^{\sum\gamma_i}$，则用户私钥为：

$$SK=\{IK_\sigma,D,(R_{i,t_i})_{i\in\{1,\cdots,l\}}\} \tag{4}$$

使用安全通道将 SK 分发给 DS，并将用户唯一标识 σ 发送至用户列表 L_σ。

4）加密 $Encrypt(PK_i,m,\Gamma)\rightarrow C$

该过程由 DO 执行，输入公共参数 PK_i、明文 m 和访问结构树 $\Gamma(\boldsymbol{M},\rho,Z)$，$\boldsymbol{M}$ 为 $l\times n$ 的矩阵，映射 ρ 将 $\boldsymbol{M}$ 的每一行 M_i 与每个属性名映射。对于 $\forall z_{\rho(i)}\in\Gamma$，$DO$ 计算 $h=H_0(z_{\rho(1)}\|\cdots\|z_{\rho(l)})$，随机选取 $x_i,y_i,b_i,q_1,\cdots,q_l\in Z_p$ 和随机向量 $\boldsymbol{v}=(s,\nu_2,\cdots,\nu_n)\in Z_p^n$，计算 $X_{i,b_i}=g^{-H_0(x_i\|i\|b_i)}$，$Y_{i,b_i}=g^{H_0(y_i\|i\|b_i)}$ 和 $\lambda_i=\boldsymbol{M}_i\cdot\boldsymbol{v}$，则密文组件为：

$$\begin{cases}C_0=mY\\C_1=g^s\\C_{2,i}=h^{\lambda_i}X_{i,b_i}\\C_{3,i}=Y_{i,b_i}\end{cases} \tag{5}$$

其中，$i\in(1,\cdots,l)$，密文集合表示为 $C=\{(\boldsymbol{M},\rho),C_0,C_1,(C_{2,i},C_{3,i})\}$，并保留加密信息 $En(m)=(\boldsymbol{v},q_1,\cdots,q_l)$。

说明：本文方案的密文为定长，与属性数量无关。本文方案在节省 SSP 存储空间的同时提高了算法效率；将属性名隐藏于密文中，提高了算法的安全性。

5）解密 $Decrypt(PP,C,SK)\rightarrow m$

该过程由 DS 执行，输入 PP，C 和 SK。首先，通过访问结构 $(\boldsymbol{M},\rho)$ 得出访问集合 $I_{(M,\rho)}$，然后验证是否存在 $I\in I_{(M,\rho)}$，满足方程：

$$\begin{aligned}e(C_{2,i},R_i)&=e(h^{\lambda_i}g^{-H_0(x_i\|i\|b_i)},g^{\gamma_i})\\&=e(h^{\lambda_i},g^{\gamma_i})e(g^{-H_0(x_i\|i\|b_i)},g^{\gamma_i})\\&=e(h,g)^{\lambda_i\gamma_i}e(g,g)^{-H_0(x_i\|i\|b_i)\gamma_i}\end{aligned} \tag{6}$$

$$\begin{aligned}e(C_{3,i},R_i)&=e(g^{H_0(y_i\|i\|b_i)},g^{\gamma_i})\\&=e(g,g)^{H_0(y_i\|i\|b_i)\gamma_i}\end{aligned} \tag{7}$$

其中，$\sum_{i=1}^{l}\omega_i\lambda_i=s$。若满足式（6）－式（7），则验证成功，通过式（8）获得明文：

$$m=\frac{C_0}{e(D,C_1)\cdot e(H_1(sk),C_1)} \tag{8}$$

3.2.2 支持策略对比的属性撤销方案

当发生属性撤销时，访问策略将改变，这时需要更新密钥和密文。该过程由 DO 和 SSP 交互完成，DO 根据保留的加密信息 $En(m)$ 生成动态更新密钥 DK_m，并将 DK_m 发送给 SSP，SSP 接收到 DK_m 后更新生成新的密文。具体过程如下：

Step 1 动态更新密钥 $DK_mGen(En(m),\Gamma',\Gamma)$

输入 $En(m)$、新的访问结构 $\Gamma'(\boldsymbol{M}',\rho',Z')$ 和旧的访问结构树 $\Gamma(\boldsymbol{M},\rho,Z)$，输出更新密钥 DK_m。其中，$\boldsymbol{M}'$ 为 $l'\times n'$ 的矩阵，ρ' 为新的映射，Z' 为更新后的属性值。

1）访问策略对比：运行动态更新密钥算法之前进行新旧访问策略的对比，输出行索引信息集。定义 B_1 和 B_2 为行索引信息集，$n_{\rho(i),\boldsymbol{M}}$，$n_{\rho'(j),\boldsymbol{M}'}$ 分别表示矩阵 $\boldsymbol{M}$ 和 $\boldsymbol{M}'$ 中属性名的数量，若 $n_{\rho'(j),\boldsymbol{M}'} \leqslant n_{\rho(i),\boldsymbol{M}}$，则将 $\boldsymbol{M}'$ 中属性名的索引信息存入 B_1 中；若 $n_{\rho'(j),\boldsymbol{M}'} > n_{\rho(i),\boldsymbol{M}}$，则将 $n_{\rho'(j),\boldsymbol{M}'} - n_{\rho(i),\boldsymbol{M}}$ 个属性名索引信息存入 B_2 中。

2）动态更新密钥 DK_m 的生成：随机选取一个新的随机向量 $\boldsymbol{v}' \in Z_p^n$ 和 s，令 $\lambda_j' = \boldsymbol{M}_j' \cdot \boldsymbol{v}'$，其中 $\boldsymbol{M}_j'$ 表示矩阵 $\boldsymbol{M}'$ 的第 j 行。对于 $j \in (1,\cdots,l')$，分两种情况：

①若 $j \in B_1$，更新密钥 $DK_1 = (DK = h^{\lambda_j' - \lambda_i})$，置 $q_j' = q_i$；

②若 $j \in B_2$，随机选取 $x_j \in Z_p$，更新密钥 $DK_2 = (x_j, DK = h^{\lambda_j' - x_j\lambda_i})$，则更新密钥为：

$$DK_m = \begin{cases} DK_1 = (DK = h^{\lambda_j' - \lambda_i}), & j \in B_1 \\ DK_2 = (x_j, DK = h^{\lambda_j' - x_j\lambda_i}), & j \in B_2 \end{cases} \tag{9}$$

Step 2　动态更新密文 $CipherUpdate(DK_m)$

DO 将 DK_m 发送至 SSP，其接收到密钥 DK_m 后更新相应的密文。

1）若 $j \in B_1$，计算：$\begin{cases} C_{2,j}' = C_{2,i} \cdot DK_1 \\ C_{3,j}' = C_{3,i} \end{cases}$。

2）若 $j \in B_2$，计算：$\begin{cases} C_{2,j}' = (C_{2,i})^{x_j} \cdot DK_2 \\ C_{3,j}' = C_{3,i} \end{cases}$。

则新的密文 $C' = \{(\boldsymbol{M}', \rho'), C_0, C_1, (C_{2,j}', C_{3,j}')\}$。

3.2.3　一种高效的密文检索方案

DO 将数据加密上传至 SSP 后，数据都是以密文的形式存在的，即使后台系统被攻破，也能在很大程度上避免数据隐私的肆意泄露。但是加密处理导致数据失去了原有的特性，使得 DS 难以高效地对信息进行查询，因此密文搜索也是物联网数据安全保护中必不可少的一个环节。

本文提出的搜索方案的关键技术如下：

1）引入超级节点，用于索引结构的存储，以降低 SSP 系统的压力。

2）使用混合索引结构，在 B+树中嵌套二叉树，以树型结构进行搜索，提高检索效率。

3）在索引结构中引入文件标识符（FID）和用户标识符（UID），由 FID 和 UID 检索相应的密文，提高检索效率。

密文搜索过程如下：

Step 1　密文上传

1）DO 标记自己的唯一用户身份标识 UID，并为文件选取一个唯一的文件标识 FID。

2）调用 3.2.1 节中的 $Encrypt(PK_i, m, \Gamma) \rightarrow C$ 算法对文件进行加密。

3）将加密后的文件按照图 2 的格式上传至 SSP 进行保存。

FID	UID	C	$\{DataFile\}_{AES}$

图 2　数据文件的存储格式

Fig. 2　Storage format of data files

Step 2　关键词提取上传

1）DO 提取出关键字集，将其作为可搜索的数据结构集 W，记为 $W = (w_1, \cdots, w_n)$。

2）DO 分别计算每个关键字在文件中出现的次数，记为 $t_1, t_2, \cdots, t_n$。

3）DO 为每个 w_i 计算 $TF(w_i) = \frac{t_i}{N}$，其中 N 表示文件单词总数，并将 $TF(w_i)$、关键字集 W、UID 和 FID 发送至 SP。

Step 3　索引生成

本文方案使用如图 3 所示的混合索引结构（B+树中嵌套二叉树）。在 B+树中，根节点表示子树的哈希值范围，B+树的每个叶子节点由两个部分组成：一部分为关键字的哈希值，另一部分为一个二叉树。在二叉树中，一个叶子节点包含 FID，另一个叶子节点包含 UID。

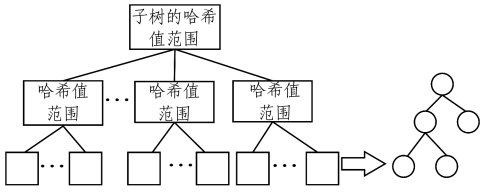

图 3　混合索引结构

Fig. 3　Hybrid indexing structure

该过程由 SP 执行，当 SP 接收到来自 DO 的 $TF(w_i)$，W，UID 和 FID 后，建立索引结构。

1）SP 计算 $IDF = \log(D/d_i + 1)$，其中 D 表示文件总数，d_i 表示关键字 w_i 在 D 份文件中出现的次数。

2）SP 计算关键字 w_i 的权重值 $\varphi = TF \times IDF$，用于对包含相同关键字的文件进行标记排序。

3）随机选取 $\alpha, s \in Z_p$，并定义消息认证函数 F，SP 为每个 w_i 生成相应的哈希函数序列 $h_i = e(g,g)^{\alpha s} \cdot e(g, H_2(w_i))^s$。

4）生成索引结构 $IX = \{UID, FID, F(\varphi, h_i)\}$。

Step 4　搜索密钥生成

该过程由 DS 和 AA 交互完成，当 DS 输入任意关键字 w' 进行搜索时，随选取机 $u, g \in Z_p$，并令 $q_u = g^u$。

1）DS 将身份标签 σ 发送给 AA，AA 确认其身份标签是否存在于用户列表 L_σ 中。

2）若 $\sigma \in L_\sigma$，AA 为 DS 生成相应的搜索密钥 $SK' = (g^{\alpha_i} \cdot q_u^{\alpha_i})$。

Step 5　单射查询函数生成

该过程由 SSP 完成，输入用户私钥 SK、用户序列 U、关键字 w' 和与之对应的搜索密钥 SK'，输出关键字 w' 的单射查询函数：

$$T_{w'^i} = \{T = H_2(w')(g^{\alpha_i} \cdot q_u^{\alpha_i})^{1/u}, D' = D^{1/u}, (R'_{i,t_i} = R_{i,t_i}^{1/u})_{\in\{1,\cdots,l\}}\}$$

Step 6　验证检索

1）输入单射查询函数 $T_{w'^i}$ 和用户序列 U，SSP 判断 U 是否满足访问结构 Γ。

2）若满足，则进行关键字匹配验证：SSP 和 SP 交互进行匹配，通过式（10）验证 $T_{w'^i}$ 中的关键字与 IX 中的关键字是否匹配：

$$F(h_i) = F(T) \tag{10}$$

3）若式（10）成立，SP 根据索引结构 IX 将相应的 FID 和

UID 返回给 SSP。

4)SSP 根据 FID 和 UID 检索到相应的密文,并将密文返回给 DS。

Step 7 密文解密

DS 接收到密文后,调用 $Decrypt(PP,C,SK)\rightarrow m$ 算法对密文进行解密,获得明文。

4 安全性和性能分析

4.1 安全性分析

本文采用灵活、细粒度的访问控制来确保数据的安全性,并从抗合谋攻击、数据安全和用户属性安全 3 个方面进行安全性分析。

4.1.1 抗合谋攻击

本文采用 N 个属性机构 $AA=(AA_1,\cdots,AA_n)$ 对 N 个属性分别进行管理,任意一个 AA_i 只能根据其管理的属性生成部分用户私钥,去除中央授权机构 CA,取消 CA 对 AA 的管理权利,从而避免了由于 CA 被攻破而造成的用户私钥泄露。若攻击者想要获得用户私钥,其必须攻破 N 个 AA,只要有一个 AA 未被攻破,攻击者都无法获得完整的私钥组件。因此,本文方案可以抵抗 $N-1$ 个属性授权机构的合谋攻击。

4.1.2 数据安全

在加密阶段,本文使用了混合加密的方式。数据的安全性主要由对称密钥 K 的安全性决定,而 K 使用本文加密方案进行加密和解密。本方案在运行解密运算前,首先通过部分访问结构 $(\boldsymbol{M},\rho)$ 获得访问集合 $I_{(\boldsymbol{M},\rho)}$,若 $\forall i\in I_{(\boldsymbol{M},\rho)}$,$u_{\rho(i)}=z_{\rho(i)}$,则表明匹配成功,此时才能进行解密。

当发生用户撤销时,被撤销用户的属性集将从属性授权机构移除,即 $U\cap S=\varnothing$,此时,当该用户再次提交属性时,AA 返回 $\perp$,即无法返回用户私钥,从而确保了数据的安全性。当发生属性撤销时,对于任一用户而言,只有该用户没有被撤销的属性仍然满足访问结构时,该用户才能对密文进行解密。

在密文搜索阶段,DS 输入关键字进行检索时,同样需要验证 DS 的属性序列是否满足访问结构,若不满足,则无法搜索到相应的密文,从而进一步确保了数据的安全性。

4.1.3 用户属性安全

用户属性的安全性也是系统安全的重要组成部分,加密方案中,密文 C 包含访问结构 Γ,访问结构又是由一系列用户的属性集 A 构成,假设 A 是以明文形式传输的,密文 C 中将包含关于属性的明文信息,若恶意用户使用各种手段获得明文,则会泄露用户属性的一些敏感信息。本文使用隐藏属性值的方法确保密文目标群体用户属性隐私信息的安全性,本方案的密文由 C_0,C_1,$C_{2,i}$ 和 $C_{3,i}$ 组成。由 $h=H_0(z_{\rho(1)}\parallel\cdots\parallel z_{\rho(l)})$ 和 $C_{2,i}=h^{\lambda_i}X_{i,b_i}$ 可以看出,属性值 $z_{\rho(i)}$ 被隐藏在 $C_{2,i}$ 中,因此即便恶意用户获得密文,也无法获得用户的明文属性信息,确保了用户属性的安全。

4.2 安全性证明

定义 1 q-PBDHE 假设

随机选取 $a,s,b_1,\cdots,b_q\in Z_p$,$T\in G_T$,攻击者给定元组 $y=(g,g^s,g^a,\cdots,g^{(a^q)},g^{(a^{q+2})},\cdots,g^{(a^{2q})};\forall 1\leqslant j\leqslant q,g^{s\cdot b_j},g^{a/b_j},g^{a^q/b_j},g^{a^{q+2}/b_j},\cdots,g^{a^{2q}/b_j};\forall 1\leqslant j,k\leqslant q,k\neq j,q^{a\cdot s\cdot b_k/b_j},\cdots,g^{a^q\cdot s\cdot b_k/b_j})$,判断 $T=e(g,g)^{a^{q+1}\cdot s}$ 是否成立。若攻击者在多项式时间内区分 $(\boldsymbol{y},e(g,g)^{a^{q+1}\cdot s})$ 和 $(\boldsymbol{y},T)$ 的优势 $Adv_A=|\Pr[A(\boldsymbol{y},e(g,g)^{a^{q+1}\cdot s})=1]-\Pr[A(\boldsymbol{y},T)=1]|\geqslant\varepsilon$ 是可忽略的,则认为在群 (e,p,G,G_T) 上的 q-PBDHE 假设成立。

设本文安全性证明基于 q-PBDHE 假设,假设在 2.4 节定义的安全模型中,不存在任意多项式时间内的攻击者选择访问结构 $\Gamma^*(\boldsymbol{M}^*,\rho^*,Z^*)$ 可攻破 3.3 节的加密方案。

证明:游戏开始前,挑战者生成 q-PBDHE 挑战 $(\boldsymbol{y},T)$:

Step 1 系统初始化阶段

1)攻击者创建并提交要挑战的访问结构 $\Gamma^*(\boldsymbol{M}^*,\rho^*,Z^*)$,其中 $\boldsymbol{M}^*$ 为 $l^*\times n^*$ 的矩阵。

2)挑战者随机选择 $m\in Z_p$,哈希函数 H_0,H_1 和 H_2,计算 $g_1=g^m$,并将公共参数 $PP=(e,p,g,g_1,G,G_T,H_0,H_1,H_2)$ 发送给攻击者。

Step 2 属性授权机构初始化

属性授权机构 AA_i 随机选取 $\alpha_i^*,y^*\in Z_p$,计算 $Y=e(g,g)^y$,令 $\alpha_i=\alpha_i^*+a^{q+1}$,计算:

$$\begin{aligned}A_i&=e(g,g)^{\alpha_i}\\&=e(g,g)^{\alpha_i^*}e(g,g)^{a^{q+1}}\\&=e(g,g)^{\alpha_i^*}e(g^a,g^{a^q})\end{aligned}\tag{11}$$

设 S 为属性授权机构的属性集,对于 AA 管理的属性名 a_i 且 $\rho^*(i)\in S$,随机选取 $\gamma_i\in Z_p$,计算:

$$R_i=g^{\gamma_i}\prod g^{a\boldsymbol{M}_{i,1}^*/b_i}\cdot g^{a^2\boldsymbol{M}_{i,2}^*/b_i}\cdot\cdots\cdot g^{a^{n^*}\boldsymbol{M}_{i,n^*}^*/b_i}\tag{12}$$

对于 AA 管理的属性名 a_i 且 $\rho^*(i)\notin S$,随机选择 $\gamma_i\in Z_p$,计算 $R_i=g^{\gamma_i}$;对于 a_i 下的属性值 a_{i,t_i},随机选取 $\beta_{i,t_i}\in Z_p$,计算 $T_{i,t_i}=g^{\beta_{i,t_i}}$,则属性授权机构 AA_i 的公钥为:

$$PK_i=\{A_i,Y,R_i,T_{i,t_i}\}\tag{13}$$

主密钥为:

$$MSK_i=\{\alpha_i,y,\gamma_i,\beta_{i,t_i}\}\tag{14}$$

并且挑战者将 PK_i 发送给攻击者。

Step 3 密钥询问阶段 1

1)攻击者向挑战者提交含有唯一身份标识 σ^* 但不满足访问结构 Γ^* 的属性集合 $U^*=(u_1^*,\cdots,u_i^*)$,询问解密密钥。

2)对于哈希函数 $H_1(x)$,每当攻击者提出密钥查询时,挑战者随机选取 $sk\in Z_p$ 并计算 $H_1(sk)$,则 $IK=(\sigma^*,H_1(sk))$。

3)对于满足访问结构 Γ^* 的属性名 u_i^*,挑战者计算 $D_i=g^{\alpha_i'}g^{a^{q+1}}h^{\gamma_i}\prod g^{a\boldsymbol{M}_{i,1}^*/b_i}\cdot g^{a^2\boldsymbol{M}_{i,2}^*/b_i}\cdot\cdots\cdot g^{a^{n^*}\boldsymbol{M}_{i,n^*}^*/b_i}$,对于 u_i^* 下的属性值 u_{i,t_i}^*,计算 $R_{i,t_i}=(T_{i,t_i}^{u_{i,t_i}^*})^{\gamma_i}$。

4)对于不满足访问结构但满足属性授权机构属性集合 S 的属性名 u_i^*,计算 $D_i=g^{\alpha_i'}g^{a^{q+1}}h^{\gamma_i}$,对于由其管理的属性值 u_{i,t_i}^*,计算 $R_{i,t_i}=(T_{i,t_i}^{u_{i,t_i}^*})^{\gamma_i}$。

5)挑战者随机选取 $d_i\in Z_p$,计算:$D=\prod\limits_{i\in\{1,\cdots,l\}}g^{d_i}\cdot D_i=g^{\sum d_i\alpha_i}h^{\sum\gamma_i}$,则解密密钥为:

$$SK=\{IK,D,(R_{i,t_i})_{i\in\{1,\cdots,l\}}\}\tag{15}$$

挑战者将解密密钥发送给攻击者,攻击者获得解密密钥组件。

Step 4 挑战阶段

攻击者提交两个相同长度的消息 M_0 和 M_1，挑战者随机选取 $b\in\{0,1\}$，加密 M_b。选取 $\boldsymbol{v}^*=(s,sa+v_2^*,sa^2+v_3^*,\cdots,sa^{n^*-1}+v_{n^*}^*)\in Z_p^{n^*}$ 和 $x_i,y_i,b_i\in Z_p$，计算 $X_{i,b_i}=g^{-H_0(x_i\|i\|b_i)}$ 和 $Y_{i,b_i}=g^{H_0(y_i\|i\|b_i)}$，挑战密文组件：

$$\begin{cases}C_0^*=m_b\cdot T\cdot Y\\ C_1^*=g^s\\ C_{2,i}^*=X_{i,b_i}\left(\prod\limits_{j=1,\cdots,n^*}(h)^{\boldsymbol{M}_{i,j}^*v_j^*}\right)\\ C_{3,i}^*=Y_{i,b_i}\end{cases}\tag{16}$$

发送密文 $C_b^*=\{(\boldsymbol{M}^*,\rho^*),C_0^*,C_1^*,(C_{2,i}^*,C_{3,i}^*)_{i\in(1,\cdots,l^*)}\}$ 给攻击者。

Step 5　密钥询问阶段 2

该阶段，已经获得 M_b 的攻击者可以重复密钥询问阶段 1 继续询问解密密钥组件，挑战者返回相应的询问结果。

Step 6　解密询问阶段

攻击者使用解密密钥解密挑战密文，尝试询问解密结果，猜测与挑战密文 C_b^* 对应的消息 M_{b^*}。挑站者输出攻击者对 b 的猜想 $b^*\in(0,1)$。

1）若 $b^*=b$，挑战者输出 $\theta=0$，表明攻击者能够区分 $(\boldsymbol{y},e(g,g)^{a^{q+1}\cdot s})$ 和 $(\boldsymbol{y},T)$，$T=e(g,g)^{a^{q+1}\cdot s}$，此时攻击者的优势为 $\Pr[b^*=b|\theta=0]=\frac{1}{2}+\varepsilon$。

2）若 $b^*\neq b$，挑战者输出 $\theta=1$，表示 T 是 G_T 中的随机元素，攻击者的优势 $\Pr[b^*=b|\theta=1]=\frac{1}{2}$。

3）由 1）和 2）得到攻击者攻击 q-PBDHE 的优势：$Adv=|\frac{1}{2}\Pr[b^*=b|\theta=0]+\frac{1}{2}\Pr[b^*=b|\theta=1]-\frac{1}{2}|=\frac{1}{2}\varepsilon$。

综上所述，本文提出的方案是 q-PBDHE 安全的。

4.3　性能分析

4.3.1　理论分析

本节将从几个方面对本文提出的方案进行定量分析，主要研究内容为密文长度和私钥长度等。为简化表示，定义 $|G|$ 和 $|G_T|$ 分别表示 G 和 G_T 中的元素长度，$|Z_P|$ 表示 Z_p 的长度，I 表示属性数量，L_i 表示所有属性的属性值。

由表 1 可以看出，一方面，本文方案中的密文为定长，密文长度与属性数量无关，节约了 SSP 的存储空间，提高了效率。另一方面，虽然本方案的公钥长度、私钥长度等都与属性数量相关，但与文献[15-16]相比，仍然具有一定的优势。

表 1　方案比较

Table 1　Comparison of schemes

长度	文献[15]	本文
公钥	$(3I+1)\|G\|+\|G_T\|$	$(I+2)\|G\|+\|G_T\|$
主密钥	$(2I+3)\|Z_p\|$	$(I+2)\|Z_p\|$
私钥	$(I+2)\|G\|$	$I\|G\|$
密文	$(I+2)\|G\|+I\|G_T\|$	$2\|G\|+\|G_T\|$

4.3.2　实验仿真

本方案的仿真平台为：Inter(R)Core (TM)i7-4770 CPU，内存为 16GB；操作系统为 Windows7 64 位；采用 PBC 库；椭圆曲线使用 *Type A*：$y^2=x^3+x$。

为确保实验数据的可靠性，本文实验的基本参数设置如表 2 所列。

表 2　实验参数

Table 2　Experiment parameters

参数名称	参数值
用户属性数量	[0,50]
系统属性数量	[0,50]
用户的个数	(0,400]
AES 密钥/bit	128
加密文本大小/k	568
关键字个数	[0,20]

本节将从加密时间、解密时间、索引建立时间和重加密时间方面对本文方案与文献[15]和文献[16]提出的方案进行仿真比较，具体结果如下。

图 4 和图 5 比较了加、解密时间和属性数量的关系。当文件大小为 568k，属性数量在 50 以内时，加密时间和解密时间与属性值的数量呈正相关。在相同属性数下，本方案的加、解密时间少于文献[15-16]的加解密时间，并且随着属性数量的增多，加、解密的时间优势更加明显。这是由于本文方案的密文组件比文献[15-16]的少，并且本文方案的密文长度为定值，这使其在解密过程中只需运行对运算。

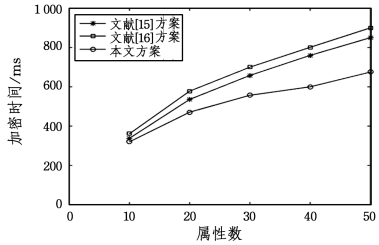

图 4　加密时间

Fig. 4　Encryption time

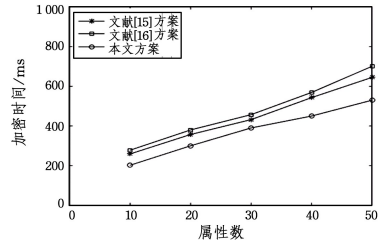

图 5　解密时间

Fig. 5　Decryption time

图 6 描述了当关键字数量在 20 以内时，索引生成时间与关键字数量的关系。由于本文使用了混合索引的方式建立索引，并且索引的建立由引入的 SP 执行，因此索引建立的时间较文献[15-16]的方案有较大的优势。

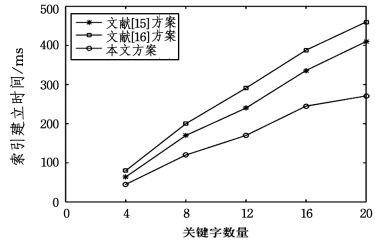

图 6　索引生成时间

Fig. 6　Index generation time

图 7 描述了重加密时间与属性数量的关系。可以看出，当属性数量在 50 以内时，相对于文献[15-16]的方案，本文方案在重加密时间上有明显优势，这是因为本文方案在重加密过程中使用了访问策略对比的方法，提高了算法的细粒度，并且加快了系统的运行速度。另外，由于在重加密过程中只运行了少量的幂运算，因此重加密的运算效率比加、解密的运算效率高。

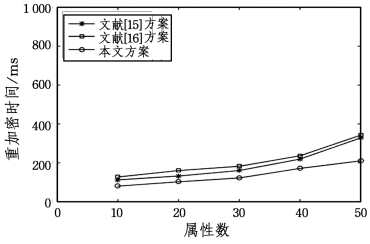

图 7 重加密时间

Fig. 7 Re-encryption time

结束语 本文方案使用去除属性授权中心、密文定长和隐藏访问模式等方法，实现了物联网搜索技术中数据的安全保护。本文提出了支持策略对比的属性撤销方案，明显减少了重加密时间。同时，密文更新过程由搜索中心后台完成，减轻了加密端的系统压力。文中提出一种高效的密文检索方案，提高了密文检索效率。

当文件数量不断增大，不同文件具有相同关键字时，密文检索的准确性将受到一定的影响，因此所算方案存在一定的准确性问题，下一步的研究重点将围绕该问题展开。

参 考 文 献

[1] WANG J H, LIU C Y, FANG B X. A Survey of Research on Data Privacy Protection for Internet of Things Search[J]. Journal of Communications, 2016, 37(9): 142-153. (in Chinese)
王佳慧，刘川意，方滨兴. 面向物联网搜索的数据隐私保护研究综述[J]. 通信学报，2016，37(9)：142-153.

[2] GORLATYKH A, ZAPECHNIKOV S. Building access tree for attribute-based encryption schemes over multidimensional data objects[C]// IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering. IEEE, 2018: 1496-1499.

[3] CANARD S, PHAN D H, TRINH V C. Attribute-based broadcast encryption scheme for lightweight devices[J]. IET Information Security, 2018, 12(1): 52-59.

[4] LUAN I, PETKOVIC M, NIKOVA S, et al. Mediated Ciphertext-Policy Attribute-Based Encryption and Its Application [C]// Information Security Applications, International Workshop. Wisa 2009, 2009.

[5] YANG F, YUAN Q, DU S, et al. Reserving relief supplies for earthquake: a multi-attribute decision making of China Red Cross[J]. Annals of Operations Research, 2016, 247(2): 759-785.

[6] EWENIKE S, BENKHELIFA E, CHIBELUSHI C. Cloud Based Collaborative Software Development: A Review, Gap Analysis and Future Directions[C] // IEEE/ACS, International Conference on Computer Systems and Applications. IEEE, 2018: 901-909.

[7] WANG S, ZHOU J, LIU J K, et al. An Efficient File Hierarchy Attribute-Based Encryption Scheme in Cloud Computing[J]. IEEE Transactions on Information Forensics & Security, 2017, 11(6): 1265-1277.

[8] XU X, ZHANG Q, ZHOU J. NC-MACPABE: Non-centered multi-authority proxy re-encryption based on CP-ABE for cloud storage systems[J]. Journal of Central South University, 2017, 24(4): 807-818.

[9] GAO W, WANG G, CHEN K, et al. Efficient identity-based threshold decryption scheme from bilinear pairings[J]. Frontiers of Computer Science, 2018, 12(2): 1-13.

[10] GUO F, MU Y, SUSILO W, et al. Optimized Identity-Based Encryption from Bilinear Pairing for Lightweight Devices[J]. IEEE Transactions on Dependable & Secure Computing, 2017, 14(2): 211-220.

[11] MALLUHI Q M, TRINH V C. A Ciphertext-Policy Attribute-based Encryption Scheme with Optimized Ciphertext Size And Fast Decryption[C]// ACM on Asia Conference on Computer and Communications Security. ACM, 2017: 230-240.

[12] ZIRTOL K A, NOROOZI M, ESLAMI Z. Multi-user searchable encryption scheme with general access structure[C]// International Conference on Knowledge-Based Engineering and Innovation. IEEE, 2016: 399-404.

[13] MEI Z, ZHU H, CUI Z, et al. Executing multi-dimensional range query efficiently and flexibly over outsourced ciphertexts in the cloud[J]. Information Sciences, 2018, 432(1): 79-96.

[14] LIN S, ZHANG R, MA H, et al. Revisiting Attribute-Based Encryption With Verifiable Outsourced Decryption [J]. IEEE Transactions on Information Forensics & Security, 2017, 10(10): 2119-2130.

[15] WANG N, FU J, BHARGAVA B K, et al. Efficient Retrieval over Documents Encrypted by Attributes in Cloud Computing [J]. IEEE Transactions on Information Forensics and Security, 2018, 13(10): 2653-2667.

[16] FAN K, WANG X, SUTO K, et al. Secure and Efficient Privacy-Preserving Ciphertext Retrieval in Connected Vehicular Cloud Computing[J]. IEEE Network, 2018, 32(3): 52-57.

[17] MA H, ZHANG R, WAN Z, et al. Verifiable and Exculpable Outsourced Attribute-Based Encryption for Access Control in Cloud Computing[J]. IEEE Transactions on Dependable & Secure Computing, 2017, 14(6): 679-692.

[18] YAN X X, LIU Y, LI Z C, et al. Multi-attribute attribute-based encryption scheme supporting dynamic update of policies[J]. Journal of Communications, 2017, 38(10): 94-101. (in Chinese)
闫玺玺，刘媛，李子臣，等. 支持策略动态更新的多机构属性基加密方案[J]. 通信学报，2017，38(10)：94-101.

[19] CHI P W, LEI C L. Audit-Free Cloud Storage via Deniable Attribute-based Encryption[J]. IEEE Transactions on Cloud Computing, 2018, 6(2): 414-427.

[20] HAN J, YANG Y, LIU J K, et al. Expressive attribute-based keyword search with constant-size ciphertext[J]. Soft Computing, 2018, 22(15): 5163-5177.