XXXX

面向可信数据空间的基于属性加密的数据安全托管方案

摘　要：可信数据空间是促进数据流通利用，充分释放数据要素价值的重要基础设施。可信数据空间的构建面临可信数据空间运营方、数据托管方、数据提供方、数据使用方等多个参与方之间的协同问题，数据以外包方式流通过程中的隐私保护问题，以及数据“三权”（数据持有权、数据使用加工权、数据产品经营权）分置在可信数据空间中的实现问题。针对上述问题，本文提出了一种面向可信数据空间的基于属性加密的数据安全托管方案。方案设计了可信数据运营方对数据托管方的授权机制，并将数据托管方授权证书嵌入基于属性加密的访问控制过程中，在访问控制过程中实现对数据托管方的认证；在基于属性访问控制的基础上增加基于权限的访问控制结构，实现可信数据空间中的“三权”分置。安全性分析及实验证明，本方案实现开销较小且能够提供不低于CP-ABE的安全性，具备在可信数据空间中良好的应用价值。

关键词：数据流通；访问控制；可信数据空间；CP-ABE

中图分类号：TP393

文献标志码：A

doi: 10.11959/j.issn.1000−0801.

A Data Security Hosting Scheme Based on Attribute-Based Encryption for Trusted Data Space

Abstract: The trusted data space is an important infrastructure for promoting data circulation and utilization, and fully releasing the value of data elements. The construction of trusted data spaces faces issues such as coordination among multiple participants including trusted data space operators, data custodians, data providers and data users; privacy protection in the process of data circulation through outsourcing; and the implementation of “the separation of three rights” (right to hold data resources, right to use and process data, right to operate data products) of data in trusted data spaces. To address the above issues, this paper proposes an attribute-based encryption (ABE)-based data security custody scheme for trusted data spaces. The scheme designs an authorization mechanism for trusted data operators to data custodians, and embeds the data custodian's authorization certificate into the access control process based on ABE, thereby realizing the authentication of data custodians during access control. In addition, on the basis of attribute-based access control, a permission-based access control structure is added to realize the separation of three rights in the trusted data space. Security analysis and experiments show that the proposed scheme has low implementation overhead, can provide security not lower than that of CP-ABE , and has good application value in trusted data spaces.

Key words: Data circulation, Access control, Trusted data space, CP-ABE

收稿日期：XXXX−XX−XX；修回日期：XXXX−XX−XX

1 引言

作为数据要素流通的关键载体，可信数据空间构建了一种多方协同的新型数据治理模式，其核心在于通过技术赋能与机制创新，实现“数据可用不可见、可控可计量”。从参与主体来看，可信数据空间涵盖运营方、数据提供方、数据使用方及数据服务方四大核心角色，其中数据服务方进一步细分为数据运营、托管与开发等专业化主体。这种多主体参与模式对构建可信协作框架、全生命周期隐私保护体系提出了新的要求。可信数据空间需深度融合“三权”（数据资源持有权、数据使用加工权、数据产品经营权）分置的制度设计，通过精细化的授权机制实现数据权属清晰化、流转规范化，这为技术方案的落地提出了严峻挑战。在技术层面，基于属性的加密（ABE）机制[4-5]凭借其细粒度的访问控制能力，成为解决云存储环境下数据安全共享的重要技术路径。

基于属性加密的访问控制技术作为数据安全领域的关键分支，一直以来都吸引着广大研究者的目光。Goyal提出密钥策略的基于属性加密方案（key-policy attribute-based encryption，KP-ABE）[9]，通过用户构建的访问树与文件属性的匹配来决定用户是否能够解密并访问文件。为了提升文件拥有者对访问权限的控制，Benthcourt等人提出密文策略的基于属性加密方案（ciphertext-policy attribute-based encryption，CP-ABE）[10]。为了提升CP-ABE的安全性，文献[11-13]研究了多授权中心的CP-ABE方案，通过增加授权中心分散计算任务，来抵御授权不可信问题，并同时提高计算效率，多授权中心方案的实现需要更多的硬件投入。文献[14-15]等研究CP-ABE机制在物联网领域的数据安全共享方案。

2 相关理论知识

2.1 双线性对

定义1：设q是一个大素数，G_1是阶为q的循环群，生成元为g。双线性映射$e:G_1\times G_1\to G_2$是一个双线性对[16]，$\forall m,n\in G_1$，选择任意的$\alpha,\beta\in Z_p$（Z_p为素数p阶循环群），双线性对具有以下性质：

1） 双线性性：$e(m^\alpha,n^\beta)=e(m,n)^{\alpha\beta}$。

2） 对称性：$\forall m,n\in G_1,e(m,n)=e(n,m)$。

3） 非退化性：$e(g,g)\neq 1$。

根据上述性质可知，对生成元g有：

$$e(g^\alpha,g^\beta)=e(g,g)^{\alpha\beta}=e(g^\beta,g^\alpha) \tag{1}$$

2.2 可信数据空间

可信数据空间作为支撑数据要素安全流通的新型基础设施，其核心内涵在于通过构建技术、制度与治理相结合的信任体系，实现跨主体、跨领域数据的可控共享与合规使用。

在可信数据空间的数据托管服务中，各个角色的分工为：

1） 可信数据空间运营方：保障可信数据空间安全合规运营，对要提供服务的数据托管方进行认证和授权，对参与数据提供、数据使用的用户进行管理。

2） 数据托管方：在可信数据空间中提供数据托管基础设施的主体，在获得可信数据空间运营方授权后，提供数据托管服务。

3）数据提供方：在可信数据空间中提供数据资源的主体；

4） 数据使用方：在可信数据空间中使用数据资源的主体，可根据权限访问托管存储的数据。

2.3 Square-CDH问题

定义2：Square-CDH问题：设λ为安全参数，G为素数p阶群，g为G的生成元，$e:G\times G\to G_T$

为一个双线性映射，给定$g, g^{\alpha} \in G$，其中$\alpha \in Z_p$，计算g^{α^2}，概率多项式时间算法$\mathcal{A}$计算出G上的Square-CDH问题的概率定义为：$\Pr\left[\mathcal{A}\left(g, g^{\alpha}\right) = g^{\alpha^2}\right]$。

定义3：在计算性DH问题难解的前提下，不存在多项式时间算法□能以不可忽略的概率求解Square-CDH问题，即：$\Pr\left[\mathcal{A}\left(g, g^{\alpha}\right) = g^{\alpha^2}\right] \leq \mathrm{neg}(\lambda)$。其中，$\mathrm{neg}(\lambda)$为可忽略函数。

3　DSC-ABE模型

3.1　方案模型描述

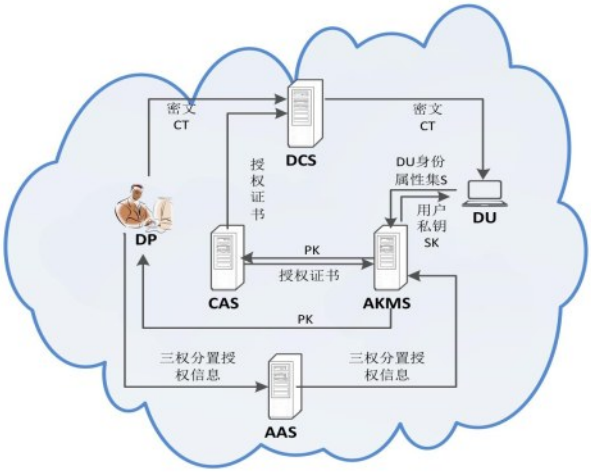

图1　DSC-ABE系统模型

下面对方案的整体流程进行概述：

1) DCS授权：DCS加入可信数据空间，向CSA申请服务注册。CAS对DCS进行背景、能力审核后，对合格的申请者发送授权证书。

2) 文件托管：DP将文件及其指定的访问树（由身份属性子树和权限子树共同构成）加密后存储到DCS，并将具有“三权”授权信息同步给AAS。授权信息具体包括DU对应的权限及授权有效期。DCS收到托管的文件后，在文件密文状态下，再使用自己的授权证书进行进一步加密。

3) 文件访问：DU申请访问文件时，首先向AKMS申请私钥。AKMS根据用户身份属性集信息及其所拥有的权限信息，共同形成完整的DU属性集合。AKMS使用PK与待访问DCS的授权证书为DU计算私钥SK。DU将文件下载到本地，使用SK对文件进行解密，只有DU的属性集匹配文件的访问策略，且DCS是合规的服务方时，DU能够解密出明文信息。

4) 密钥更新：当DCS授权证书，或对DU的授权内容、授权时间等信息进行更新时，通过构建重加密密钥RK，实现在不解密的情况下直接基于密文实现证书和密钥的更新。

3.2　符号使用及定义

本文中涉及的符号和定义如表1所示。

表1　符号定义及说明

符号	说明
PK	Public Key 公钥
MK	Master Key 主密钥
SK	用户私钥
RK	重加密密钥
T, LT	树形访问结构, 访问树T的叶子节点集合
AMS	Attribute Manage Sever属性管理服务器
DP	Data Provider 数据提供者
DU	Data User 数据使用者
DCS	Data Custody Sever 数据托管服务器
CAS	Custody Authorization Server托管授权服务器
AAS	Access Authorization Server访问授权服务器
AKMS	Access Key Manage Sever访问密钥管理服务器
M, CT	要加密的明文, 明文加密后形成的密文
cert	授权证书，拥有此证书的CAS为合规数据托管服务方
cert'	授权子证书，用于方案安全性验证

3.3　安全性定义

方案模型中，假设数据托管方是不完全可信的。即数据托管方不会主动泄露用户信息，且能够安全保存自己所拥有的授权证书，但是安全机制不完善造成用户隐私信息泄露、访问权限泄露。同时，由于可信数据空间中数据流通可能带来经济收益，数据托管方还会面临仿冒问题，即未得到授权的托管方通过伪造授权证书、窃取文

件等方式仿冒合法托管方提供数据托管服务；假设可信数据空间运营方是可信的，具有计算密钥、保护密钥的能力，可为数据托管方和DU提供所需的授权及密钥；假设数据使用方不完全可信，DU能够按照承诺对自己获得授权的数据提供应有的保护，但若访问机制不完善造成DU越权访问到其他数据，可能会造成数据泄露；假设DP与DU不会进行进行合谋。

下面定义安全游戏Ω：定义敌手AD，对于选定变量，它通过以下阶段来伪造一个私钥尝试去解密存储在DSC中的密文，以获取授权证书*cert*，实现攻击。在此过程中，如果敌手AD伪造成功的概率满足：

$$\Pr\left[\begin{array}{l}\left(SK',q',Fcert\right)\leftarrow \square \mathrm{D}\left(PK,MK,\mathrm{S},1^{\lambda}\right)\\ \left(\mathrm{CT},cert\right)\leftarrow \mathrm{Dec}\left(SK',q',Fcert,\mathrm{CT}'\right)\end{array}\right]\leq$$

$$\mathrm{neg}neg\left(\lambda\right) \qquad (2)$$

其中，$\mathrm{neg}(\lambda)$为可忽略的函数，则证明敌手AD无法获取*cert*信息并解密密文，方案是安全的。

4 方案详细描述

4.1 方案初始化

首先，AKMS执行Setup（λ）→*PK*，*MK*算法设置初始参数，输入安全参数λ，生成安全参数：

$$PK=G_0,g,h=g^{\beta},e\left(g,g\right)^{\alpha},MK=\left(\beta,g^{\alpha}\right) \qquad (3)$$

其中，G_0是素数p阶双线性群，g为生成元，方案中选择随机数$\alpha,\beta\in Z_p$。

接下来，进行数据托管方授权。首先，CAS向AKMS申请获取PK；CAS执行算法CertGen（*PK*）→*cert*，*cert*’并选择随机数$u\in Z_p$，其中：

$$cert=g^{u},cert=g^{u^2} \qquad (4)$$

以生成证书。DCS获得*cert*后开展托管服务。

4.2 访问树构建

DP为自己待托管的文件构建树形访问控制策略T，本方案中，访问树分为身份属性子树和权限属性子树。其中，身份属性子树记为T_{id}，权限子树记为T_{au}，两棵子树以AND为根节点进行连接。

T_{id}以身份信息相关的属性为叶子节点，以门限值作为非叶子节点构建访问策略。T_{au}子树涉及到的属性包括权限和时间两类，用*auc*， *aus*和*auj*分别表示持有权、使用加工权和产品经营权，time_{auc}，time_{aus}，time_{auj}。其中，每类权限与其对应的授权时间之间用AND连接，不同权限之间的子树使用其他所需门限值连接。

以医疗数据为例，构建完整访问树如图2所示。

4.3 文件加密托管

DP将需要共享的文件加密后托管到DCS，需要对文件及访问策略进行加密，具体流程如下：

1) DP向AKMS发出文件加密申请；

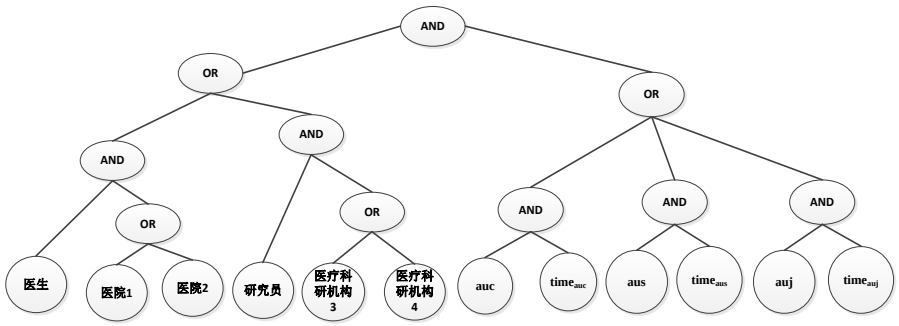

图2 访问树示例

2) DP 获得加密所用 PK，指定本次要托管的文件 M，并为其构建访问树 T，执行加密算法 Encrypt（PK，M，T）→CT：

$$\mathrm{CT}=\left(\mathrm{T},\tilde{C}=Me(g,g)^{as},C=h^{s},\forall y\in Y:C_y=g^{q_y(0)},C_y'=H(att(y))^{q_y(0)}\right). \quad (5)$$

q_x 表示节点 x 的多项式，对根节点 R，选择随机数 $s\in Z_p$，$q_R(0)=s$。$H(i)$ 表示属性 i 转化成二进制值的函数。DP 将 CT 外包到 DCS 进行存储，CT 为密文，DCS 无法看到明文内容。

3) DCS 在接收到 DU 对 CT 的访问请求后执行以下算法 CertEn（CT，cert）→CT'：

$$\mathrm{CT}'=\left(\mathrm{T},\tilde{C}=Me(g,g)^{as},C'=e(cert,C)=e(g^{2u},h^{s})=e(g,g)^{\beta su},C_y=g^{q_y(0)},C_y'=H(att(y))^{q_y(0)}\right)$$

(6) 完成计算后，将 CT' 反馈给 DU

DP 为文件 M 指定访问权限如表 2 所示。

表 2　文件授权表

DU	持有权		加工使用权		产品经营权	
	授权情况	有效时间	授权情况	有效时间	授权情况	有效时间
DU_1	*auc*	$time_{auc}$	******	******	******	******
DU_2	******	******	*aus*	$time_{aus}$	******	******
DU_3	******	******	*aus*	$time_{aus}$	*auj*	$time_{auj}$
.....						

如表所示，以矩阵形式进行访问授权，具有 *auc*，*aus*，*auj* 的表示进行授权，*表示无此项权限。对于具有授权的项，同时指定授权有效时间。根据文件授权表可构建用户的授权属性集合 S，即 DU_1 的 S 为 {*auc*，$time_{auc}$，******，******，******，******}。

4.4　密钥分配

用户密 DU 要访问存储在 DCS 的文件，根据自己的身份信息构建属性集合 S_{id} 向 AKMS 申请密钥，具体流程如下。

1) 用户 DU 根据身份信息构建身份属性集合 Sid，并向 AKMS 申请访问 DCS 的密钥；

2) AKMS 向 AAS 查询此 DU 的的授权集合 Sau。AAS 查询表 2，形成集合 Sau 返回给 AKMS；

3) AKMS 将 Sid 与 Sau 取合集，形成完整属性集 S；

4) AKMS 向 CAS 查询 DCS 的授权证书 cert；

5) AKMS 根据 S 及 cert'，执行以下算法 KeyGen（MK，S，cert'）→SK：

$$SK=\left(e(g^{(\alpha+r)/\beta},cert')=e(g^{(\alpha+r)/\beta},g^{u^2})=e(g,g)^{(\alpha+r)/u^2},\forall j\in I:D_j=g^{r}\cdot H(j)^{r_j},D_j'=g^{r_j}\right) \quad (7)$$

其中，随机数 $r\in Z_p$，对集合 S 中的每个属性 $j\in S$，选择随机数 $r_j\in Z_p$。

4.5　用户解密

DU 获取到密钥后与密文后，执行以下算法进行解密 Decrypt（CT'，*SK*）→M：

x 是访问树中的节点，i 是节点对应的属性，叶子节点的解密非递归算法为 DecryptNodeL（CT'，*SK*，x），非叶子节点解密的算法为递归算法 DecryptNodeNL（CT'，*SK*，x）。

如果 $i\in S$，那么执行解密操作：

$$\mathrm{DecryptNodeL}(\mathrm{CT}',SK,x)=\frac{e(D_i,C_x)}{e(D_i',C_x')}=e(g,g)^{rq_x(0)} \quad (8)$$

若 $i\notin S$，则 DecryptNodeNL（CT'，*SK*，x）=⊥。

当 x 为非叶子节点时，调用递归算法 DecryptNodeNL（CT'，*SK*，x）：

对 x 的所有孩子节点 z，调用 DecryptNodeL（CT'，*SK*，z），并将其输出结果存储为 F_z。让 Ix 是子节点 z 的 kx（非叶子节点的门限值）大小的任意集合，使得 $F_z\neq\perp$。如果找不到此集合，那么节点就会不满足函数，返回⊥。

否则，进行如下计算并返回结果：

$$F_x=\prod_{z\in I_x}F_z^{\Delta_{i,I_x'(0)}}=\prod_{z\in I_x}\left(e(g,g)^{r\cdot q_{parent(z)}(index(z))}\right)^{\Delta_{i,I_x'(0)}}=$$

$$e(g,g)^{rq_x(0)}\cdot\prod_{z\in I_x}\left(e(g,g)^{r\cdot q_{parent(z)}(index(z))}\right)^{\Delta_{i,I_x'(0)}} \quad (9)$$

其中，Index（z）是节点z作为x子节点的唯一索引值。

解密时，对T的根节点R调用函数，如果用户的属性集合S能满足访问树，最终可获得以下结果：

$$\mathrm{A=DecryptNodeNL(CT',}SK,r)=e(g,g)^{rq_R(0)}=e(g,g)^{rs} \quad (10)$$

再经过以下计算进行解密：

$$\tilde{c}/\left((\mathrm{C'}*\mathrm{D})/\mathrm{A}\right)=\tilde{c}\Big/\left(e(g,g)^{\beta su}\cdot e(g,g)^{(\alpha+r)u/\beta}\Big/e(g,g)^{rs}\right)=\mathrm{M} \quad (11)$$

方案执行完毕。

4.6 授权证书更新

当DCS服务达到期限时，需要重新对DCS服务资质进行验证，对可继续进行托管服务的DCS更新授权证书，更新流程如下：

CAS查询DCS的$cert$的随机数u，执行算法CertGenUp：

CertGenUp（u）→$cert_{\mathrm{new}}$，$cert'_{\mathrm{new}}$，RK

选择新的随机数$v\in Z_p$，$RK=v/u$，则：

$$cert_{new}=cert^{RK}=(g^u)^{v/u}=g^v \quad (12)$$

$$cert'_{new}=(cert')^{1/RK}=(g^{1/u})^{u/v}=g^{1/v} \quad (13)$$

RK和$cert_{\mathrm{new}}$交付DCS后进行以下处理：

对于用户新托管的密文，使用新的授权证书$cert_{\mathrm{new}}$执行算法CertEn；

对于正在托管，已经执行了CertEn算法的文件，执行以下算法进行更新：

CT'$_{\mathrm{Up}}$（CT'，RK）→CT'$_{\mathrm{new}}$，有：

$$\mathrm{CT'_{new}}=\left(\mathrm{T},\tilde{C}=Me(g,g)^{as},(C')^{RK}=e(g,g)^{\beta sv},C_y=\right.$$

$$\left.g^{q_y(0)},C'_y=H(att(y))^{q_y(0)}\right) \quad (14)$$

RK和$cert'_{\mathrm{new}}$交付AKMS后进行以下处理：

对于新的密钥申请，使用$cert'_{\mathrm{new}}$执行密钥分配算法KeyGen；

对于已有密钥，执行如下算法进行更新：

KeyGenUp（SK，$cert'_{\mathrm{new}}$）→SK_{new}，有：

$$SK_{\mathrm{new}}=\left(D^{cert'}=(e(g,g)^{(\alpha+r)/\beta u})^{u/v}=\right.$$

$$\left.e(g,g)^{(\alpha+r)/\beta v},\forall j\in I{:}D_{\mathrm{j}}=g^r\cdot H(j)^{r_j},D'_{\mathrm{j}}=g^{r_j}\right) \quad (15)$$

通过以上算法，可在在不解开原有数据的情况下实现授权证书的高效更新。

5 安全性证明与实验测试

5.1 安全性证明

首先对DSC-ABE进行整体安全性分析，本方案中，引入授权证书机制来实现对可信数据空间的运营方对DCS的授权。基于授权证书实现DCS防仿冒，在本方案设计的整体流程中，用户明文不会泄露，授权证书不会泄露，且用户无法突破“三权”分置授权的限制对文件进行解密。下面分别对方案中三处细节进行详细安全性分析：

防DCS仿冒：授权证书一方面由DCS嵌入到外包存储的密文中，即方案在密文状态下直接开展计算，不泄露明文；另一方面由AKMS嵌入到用户的解密密钥中。在解密过程中，只有得到授权的合规DCS提供的密文，才能够与用户密钥相匹配，解密得到明文。否则，若仿冒DCS提供未嵌入授权证书的密文，用户得到嵌入了证书的解密密钥，则用户不能成功解密。若用户的密钥为伪造，未经过AKMS嵌入授权证书，则也无法

成功解密由合规DCS提供的密文。

DCS授权证书安全性：在本方案中，DCS授权证书由DCS，CAS和AKMS直接拥有授权证书，CSA和AKMS均为可信服务器，不会泄露授权证书，DCS为了自己的经济利益会有效保护自己的授权证书。*SK*密钥中嵌入了证书信息，基于算法安全性，DU无法从*SK*中获得嵌入的授权证书。

用户越权访问安全性：DP对用户数据持有权限、使用加工权和产品经营权的授权直接由可信服务器AAS进行管理，作为属性信息加入到用户身份属性集合中。基于算法安全性，用户无法破解并修改嵌入密钥中的属性信息，因此，用户无法实现越权访问，有效实现“三权分置”授权机制。

下面对方案算法进行安全性证明：

定理1：如果存在一个敌手A可以在不可忽略的概率多项式时间内赢得3.3所示安全游戏，那么就存在一个算法B能够以不可忽略的概率获取证书*cert*，实现攻击。

接下来，证明如何构建一个算法B能够以不可忽略的概率获取*cert*。

B执行Setup算法，得到*PK*，*MK*，并将其发送至敌手A；

A伪造一个对存储在可信空间的CT的访问请求*query*；B收到*query*后，执行CertEn算法，完成计算后，将CT’反馈给A；

A根据身份信息构建身份属性集合$S_{id`}$，并向AKMS申请访问DCS的密钥，伪造一个证书*Fcert*，A执行KeyGen算法，利用A自身随机数u'，得到攻击私钥SK；

游戏的最后，A利用攻击私钥SK对CT`进行解密，得到CT以及证书cert，实现攻击。若A赢得游戏，则A执行DecryptNodeNL算法，由此可以计算出$g^{u'^2}$，则B算法能够在概率多项式时间内解决Square-CDH问题。根据定义3，由于Square-CDH问题是难解的，因此定理1不成立，即不存在概率多项式时间内的敌手A能够以不可忽略的概率赢得3.2中所示的安全游戏，进而A无法对明文进行解密，且无法判断*cert*，因此挑战失败，DCS-ABE方案被证明安全。

5.2　方案开销评估

本方案在CP-ABE算法基础上实现对数据托管方的授权及对用户的授权，下面对主要算法及操作进行计算复杂度分析。

1) 初始参数设置：此步骤执行Setup算法，生成PK和MK，需要选择两个随机数，进行三次幂计算，幂的数值为随机常数，计算复杂度为O(l)，与CP-ABE方案相同。

2) 数据托管方授权：授权过程中执行算法CertGen，需要选择一个随机数，执行两次幂的计算，幂的值为随机常数，计算量恒定，计算复杂度为O(l)。当需要授权的数据托管方有n个时，计算时间随着托管方数量的增加线性提高，计算复杂度为O(n)。

3) 文件加密托管：在此过程中需要执行Encrypt算法与CertGen算法。

Encrypt算法需要对明文和访问树进行加密，加密的时间复杂度与明文长度以及访问树复杂度相关。明文以对称加密的方式进行计算，只需要一个乘的操作，明文书数据量越大，所需的计算时间越长。访问树的构建与叶子节点数量有关，若文件访问树相关的属性集合数量为*m*，即访问树中叶子节点数量为*m*，则访问树的计算复杂度为*O*(*m*)。与CP-ABE方案相比，DCS-ABE方案中增加访问授权及授权时间作为叶子节点构建权限访问子树，增加的叶子节点数量为6个，是固定值，此部分计算时间复杂度为*O*(l)。因此，增加的权限子树不会提升的计算复杂度；CertEn算法使用*cert*对原有密文进行乘操作，计算开销与

原始密文长度相关。

4) 密钥分配：KeyGen 算法为用户计算密钥，计算的时间开销与用户所拥有的属性数量正相关。用户属性集 S 中属性的数量表示为|S|，那么 KeyGen 计算的时间复杂度为 O(|□|)。与 CP-ABE 方案相比，DSC-ABE 方案中，用户属性信息增加权限相关属性信息，此部分属性不多于 6 个，计算时间复杂度为 O(1)。同时，DSC-ABE 方案中增加了 cert’ 的嵌入步骤，为一次幂的操作，计算开销固定，不因其他因素变化而增加，此部分操作的计算时间复杂度也为 O(1)。因此，与原 CP-ABE 方案比，本方案中增加的计算量为 O(1) 级别，未提升计算的时间复杂度。

5) 用户解密：用户执行 Decrypt 解密明文，计算开销与访问树复杂度相关，计算时间复杂度与访问树的加密复杂度一致，为 O(m)。

6) 授权证书更新：授权证书更新需要执行 CertGenUp，CT’ $_{Up}$，KeyGenUp 三个算法。算法 CertGenUp 主要涉及随机数的选择和乘除法的计算，计算量固定，计算复杂度为 O(1)。

CT’ $_{Up}$ 采用加密的方式在原始密文基础上直接进行计算，执行一次幂的操作，计算开销与原始密文长度相关，此算法操作计算量与算法 CT’ $_{En}$ 复杂度相同。算法 $KeyGen_{Up}$ 需要进行一次幂的操作，幂的值为确定值，与 *SK* 中属性值的数量等因素无关，计算复杂度为 *O*(1)。

5.3 仿真实验

为了证明上述算法开销评估的真实性，本文对于 DSC-ABE 方案性能指标，根据方案所描述的具体算法流程，并与现存的访问控制方案 [18] [19] [20] 进行仿真实验对比，进一步证明本方案在性能方面的优势。本文 DSC-ABE 方案的仿真实验在 Win10 操作系统下通过 Java 语言实现，对可信数据空间进行了原型仿真，采用中移动某省公司能开平台日志数据作为测试文本数据，数据属性自定义，模拟多个小型代理服务器作为数据托管方进行仿真实验。

首先考虑数据托管方个数对方案性能的影响，实验结果如图 3 所示：

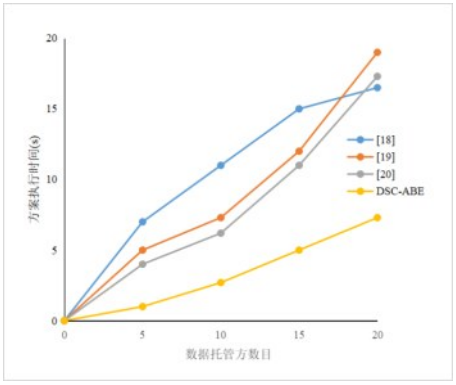

图 3　不同数据托管方下 ABE 方案性能对比

接下来考虑明文长度对方案性能的影响，实验结果如图 4 所示：

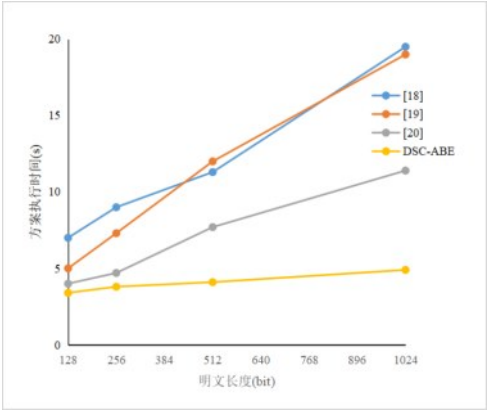

图 4　不同明文长度下 ABE 方案性能对比

最后考虑属性个数对方案性能的影响，实验结果如图 5 所示：

通过上述实验可知，在 DSC-ABE 中，方案执行时间与明文长度和数据托管方均为线性关系，而由于访问树的构建，方案执行时间与属性个数呈亚线性关系，符合上述算法开销评估，且相比现有方案，DSC-ABE 更为轻量，执行时间更短，具备较高的应用价值。

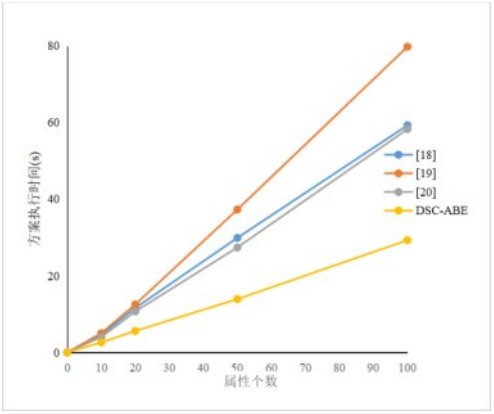

图5　不同属性个数下ABE方案性能对比

6　总结

基于密文的计算、访问控制、检索等技术，能够实现数据的可用不可见，保护数据在计算、存储过程中的隐私性。其中，基于属性的密文访问控制技术能够为外包存储的数据提供密文存储能力，以及基于属性的访问控制能力，为可信数据空间中数据的隐私保护以及信任关系的构建提供基础。本文将可信数据空间中的信任关系构建及“三权”分置的实现融入基于属性的密文访问控制过程中，在数据安全外包存储的同时，实现数据托管方的可信验证，与基于“三权”分置模式的访问授权。本文方案是对可信数据空间可信生态环境建立的探索，具体方案的应用以及本方案与可信数据空间中其他参与方的配合机制，还需要在可信数据空间发展过程中进行讨论和优化。

参考文献：

[1] 吴国威，樊宁，汪来富，等. 云环境下基于属性加密体制算法加速方案[J]. 电信科学，2019，35(11)：7. DOI：10.11959/j.issn.1000-0801.2019145.

[2] 潘洁，侯慧芳，陈曦，等. 面向算力网络的多方安全协同线性回归研究[J]. 电信科学，2024，40(8)：162-171. DOI：10.11959/j.issn.1000-0801.2024204.

[3] V.Goyal，O.Pandey，A.Sahai，B.Waters，“Attribute-based encryption for fine grained access control of encrypted data，” CCS.，89－98，2006.

[4] J. Bethencourt， A. Sahai， B. Waters，“Ciphertext-policy attribute-based encryption，” IEEE S&P.，321－334，2007.

[5] M. Chase，“Multi-authority attribute based encryption，” Theory of Cryptography，pp. 515－534，2007

[6] M. Chase and S. Chow，“Improving privacy and security in multiauthority attribute-based encryption，” in Proceedings of the 16th ACM conference on Computer and communications security，2009，pp. 121－130.

[7] Ming Li，Shucheng Yu，Yao Zheng，Kui Ren，and Wenjing Lou，“Scalable and Secure Sharing of Personal Health Records in Cloud Computing Using Attribute-Based Encryption，” IEEE Transactions on parallel and distributed systems，Volume 24，No.1，2013.

[8] Thakur A ，Ranga V ，Agarwal R .Revocable and Privacy-Preserving CP-ABE Scheme for Secure mHealth Data Access in Blockchain[J].Concurrency & Computation：Practice & Experience，2025，37.DOI：10.1002/cpe.70064.

[9] Luo W ，Lv Z ，Yang L ，et al.FOC-PH-CP-ABE：An Efficient CP-ABE Scheme With Fully Outsourced Computation and Policy Hidden in the Industrial Internet of Things[J].IEEE sensors journal，2024(18)：24.DOI：10.1109/JSEN.2024.3432276.

[10] Zhou T ，Tang Z ，Zeng S ，et al. Deduplication-enabled CP-ABE with revocation[J].Peer-to-Peer Networking and Applications，2025，18(2).DOI：10.1007/s12083-024-01863-z.

[11] Sensors，Journal of.Retracted：Efficient and Secure Key Management and Authentication Scheme for WBSNs Using CP-ABE and Consortium Blockchain[J].Journal of Sensors，2023. DOI：10.1155/2023/9865746.

[12] 徐小龙，张栖桐，周静岚.NC-MACPABE：Non-centered multi-authority proxy re-encryption based on CP-ABE for cloud storage systems[J]. 中南大学学报：英文版，2017，24(4)：12. DOI：10.1007/s11771-017-3483-z.

[13] Li J ，Yao W ，Han J ，et al.User Collusion Avoidance CP-ABE With Efficient Attribute Revocation for Cloud Storage[J].IEEE Systems Journal， 2017： 1-11. DOI： 10.1109/JSYST.2017.2667679.

[14] Wang H ，He D ，Shen J ，et al.Fuzzy matching and direct revocation：a new CP-ABE scheme from multilinear maps[J].Soft Computing，2017.DOI：10.1007/s00500-017-2488-8.