Operations Research and Fuzziology 运筹与模糊学, 2021, 11(2), 168-176
Published Online May 2021 in Hans. http://www.hanspub.org/journal/orf
https://doi.org/10.12677/orf.2021.112021

Hans汉斯

基于Petri网的谓词加密方案设计

谢天皓[1]，彭长根[1,2]

[1]贵州大学公共大数据国家重点实验室(计算机科学与技术学院)，贵州 贵阳
[2]贵州大学密码学与数据安全研究所，贵州 贵阳
Email: 906912081@qq.com

收稿日期：2021年4月8日；录用日期：2021年5月13日；发布日期：2021年5月20日

摘 要

现有的谓词加密方案有谓词组合动态转换、参数重用和确定性有限自动机等方法，针对他们在解密时对于解密权限的判断都需要大量的开销而效率不足的问题，基于Petri网提出了一种新的谓词加密方案，对比原有的有限状态自动机方案，可利用Petri网的可达性预测特点对申请者的解密权限进行预测，从而节省开销，提升运算速率。首先形式化定义了该方案，给出了对解密许可的定义；其次利用双线性对实现了Petri网方案的具体构造；最后给出了本方案的分析与正确性证明，并与有限状态机方案做了效率分析对比。

关键词

谓词加密，双线性对，Petri网

A Predicate Encryption Scheme Based on Petri.net

Tianhao Xie[1], Changgen Peng[1,2]

[1]State Key Laboratory of Public Big Data of Guizhou University (College of Computer Science and Technology), Guiyang Guizhou
[2]Institute of Cryptography and Data Security, Guizhou University, Guiyang Guizhou
Email: 906912081@qq.com

Received: Apr. 8th, 2021; accepted: May 13th, 2021; published: May 20th, 2021

Abstract

The existing predicate encryption schemes include predicate combination dynamic conversion, parameter reuse and deterministic finite automata. They need a lot of cost to determine the de-

文章引用: 谢天皓, 彭长根. 基于 Petri 网的谓词加密方案设计[J]. 运筹与模糊学, 2021, 11(2): 168-176.
DOI: 10.12677/orf.2021.112021

cryption authority when decrypting, which may lead to congestion when a large number of users apply for decryption. Based on Petri.net a new predicate encryption scheme is proposed. Firstly, the scheme is formally defined and the decryption permission is defined. Secondly, it is implemented by bilinear pairing based on Petri.net. Finally, the analysis and correctness of the scheme are given. Compared with the traditional predicate encryption scheme, this scheme can make use of Petri.net. In order to save the cost and improve the operation speed, the reachability prediction feature of the applicant is used to predict the decryption authority of the applicant.

Keywords

Predicate Encryption, Bilinear Pair, Petri.net

1. 引言

传统的公钥加密体制是粗粒度的，当数据拥有者使用给定的公钥 *PK* 加密消息 *M* 时，只有拥有对应公钥 *PK* 的私钥 *SK* 的用户才能解密密文，将其还原为明文。因此只在个体之间的交互较好，加密后的数据是发给一个数据拥有者指定的接收者，数据拥有者相当于是已经知道对方的身份。然而，传统的公钥加密很难做到只让指定的人访问指定的数据，而谓词加密就可以，这个新的加密机制对加密数据能提供更加细粒度的访问控制，因此对谓词加密的研究是很有必要的。上述情况基于属性加密也能做到，且引入属性加密证实为了解决此问题，谓词加密最初是其延伸，后为防止属性匹配的过程中暴露访问者的隐私或者敏感信息，才将谓词加密作为隐藏访问结构的属性加密的方法。不过谓词加密目前没有普遍应用，原因在于如果要保证策略的私密性，谓词加密会比较复杂且效率难以保证，而且概念提出较晚，近几年才起步研究，因此大多都只停留在理论研究阶段。基于状态转移模式的谓词加密可以通过对密文属性以及密钥属性基于双线形群进行特定的编码，将明文的策略转化为一些向量来表示，从而保证其谓词的私密性，通过不同的结构来表示状态转移，还能通过不同结构固有的优势来转换为谓词加密的额外应用。

在传统的公钥密码学中，实体的身份和公钥通常是通过由证书权威颁发的公钥证书来绑定。然而，证书的存储和管理需要很高的计算和存储开销，大大加重了系统负担。为了简化公钥管理过程，基于身份加密(identity-based encryption, IBE)的概念于1984年由Shamir [1]提出，它避免了传统公钥密码系统建立和管理公钥基础设施的困难。但是，基于身份的加密仍然存在例如如何获取接收者的公钥，以及接收者公开信息使得自己隐私暴露等问题。因此，2005年Sahai与Waters [2]提出了基于属性加密(Attribute-based encrypti, ABE)的概念。

基于属性加密体制虽然实现了细粒度的访问控制，但是属性策略的匹配仍然具有暴露访问者部分隐私的风险。2007年，Katz、Sahai与Waters [3]首次提出了谓词加密(Predicate Encryption, PE)的概念，并构造了一个支持内积查询、析取和多项式等式的内积谓词加密方案。不过，此时提出的谓词匹配方案是以明文形式进行的，这就仍然存在在谓词匹配过程中泄露访问者隐私的问题，这在公钥内积加密方案中是不可避免的。2009年，Shen，Shi和Waters [4]首先提出了谓词私密性的概念，目的就是使得在谓词匹配中，也可以像密文一样不透露任何信息，其最大的特点就是在使得明文具有私密性的同时还保护了谓词的私密，但是此方案的运算开销是非常大的，而且只具有选择性安全，难以应用。2010年，Carlo Blundo等[5]

提出了一个语义安全的构造，使得谓词匹配时不显示任何与关联模式有关的信息，并大大提高了构造的效率。2012年，Yoshino等[6]提出一个基于三元组群的对称内积谓词加密方案，该方案满足在非交互假设下的选择性安全模型，相比文献[4]中的四元组群，该方案中明文空间更大，且在适当的安全参数下能够更好地抵抗整数分解攻击。2015年，Gay等[7]针对多维范围和多维子集查询，构造了一种基于格的谓词加密方案。该方案具有选择性安全和弱属性隐藏的特点，其安全性基于容错学习问题(LWE)假设。2017年，Katz [8]等基于双线性群中的标准假设提出了两个新的构造；这些构造具有非常有效的解密算法(分别由一个和两个配对计算组成)且密钥短，并且证明了无随机预言构造的选择性安全性，还通过描述到更复杂原语的几个黑盒转换来证明子集谓词加密的有用性。2018年，Datta [9]等提出了一个基于模拟的自适应的强部分隐藏(PHPE)方案，用于谓词计算公共属性上的算术分支程序(ABP)，然后是私有属性上的内积谓词，证明了该方案对任意先验有界密文和无界(多项式)解密密钥数都是安全的，这是基于仿真的自适应安全框架中的最佳方案。这直接意味着，他们的方案还实现了基于不可区分性的强部分隐藏安全性，以防对手请求无限(多项式)数量的密文和解密密钥。2019年，Nuttapong [10]提出大量谓词组合无边界、动态转换以及用有限状态自动机(DFA)的谓词加密方法，这是首次提出使用状态转移的方式来进行谓词加密。

由于谓词加密理论的研究起源于理论研究，而且具有高度的复杂性，在行业中无法广泛应用，所以许多开放问题仍然存在，例如文献[10]中提出的多种方案，由于其结构特点，它在可预测性、并发访问控制等方面较薄弱，因此，改进这种高效、灵活的机制对推动云存储的普及起着重要的作用，如何设计谓词加密方案让其在变得更为高效的同时保证谓词的私密性以及可证明安全性的基础上让其得到更广泛的应用，已经逐渐成为近年来谓词加密的主要发展方向。

2. 预备知识

2.1. Petri.net

Petri 网是对离散并行系统的数学表示。它是 20 世纪 60 年代由卡尔·A·佩特里发明的，适合于描述异步的、并发的计算机系统模型。Petri 网既有严格的数学表述方式，也有直观的图形表达方式，既有丰富的系统描述手段和系统行为分析技术，又为计算机科学提供坚实的概念基础。经典的 Petri 网是简单的过程模型，由两种节点：库所和变迁，有向弧，以及令牌等元素组成的。Petri 网具有如下特点：

1) Petri 网的元素：库所(Place)、变迁(Transition)、有向弧(Connection)、令牌(Token)。

2) Petri 网需要满足以下规则：有向弧是有方向的、两个库所或变迁之间不允许有弧、库所可以拥有任意数量的令牌。

3) Petri 网的行为：如果一个变迁的每个输入库所(input place)都拥有令牌，该变迁即为被允许(enable)。一个变迁被允许时，变迁将发生(fire)，输入库所(input place)的令牌被消耗，同时为输出库所(output place)产生令牌。如果变迁的发生是完整的，那没有一个变迁只发生了一半的可能性。如果出现一个变迁，其输入库所的个数与输出库所的个数不相等，令牌的个数将发生变化。如果 Petri 网络是静态的，那将不会在一个变迁后忽然产生另一个变迁或者库所而改变 Petri 网结构。如果 Petri 网的状态由令牌在库所的分布决定，那么变迁发生完毕、下一个变迁等待发生的时候才有确定的状态，正在发生变迁的时候没有确定的状态。两个变迁争夺一个令牌的情形被称之为冲突。当发生冲突时，由于 Petri 网的时序不确定，因此具体哪个变迁发生也不确定。弧的个数决定了消耗/产生的令牌的个数。

2.2. 符号解释

$Z_N^{m\times l}$ 表示所有维度为 $m\times l$ 的矩阵集，元素在 Z_N 中。当 $m=1$ 且 $l>1$ 时，则也表示一个行向量，反之当 $m>1$ 且 $l=1$ 时，也表示一个列向量。若向量 $s\in Z_N^{1\times a}$，向量 $t\in Z_N^{1\times a}$，则 $(s,t)\in Z_N^{1\times(a+b)}$，表示 (s,t) 的意

思为将 **t** 接在 **s** 后组成一个新的行向量，相当于串联它们。

3. 谓词 Petri 网形式化定义

设输入 Petri 网的密文属性为 $Y=\{y_0,y_1,\cdots,y_l\}$，密钥属性为 X，为了表示转换过程中的步骤，我们将谓词 Petri 网 Pe 定义为如下六元组：$Pe=\{P,T,Token,p_0,p_{\mu-1},Trans\}$。他们分别代表：

库所集 $P=\{p_0,p_1,\cdots,p_l\}$，包含了 Petri 网中的所有库所，六元组中的 p_0 与 $p_{\mu-1}$ 就包含在其中，其中 p_0 代表起始库所，$p_{\mu-1}$ 代表接受的终态库所。

转换集 $Trans=\{t_0,t_1,\cdots,t\}_b$，包含了 Petri 网中所有库所之间的转换，它们是有向的，由于数量和库所数量没有明确的对应关系，因此此处假设有 b 个转换数量。

令牌集 $Token=\{tk_0,tk_1,\cdots,tk_l\}$，代表 Petri 网中进行转换所需要的所有令牌。

过渡表 $T\in P\times Trans\times Token\times P\times X$，收集已经通过检测的密钥，其为转换前后库所、对应转换、令牌以及相应的密文属性的笛卡尔积。

当输入 $Y=\{y_0,y_1,\cdots,y_l\}$ 后，有一系列的库所 $\left(p^{(0)},p^{(1)},\cdots,p^{(l)}\right)$、一系列转换 $\left(t^{(0)},t^{(1)},\cdots,t^{(l)}\right)$ 以及对应的一系列令牌 $\left(tk^{(0)},tk^{(1)},\cdots,tk^{(l)}\right)$ 满足对所有 $i\in[0,l-1]$ 都存在 $\left(p^{(i)},t^{(i)},tk^{(i)},p^{(i+1)},x^{(i)}\right)\in T$，满足 $P\left(x^{(i)},y_{(i)}\right)=1$ 且 $p^{(0)}=p_0,p^{(l)}=p_{\mu-1}$ 时，预测 Petri 网 Pe 接受输入 Y，此时有 $P(Pe,Y)=1$，允许解密。

4. 具体方案

使用文献[10]的对密文以及密钥的一次编码方案，我们先获得索引 $K=(N,Par)$，其中 Par 代表一些指定参数；接着运行 $Param(Par)$，得到公共变量数 n，根据 n 获取变量 $b=(b_1,b_2,\cdots,b_n)$，向量将插入到后续的编码中；继续运行密文一次编码函数 $EncCt(y,N)$，得到 w_1,w_2 两个参数以及关键向量 $c\left(s,s^*,b\right)$，其中 w_1,w_2 分别为 s,s^* 的元素个数，即 $s=(s_0,s_1,\cdots,s_{w1})$，$s^*=\left(s_0^*,s_1^*,\cdots,s_{w2}^*\right)$；最后运行密钥一次编码函数 $EncKey(X,N)$，得到参数 m_1,m_2 以及关键向量 $k\left(r,r^*,b\right)$，其中 m_1,m_2 分别为 r,r^* 的大小，类似密文编码中的定义。根据文献 10 的描述，谓词加密目前是默认安全的，因此本文也没有给出安全性证明。

当 $P(X,Y)=1$ 时就可配对 c 与 k，获得 αs_0，也就说明 c_ir_u 与 k_js_t 存在线性组合性质。我们使用符合双系统组的双线性对 $G=(G_1,G_2)$，其中 g_1,g_2 是他们的生成元，加密公钥为 $\left(gb_2,e(g_1,g_2)\alpha\right)$，$e$ 是恒元。加密所得的密文 Y 由 gc_2 与 gs_2 组成，密钥 X 由 gk_1 与 gk_2 组成，通过对 c 与 k 解密来获得 αs_0。

以上内容是文献 10 中对一般的谓词加密所需要进行的步骤，为了让它适应 Petri 网，从而进行联动，我们将对密文属性以及密钥属性进行如下二次编码：首先定义函数 $Param'(Par)$，它的返回结果是 $Param(Par)$ 的返回值的三倍并加 1，假如 $Param(Par)$ 返回值为 n，那我们就得到的是 $3n+1$，并且根据 $Param(Par)$ 得到的 b 进行如下定义：$b_1=\left(b_{1,1},b_{1,2},\cdots,b_{1,n}\right)$，$b_2=\left(b_{2,1},b_{2,2},\cdots,b_{2,n}\right)$，$b'=\left(b_1,b_2,h_0,g_1,h_1,g_2,h_2\right)$。其次，定义密文属性二次编码函数 $EncCt'(Y,N)$，其作用是：解析密文 $Y=\{y_0,y_1,\cdots,y_l\}$，对 $i\in[1,l]$，运行密文属性一次编码函数 $EncCt\left(y_i,N\right)$，得到多项式向量 $c^{(i)}$，将其分割为两份，分别为 $c^{(1,i)}=c^{(i)}\left(s^{(1,i)},s^{*(1,i)},b_1\right)$ 与 $c^{(2,i)}=c^{(i)}\left(s^{(2,i)},s^{*(2,i)},b_2\right)$，其中，两组 s 具有如下关系：

$$s_0'=\begin{Bmatrix}s_0^{(1,i+1)}, & i=0\\ s_0^{(1,i+1)}=s_0^{(2,i)}, & i\in[1,l-1],i\in N\\ s_0^{(2,i)}, & i=l\end{Bmatrix} \tag{4.1}$$

然后定义 $c_0'=h_0s_{re}^{(0)}$ 以及当 $i\in[1,l]$ 时，

$$c_i'=h_1s_{re}^{(i-1)}+g_1s_0'^{(i-1)}+h_2s_{re}^{(i)}+g_2s_0'^{(i)},$$

此函数最后输出

$$c' = \left(c'_0, c'_1, \cdots, c'_l, \left(c^{(1,i)}, c^{(2,i)}\right)_{i\in[1,l]}\right) \tag{4.2}$$

最后要做的就是对密钥属性进行二次编码，构造函数 $EncKey'(Pe, N)$，令 $\mathrm{Pe} = \{P, T, Token, p_0, p_{\mu-1}, Trans\}$，$t = \left\{\left(p_{v_t}, t_{v_t}, tk_{v_t}, p_{\omega_t}, x_t\right)_{t\in[1,m]}\right\}$，其中 v_t 表示第 t 次转换中的起始，ω_t 则代表到达，对所有 $t\in[1,m]$，运行 $EncKey(x_t, N)$ 得到多个向量 $K(t)$，类似于密文编码，我们将其分为两组，分别为

$$k^{(1,t)} = k^{(t)} \cdot \left(r^{(1,t)}, r*^{(1,t)}, b_1\right) \tag{4.3}$$

与

$$k^{(2,t)} = k^{(t)} \cdot \left(r^{(2,t)}, r*^{(2,t)}, b_2\right) \tag{4.4}$$

然后进行如下定义：

$$\begin{Bmatrix} k'_0 = -u_0 + r_{re}^{(0)} h_0 \\ k'_{1,t} = u_{v_t} + r_{re}^{(t)} h_1 \\ k'_{2,t} = -u_{\omega_t} + r_{re}^{(t)} h_2 \end{Bmatrix}$$

最后得到输出结果：

$$k' = \left(k'_0, \left(k'_{1,t}, k'_{2,t}, k^{(1,t)}, k^{(2,t)}\right)_{t\in[1,m]}\right) \tag{4.5}$$

密文密钥都经过二次编码后得到了许多对应的参数，最后验证是否有

$$k'_i = k'_0 = c'_i = c_0 = 0$$

成立，如果成立则 $P(X, Y) = 1$，允许解密，否则为 0，不允许解密。

5. 方案分析

在本节中，我们将对方案的一些性质进行分析，并通过运算的方式给出方案的正确性证明。

5.1. 正确性分析

密文加密中，通过一定变换，可将 c_0 替换并计算为：

$$H_0 s_{re}^{(0)} = \begin{pmatrix} 0 & a & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \vdots & 0 & \cdots & a_l \\ 0 & \cdots & \cdots & 0 \end{pmatrix} \left(L_1^{d'_2}\right)^{\mathrm{T}} = 0$$

将 c'_i 替换并计算为：

$$H_1 \left(s_{re}^{(i-1)}\right)^{\mathrm{T}} = \begin{pmatrix} a & \cdots & 0 & 0 \\ \vdots & \ddots & \vdots & \vdots \\ 0 & \cdots & a_l & \vdots \\ 0 & \cdots & \cdots & 0 \end{pmatrix} \begin{pmatrix} 0 \\ \vdots \\ 0 \\ L_1^{d_2} \\ 0 \\ \vdots \\ 0 \end{pmatrix} = \left(L_i^{d_1}\right)^{\mathrm{T}}$$

$$H_2\left(s_{re}^{(i)}\right)^{\mathrm{T}}=-\overbrace{\begin{pmatrix}0&a&\cdots&0\\ \vdots&\vdots&\ddots&\vdots\\ \vdots&0&\cdots&a\\ \vdots&a&\cdots&0\\ \vdots&\vdots&\ddots&\vdots\\ 0&0&\cdots&a\\ a&0&\cdots&0\\ 0&a&\cdots&0\\ \vdots&\vdots&\ddots&\vdots\\ 0&0&\cdots&a\end{pmatrix}}^{l+1}\begin{pmatrix}0\\ \vdots\\ 0\\ L_1^{d_2}\\ 0\\ \vdots\\ 0\end{pmatrix}=-\left(L_i^{d_i'}\right)^{\mathrm{T}}-\left(L_{i+1}^{d_i'}\right)^{\mathrm{T}}-\left(L_{2l+i+1}^{d_i'}\right)^{\mathrm{T}}$$

$$G_2\left(s_0'^{(i)}\right)^{\mathrm{T}}=\begin{pmatrix}0&\cdots&\cdots&0\\ \vdots&\ddots&\iddots&\vdots\\ \vdots&\iddots&\ddots&\vdots\\ 0&\cdots&\cdots&0\\ 0&a&\cdots&0\\ \vdots&\vdots&\ddots&\vdots\\ 0&0&\cdots&a\\ a&0&\cdots&0\\ 0&a&\cdots&0\\ \vdots&\vdots&\ddots&\vdots\\ 0&0&\cdots&a\end{pmatrix}\begin{pmatrix}0\\ \vdots\\ 0\\ L_1^{d_2}\\ 0\\ \vdots\\ 0\end{pmatrix}=\left(L_{l+i}^{d_1'}\right)^{\mathrm{T}}+\left(L_{2l+i+1}^{d_1'}\right)^{\mathrm{T}}$$

联立以上四个等式以及对 c_i' 的定义，有

$$\left(L_i^{d_i'}\right)^{\mathrm{T}}-\left(L_i^{d_i'}\right)^{\mathrm{T}}-\left(L_{i+1}^{d_i'}\right)^{\mathrm{T}}-\left(L_{2l+i+1}^{d_i'}\right)^{\mathrm{T}}+\left(L_{l+i}^{d_1'}\right)^{\mathrm{T}}+\left(L_{2l+i+1}^{d_1'}\right)^{\mathrm{T}}=0$$

至此已经证明了 $c_i'=c_0=0$，而此时正好代表对密文属性进行了正确验证。

类似的，在密钥二次编码函数中，根据密钥一次编码中的性质，可以通过一定变换，可将 k_0 替换并计算。同样可以得到如下等式：

$$\left(L_i^{d_i'}\right)^{\mathrm{T}}-\left(L_i^{d_i'}\right)^{\mathrm{T}}-\left(L_{i+1}^{d_i'}\right)^{\mathrm{T}}-\left(L_{2l+i+1}^{d_i'}\right)^{\mathrm{T}}+\left(L_{l+i}^{d_1'}\right)^{\mathrm{T}}+\left(L_{2l+i+1}^{d_1'}\right)^{\mathrm{T}}=0$$

此时正好代表对密钥属性进行了正确验证，也就是 $k_i'=k_0'=0$。联立刚才所证明的 $c_i'=c_0=0$，就能得到谓词表达式的条件 $k_i'=k_0'=c_i'=c_0=0$，使得布尔函数返回值为 1，允许解密。解密完成后将对应的密文属性 Y 与密钥属性 X 加入过渡表。

5.2. 性质分析

我们先做如下定义：

对 $i\in[0,l]$，令 $Y_i=\left(y_{i+1},\cdots y_l\right)$，也就是密文 Y 中的后面 $l-i$ 个向量，如果 $Y_0=Y$ 则代表 Y_l 为空。对 $k\in[0,u-1]$，定义 $V_k=\left\{i\in[0,l]\middle|Pe\text{接受}Y_i\right\}$，令 $V_k^{+1}=\left\{i+1\middle|i\in V_k\right\}$，相反，令 $U_i=\left\{i\in[0,l]\middle|Pe\text{接受}Y_i\right\}$。

接下来，我们不难证明：若 Pe 不接受 Y，则有

$$0\notin V_0,0\notin U_0$$

对 $t\in[1,m]$， $i\in[1,l]$ 有 $i\in\left(V_{\omega_t}\cup V_{v_t}^{+1}-V_{\omega_t}\cap V_{v_t}^{+1}\right)\Rightarrow P(x_t,y_i)=0$ 或者说

$$P(x_t,y_i)=1\Rightarrow\left(v_t\in U_{i-1}\wedge\omega_t\in U_i\right)\vee\left(v_t\notin U_{i-1}\wedge\omega_t\notin U_i\right)。$$

简单的说，如果预测 Petri 网 Pe 不接受密文属性 Y，那也就意味着在某一库所的子密文属性与子密钥属性匹配失败，由于 Petri 网本身的性质，也不会存在同一个变迁指向两个库所的情况，因此只能代表失败，而不可能是另一条正确变迁到这个拒绝库所。

6. 方案对比分析

目前对于状态转移类型的谓词加密分析比较尚比较缺乏，因此本节在进行效率分析时，除了与原有的对于原有的 DFA 方案，与近期提出的几个谓词加密与属性加密方案进行了对比。

本方案中主要是对密文属性和密钥属性进行运算，而公钥是事先就已根据用户属性计算得来，因此此时公钥的计算复杂度为 O(1)。在 DFA 方案中，收到一个解密请求之后对其密钥属性的验证需要进行复杂度为 O(t)的密文编码，而后解析 DFA，对密文编码和密钥编码所产生的向量各自进行若干次矩阵相乘运算，还需要再次对编码数据进行多次复杂度为 O(m)的迭代检测属性。而 Petri 网方案在进行类似的密文编码与若干矩阵相乘运算后，可以通过 Petri 网的可达性预测模型来代替 DFA 方案中后续的迭代属性验证，即使使用效率最低的穷举法其复杂度也与迭代法相同都为 O(m)，因此只需改变可达性预测的方法，就可以降低复杂度，进而提升效率。本方案的加密复杂度与类似的加密方案对比如表 1 所示，可以看出在第一次加密中，虽然没有明显超越，但是复杂度也是最低的一部分。

Table 1. Comparison of accept first encryption
表 1. 第一次加密接受比较

方法	PK	SK	CT	加密
OT10	O(TR)	O(tR)	O(m)	完全
OT12	O(1)	O(t)	O(m)	完全
YWB18	O(1)	O(t)	O(m)	选择性
CP-NSP	O(T)	O(1)	$O(T^3m)$	完全
ACM	O(un)	O(1)	O(m)	完全
DFA	O(1)	O(t)	O(m)	完全
This work	O(1)	O(t)	O(m)	完全

而当利用可达性预测拒绝访问时，就能体现出明显的优势，其效率对比如表 2 所示：

Table 2. Comparison of refuse first encryption
表 2. 第一次加密拒绝比较

方法	PK	SK	CT	加密
OT10	O(TR)	O(tR)	O(m)	完全
OT12	O(1)	O(t)	O(m)	完全
YWB18	O(1)	O(t)	O(m)	选择性
CP-NSP	O(T)	O(1)	$O(T^3m)$	完全
ACM	O(un)	O(1)	O(m)	完全
DFA	O(1)	O(t)	O(m)	完全
This work	O(1)	O(t)	O(1)	完全

另外，过渡表的加入也可以允许已经通过检测的密钥直接获得解密权限，不需要再次进行复杂的检测运算，其复杂度对比如表 3 所示，对于已检测过的用户而言，本文可以直接检测其是否在过渡表内，对过渡表进行简单的搜寻操作即可，如果在则直接允许解密，在指定访问类型的用户量不大的情况下，效率是有显著提升的。

Table 3. Comparison of accept second encryption
表 3. 第二次加密比较

方法	PK	SK	CT	加密
OT10	O(TR)	O(tR)	O(m)	完全
OT12	O(1)	O(t)	O(m)	完全
YWB18	O(1)	O(t)	O(m)	选择性
CP-NSP	O(T)	O(1)	$O(T^3m)$	完全
ACM	O(un)	O(1)	O(m)	完全
DFA	O(1)	O(t)	O(m)	完全
This work	O(1)	O(1)	O(1)	完全

7. 结束语

参照原有 DFA 方案，分别设计出适用于 Petri 网环境的对于密文属性与密钥属性的一次编码方案、二次编码方案，方法同样是基于一个双线性群，在输入密钥属性和密文属性以及给定参数之后，将其转换为特定的向量组，在向量组上进行一定的运算之后，判断其中的四个向量是否满足一个特定的线性方程，作为 $P(X,Y)$ 是否等于 1 的判定基准，也就是能否解密的标准。通过 Petri 网的可达性预测特点成功解决了确定有限状态自动机在大量拒绝访问条件下浪费大量计算资源的问题，使得预测不通过的用户直接被拒绝访问，避免大量的密文密钥属性验证而带来的资源浪费，同时引入过渡表后，对于可访问者较少的情况，也能有效降低计算开销。

参考文献

[1] Hellmanme, D. (1976) New Directions in Cryptography. *IEEE Transactions on Information Theory*, **86**, 644-654. https://doi.org/10.1109/TIT.1976.1055638

[2] Sahai, A. and Waters, B. (2005) Fuzzy Identity-Based Encryption. *Advances in Cryptology*: *EUROCRYPT* 2005, **3494**, 457-473. https://doi.org/10.1007/11426639_27

[3] Katz, J. and Sahai, A. (2007) Brent Waters: Predicate Encryption Supporting Disjunctions. *IACR Cryptology ePrint Archive*, 404.

[4] Ling, S., Nguyen, K., Wang, H.X. and Zhang, J.Y. (2019) Server-Aided Revocable Predicate Encryption: Formalization and Lattice-Based Instantiation. *The Computer Journal*, **62**, 1849-1862. https://doi.org/10.1093/comjnl/bxz079

[5] Yoshino, M., Kunihiro, N., Naganuma, K. and Sato, H. (2012) Symmetric Inner-Product Predicate Encryption Based on Three Groups. *ProvSec* 2012: *Provable Security*, **7496**, 215-234. https://doi.org/10.1007/978-3-642-33272-2_14

[6] Datta, P., Dutta, R. and Mukhopadhyay, S. (2019) Succinct Predicate and Online-Offline Multi-Input Inner Product Encryptions under Standard Static Assumptions. *Journal of Information Security and Applications*, **48**, Article ID: 102353. https://doi.org/10.1016/j.jisa.2019.06.009

[7] Gay, R., Méaux, P. and Wee, H. (2015) Predicate Encryption for Multi-Dimensional Range Queries from Lattices. *Public Key Cryptography*, **35**, 752-776. https://doi.org/10.1007/978-3-662-46447-2_34

[8] Katz, J., Maffei, M., Malavolta, G. and Schröder, D. (2017) Subset Predicate Encryption and Its Applications. *CANS* 2017: *Cryptology and Network Security*, **11261**, 115-134. https://doi.org/10.1007/978-3-030-02641-7_6

[9] Datta, P., Okamoto, T. and Takashima, K. (2018) Adaptively Simulation-Secure Attribute-Hiding Predicate Encryption.

Advances in Cryptology: *ASIACRYPT* 2018, **11273**, 640-672. https://doi.org/10.1007/978-3-030-03329-3_22

[10] Attrapadung, N. (2019) Unbounded Dynamic Predicate Compositions in Attribute-Based Encryption. *Advances in Cryptology*: *EUROCRYPT* 2019, **11476**, 34-67. https://doi.org/10.1007/978-3-030-17653-2_2