基于 EBPN 模型的电子商务结构化安全验证方法

宋浩天,刘 伟
(山东科技大学 计算机科学与工程学院,山东 青岛 266590)

摘 要:电子商务借助互联网中 app 等形式作为媒介,打破传统的面对面交易体系,使系统变得更复杂,以至易产生更多恶意行为,这些恶意行为可能来自用户,商家或第三方平台等。第三方支付平台的加入使程序逻辑设计更复杂,更多结构化问题产生。为此,首先,根据基于 Petri 网的电子商务业务流程网模型,对电子商务系统进行建模推演,提出了电子商务业务流程关键网模型。其次,根据基于角色访问控制策略中的基数约束和职责分离进行改进,提出了电子商务模型中的行为分离和变迁约束。通过对序列发生的充分性进行补充,完善在 Petri 网模型中变迁序列发射的充分必要条件,为构建线性不等式提供了条件。最后通过构造线性规划的方法对网模型系统中的变迁约束和行为分离进行求解,进而验证系统的结构化安全性。

关键词:电子商务;Petri 网;基数约束;整数线性规划;结构化安全
中图分类号:TP311 **文献标识码**:A **文章编号**:1673-629X(2024)10-0100-10
doi:10.20165/j.cnki.ISSN1673-629X.2024.0186

Electronic Commerce Structured Security Verification Method Based on EBPN Model

SONG Hao-tian, LIU Wei
(School of Computer Science and Engineering, Shandong University of Science and Technology, Qingdao 266590, China)

Abstract: With the help of apps and other forms on the Internet as a medium, e-commerce breaks the traditional face-to-face transaction system, making the system more complex and prone to more malicious behaviors, which may come from users, merchants or third-party platforms. The addition of the third-party payment platform makes the program logic design more complicated and more structural problems arise. Therefore, first of all, according to e-commerce business process network model based on the Petri net, the e-commerce system is modeled, and the e-commerce business process critical network model is proposed. Secondly, according to cardinality of constraints and separation of duties based on the role access control strategy for improvement, the separation of action and constraints of transition in the e-commerce system is put forward. By supplementing the sufficiency of sequence generation, the sufficient and necessary conditions of transition sequence emission in Petri net model are perfected, which provides the conditions for constructing linear inequalities. Finally, the constraints of transition and separation of action in the network model system are solved by constructing linear programming method, and then the structural security of the system is verified.

Key words: electronic commerce; Petri net; cardinality of constraint; integer linear programming; structural security

0 引 言

电子商务突然兴起带动了各行各业经济的发展,成为许多国家新的经济增长点[1]。然而,因为网络安全的不稳定性,三方并行的电子商务模型使得模型在结构上存在诸多安全隐患。不同于程序结构体系中的程序 bug,通信拥堵以及安全协议之类问题[2],业务流程模型存在的隐患在整个系统设计之初就已经产生,简称其为结构化问题[2]。现在对于商务网络安全的研究主要集中在工作流程中的访问控制[3]、死锁的避免[3]等问题上,这些问题的根源来自于结构化安全。

Petri 网是一种用于说明并发性和建模分布式系统的合适数学工具,因其特殊数学性质,其多应用于工作流的建模分析和安全分析[4-5]。众多形式化验证使用Petri网进行建模,如基于Petri网的访问控制验证

收稿日期:2023-12-15 **修回日期**:2024-04-17
基金项目:山东省教育教学研究重点课题(2023JXZ001)
作者简介:宋浩天(1999-),男,硕士研究生,研究方向为 Petri 网、访问控制;通信作者:刘 伟(1977-),男,教授,博导,研究方向为过程挖掘、Petri 网理论与应用。

方法已被提出和解决[3],基于逻辑博弈 Petri 网的 B2C 电子商务信用机制风险行为分析也完成了一些工作[6],基于扩展逻辑 Petri 网的业务流程建模提出了一种对电子商务进行建模的方法[7]。然而这些均没有涉及到电子商务结构化问题的研究。

电子商务系统构建十分复杂,参与者之间通信通过 web 服务和 API 进行通信,如 Cashier-as-a-Service (CaaS)[2]。电子商务业务流程网(E-commerce Business Processes Net,EBPN)是现有的基于 Petri 网对三方并行的电子商务系统进行建模的形式化工具,该模型将参与者通信的 API 设计为变迁(transitions),将电子商务平台中出现的交易参数定义为不同令牌(token)的种类,因此该网设计十分契合电子商务系统。EBPN 网的性质和一般 Petri 网相比并无二致,不同之处在于 EBPN 网处于三维的维度分析问题。然而,EBPN 网对于漏洞的探索需要依靠违规序列来验证,文献[2]中提供了两个方法来验证问题安全性。一是通过对电子商务业务流程案例的分析,构建了导致安全问题的违法行为,然后将其转换为 EBPN 模型的行为序列,最后手动代入模型中,判断是否可以执行成功。该方法提前确认违规行为的部分序列变迁,然后需要从初始状态开始,完整遍历整个图,来确定所有的模型可达变迁中包含此变迁。该方法需要人工模拟 EBPN 网中所有可能序列并对比是否存在某个或某些可达序列中包含违规行为的部分变迁序列,在实现操作上会十分繁琐和冗杂。二是通过使用状态分析法来分析结构安全性,在步骤上也十分复杂。两种方法对于单一的可能产生违规的变迁发生验证是可行的,但验证多个结构化安全问题效率却十分低下。

该文定义一种新模型电子商务业务流程关键网(Critical E-commerce Business Process Net,CEBPN),其在最初的 EBPN 网定义关键库所和变迁,用于解决电子商务系统中的结构化问题,拓展了原始的 EBPN 网。引入一种来自于基于角色的访问控制(Role Based Access Control)[8] 策略中的基数约束(Constraints of Cardinality)[9] 和职责分离约束(Separation of Duties)[8] 来验证 EBPN 模型中的结构化安全问题。RBAC 策略是一种基于角色的访问控制模型,建模通常基于 Petri 网这一数学分析工具。根据关键变迁和库所设计线性规划求解方法。提出方法可以高效率地验证电子商务系统是否存在结构化问题。

1　动机实例

某网站发布的信息,大学生利用漏洞免费吃肯德基获刑。问题(1):在校大学生徐某等人通过在 APP 客户端用套餐优惠券下单,进入待支付状态后暂不支付,之后在微信客户端对优惠券进行退款操作,然后再将之前客户端的订单取消,这时候客户端上竟可以重新获取优惠券,此种方式分文未付骗取了一份优惠券。发现这个漏洞后,徐某除了自己这样点餐操作,还将诈骗得来的套餐产品通过线上交易软件低价出售给他人,从中获利。徐某的行为造成某企业损失 5.8 万余元。问题(2):另外是先在 APP 客户端下单待支付,然后在支付界面进行付款时,竟然可以再使用 APP 客户端中点餐界面添加商品,这时便可以以很小代价付款获得价值远超过付款金额的商品。这两个例子是典型的电子商务的结构化问题,它产生于模型建立之初的结构化漏洞。

结构化安全问题突出体现在系统的多个参与者之间的通信,由于电子商务平台系统越发的复杂,各个用户之间的 API 数量激增,使得系统容易在原始结构上产生各种各样的结构化安全问题。各种的结构化安全问题可能造成不同参与者在系统运行流程上的事故,进而产生一定的经济损失。如何正确验证复杂电子商务系统是否存在相应的结构化安全问题是当前需要解决的问题。

2　相关概念

2.1　Petri 网

$$N=(P,T,F,W) \tag{1}$$

是一个网[10-11],其中:

(1) P 是由圆表示的一个库所的有限集合;

(2) T 是由黑色矩形表示的一组有限的变迁集 $P\cap T=\varnothing$ 且 $P\cup T\neq\varnothing$;

(3) $F\subseteq(P\times T)\cup(T\times P)$ 是一组有向弧;

(4) $W:F\subseteq(P\times T)\cup(T\times P)\to N$ 是每个弧上的权重函数,如果 $(x,y)\in F$,那么满足 $W(x,y)>0$。否则,$W(x,y)=0$。

2.2　EBPN 模型及性质

EBPN 是一个描述电子商务业务流程的形式化模型[2,12],该模型能够很好地对存在第三方交易平台的电子商务业务流程进行建模。给出 EBPN 定义及关联矩阵相关性质。

$$\mathrm{EN}=(P,T;F,D,W,S,G) \tag{2}$$

是一个 EBPN 模型且满足以下条件:

(1) P 是一个库所的有限集合;

(2) T 是一个变迁的有限集合;

(3) $F\subseteq(P\times T)\cup(T\times P)$ 是一组有向弧的集合;

(4) D 是拥有不同类型 token 的有限集合,每一个 $d\in D$ 是由表示交易参数的单词表示;

(5) $W:F\to(a_1\,d_1,a_2\,d_2,a_3\,d_3,\cdots,a_k\,d_k)^k$,表示

每个有向弧上的权重，$a_k\ d_k \in \{0,1\}$，$d_k \in D$ 并且 $k > 0$ 是 D 中数据元素的数量；

(6) S 是系统运行过程中的关键元素，并且 $S \in D$；

(7) $G: T \rightarrow \Pi$ 是一个谓词函数，该函数为每个变迁 t 指定一个谓词，其中 Π 是 D 上的布尔表达式集。

EBPN 模型的三维关联矩阵(Three-dimensional Incidence Matrix)[2]：

假设 EN=($P,T;F,D,W,S,G$) 为一个 EBPN 网，$P=(p_1,p_2,\cdots,p_n)$ 表示库所集合，$T=(t_1,t_2,\cdots,t_m)$ 表示变迁集合，$D=(d_1,d_2,\cdots,d_l)$ 表示拥有不同 token 类型的有限集合，$n,m,l \in \mathbb{N}^+=(1,2,3,\cdots)$。EBPN 网三维输出矩阵可以表示为 $[N^+]=[N^+]_{n\times m\times l}$，输入矩阵可以表示为 $[N^-]=[N^-]_{n\times m\times l}$，关联矩阵可以表示为 $[N]=[N]_{n\times m\times l}$，其中 $N_{ijk}=N_{ijk}^+-N_{ijk}^-$。

$$N_{ijk}^+=\begin{cases}W(t_j,p_i)_k,(t_j,p_i)\in F,i\in\{1,2,\cdots,n\}\\ \quad j\in\{1,2,\cdots,m\},k\in\{1,2,\cdots,l\}\\ \text{一个 1 维的 0 向量，其他}\end{cases} \tag{3}$$

$$N_{ijk}^-=\begin{cases}W(p_i,t_j)_k,(p_i,t_j)\in F,i\in\{1,2,\cdots,n\}\\ \quad j\in\{1,2,\cdots,m\},k\in\{1,2,\cdots,l\}\\ \text{一个 1 维的 0 向量，其他}\end{cases} \tag{4}$$

普通 Petri 网关联矩阵中定义的是库所与变迁的关系，而在 EBPN 加入了不同类型的 token，因此 EBPN 网关联矩阵的形式化定义在上述定义里进行了详细的说明。$\boldsymbol{W}(p_i,t_j)$ 是一个 l 维的向量，而 $W(p_i,t_j)_k$ 是 $\boldsymbol{W}(p_i,t_j)$ 中的第 k 个标量。N_{ijk} 是矩阵 $[N]=[N_{ijk}]_{n\times m\times l}$ 中一个标量，因此 N_{ijk}^+ 和 N_{ijk}^- 都是标量。

2.3 基于角色的访问控制和授权约束

基于角色的访问控制(Role-Based Access Control，RBAC)是一种建模策略，已成为一种被广泛接受的替代传统任意和强制访问控制的方法[8]。RBAC 的一个基本方面是授权约束(authorization constraints)，安全需求表示为用户和角色的授权约束[3]。基本的授权约束包括基数约束(Constraints of Cardinality，CoCs)[3]和职责分离(Seperation of Duties，SoDs)[9]。

2.4 结构化安全

EBPN 是一个描述电子商务业务流程的形式化模型[2,12]，该模型能够很好地对存在第三方交易的平台进行业务流程建模，而结构化安全源自于其业务结构的设计[2,13-15]。

该文以改进 Petri 网为基础的形式化模型 EBPN 为主体，通过添加关键库所和变迁，提出电子商务业务流程关键网(Critical E-commerce Business Processes Net，CEBPN)。基于此模型，定义了新的变迁约束(Constraints of Transitions，CoTs)和行为分离(Separate of Actions，SoAs)，通过整数线性规划(Integer Linear Programming)提出基数约束安全验证法和行为分离安全验证法来验证结构化安全问题的存在。

定义 1 电子商务业务流程关键网(CEBPN)定义为：

$$\text{CEBPN}=(\text{EN},C) \tag{5}$$

其中，EN 表示一个基本的 EBPN 网，C 表示网中定义的需要验证结构化安全的关键库所和变迁。

3 模型建立

利用 EBPN 网的建模方法对电子商务系统进行建模。依次给出系统中的控制结构和数据结构。最后将两个结构整合获得数据结构和控制结构完整的模型。具体 EBPN 网的构建方法请参考文献[2]。

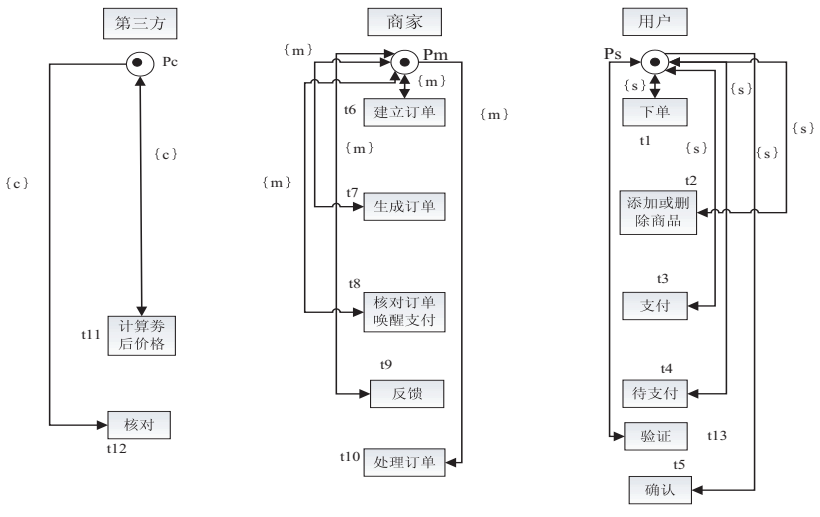

图 1 实例的控制结构

图 1 根据动机实例问题中订餐交易系统流程给出 EBPN 控制结构,分别构建第三方交易平台,商家以及用户三个控制结构,其中每一个控制结构都拥有属于自己的 API,不同的 API 由变迁 t 表示。商家拥有的操作界面包括建立订单,生成订单等,用户拥有的操作界面包括下单点餐,添加或删除商品,支付等,第三方交易平台拥有计算券后价格以及核对的 API。每一个控制结构中的变迁 t 都由一个控制库所连接,例如用户的 API 均由库所 p_s 连接,代表每一个控制结构都随时准备好参与变迁的发生,即每一个控制结构都保持正常运行状态,随时准备参与到事务中。p_c , p_m , p_s 中的 token 种类分别定义为 c,m 和 s ,这三个库所中的 token 是恒存的,表示每一个控制结构可以随时处理属于该结构的事务。因此,$\{s,m,c\}$ 是一组控制参数,这意味着三个参与者总是准备好开始新的事务。

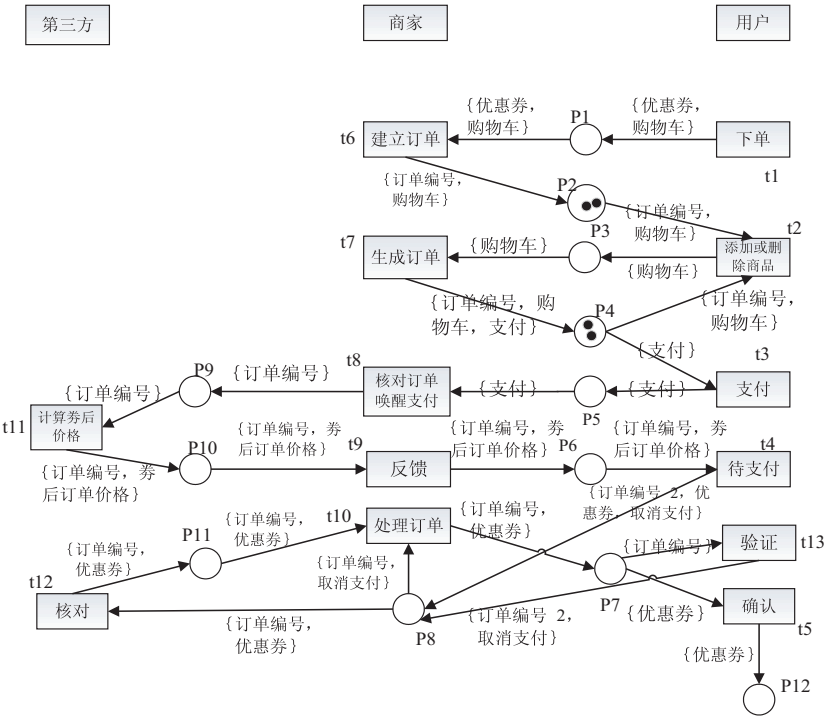

图 2　实例的数据结构

图 2 是实例问题的数据结构,图中定义了一组交易的参数{订单编号,购物车,优惠券,支付,券后订单价格,取消支付}作为 EBPN 网中不同 token 种类。流程开始时,用户在订单界面发送请求下单,用户端将购物车的信息发送给商家,并且用户是以优惠券下单,因此将优惠券作为 token 发送给商家。注意,这里的优惠券仅代表以优惠券下单这个信号,并非已经消费了优惠券。在生成订单后,支付的 API 是在用户界面发生,通过库所 p_4 发送订单标号和购物车两个 token,用以添加删除商品。用户在经过商家和第三方的费用核对后进入待支付状态,待支付状态属于用户和第三方联系的状态,用户可以选择取消支付,即将取消支付的 token 通过库所 p_8 发送。

将图 1 和图 2 整合得到完整的 EBPN 模型(见图 3)。不同种类的 token 在 EBPN 网标记 M 中的顺序为 $\{s,m,c$ 订单编号,购物车,优惠券,支付,券后订单价格,取消支付$\}$,将 token 种类简化得到标记中 token 的顺序为 $\{s,m,c,d,y,g,z,h,q\}$ 。注意,每个模型图中每个弧上消耗的 token 都以其简写字母表示,弧上的权值已在图中标明。标记中库所集 P 的顺序为 $\{p_s$, $p_m,p_c,p_1,p_2,p_3,p_4,p_5,p_6,p_7,p_8,p_9,p_{10},p_{11},p_{12}\}$。

EBPN 模型初始的简写标记为:

$$M_0 = [s(1),m(1),c(1),\mathbf{0},d(1)g(1),\mathbf{0},d(1)g(1),\mathbf{0},\mathbf{0},\mathbf{0},\mathbf{0},\mathbf{0},\mathbf{0},\mathbf{0},\mathbf{0}]$$

其中,粗体数字 **0** 表示 9 维的零向量。将各个库所中不同种类的 token 数量简化,已有的 token 写出它的字母代号和数量,其余的 token 则默认数量为零,例如 $s(1)$ 表示 p_s 库所中只拥有 1 个 s 型 token,其余种类的 token 数量为 0。因此给出该模型系统简略的初始标记为:

$$M_0 = [p_s(s),p_m(m),p_c(c),p_2(d,g),p_4(d,g)]$$

下文关联矩阵同样使用该表示方法。文章结尾给出该 EBPN 模型的输出矩阵 $[N^+]$,输入矩阵 $[N^-]$ 和关联矩阵 $[N]$ 。

注意,图 3 中给出的完整的 CEBPN 模型是有界网,因此其可达图是可以穷举的,在下文中使用状态分析法来验证文中方法的有效性,也证明了模型的可达性是可以穷举的。

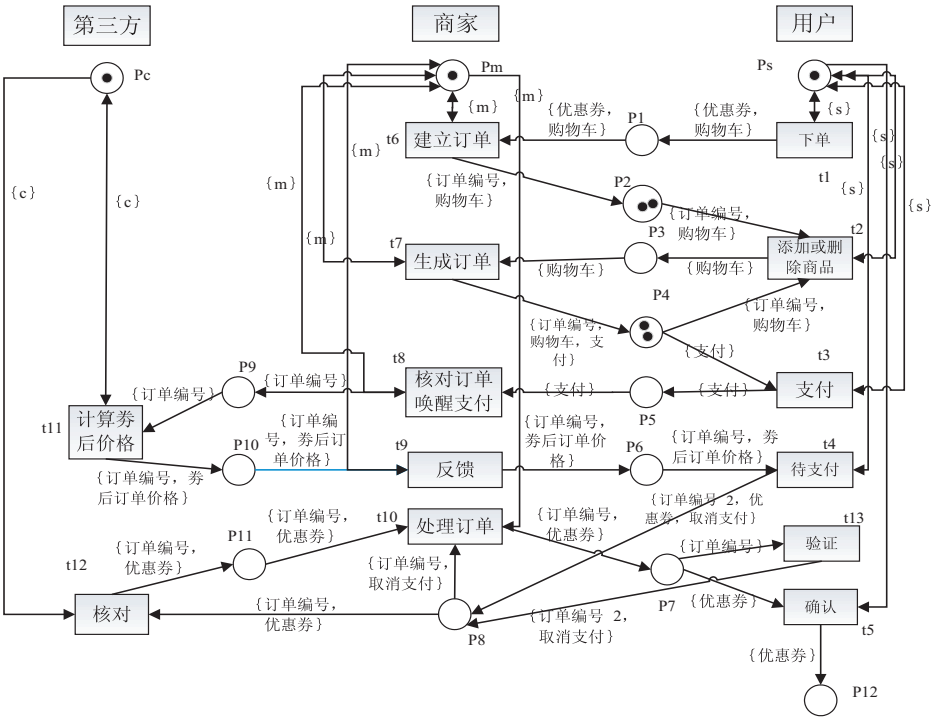

图 3 完整的实例 CEBPN 模型

4 基于 CEBPN 网的约束定义和约束证明

4.1 约束定义

关于 EBPN 模型分析，文献[2]给出两种方法：行为序列法(Behavioral Sequence Method)和状态分析法(State Analyzing Method)。两种方法都可以验证 EBPN 网中的结构化安全问题。然而，行为序列法对于比较复杂的网系统，在实际操作上是十分复杂的。对于状态分析法，该方法只考虑了状态方程的求解，而发生序列所满足的状态方程仅仅只是初始状态 M_0 到不安全状态 M 的必要条件，缺少发生的充分条件[16]。

推论 1 给出一个标记 M 可由初始标记 M_0 经过发射有限长度的变迁序列 σ 而得到的充分必要条件。

推论 1 存在 λ 个整数向量 $\vec{\sigma_1}, \vec{\sigma_2}, \cdots, \vec{\sigma_\lambda}$($\lambda \in \mathbb{Z}^+$)，$\lambda \leqslant |\sigma|$，使得以下线性不等式满足条件：

$$\begin{aligned} & M_0 \geqslant [N^-]^{\mathrm{T}} \cdot \vec{\sigma}_1 \\ & M_0 + [N^-]^{\mathrm{T}} \cdot \vec{\sigma}_1 \geqslant [N^-]^{\mathrm{T}} \cdot \vec{\sigma}_2 \\ & \cdots \\ & M_0 + [N]^{\mathrm{T}} \cdot \sum_{i=1}^{\lambda-1} \vec{\sigma}_i \geqslant [N^-]^{\mathrm{T}} \cdot \vec{\sigma}_\lambda \\ & \sum_{i=1}^{\lambda} \vec{\sigma}_i \geqslant \vec{\sigma} \end{aligned} \tag{6}$$

$[N]$ 是属于 EBPN 的三维关联矩阵，$[N^-]$ 是属于 EBPN 网的输出矩阵。这里

$$M_0 + [N]^{\mathrm{T}} \cdot \sum_{i=1}^{\lambda} \vec{\sigma}_i = M \tag{7}$$

为普通 Petri 网状态方程的变式，其中初始标记 M_0 通过一系列变迁的发生到达终止标记 M[17]。推论 1 给出的充分条件适合于有界 Petri 网，对于无界 Petri 网难以适用[3]。对于有界 Petri 网，其可达标识图的范围是有限的，因此 λ 的大小可以定义为 $L_{\min} = 12$，即变迁数，对于最大长度问题，具体可以参考文献[18-19]。

定义 2 和定义 3 提出两种基于 EBPN 网模型的约束，并利用线性规划来求解验证 EBPN 网的结构安全性。接下来给出基于 EBPN 网的行为分离和变迁约束的定义[20-21]。

定义 2 变迁约束(Constraints of Transition, CoTs)：基于 EBPN 模型的变迁约束 CoTs(t,Δ) 表示在可行变迁序列中 t 的发生次数小于 Δ 个，其中 $t \in T$，$\Delta \in \mathbb{Z}^+$，$\mathbb{Z}^+$ 表示正整数集合。

定义 3 行为分离(Separation of Actions, SoAs)：基于 EBPN 模型的行为分离 SoAs($d_x \in p_i, d_y \in p_j$) 表示对于 EBPN 中不同类型的 tokend_x, d_y($d_x, d_y \in D$)，在所有的 EBPN 的可达标识中不能同时存在 $d_x \in p_i$，$d_y \in p_j$($p_i \in P, p_j \in P$)，其中 D 表示 token 类型集合，P 表示库所集合。

4.2 定理和证明

通过线性规划来构建目标函数进行求解，验证违法变迁序列是否存在。

下面给出变迁约束和行为分离两个约束的验证方法。

定理 1 给定线性规划目标函数

$$\min \sum_{i=1}^{\lambda} \vec{\sigma}(t)$$

给出线性约束:

$$\begin{cases} M_0 \geqslant [N^-]^{\mathrm{T}} \cdot \vec{\sigma}_1 \\ M_0 + [N]^{\mathrm{T}} \cdot \vec{\sigma}_1 \geqslant [N^-]^{\mathrm{T}} \cdot \vec{\sigma}_2 \\ \cdots \\ \cdots \\ M_0 + [N]^{\mathrm{T}} \cdot \sum_{i=1}^{\lambda-1} \vec{\sigma}_i \geqslant [N^-]^{\mathrm{T}} \cdot \vec{\sigma}_\lambda \\ M_0 + [N]^{\mathrm{T}} \cdot \sum_{i=1}^{\lambda} \vec{\sigma}_i \geqslant \vec{0} \end{cases} \tag{8}$$

$$\sum_{i=1}^{\lambda} \vec{\sigma}_i(t) \geqslant \Delta \tag{9}$$

$$\vec{\sigma}_i \in N^{|\mathrm{T}|}, i = 1,2,\cdots,\lambda \tag{10}$$

如果该线性规划能够得出一个解 $\overrightarrow{\sigma_1}$, $\overrightarrow{\sigma_2}$, …, $\overrightarrow{\sigma_\lambda} \in N^{|\mathrm{T}|}(\lambda \in \mathbb{Z}^+)$,则变迁约束 CoTs($t$,Δ) 违反。

证明:(必要性)如果线性规划不等式 8 ~ 10 能够求得一个解,即存在一组变迁发生序列使得模型由初始标记发生至终止标记。由不等式 9 可得在完整变迁序列中变迁 t 的数目不小于违反变迁约束的数目 Δ,而 CoTs(t,Δ) 定义则是在所有的发生序列中,t 的发生次数要小于 Δ,因此在这种条件下求解出来的变迁序列一定是违反变迁约束的。

(充分性)运用反证法,假如模型满足变迁约束 CoTs(t,Δ) 且线性规划能够求得一个解,则说明这个解满足不等式 9,所以存在一组变迁序列,使得变迁 $t \geqslant \Delta$,这与变迁约束 CoTs(t,Δ) 相违背,因此这是与假设相反的。

定理 2　给定线性规划目标函数

$$\min \sum_{i=1}^{\lambda} \vec{\sigma}(t)$$

给出线性约束:

$$\begin{cases} M_0 \geqslant [N^-]^{\mathrm{T}} \cdot \vec{\sigma}_1 \\ M_0 + [N]^{\mathrm{T}} \cdot \vec{\sigma}_1 \geqslant [N^-]^{\mathrm{T}} \cdot \vec{\sigma}_2 \\ \cdots \\ \cdots \\ M_0 + [N]^{\mathrm{T}} \cdot \sum_{i=1}^{\lambda-1} \vec{\sigma}_i \geqslant [N^-]^{\mathrm{T}} \cdot \vec{\sigma}_\lambda \\ M_0 + [N]^{\mathrm{T}} \cdot \sum_{i=1}^{\lambda} \vec{\sigma}_i \geqslant \vec{0} \end{cases} \tag{11}$$

$$\sum_{i=1}^{\lambda} \vec{\sigma}_i(t_{e\_p_i}) \geqslant 1 \tag{12}$$

$$\sum_{i=1}^{\lambda} \vec{\sigma}_i(t_{e\_p_j}) \geqslant 1$$

$$[M_0 + [N]^{\mathrm{T}} \cdot \sum_{i=1}^{\lambda} \vec{\sigma}_i]_{ix} \geqslant 1 \tag{13}$$

$$[M_0 + [N]^{\mathrm{T}} \cdot \sum_{i=1}^{\lambda} \vec{\sigma}_i]_{jy} \geqslant 1 \tag{14}$$

如果该线性规划能够得出一个解 $\overrightarrow{\sigma_1}$, $\overrightarrow{\sigma_2}$, …, $\overrightarrow{\sigma_\lambda} \in N^{|\mathrm{T}|}(\lambda \in \mathbb{Z}^+)$,则行为分离约束 SoAs($d_x \in p_i$, $d_y \in p_j$) 违反。

其中,$t_{e\_p_i}$ 表示 p_i 库所的输入变迁,$t_{e\_p_j}$ 表示 p_j 库所的输入变迁,$[M_0 + [N]^{\mathrm{T}} \cdot \sum_{i=1}^{\lambda} \vec{\sigma}_i]_{ix}$ 表示终止标记中 p_i 库所中 d_x 种类 token 的数量,$[M_0 + [N]^{\mathrm{T}} \cdot \sum_{i=1}^{\lambda} \vec{\sigma}_i]_{jy}$ 同理[22-24]。

证明:(必要性)如果线性规划 11 ~ 14 能够求得一个解,即存在一组变迁发生序列使得模型由初始标记发生至终止标记。由不等式 11 ~ 14 可得在 p_i 库所和 p_j 库所都会得到 token 的前提下,在所有的 EBPN 网的可达标识中可能会同时存在 $d_x \in p_i, d_y \in p_j (p_i \in P, p_j \in P)$,其中 D 表示所有 token 类型的集合,P 表示所有库所的集合。而 SoAs($d_x \in p_i, d_y \in p_j$) 定义则是在所有的 EBPN 网的可达标识中不会同时存在 $d_x \in p_i, d_y \in p_j (p_i \in P, p_j \in P)$,因此在这种条件下求解出来的变迁序列一定是违反变迁约束的。

(充分性)运用反证法,假如模型满足变迁约束 SoAs($d_x \in p_i, d_y \in p_j$) 且线性规划能够求得一个解,则说明这个解满足不等式 11 ~ 14,所以可能会同时存在 $d_x \in p_i, d_y \in p_j (p_i \in P, p_j \in P)$,这与变迁约束 SoAs($d_x \in p_i, d_y \in p_j$) 相违背,因此这是与假设相反的。

对于以上提出的两种线性规划理论,单一不等式的空间复杂度可以近似看成 $n \times m \times l$,其中 n 表示 EBPN 模型中变迁的数量,m 表示 EBPN 模型中库所的数量,l 表示库所中不同类型 token 的种类数。整个线性规划的空间复杂度可以近似看为 $n \times m \times l \times \lambda$,$\lambda$ 可近似表示线性规划中不等式的数量。

5　模型实例分析

图 3 定义的实例中 CEBPN 模型图描述了第二章的结构化安全问题。首先,通过变迁约束(CoTs)和行为分离(SoAs)来定义实例模型的安全约束。针对图 3 所示的 CEBPN 模型,给出需要验证结构化安全的关键库所和变迁,如表 1 所示。

表 1　关键库所和变迁

关键库所 P	关键变迁 T
p_3	t_2
p_5	t_{10}
p_7	t_{12}
p_8	

首先给出模型完整的由初始标记到达终止标记的

状态方程：

$$M = \begin{bmatrix} (1,0,0,0,0,0,0,0,0) \\ (0,1,0,0,0,0,0,0,0) \\ (0,0,1,0,0,0,0,0,0) \\ \mathbf{0} \\ (0,0,0,1,0,1,0,0,0) \\ \mathbf{0} \\ (0,0,0,1,0,1,0,0,0) \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix} + [N]^T \cdot \sum_{i=1}^{\lambda} \vec{\sigma}_i$$

推论 1 给出了标记 M 可由 M_0 经过变迁序列 σ 的发生而获得的充分必要条件，因此满足推论 1 的任何变迁序列必然满足模型的状态方程。

5.1 实例 1

定义两个约束 SoAs(购物车 $\in p_3$，支付 $\in p_5$)，前者表示在该 EBPN 网中，由初始标记 M_0 发射到达的所有可达标记中不能同时存在 token 购物车 $\in p_3$，token 支付 $\in p_5$，后者表示由初始标记 M_0 发射到达的所有可达标记中，任意一个变迁发生序列不能存在组合变迁 t_2 发生两次以上。当模型的可达标记中同时存在 token 购物车 $\in p_3$，token 支付 $\in p_5$，意味着用户已经进入支付状态，在用户界面却仍可以更新购物车中内容，即添加或删除商品，这是一个违法行为。如果变迁序列中 t_2 发生两次以上，表示用户可以再次进入商家这个控制结构更新购物车里内容，因此属于违法序列。通过确认这两个约束是否违反来验证第 2 章中的问题(2)。注意，对于这个违法行为的定义是必须要求这两个约束都违反，即对于这两个约束的线性规划求解都可以得到一个解，才可以判定该模型系统存在该结构化安全问题。

5.2 实例 2

定义两个约束 SoAs(优惠券 $\in p_7$，取消支付 $\in p_8$)，CoTs($t_{10}t_{12}$,2)。SoAs(优惠券 $\in p_7$，取消支付 $\in p_8$)，CoTs($t_{10}t_{12}$,2) 表示用户已经经过商家的取消订单处理进入取消订单状态，此时用户与第三方支付平台仍有 API 进行操作退款，第三方将信息反馈给商家，商家将优惠券退还给用户，因此属于违法序列。通过确认这两个约束是否违反来验证第二章中的问题(1)。同上，对于这个违法行为的定义是必须要求这两个约束都违反，即对于这两个约束的线性规划求解都可以得到一个解，才可以判定该模型系统存在该结构化安全问题。

5.3 定理验证

根据定理 1 和定理 2 来分别求解上述实例安全约束。对于已提出的定理 1 和定理 2，可以使用现有的软件工具进行求解，比如 Lingo 和 Python。

根据定理 1 线性规划验证 CoTs(t_2,2) 约束：

给定线性规划目标函数

$$\min \sum_{i=1}^{\lambda} \vec{\sigma}_i(t_2)$$

给出线性约束：

$$\begin{cases} M_0 \geqslant [N^-]^{\mathrm{T}} \cdot \vec{\sigma}_1 \\ M_0 + [N]^{\mathrm{T}} \cdot \vec{\sigma}_1 \geqslant [N^-]^{\mathrm{T}} \cdot \vec{\sigma}_2 \\ \cdots \\ \cdots \\ M_0 + [N]^{\mathrm{T}} \cdot \sum_{i=1}^{\lambda-1} \vec{\sigma}_i \geqslant [N^-]^{\mathrm{T}} \cdot \vec{\sigma}_\lambda \\ M_0 + [N]^{\mathrm{T}} \cdot \sum_{i=1}^{\lambda} \vec{\sigma}_i \geqslant \vec{0} \\ \sum_{i=1}^{\lambda} \vec{\sigma}_i(t_2) \geqslant 2 \\ \vec{\sigma}_i \in N^{|\mathrm{T}|}, i = 1,2,\cdots,\lambda \end{cases}$$

对 λ 取值为 12 并对这个线性规划进行求解可以得到一个解，说明约束 CoTs(t_2,2) 违反。

根据定理 2 线性规划验证 SoAs(购物车 $\in p_3$，支付 $\in p_5$) 约束：

给定线性规划目标函数

$$\min \sum_{i=1}^{\lambda} \vec{\sigma}_i(t)$$

给出线性约束：

$$\begin{cases} M_0 \geqslant [N^-]^{\mathrm{T}} \cdot \vec{\sigma}_1 \\ M_0 + [N]^{\mathrm{T}} \cdot \vec{\sigma}_1 \geqslant [N^-]^{\mathrm{T}} \cdot \vec{\sigma}_2 \\ \cdots \\ \cdots \\ M_0 + [N]^{\mathrm{T}} \cdot \sum_{i=1}^{\lambda-1} \vec{\sigma}_i \geqslant [N^-]^{\mathrm{T}} \cdot \vec{\sigma}_\lambda \\ M_0 + [N]^{\mathrm{T}} \cdot \sum_{i=1}^{\lambda} \vec{\sigma}_i \geqslant \vec{0} \\ \sum_{i=1}^{\lambda} \vec{\sigma}_i(t_{e\_p_3}) \geqslant 1 \\ \sum_{i=1}^{\lambda} \vec{\sigma}_i(t_{e\_p_5}) \geqslant 1 \\ [M_0 + [N]^{\mathrm{T}} \cdot \sum_{i=1}^{\lambda} \vec{\sigma}_i]_{3g} \geqslant 1 \\ [M_0 + [N]^{\mathrm{T}} \cdot \sum_{i=1}^{\lambda} \vec{\sigma}_i]_{5z} \geqslant 1 \end{cases}$$

对 λ 取值为 12 并对这个线性规划进行求解可以得到一个解,说明约束 SoAs(购物车 $\in p_3$,支付 $\in p_5$)违反。

根据定理 1 线性规划验证 CoTs($t_{10}t_{12}$,2) 约束:

给定线性规划目标函数

$$\min \sum_{i=1}^{\lambda} \vec{\sigma}_i(t_{10}t_{12})$$

给出线性约束:

$$\begin{cases} M_0 \geqslant [N^-]^{\mathrm{T}} \cdot \vec{\sigma}_1 \\ M_0 + [N]^{\mathrm{T}} \cdot \vec{\sigma}_1 \geqslant [N^-]^{\mathrm{T}} \cdot \vec{\sigma}_2 \\ \cdots \\ \cdots \\ M_0 + [N]^{\mathrm{T}} \cdot \sum_{i=1}^{\lambda-1} \vec{\sigma}_i \geqslant [N^-]^{\mathrm{T}} \cdot \vec{\sigma}_\lambda \\ M_0 + [N]^{\mathrm{T}} \cdot \sum_{i=1}^{\lambda} \vec{\sigma}_i \geqslant \vec{0} \\ \sum_{i=1}^{\lambda} \vec{\sigma}_i(t_{10}t_{12}) \geqslant 2 \\ \vec{\sigma}_i \in N^{|T|}, i = 1,2,\cdots,\lambda \end{cases}$$

对 λ 取值为 12 并对这个线性规划进行求解可以得到一个解,说明约束 CoTs($t_{10}t_{12}$,2) 违反。

根据定理 2 线性规划验证 SoAs(优惠券 $\in p_7$,取消支付 $\in p_8$) 约束:

给定线性规划目标函数

$$\min \sum_{i=1}^{\lambda} \vec{\sigma}_i(t)$$

给出线性约束:

$$\begin{cases} M_0 \geqslant [N^-]^{\mathrm{T}} \cdot \vec{\sigma}_1 \\ M_0 + [N]^{\mathrm{T}} \cdot \vec{\sigma}_1 \geqslant [N^-]^{\mathrm{T}} \cdot \vec{\sigma}_2 \\ \cdots \\ \cdots \\ M_0 + [N]^{\mathrm{T}} \cdot \sum_{i=1}^{\lambda-1} \vec{\sigma}_i \geqslant [N^-]^{\mathrm{T}} \cdot \vec{\sigma}_\lambda \\ M_0 + [N]^{\mathrm{T}} \cdot \sum_{i=1}^{\lambda} \vec{\sigma}_i \geqslant \vec{0} \\ \sum_{i=1}^{\lambda} \vec{\sigma}_i(t_{e\_p_7}) \geqslant 1 \\ \sum_{i=1}^{\lambda} \vec{\sigma}_i(t_{e\_p_8}) \geqslant 1 \\ [M_0 + [N]^{\mathrm{T}} \cdot \sum_{i=1}^{\lambda} \vec{\sigma}_i]_{7y} \geqslant 1 \\ [M_0 + [N]^{\mathrm{T}} \cdot \sum_{i=1}^{\lambda} \vec{\sigma}_i]_{8q} \geqslant 1 \end{cases}$$

对 λ 取值为 12 并对这个线性规划进行求解可以得到一个解,说明约束 SoAs(优惠券 $\in p_7$,取消支付 $\in p_8$) 违反。

通过上述线性规划的求解可以看出,关键库所和变迁中是存在结构化安全问题的。利用文献[2]中提出的状态分析法验证关键库所和变迁是否同求解结果一致。

对于实例 1,给出确定的违法标识

$M_1 = [s(1),m(1),c(1),\mathbf{0},\mathbf{0},g(1),\mathbf{0},z(1),\mathbf{0},\mathbf{0},\mathbf{0},\mathbf{0},\mathbf{0},\mathbf{0}]$

因此状态方程可列为:

$$M_1 = \begin{bmatrix} (n,0,0,0,0,0,0,0,0) \\ (0,n,0,0,0,0,0,0,0) \\ (0,0,n,0,0,0,0,0,0) \\ \mathbf{0} \\ \mathbf{0} \\ (0,0,0,0,0,1,0,0,0) \\ \mathbf{0} \\ (0,0,0,0,0,0,1,0,0) \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix} =$$

$$\begin{bmatrix} (n,0,0,0,0,0,0,0,0) \\ (0,n,0,0,0,0,0,0,0) \\ (0,0,n,0,0,0,0,0,0) \\ \mathbf{0} \\ (0,0,0,1,0,1,0,0,0) \\ \mathbf{0} \\ (0,0,0,1,0,1,0,0,0) \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix} + [N]^T \cdot \begin{bmatrix} t_1 \\ t_2 \\ t_3 \\ t_4 \\ t_5 \\ t_6 \\ t_7 \\ t_8 \\ t_9 \\ t_{10} \\ t_{11} \\ t_{12} \\ t_{13} \end{bmatrix}$$

对该方程求解可得唯一解:

$\boldsymbol{T} = [t_1,t_2,t_3,t_4,t_5,t_6,t_7,t_8,t_9,t_{10},t_{11},t_{12},t_{13}]^{\mathrm{T}} = [1,2,1,0,0,1,1,0,0,0,0,0,0]^{\mathrm{T}}$

因此,可以得到变迁发生序列 $\sigma = t_1t_6t_2t_7t_2t_3$,使得 $M_0 \xrightarrow{\sigma} M_1$。

对于实例 2,给出确定的违反规则的违法标识

$M_2 = [s(1),m(1),c(1),\mathbf{0},\mathbf{0},\mathbf{0},d(1)g(1),\mathbf{0},\mathbf{0},y(1),d(2)y(1)q(1),\mathbf{0},\mathbf{0},\mathbf{0},y(1)]$

因此,状态方程可列为:

$$M_2 = \begin{bmatrix} (n,0,0,0,0,0,0,0,0) \\ (0,n,0,0,0,0,0,0,0) \\ (0,0,n,0,0,0,0,0,0) \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \\ (0,0,0,1,0,1,0,0,0) \\ \mathbf{0} \\ \mathbf{0} \\ (0,0,0,0,1,0,0,0,0) \\ (0,0,0,2,1,0,0,0,1) \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \\ (0,0,0,0,1,0,0,0,0) \end{bmatrix} = \begin{bmatrix} (n,0,0,0,0,0,0,0,0) \\ (0,n,0,0,0,0,0,0,0) \\ (0,0,n,0,0,0,0,0,0) \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \\ (0,0,0,1,0,1,0,0,0) \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix} + [N]^T \cdot \begin{bmatrix} t_1 \\ t_2 \\ t_3 \\ t_4 \\ t_5 \\ t_6 \\ t_7 \\ t_8 \\ t_9 \\ t_{10} \\ t_{11} \\ t_{12} \\ t_{13} \end{bmatrix}$$

对该方程求解可得唯一解

$\boldsymbol{T} = [t_1,t_2,t_3,t_4,t_5,t_6,t_7,t_8,t_9,t_{10},t_{11},t_{12},t_{13}]^{\mathrm{T}} = [1,1,1,1,1,1,1,1,1,2,1,2,2]^{\mathrm{T}}$

因此,可以得到变迁发生序列 $\sigma = t_1t_6t_2t_7t_3t_8t_{11}t_9t_4t_{12}t_{10}t_5t_{13}t_{12}t_{10}t_{13}$ 使得 $M_0 \xrightarrow{\sigma} M_1$。

通过状态方程的求解方法,验证了结果的存在性。并且所得的变迁发生序列与线性规划求解的并无二致,足以证明提出的方法的可行性。注意,初始标识中的 n 表示无限量,和经过变迁序列的发生可以到达的标识中的 n 属同一种类型,都表示无限量,表明三个不同域分别联系着该域中的 API。

验证线性规划解的唯一性使用了 Petri 网中的状态分析法,因此其时间复杂度可以理解为状态方程的复杂度,即 $n \times m \times l$。对于模型存在的结构化安全问题,可以从设计基础上对原模型进行修改以解决模型中的结构化安全问题。图 4 给出正确的模型案例,模型中的结构化安全问题得以解决。

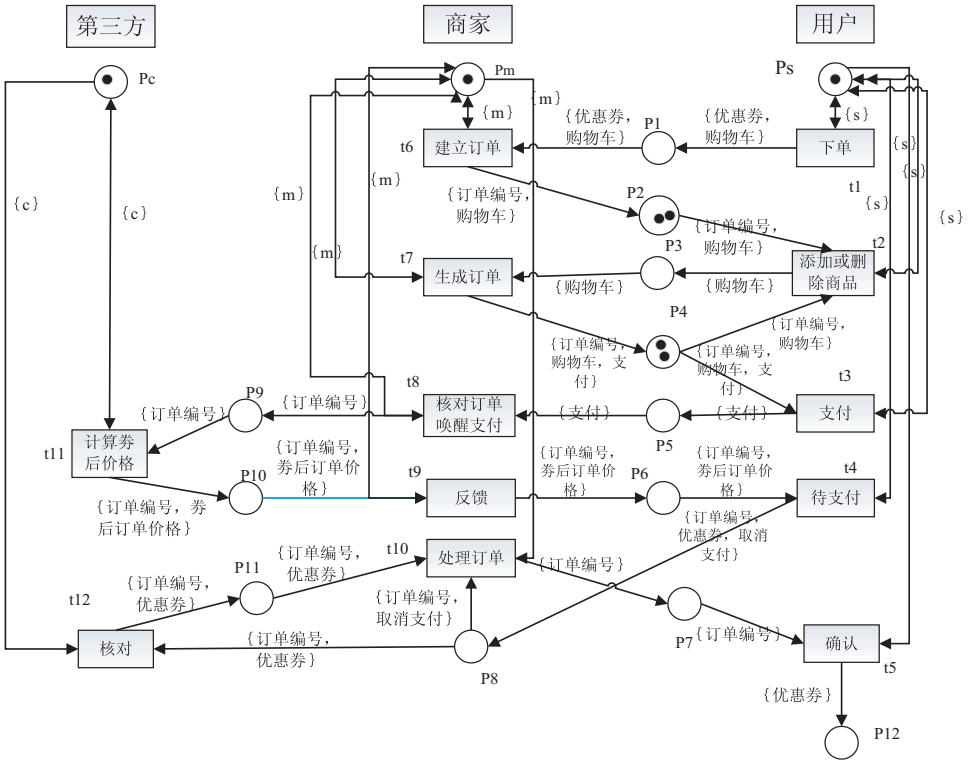

图 4 修正的实例 CEBPN 模型

6　结束语

对于解决电子商务业务流程中的结构化安全问题,提出新的解决方法。在基于 EBPN 网的基础上加入了关键库所和关键变迁元素构建了 CEBPN,并融合了基于角色的访问控制(RBAC)中的访问授权约束职责分离(SoDs)和基数约束(CoCs),定义了新的基于 EBPN 网的授权约束变迁约束(CoTs)和行为分离(SoAs),并利用整数线性规划(ILP)和 Petri 网的状态方程的性质对目标约束进行描述求解以证明模型的结构化安全性。在面对使用 EBPN 建模的电子商务业务流程模型时,运用行为分离约束和变迁约束可以检验每一个变迁或库所可能造成的结构化安全问题和非法数据状态,并且可以借助已有的编程软件直接得出结果。该方法对比已经提出的行为序列法[2]和状态分析法[2]来说更全面严谨且容易理解和求解,特别是对于不仅仅只存在少数变迁和库所的复杂的 EBPN 网模型。

参考文献:

[1] WANG L, CHAI Y, LIU Y. Analysis of e-commerce transaction system´s division of labor based on essential services quantity[J]. International Journal of Crowd Science, 2017, 1(3): 197-209.

[2] YU W, DING Z, LIU L, et al. Petri net-based methods for analyzing structural security in e-commerce business processes[J]. Future Generation Computer Systems, 2020, 109: 611-620.

[3] YANG B, HU H. Analysis of authorization constraints via integer linear programming[J]. IEEE Transactions on Knowledge and Data Engineering, 2021, 35(3): 2258-2271.

[4] JIE X, ZHANG D, LU L, et al. Dynamic authentication for cross-realm SOA-based business processes[J]. IEEE Transactions on Services Computing, 2012, 5(1): 20-32.

[5] CHEN C, HU H. Static and dynamic partitions of inequalities: a unified methodology for supervisor simplification[J]. IEEE Transactions on Automatic Control, 2019, 64(11): 4748-4755.

[6] WEI L, XIN F, ZHANG F X, et al. Analytic of B2C e-commerce credit mechanism mixed strategy risk behavior based on logical game petri nets[J]. IEEE Access, 2018, 6: 29109-29131.

[7] LIU W, WANG P, DU Y, et al. Extended logical petri nets-based modeling and analysis of business processes[J]. IEEE Access, 2017, 5: 16829-16839.

[8] AHN G J, SANDHU R. Role-based authorization constraints specification[J]. ACM Transactions on Information and System Security, 2000, 3(4): 207-226.

[9] HARIKA P, NAGAJYOTHI M, JOHN J C, et al. Meeting cardinality constraints in role mining[J]. IEEE Transactions on Dependable & Secure Computing, 2015, 12(1): 71-84.

[10] 吴哲辉. Petri 网导论[M]. 北京:机械工业出版社,2006.

[11] 李志武,周孟初. 自动制造系统建模、分析与死锁控制[M]. 北京:科学出版社,2009.

[12] YU W Y, YAN C G, DING Z J, et al. Modeling and validating e-commerce business process based on Petri nets[J]. IEEE Transactions on Systems, Man, and Cybernetics: Systems, 2013, 44(3): 327-341.

[13] GE J, TIAN Y, LIU L, et al. Understanding e-commerce systems under massive flash crowd: measurement, analysis, and implications[J]. IEEE Transactions on Services Computing, 2017, 13(6): 1180-1193.

[14] CEBECI S E, NARI K, OZDEMIR E. Secure e-commerce scheme[J]. IEEE Access, 2022, 10: 10359-10370.

[15] CABANILLAS C, RESINAS M, RUIZ-CORTÉS A. A mashup-based framework for business process compliance checking[J]. IEEE Transactions on Services Computing, 2020, 15(3): 1564-1577.

[16] LI Q, LIU W, GUAN M Z. Modeling and analysis of emergency decision making based on logical probability game petri net[J]. Computer Science, 2022, 49(4): 294-301.

[17] YU W Y, YAN C G, DING Z J, et al. Modeling and verification of online shopping business processes by considering malicious behavior patterns[J]. IEEE Transactions on Automation Science & Engineering, 2016, 13(2): 647-662.

[18] YANG B, HU H. Dynamic implementation of security requirements in business processes[J]. IEEE Transactions on Dependable and Secure Computing, 2020, 19(2): 1352-1363.

[19] BASILE F, CHIACCHIO P, DE TOMMASI G. On K-diagnosability of Petri nets via integer linear programming[J]. Automatica, 2012, 48(9): 2047-2058.

[20] WANG M, DING Z, ZHAO P. Vulnerability evaluation method for E-commerce transaction systems with unobservable transitions[J]. IEEE Access, 2020, 8: 101035-101048.

[21] SOURI A, RAHMANI A M, NAVIMIPOUR N J, et al. A symbolic model checking approach in formal verification of distributed systems[J]. Human-Centric Computing and Information Sciences, 2019, 9(1): 4.

[22] 刘　伟,史晓浩,孙红伟. 基于逻辑混合 Petri 网的混合系统建模与分析[J]. 山东科技大学学报:自然科学版,2021,40(4): 65-75.

[23] 陈金栋,刘　伟,冯　新,等. 基于逻辑工作流网的有限无死锁组合[J]. 山东科技大学学报:自然科学版,2020,39(5): 89-97.

[24] 程学珍,王常安,李继明,等. 基于自适应神经模糊 Petri 网的电机故障诊断[J]. 山东科技大学学报:自然科学版,2020,39(3): 109-117.