文章编号:1007－6735(2023)06－0636－09　DOI: 10.13255/j.cnki.jusst.20220425001

一种细粒度数据权限控制框架

邓　斌[1,2]，　丁家满[1,2]，　姜　瑛[1,2]，　贾连印[1,2]，　江　虹[1,2]
（1. 昆明理工大学 信息工程与自动化学院，昆明 650500；2. 云南省人工智能重点实验室，昆明 650500）

摘要： SaaS 作为一种热点服务模式，其服务框架的权限控制方面存在数据粒度粗及配置灵活度不足等问题。为此，在 RBAC 模型的理论基础之上，结合 SaaS 的特点提出了一种细粒度数据权限控制模型（fine-grained data permission control，FDPC）。该模型首先将数据对象与企业组织结构映射成不同粒度的数据权限；其次依据企业授权业务需求，采用将功能权限集合和数据权限集合中的对象两两组合的方式，形成组合权限对象，并将其分配给不同角色，从而达到灵活授权和细粒度数据控制的目的；最后选用 Spring Security 和 MyBatis 框架，依据 AOP 切片原理，采用结构化查询语句拼接方式，对 FDPC 模型进行实现，构建权限控制框架。通过在实际业务系统中应用，结果表明该框架合理可行，有效地提高了权限控制的灵活性。

关键词： 数据权限；访问控制；细粒度；RBAC 模型
中图分类号： TP 31　　**文献标志码：** A

A fine-grained data permission control framework

DENG Bin[1,2]，　DING Jiaman[1,2]，　JIANG Ying[1,2]，　JIA Lianyin[1,2]，　JIANG Hong[1,2]
(1. *Faculty of Information Engineering and Automation, Kunming University of Science and Technology, Kunming 650500, China*; 2. *Artificial Intelligence Key Laboratory of Yunnan Province, Kunming 650500, China*)

Abstract: As a hot service mode, SaaS has some problems such as coarse data granularity and insufficient configuration flexibility in the permission control of its service framework. Therefore, a fine-grained data permission control model (FDPC) was proposed based on the theory of the role based access control（RBAC）model and the characteristics of SaaS. Firstly, data objects and enterprise organizational structure were mapped by the model into data permissions of different granularities. Secondly, the method of combining the objects in the functional permission set and the data permission set in pairs to form a combined permission object was adopted according to the business requirements of enterprise authorization. Different roles were assigned to achieve the purpose of flexible authorization and fine-grained data control. Finally, Spring Security and MyBatis frameworks were selected. According to the AOP slicing principle, the FDPC model was implemented by structured query statement splicing and the authority control framework was constructed. Through the application in the actual business system, results show that the framework is reasonable and feasible, and the flexibility of

收稿日期: 2022−04−25
基金项目: 国家自然科学基金资助项目（62162038）
第一作者: 邓　斌（1997−），男，硕士研究生．研究方向：软件工程．E-mail：1270834910@qq.com
通信作者: 江　虹（1965−），男，讲师．研究方向：软件工程．E-mail：702104728@qq.com

authority control is effectively improved.

Keywords: *data permission control*; *access control*; *fine-grained*; *RBAC model*

权限控制是保证系统安全的重要手段，其中数据权限控制是系统安全的重要方面[1]。对用户权限加以严格的控制是软件系统安全性的重要保障。美国标准与技术研究院（NIST）提出的基于角色的访问控制 (role based access control, RBAC) 是一种很受欢迎的访问控制模型[2]。使用 RBAC 可以轻松实现最小权限并减轻权限管理的工作量，它是处理大型组织统一资源访问控制的一个很高效的方法[3]。RBAC 主要基于 3 个因素分配权限：用户、角色和资源。具体来说就是给用户分配一个系统角色[4]，不同的系统角色可以访问或操作不同的资源。通过建立角色体系，将用户和资源进行分离，来保证权限分配的实施。

权限主要分为两种：功能权限和数据权限[5]。目前，Spring Security 等框架已经较好地解决了功能权限的问题[6]，但对于数据权限而言，传统模型中使用分支判断等硬编码方式把权限逻辑加入到业务代码里的方式是使用最多的，但是硬编码形式的缺点也是显而易见的，表现为耦合度高、评估困难、系统组件重用率低、后续系统维护成本高，甚至导致整个系统整体结构变动[7]。

随着云平台技术的发展，SaaS 模式成为了一种热点服务模式，权限控制愈发复杂。文献 [8] 利用防串通访问控制多项式（ACP）和 Atallah 的高效密钥管理的双层密钥管理方案，为数据和用户级别提供访问控制手段。文献 [9] 利用有色 Petri 网（CPN）对模型进行约束建模，设置数据访问控制策略，并用于医疗信息系统。文献 [10] 引入信任链解决数据访问控制的问题。文献 [11] 通过从业务流程中提取业务级的控制要求来构建模型，同时建立模型与业务流程间的双向绑定，达到了数据权限控制的目的。文献 [12] 采用去中心化的标识符和可验证的凭证来描述实体身份，并利用区块链和智能合约进行访问控制。文献 [13] 在基于加密任务角色的访问控制中引入综合信任属性来保护云存储系统的数据访问。文献 [14] 通过数据与组织结构的映射，使拥有同一个角色的不同用户根据其所在部门获取对应数据。上述方法在不同领域和特定环境下都取得了良好的数据控制访问效果，但大都仅以功能权限和数据权限中的一项作为权限控制目标，或者将数据权限作为角色的一个整体，导致权限粒度粗及配置灵活度不足问题的出现。

针对上述问题，本文通过 RBAC 理论分析，提出了细粒度数据权限控制模型（fine-grained data permission control，FDPC），将功能和数据权限组合，使同一个角色的每个功能都有其对应的数据权限。最后使用 Spring Security 等框架，采用面向切面编程（aspect-oriented program，AOP）[15] 对代码进行解耦，实现了细粒度数据权限的灵活配置。

1　关键定义

权限（permissions）：系统提供的各种功能及其对应的数据。

功能权限（functional permissions）：用户有权在应用程序系统上执行的操作，通常成树形结构，并按类型可分为目录、菜单和按钮。

数据权限（data permissions）：用户在运行功能的时候可以查看的数据集合。

企业组织结构（enterprise organizational structure）：业务领域中，根据管理层次形成的结构，一般呈树状，因此，亦可称之为部门树。部门树如图 1 所示。

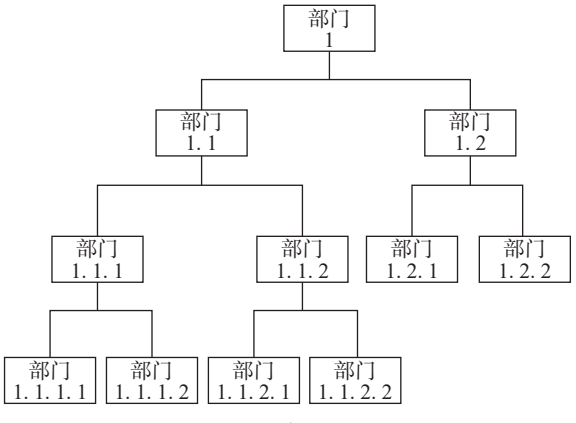


图 1　部门树

Fig.1　Departmental tree

用户与组织结构：系统中的用户必须归属于组织结构中的某一个部门。

数据创建者(data creator):数据中某条记录创建的用户。

数据归属(data attribution):业务中产生的所有数据记录必须归属于某个部门以及数据的创建者。

权限分配:将权限分配给角色的交互过程。

角色分配:将角色分配给用户的交互过程。

2 FDPC 模型

本文提出的细粒度数据权限控制模型首先根据组织结构定义数据权限,其次采用功能权限和数据权限交错组合,最后针对实际业务逻辑,对模型加以约束。细粒度数据权限控制模型如图 2 所示。图中:FP 为功能权限集合,包含 $fp_1, fp_2, \cdots, fp_m$;DP 为数据权限集合,包含 $dp_1, dp_2, \cdots, dp_n$;FDP 为功能关联数据的权限对象集合;$Role$ 为角色集合,包含 $role_1$,$role_2$,…,$role_n$;$User$ 为用户集合,包含 $user_1$,$user_2$,…,$user_n$。

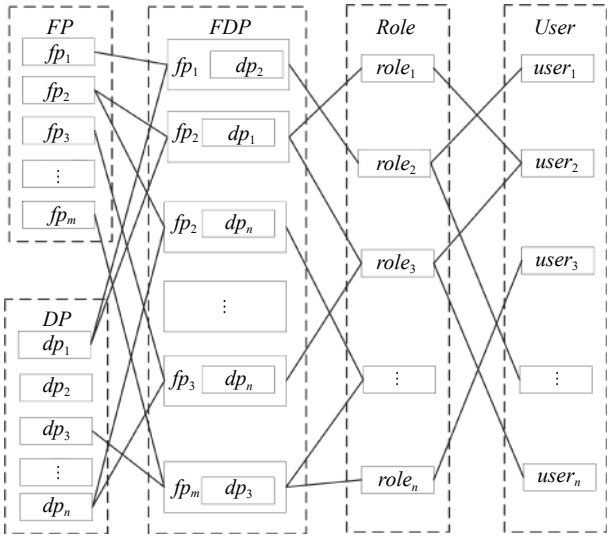


图 2 FDPC 模型示意图

Fig.2 Schematic diagram of FDPC model

2.1 数据权限定义

系统中的数据对象按照其对应的数据类型构成数据集合,在数据库中,通常不同业务类型的数据保存在不同的业务表内,表中的每条记录为一个数据对象。

数据权限一般与企业组织结构相关联,将数据对象与组织结构进行映射,形成不同范围的数据权限。本文为了简化数据权限的映射过程,在不降低灵活性,又突出细粒度数据权限的同时,根据企业组织结构,将数据权限分为 5 种,即全部数据权限、部门及以下数据权限、部门数据权限、仅本人数据权限以及自定义数据权限。

2.2 功能与数据组合控制

在分配某一角色权限时,可以对其可使用的功能权限加以约束,对于不同的功能权限,允许单独设置其对应的数据权限。

将功能权限集合与数据权限集合进行两两组合,形成功能关联数据的权限对象集合,表示为

$$\begin{aligned} FDP = FP \times DP = \\ \{\langle fp, dp\rangle | fp \in FP \wedge dp \in DP\} = \\ \{\langle fp_1, dp_1\rangle, \langle fp_1, dp_2\rangle, \cdots, \langle fp_1, dp_n\rangle, \langle fp_2, dp_1\rangle, \\ \langle fp_2, dp_2\rangle, \cdots, \langle fp_m, dp_n\rangle\} \end{aligned}$$

权限分配时,只需要分配权限集合 P 给某个角色即可,其中集合 $P \subset FDP$。

2.3 模型约束

基于上述设计的策略,为更进一步完善模型,使其更加符合实际的业务逻辑,将使用以下规定对其加以约束。

规定 1 针对实际逻辑,允许数据记录的创建者可以访问该条记录。

规定 2 只允许对角色进行授权,然后再将角色分配给用户,不允许直接对用户进行权限的分配。

3 FDPC 框架实现

针对上述提出的 FDPC 模型,下面将通过对物理模型的建立,以及权限控制流程进行分析,并对 FDPC 模型进行实现。

3.1 物理模型

根据数据权限控制的需求,建立认证鉴权的相关对象以及它们之间的联系,并完成物理模型的建立。认证授权的物理模型如图 3 所示。

user 为系统用户表,role 为系统角色,用户和角色之间为多对多的关系,在物理模型中增加 user_role 中间表进行关联;dept 为部门表,用户必须归属于某一个部门之下,因此 user 中包含 dept 的主键;business 为需要过滤数据的业务表,其中每条数据记录必须归属于某一个部门;function 为功能表,它和角色为多对多的关系,使用 role_function 与角色进行关联,role_function 中包含了数据权限的标识符 data_scope,如果 data_scope 为自定义数据权限,将通过实体 role_function_dept 与部门进行关联。

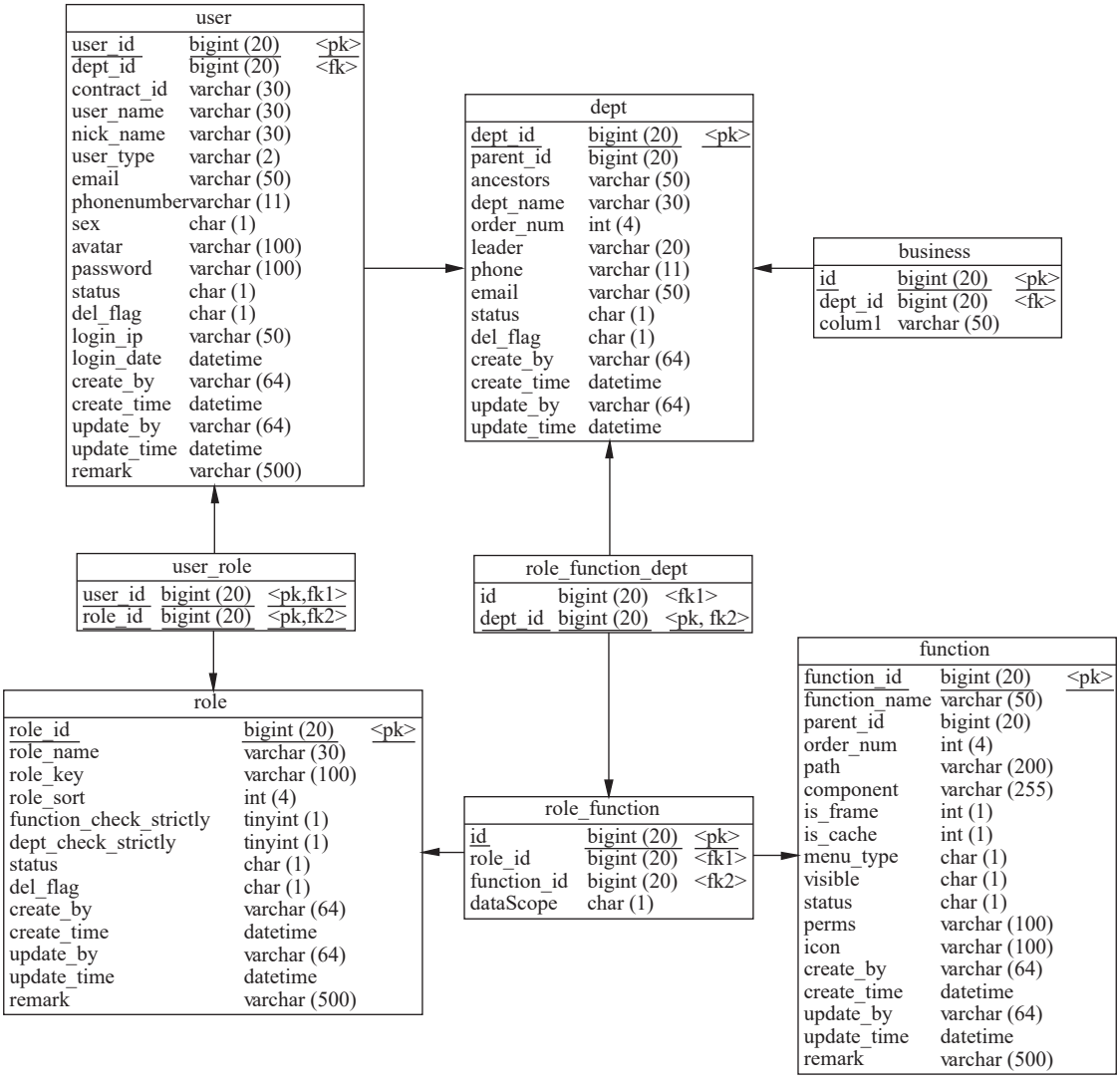


图 3　物理模型示意图

Fig.3　Schematic diagram of the physical model

3.2　权限控制流程

当用户第一次登录的时候， Spring Security 会拦截客户端发送的登录申请，并将截获的请求数据，如账号和注册密码等信息使用相同的安全加密流程予以加密，之后再将安全加密后的注册密码和数据库系统中存储的注册密码加以对比，一旦结果不相同则返回到登录页面。如果登录成功， Spring Security 会根据登录的信息创建一个令牌，并将这个令牌作为数据返回到客户端，同时令牌还会被存储到缓存服务器中。客户端通过 Spring Security 返回过来的功能权限对菜单栏进行动态渲染，为不同权限的用户展示不同的菜单栏。

如果用户不是第一次登录，只需要在发送请求的时候，把令牌加入请求头发送到服务器，不需要验证账号密码，即可从令牌里面提取出用户账号名进而从缓存服务器中获取新的令牌。如果用户账号名一样，将通过令牌里面的时间戳检查令牌是否已过期，如果已过期，将重新创建令牌。认证的时序图如图 4 所示。

因为令牌里面有基本的用户信息和时间戳，所以在登录成功之后，用户在每次访问其他界面和资源的时候都必须要在请求头加入令牌。这样可以通过令牌的信息获取该用户是否有权限访问该资源，若没有权限则显示无权访问。所以利用这种方法对身份和权限进行统一认证是安全有效的。鉴权流程如图 5 所示。

通过用户身份验证和权限验证整个流程的详细描述，可以确定数据权限控制的关键过程在于对数据库的数据查询，从而可以采用 SQL 拼接的手段来实现对数据权限的过滤。

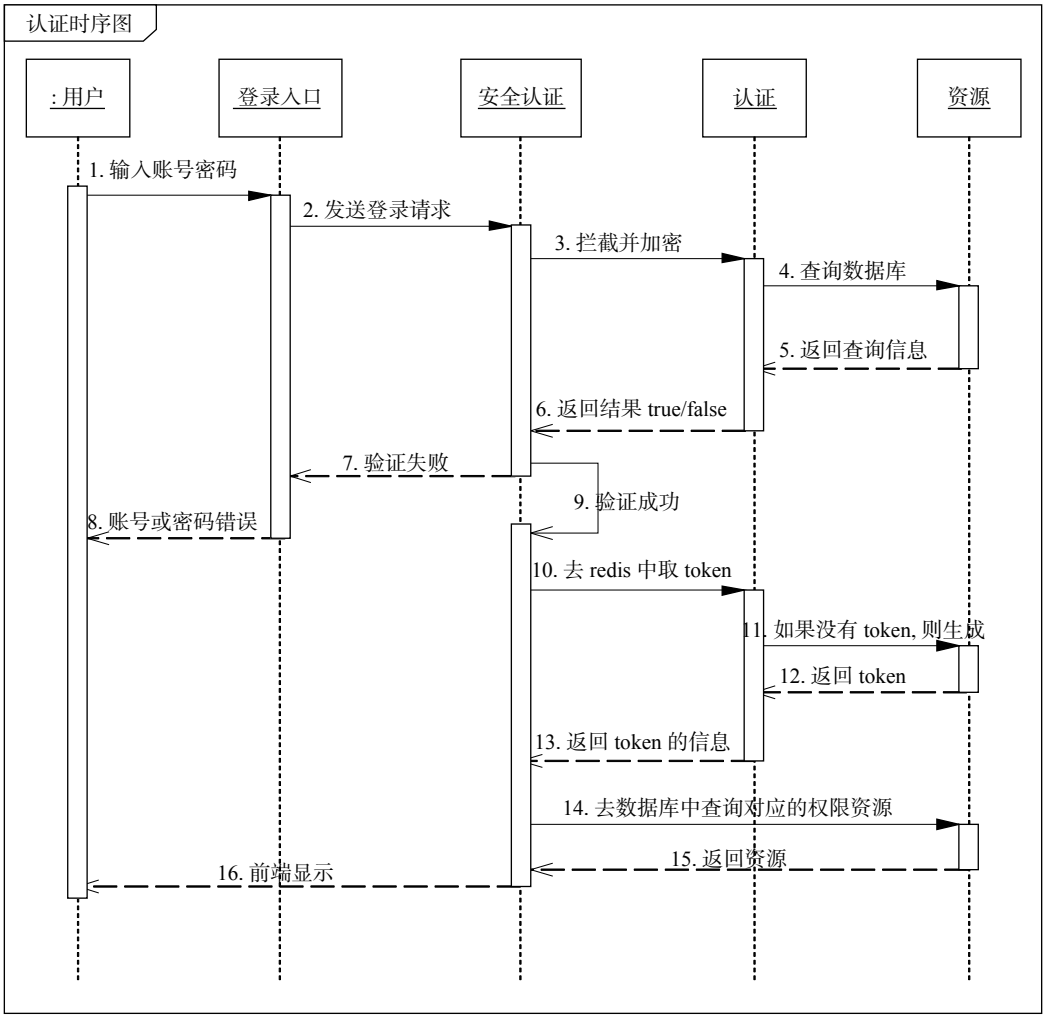


图 4 认证时序图

Fig.4 Authentication sequence diagram

3.3 数据权限控制实现

通过上述物理模型的建立和权限控制流的分析，为数据权限控制的实现提供了可靠的理论依据。实现的关键步骤将采用 SQL 拼接的方式进行数据过滤，因此必定会造成逻辑代码重复问题。为了解耦数据过滤时的代码，通过深入研究 MyBatis 框架中传参的机制，依据 AOP 思想，采用 SQL 拼接的方式实现数据权限控制。其中涉及两个重要的类，分别为 DataScope 和 DataScopeAspect。DataScope 是一个注解类，将其注解到需要数据权限过滤的方法上作为切入点；DataScopeAspect 是数据过滤的切片，其中包含数据权限标识符和实现的对应方法。数据过滤模块类图如图 6 所示。

首先在验证功能权限成功之后，使用自定义注解@DataScope 作为切入点，将其注解添加到需要进行数据过滤的业务层方法上，然后通过置入数据过滤切片。根据请求功能的权限标识符获取其内部包含的数据权限标识，通过数据权限标识判断数据权限范围并将其对应的 SQL 语句作为参数保存在功能的请求实体里面返回。在数据持久层里面使用${params.dataScope}取出数据过滤的 SQL 语句，将其进行拼接，实现数据过滤。数据权限控制实现流程如图 7 所示。

通过进一步了解 Mybatis 的实现原理，发现数据过滤切片中存在 SQL 注入风险。造成此问题的原因在于数据过滤前没有把 params.dataScope 参数值清空，因此在切片类中增加一个 clearDataScope 方法，在拼接权限 SQL 语句前先调用此方法，对 params.dataScope 参数值进行清空，防止 SQL 注入。

4 框架应用实例

为了验证框架的有效性，基于 FDPC 框架开发了温泉度假村客户关系管理系统，其目标是全方

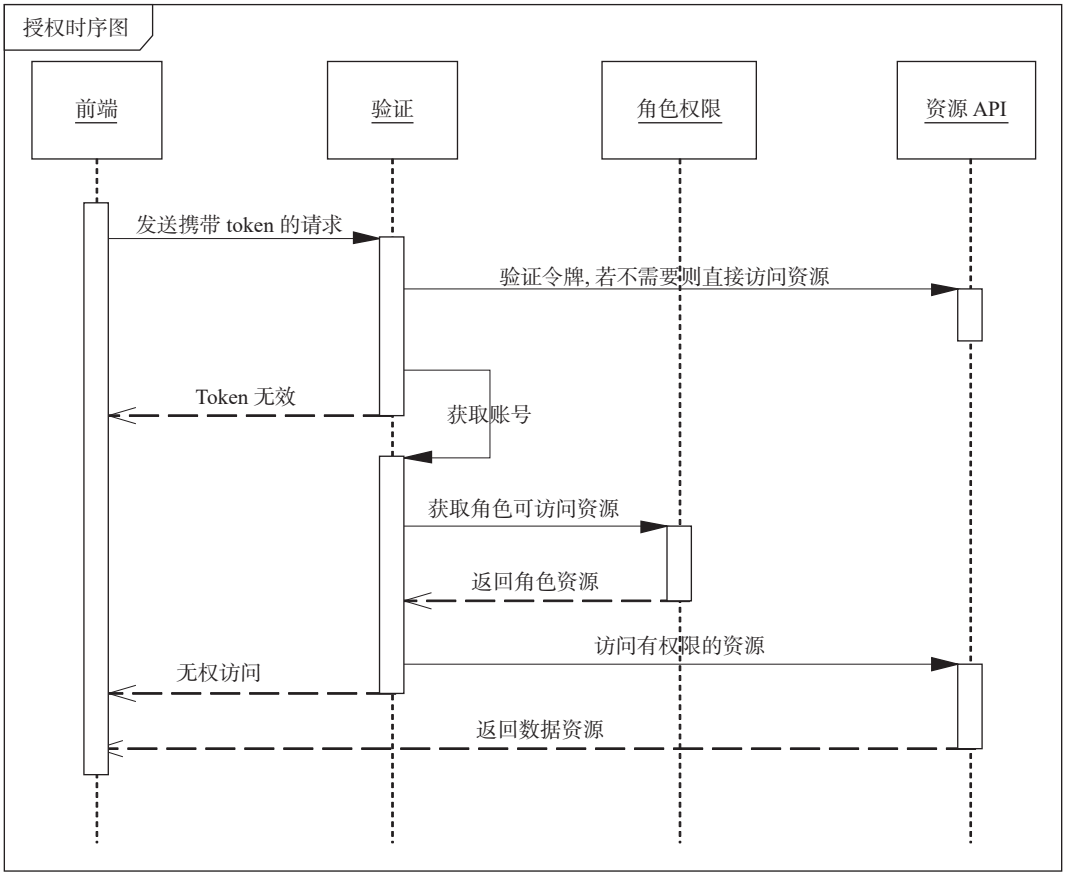


图 5　鉴权时序图

Fig.5　Authentication sequence diagram

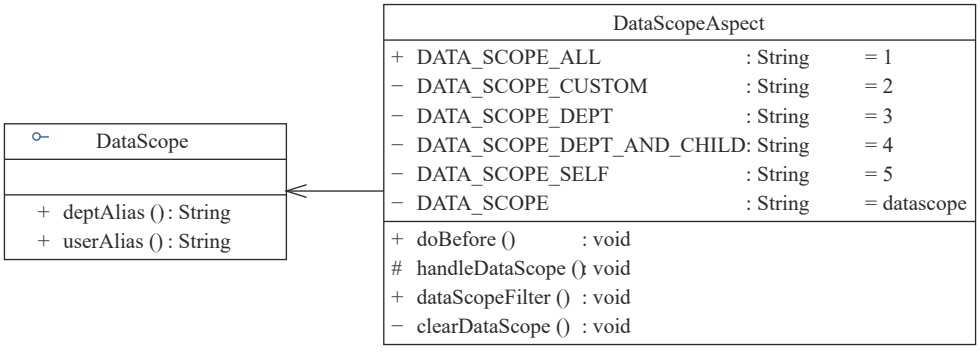


图 6　数据过滤模块类图

Fig.6　Class diagram of data filtering module

位管理客户信息，基于此协同办公管理和客户关系维护等内容，实现客户、员工、财务、市场工作的平台化，使系统规范统一。根据业务需求和用户需求结合实际应用进行用例建模，客户关系管理系统用例图如图 8 所示。

其中角色大致分为 5 类，分别为系统管理者、市场部、行政部、客服部和财务部，其权限如下：系统管理者对系统进行维护，拥有所有功能和数据的权限；市场部主要负责客历采集和分类，对于客历信息查询拥有全部数据权限，对于合约公司查询以及账单查询只拥有部门及以下数据权限；行政部主要负责合约公司管理，对于合约公司信息拥有全部数据权限，对于分析结果和账单查询仅拥有部门及以下数据权限；客服部主要负责合约和客历的分析和结果归档，以及客历关怀，因此对其拥有全部数据权限，对于账单查询只拥有部门及以下数据权限；财务部主要负责账单管理，对于账单查询拥有全部数据权限。

以客户关系管理系统为例，市场部这一角色对于客历信息查询这一功能的数据权限为全部数

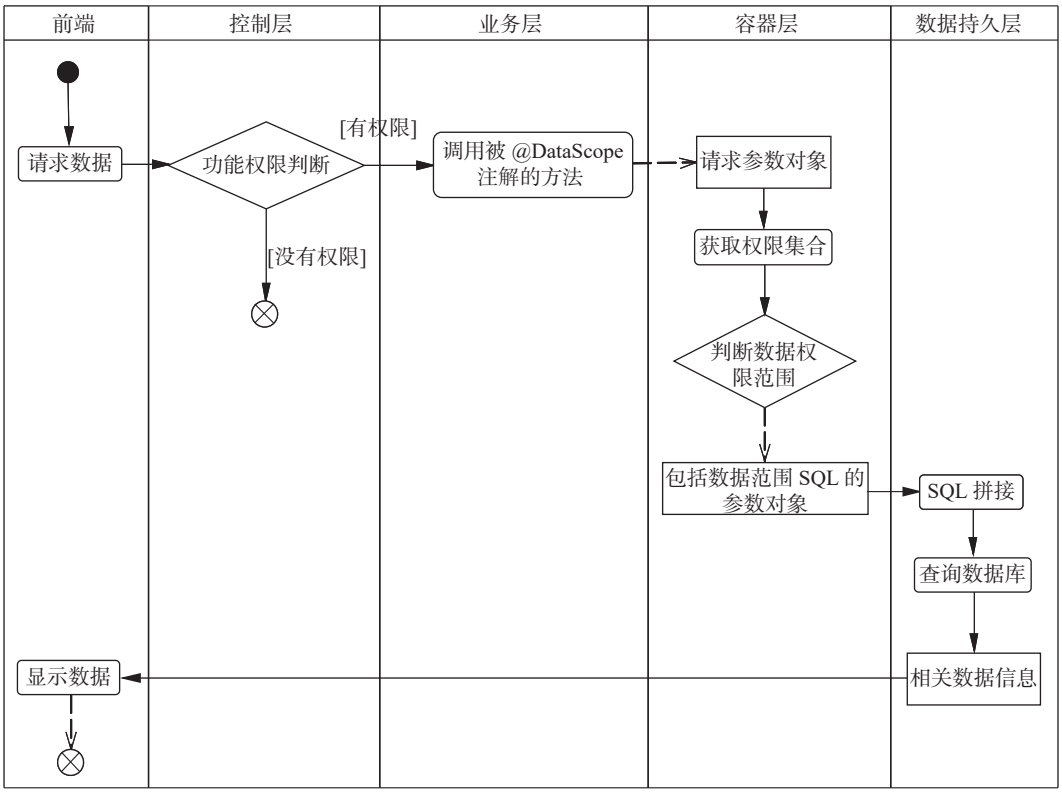


图 7　数据过滤活动图

Fig.7　Data filtering activity diagram

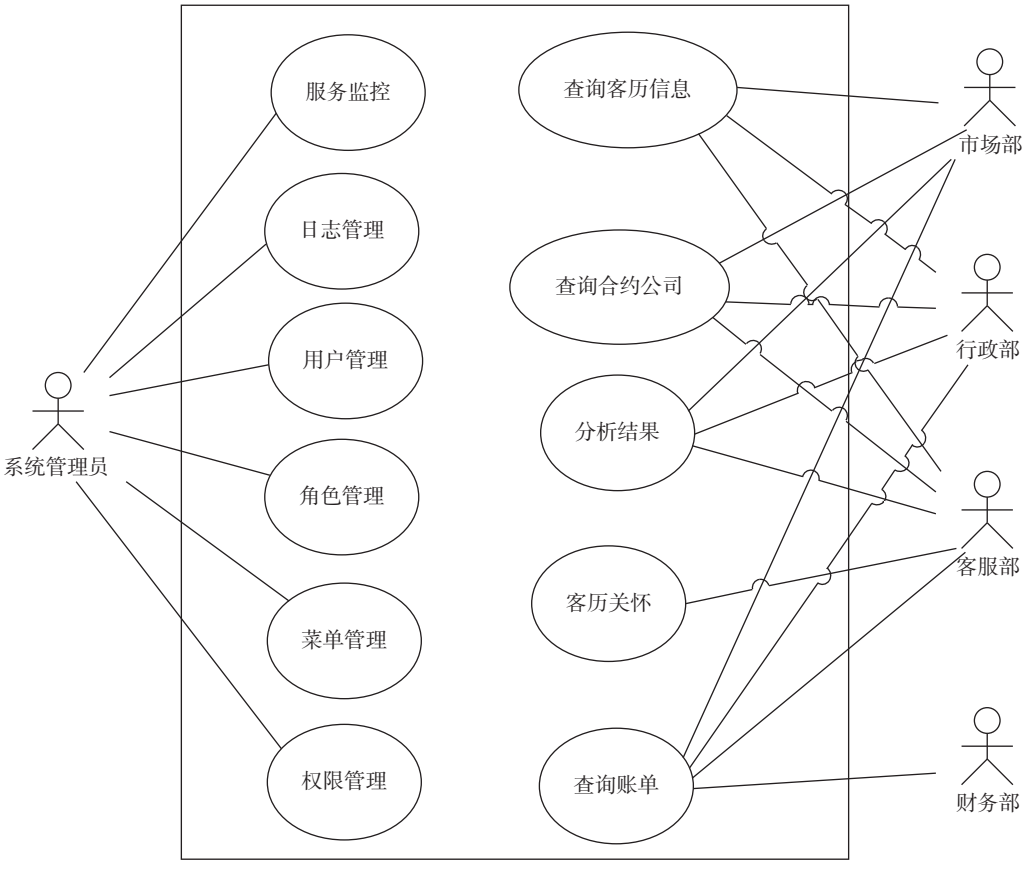
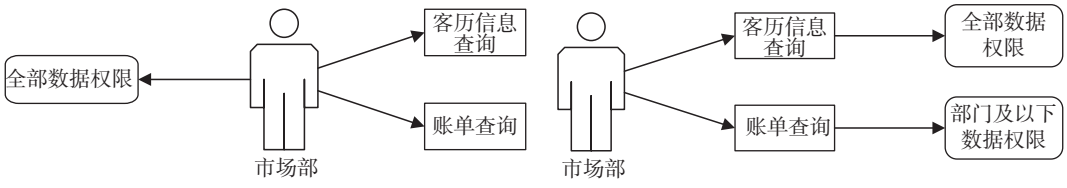


图 8　客户关系管理系统用例图

Fig.8　Use case diagram of customer relationship management system

据权限，而对于账单查询的数据权限仅限于部门及以下。在传统模型中，市场部这一角色有且仅有一个数据权限，角色所能操作的功能都只能获取同一个范围的数据，显然无法实现上述案例描述。而在 FDPC 模型中，可以为客历信息查询和账单查询两个功能分配对应的数据权限，从而满足上述案例描述。模型对比示意图如图 9 所示。

对于角色授权，首先对角色分配菜单权限（功能权限），然后对分配的功能权限分配对应的数据权限，如果权限范围为自定义数据权限，根据需求来选择对应部门。角色授权界面如图 10 所示。



(a) 基于传统模型的案例描述　　(b) 基于 FDPC 的案例描述

图 9　模型对比示意图

Fig.9　Schematic diagram of model comparison

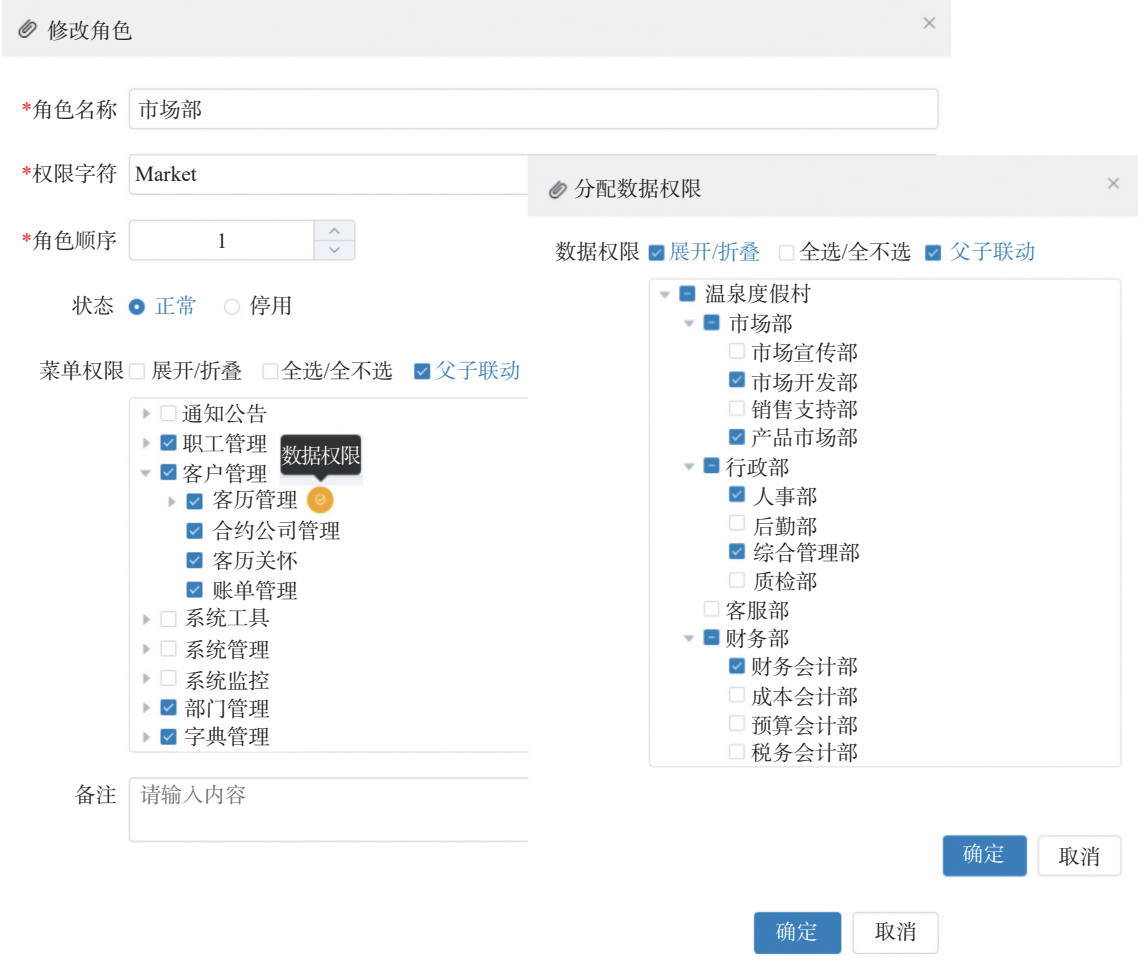


图 10　角色授权界面

Fig.10　Role authorization interface

5　结　论

在 RBAC 的基础上，提出了一种细粒度数据权限控制模型 FDPC。该模型通过将数据对象与企业组织结构映射成不同粒度的数据权限，并将其与功能权限对象组合的方式，达到了不同粒度数

据权限的灵活控制目的。为验证SaaS模式应用的有效性，选用Spring Security和MyBatis，依据AOP切片原理，采用结构化查询语句拼接方式，实现了一种细粒度数据权限控制框架，解决了SaaS为模式的服务框架权限控制方面存在的数据粒度粗及配置灵活度不足等问题。在业务系统中应用结果表明该框架合理可行，能满足实际应用中灵活授权和细粒度数据控制的需求。

参考文献：

[1] 吕卫锋, 郑志明, 童咏昕, 等. 基于大数据的分布式社会治理智能系统 [J]. 软件学报, 2022, 33(3): 931–949.

[2] NAZERIAN F, MOTAMENI H, NEMATZADEH H. Emergency role-based access control (E-RBAC) and analysis of model specifications with alloy[J]. Journal of Information Security and Applications, 2019, 45: 131–142.

[3] 黄毅, 李丽娟. 基于RBAC模型中角色继承关系的改进 [J]. 科学技术与工程, 2010, 10(4): 1066–1069.

[4] GHAFOORIAN M, ABBASINEZHAD-MOOD D, SHAKERI H. A thorough trust and reputation based RBAC model for secure data storage in the cloud[J]. IEEE Transactions on Parallel and Distributed Systems, 2019, 30(4): 778–788.

[5] UDDIN M, ISLAM S, AL-NEMRAT A. A dynamic access control model using authorising workflow and task-role-based access control[J]. IEEE Access, 2019, 7: 166676–166689.

[6] 谭作文, 张连福. 机器学习隐私保护研究综述 [J]. 软件学报, 2020, 31(7): 2127–2156.

[7] YANOVSKI J, DANG H H, JUNG R, et al. GhostCell: separating permissions from data in Rust[C]// Proceedings of the ACM on Programming Languages, 2021, 5(ICFP): 1-30.

[8] YU X Y, HAAKENSON B, PHILLIPS T, et al. User-friendly design of cryptographically-enforced hierarchical role-based access control models[C]//Proceedings of the 29th International Conference on Computer Communications and Networks (ICCCN). Honolulu: IEEE, 2020: 1–9.

[9] DE CARVALHO JUNIOR M A, BANDIERA - PAIVA P. Towards unobtrusive patient - centric access - control for Health Information System[J]. Concurrency and Computation:Practice and Experience, 2020, 32(22): e5845.

[10] PASOMSUP C, LIMPIYAKORN Y. HT-RBAC: a design of role-based access control model for microservice security manager[C]//Proceedings of 2021 International Conference on Big Data Engineering and Education (BDEE). Guiyang: IEEE, 2021: 177–181.

[11] PILIPCHUK R, HEINRICH R, REUSSNER R H. Automatically extracting business level access control requirements from BPMN models to align RBAC policies[C]//Proceedings of the 7th International Conference on Information Systems Security and Privacy. 2021: 300–307.

[12] ZHONG T, CHANG J S, SHI P C, et al. Dyacon: JointCloud dynamic access control model of data security based on verifiable credentials[C]//Proceedings of 2021 IEEE Intl Conf on Parallel & Distributed Processing with Applications, Big Data & Cloud Computing, Sustainable Computing & Communications, Social Computing & Networking. New York: IEEE, 2021: 336–343.

[13] ALSHAMMARI S T, ALBESHRI A, ALSUBHI K. Integrating a high-reliability multicriteria trust evaluation model with task role-based access control for cloud services[J]. Symmetry, 2021, 13(3): 492.

[14] 程学林, 杨小虎, 卓崇魁. 基于组织架构的数据权限控制模型研究与实现 [J]. 计算机科学, 2021, 48(S1): 558–562.

[15] ALI A, ALMAIAH M A, HAJJEJ F, et al. An industrial IoT-based blockchain-enabled secure searchable encryption approach for healthcare systems using neural network[J]. Sensors, 2022, 22(2): 572.

（编辑：丁红艺）