**细粒度权限管理系统**

**用户使用手册**

**复旦大学**

**2021年4月**

# 目录

[目录 2](#_Toc69215524)

[1 概述 3](#_Toc69215525)

[2 功能简介 3](#_Toc69215526)

[3 系统登录 4](#_Toc69215527)

[4 系统主界面 4](#_Toc69215528)

[5 细粒度权限管理平台业务流程说明 5](#_Toc69215529)

[6 细粒度权限管理平台业务功能操作说明 6](#_Toc69215530)

[6.1 组织管理 6](#_Toc69215531)

[6.1.1 新增组织申请 6](#_Toc69215532)

[6.1.2 新增组织申请查询 6](#_Toc69215533)

[6.1.3 审批新组织 7](#_Toc69215534)

[6.1.4 提交part-pk和最终确认 8](#_Toc69215535)

[6.1.5 组织的查询 9](#_Toc69215536)

[6.2 属性管理 10](#_Toc69215537)

[6.2.1 声明用户属性 10](#_Toc69215538)

[6.2.2 声明组织属性 10](#_Toc69215539)

[6.2.3 用户申请属性 13](#_Toc69215540)

[6.2.4 用户审批属性 13](#_Toc69215541)

[6.2.5 组织属性的同步 14](#_Toc69215542)

[6.3 文件共享管理 14](#_Toc69215543)

[6.3.1 加密分享 14](#_Toc69215544)

[6.3.2 查询解密 15](#_Toc69215545)

# 概述

访问控制是通过某种途径显式地允许或限制用户访问能力及范围的一种方法，是系统保密性、完整性、可用性和合法使用性的重要基础，是网络安全防范和资源保护的关键策略之一。为了保证网络资源受控地、合法地被使用，访问控制确保只有拥有相应权限的用户才能够对信息系统中的数据进行访问、复制、修改、删除等操作，防止非法用户访问系统资源、预防合法用户越权访问。用户访问资源时必须依据自己的权限大小进行访问，不能实施超越自己权限的访问行为，因为访问控制采用最小特权原则：用户申请权限时，系统管理员根据每个用户的特点为其分配完成自身任务的最低权限，用户不会获得超越完成其工作范围的任何权力。通俗地讲，访问控制解决的是“你能做什么、你有什么样的权限”的问题。访问控制的基本目标是：防止未授权用户非法访问受保护的数据资源，也让授权用户可以合理访问受保护的资源，当然这也是一个安全系统所必须具备的特性[14]。在访问控制实施时，一般有三个步骤：(1)验证用户身份；(2)选用控制策略与管理控制策略；(3)管理非法用户或合法用户的越权操作。

随着计算机网络系统跨行业、扩平台的复杂化，大规模用户为了应对各种应用需要，对权限的需求变得多样化、更加细粒度，即用户已经不满足于获得单一的、粗糙的、千篇一律的访问权限，而是在不同时刻、不同状态、不同需要时获得不同的、细粒度的权限，细粒度权限管理对访问控制提出了新的需求。

智慧城市海量数据管理需求不一、信息系统权限体系复杂，更是需要一种有效可靠的细粒度权限管理系统来支持整个智慧城市平台的安全，为智慧城市信息系统提供可靠的信息安全保障服务。

# 功能简介

细粒度权限管理系统包括以下模块：

**数据层**：数据层存储了整个区块链系统的数据存储和其操作对象的抽象结构，通过对数据结构的分析我们可以管中窥豹，一览方案的目的和流程。同时基于面向对象分析与设计的思想，数据结构往往也能反映客观世界的本来面目，采用现实生活中的思维方式去看待数据结构往往能够获得不一样的收获。

**合约层：**合约层致力于提供通用的插件供所有需要身份体系或细粒度属性控制的区块链平台使用，所以几乎所有的逻辑控制和业务功能都是由合约层控制的。同时合约层的设计方案和实际处理也考虑到了本方案“去中心化”的底线，不存在任何一个地方使用随机数、网络调用第三方平台等会引入所谓“可信”第三方平台的因素。

**应用层：**Nodejs前端部分提供了最简易的页面操作；Java Web后端用于对前端请求进行预处理，封装成指定的数据格式，然后搭配Fabric提供的Java SDK来调用智能合约完成对链上数据的相关操作。

# 系统登录

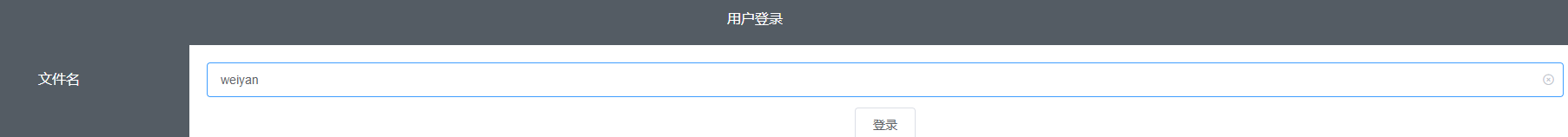


图 3‑1 系统登陆

系统用户输入存放在本机的文件名即可登录，这里考虑到仿真需要和本地文件存储并没有设置密码，可以根据实际使用场景修改。输入文件名后，点“登录”按钮，如图3-1。如果文件名正确无误，则进入系统界面，否则会报告错误信息。

# 系统主界面

登录后系统主界面如下：

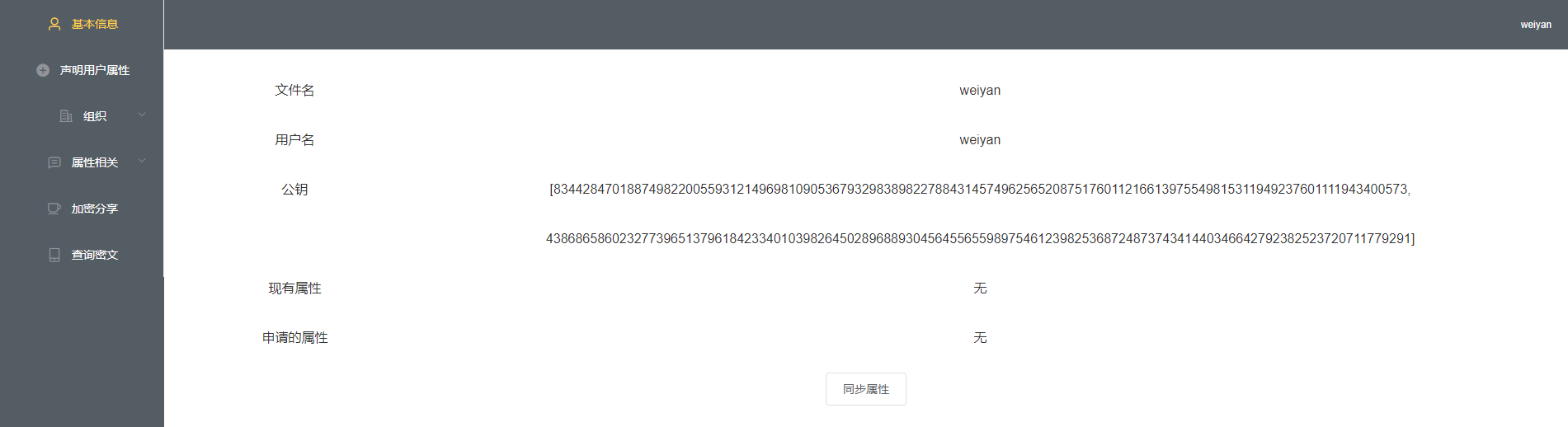


图 4‑1 系统主界面

系统主界面是我们所有功能集成的统一使用界面，左侧导航栏分布所有的大小功能，右上角标识了文件名，默认页面展示用户的基本信息，如图4-1所示。

# 细粒度权限管理平台业务流程说明

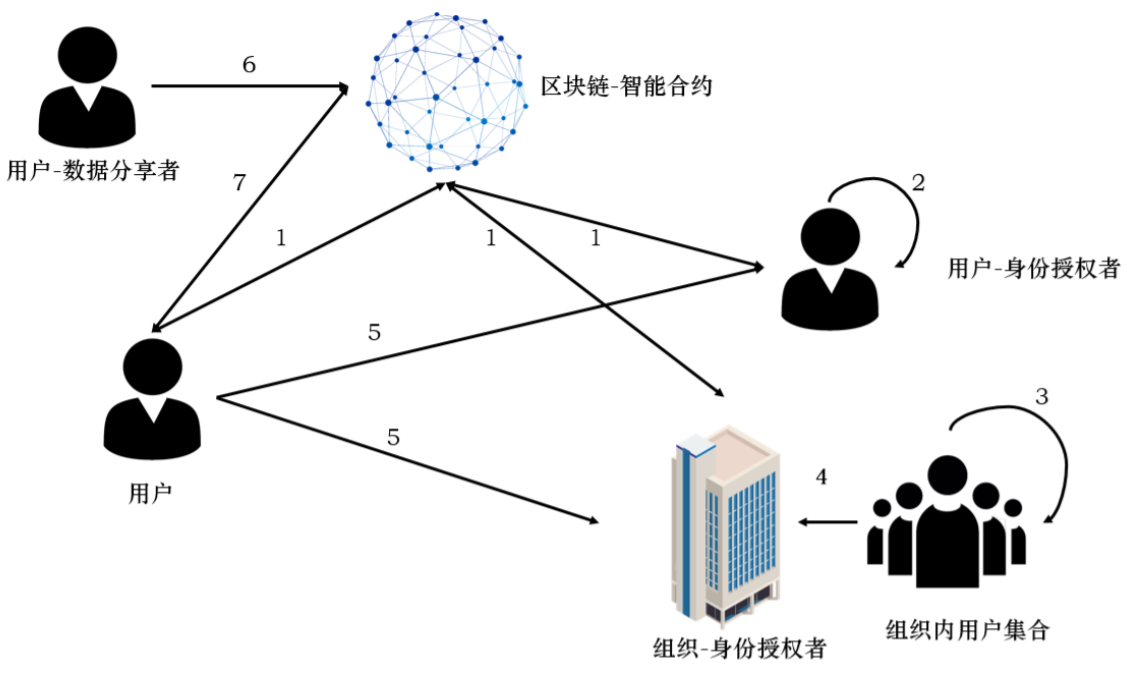


图 5‑1 业务流程图

主要业务流程为：

1、所有的用户或组织可以通过区块链和智能合约获取链上数据（包括但不限于公共参数），并将初始化产生的公钥、加密数据产生的密文等公开上链。用户与用户之间、用户与组织之间的所有通信皆通过智能合约发起。

2、用户自身选择 GID 并执行用户初始化或声明新属性等操作，将产生的用户公钥或用户属性上链。

3、多个用户之间协作进行组织初始化和组织声明新属性操作，该过程产生中间参数。其中总用户数量为 n，门限秘密分享的阈值为 t，恶意用户的占比不超过 n − t。

4、组织通过智能合约收集协作用户产生的中间参数，将其组合计算组织公钥或组织属性，并公开上链。

5、用户（申请属性）向其他用户或组织申请其下属特定的属性，由用户或组织下属的用户自行判断是否通过其申请，若通过则返回相应的秘密参数。用户（申请属性）根据秘密参数自行计算属性私钥并保存。

6、用户（数据分享）设定特殊的访问控制策略，通过属性加密算法将数据加密得到密文后上链。

7、用户从区块链中获得密文，如果满足密文的访问控制策略，则可将其解密获得明文数据。

# 细粒度权限管理平台业务功能操作说明

## 组织管理

负责管理和维护属性授权组织的相关信息。

### 新增组织申请

考虑到初始并没有组织，于是我们先从组织申请开始。申请新组织需要填写组织名称、阈值t和成员总数n、并将目标组织内用户的名称填写进去，点击提交申请即可，见图6-1：

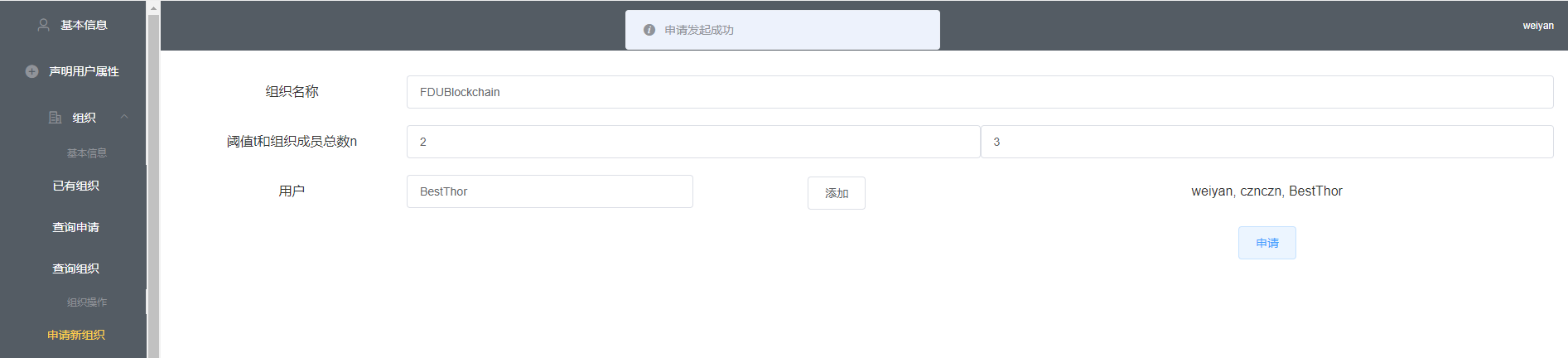


图 6-1 新增组织申请

### 新增组织申请查询

组织申请查询是组织申请过程中非常关键的一个功能，申请查询可以观察目前各个成员对该组织申请的处理进度，方便沟通，下面的流程展示中将把申请查询的结果融合到一起，实际上申请查询是独立的一个模块。见6-2所示：

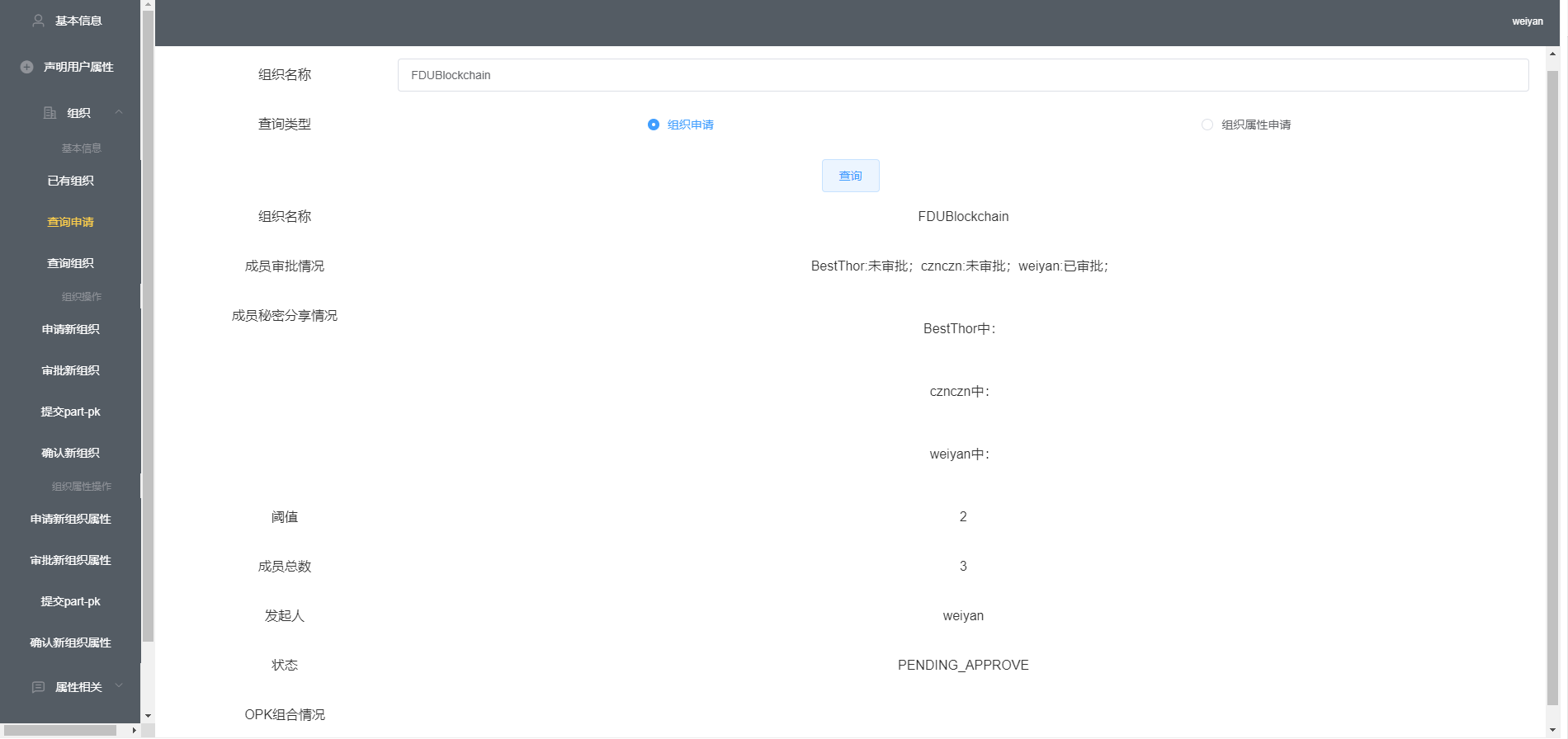


图 6-2 新增组织申请查询

### 审批新组织

如果一个成员想要同意加入某组织，只需要在审批新组织中输入该组织的名称，点击加入即可。当所有成员都确认加入后，该组织申请就可以进入到下一阶段。同样的，该操作由后端自动完成几个步骤的融合，加入+分享秘密同时进行，见图6-3：



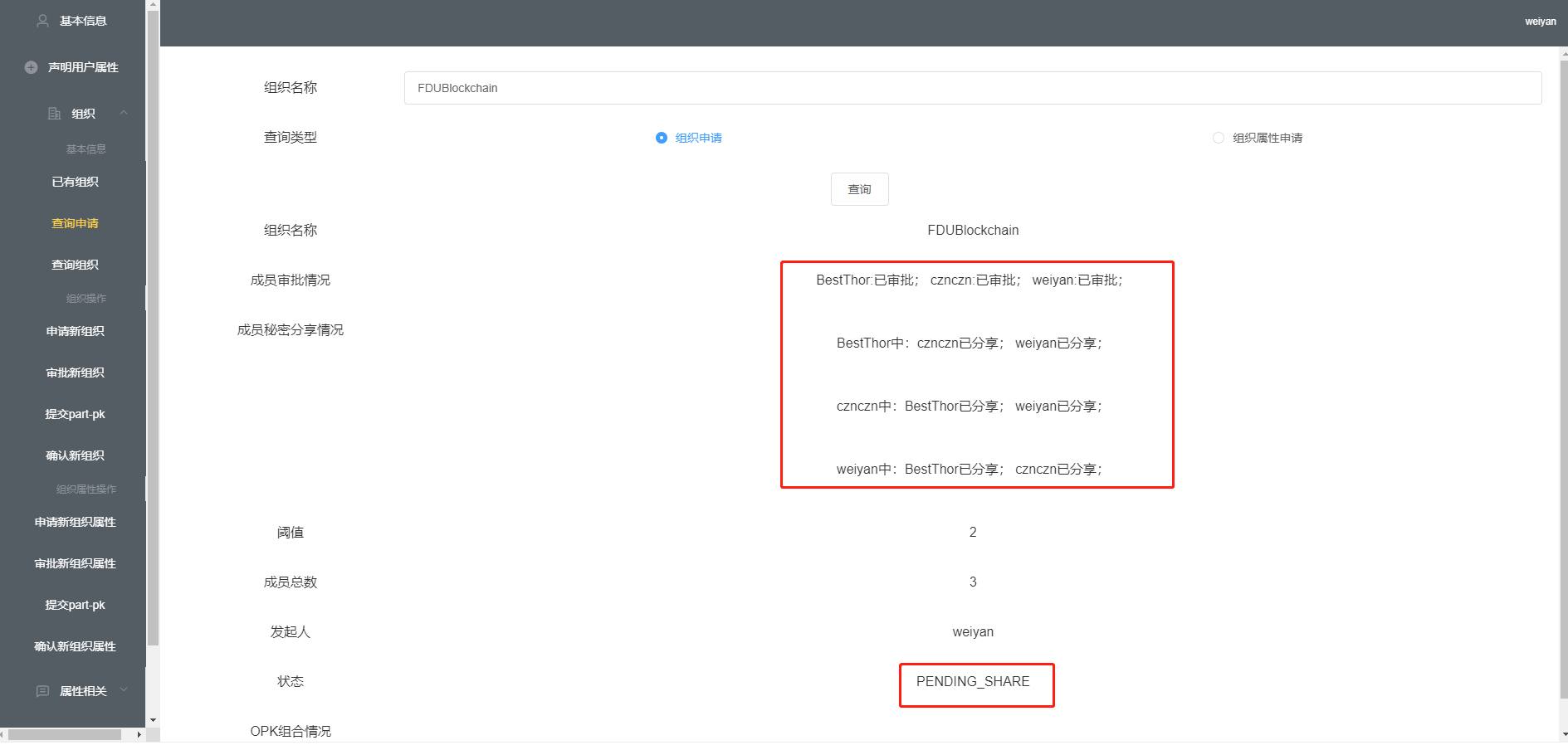
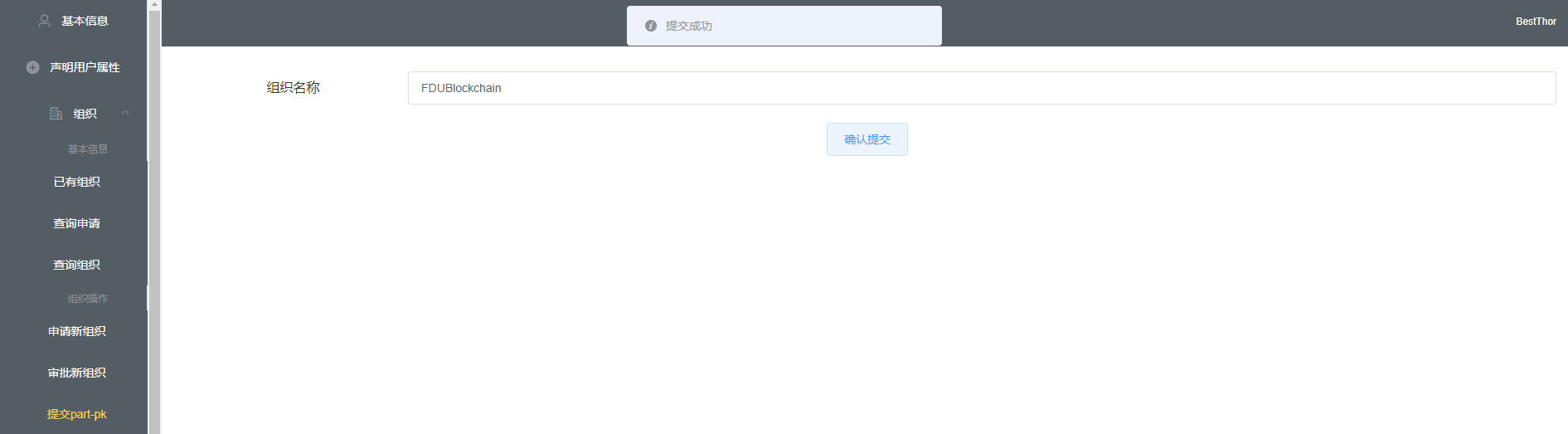


图 6-3 审批新组织与组织申请查询

### 提交part-pk和最终确认

当所有成员都同意加入到该组织后，组织申请的状态进入到PENDING\_SHARE, 此时需要所有成员提交part-pk. 该操作之所以不能整合到加入步骤是因为组织的加入是异步的，而part-pk需要上一步中所有成员产生的秘密share。同样的，用户点击提交后后端会自动整合share并生成part-pk，最终上传至合约。当所有用户提交完毕后，需要一个组织内成员执行确认新组织的操作，同样是异步的原因。这两步和过程中的组织申请状态见图6-4与6-5：



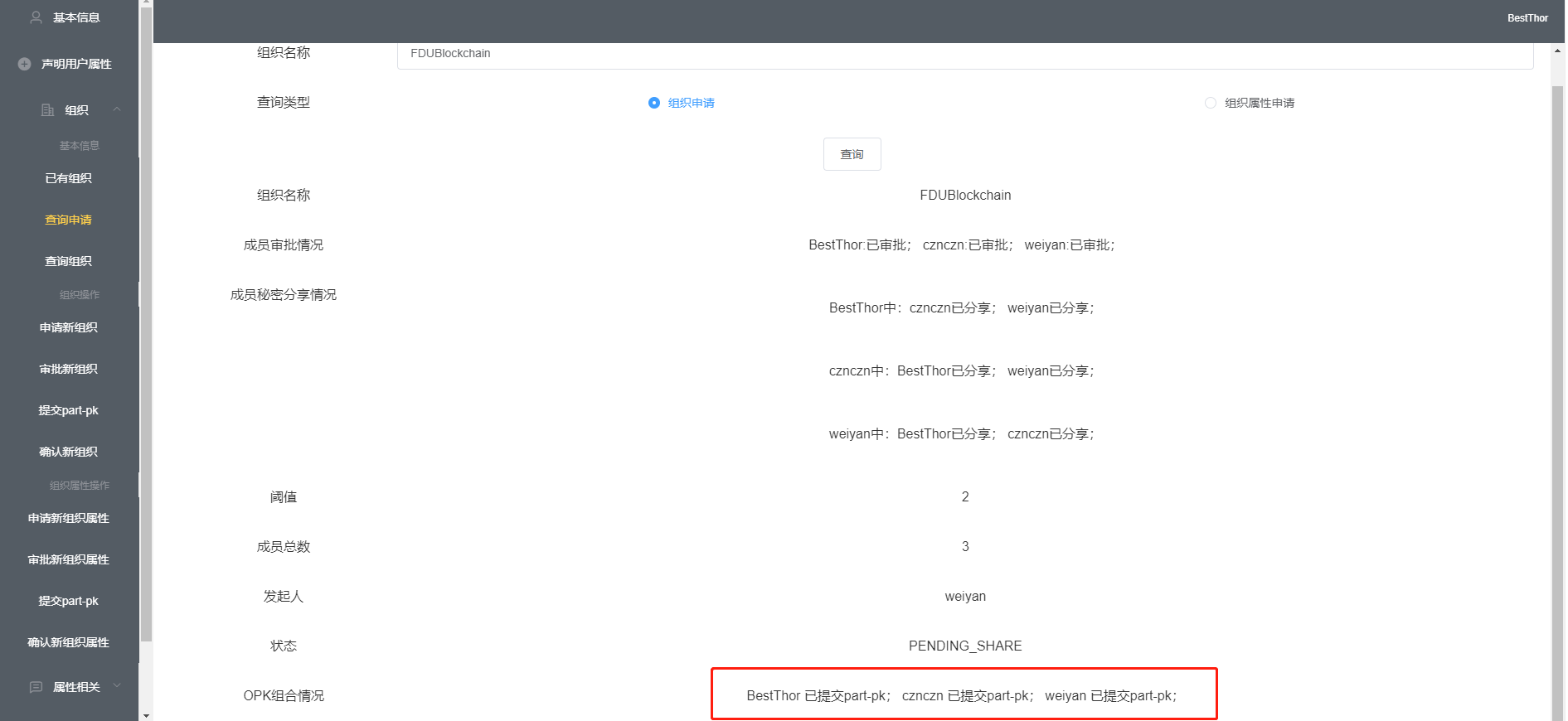
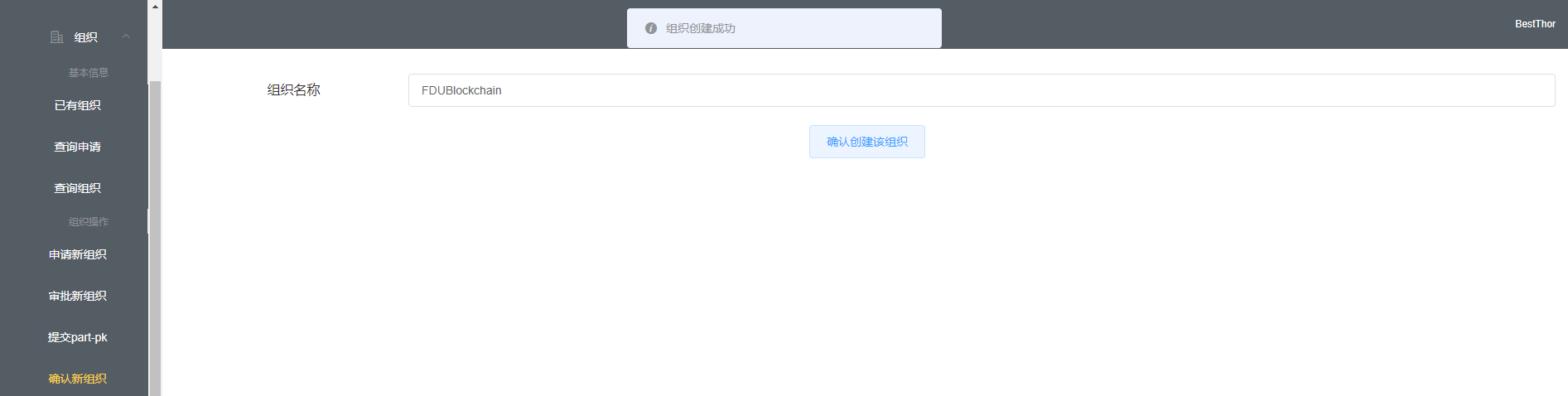
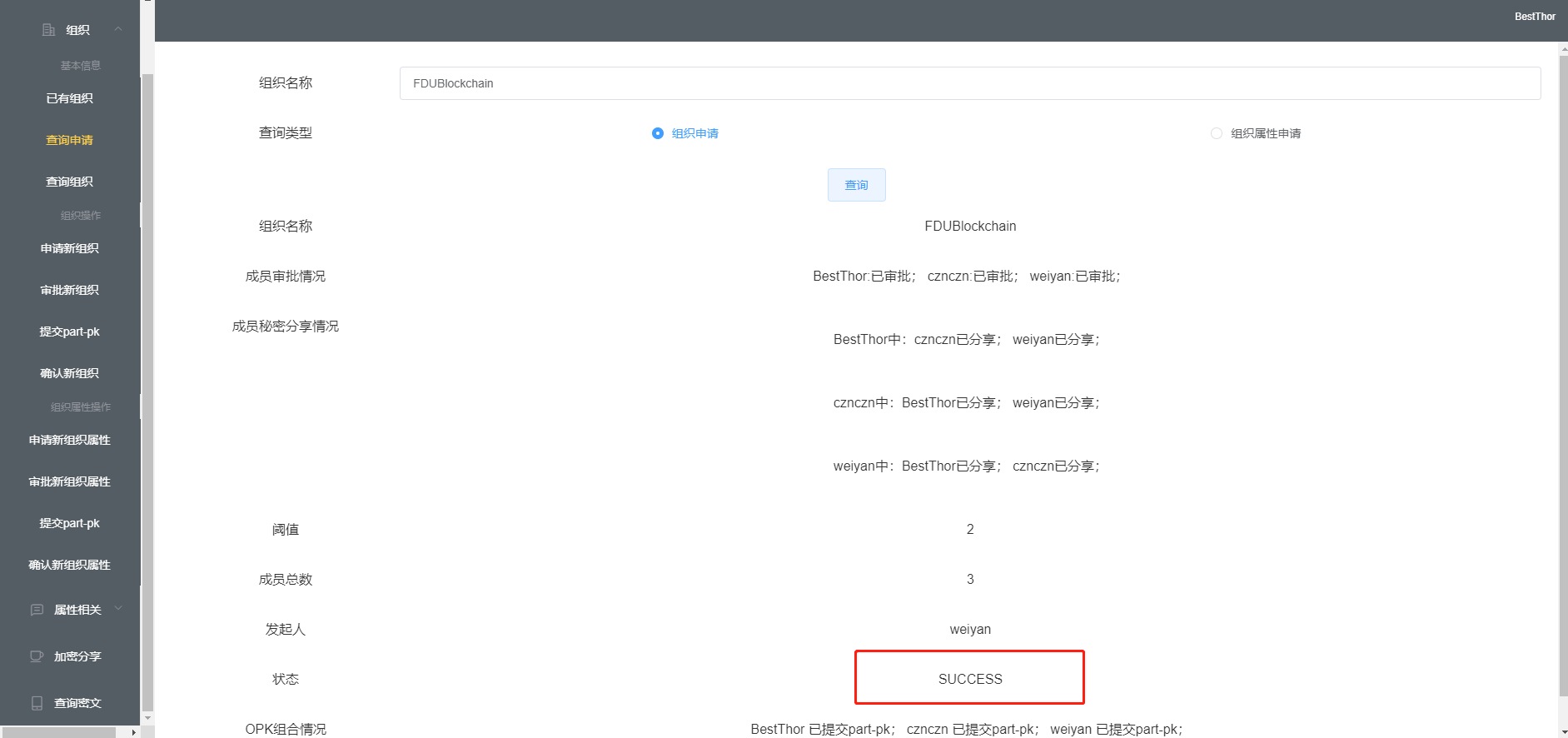


图 6-4 提交part-pk和组织申请查询





**图 6-5 确认新组织和组织申请查询**

### 组织的查询

当组织FDUBlockchain拥有了自己的第一个属性FDUBlockchain:member之后，我们可以通过查询组织来查看一个组织的基本信息，包括其基本的参数、OPK和拥有的属性等，如图6-6所示：

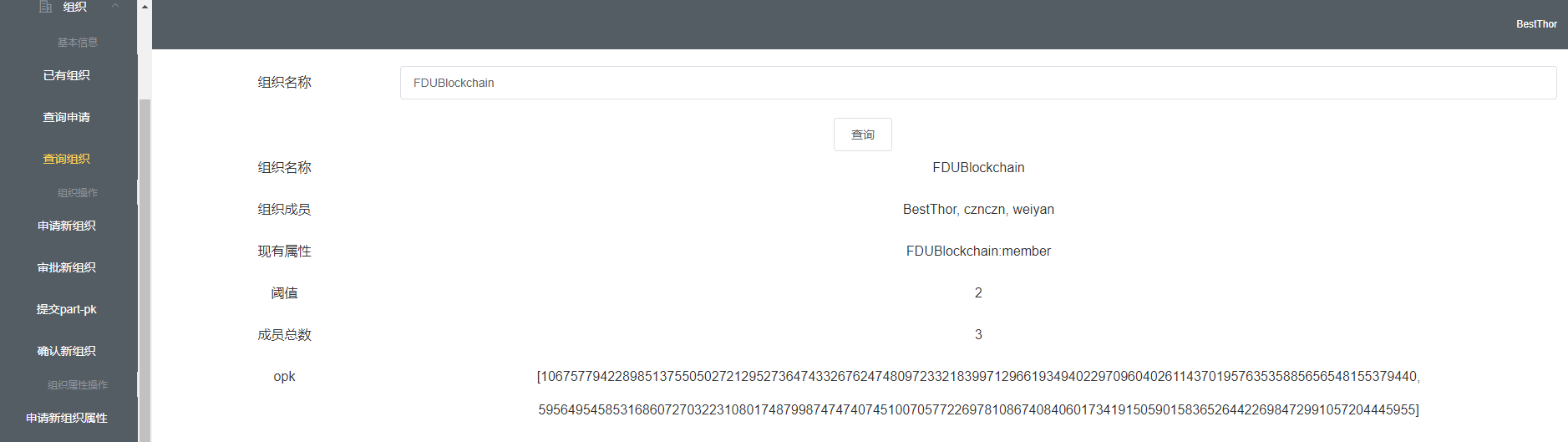


图 6-6 查询组织的基本信息

## 属性管理

负责管理和维护用户和授权授权组织的属性信息。

### 声明用户属性

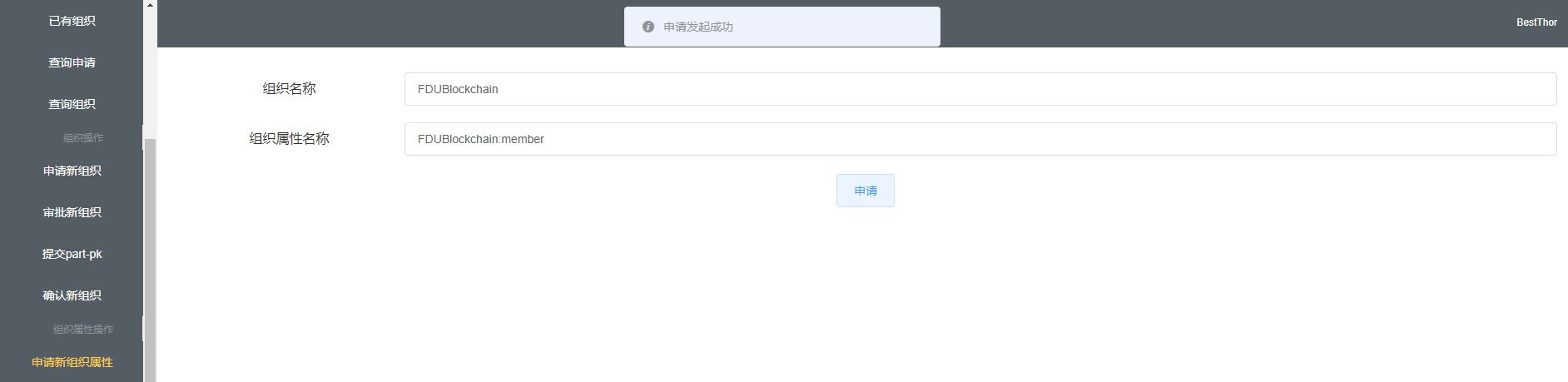
用户注册完成自己的身份后，就可以“发行”自己的属性，例如朋友、家人、同学、粉丝，甚至是偶像、QWERTY这种或许永远不会被申请的属性。在声明属性页面也十分简单，只需要填写想要声明的属性名称，只要不与现有属性重合且符合格式要求即能申请成功，后端会简化用户操作，自动顺序请求两个不同的合约完成属性声明操作，如图6-7所示：

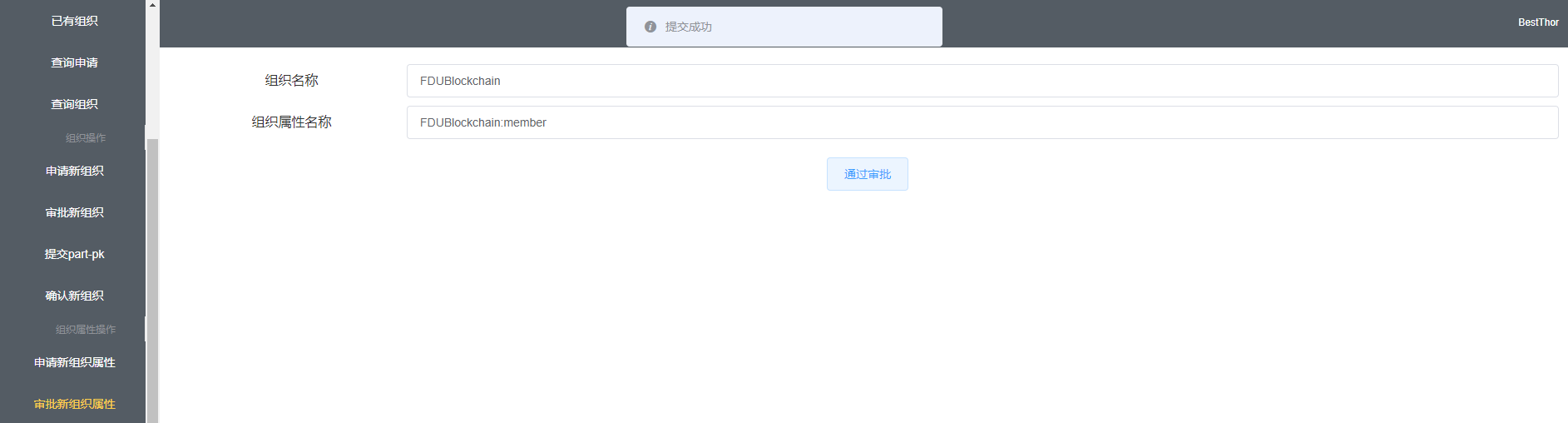


图 6-7 声明用户属性

### 声明组织属性

新组织属性的申请流程与新组织的申请流程基本完全一致，在各个步骤中都需要输入想要申请的组织属性名称，为了补全新组织申请中过程的申请状态，这里用图6-8到图6-10简要展示下组织属性的申请流程：





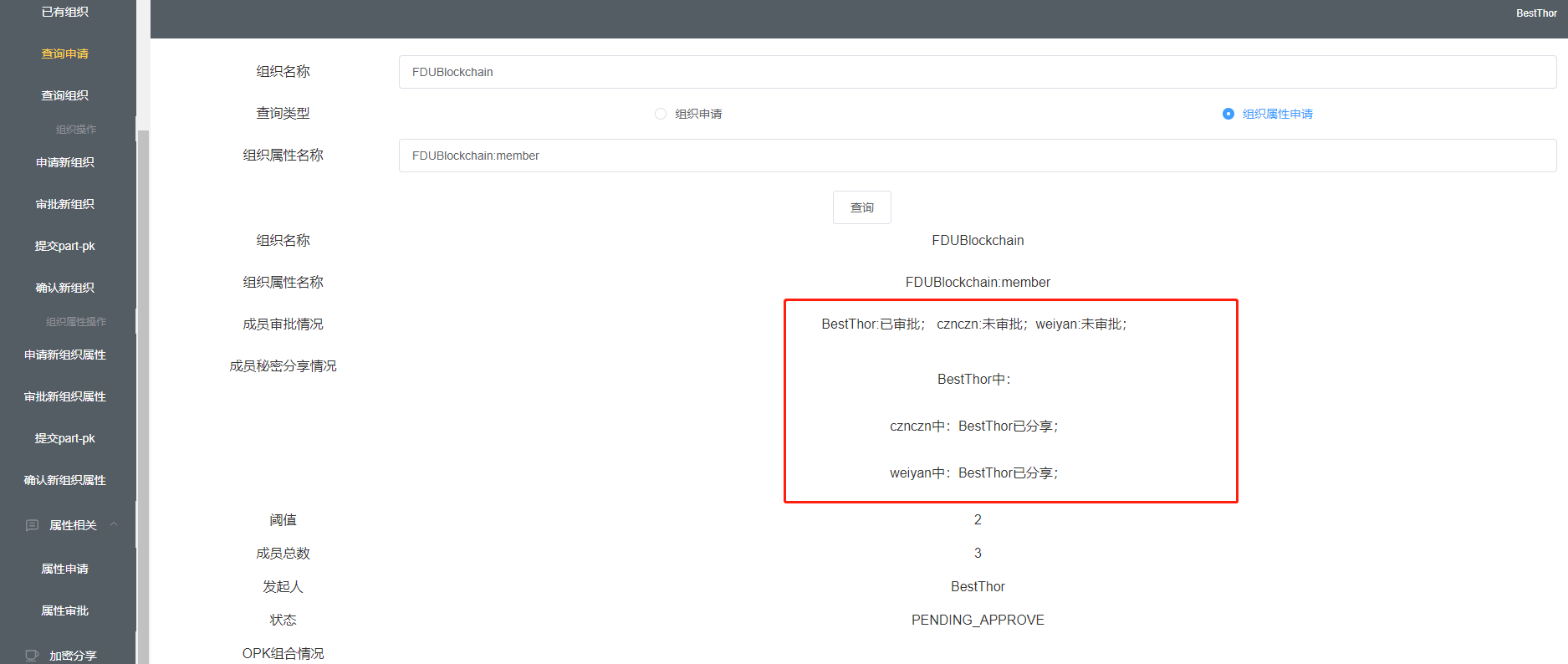
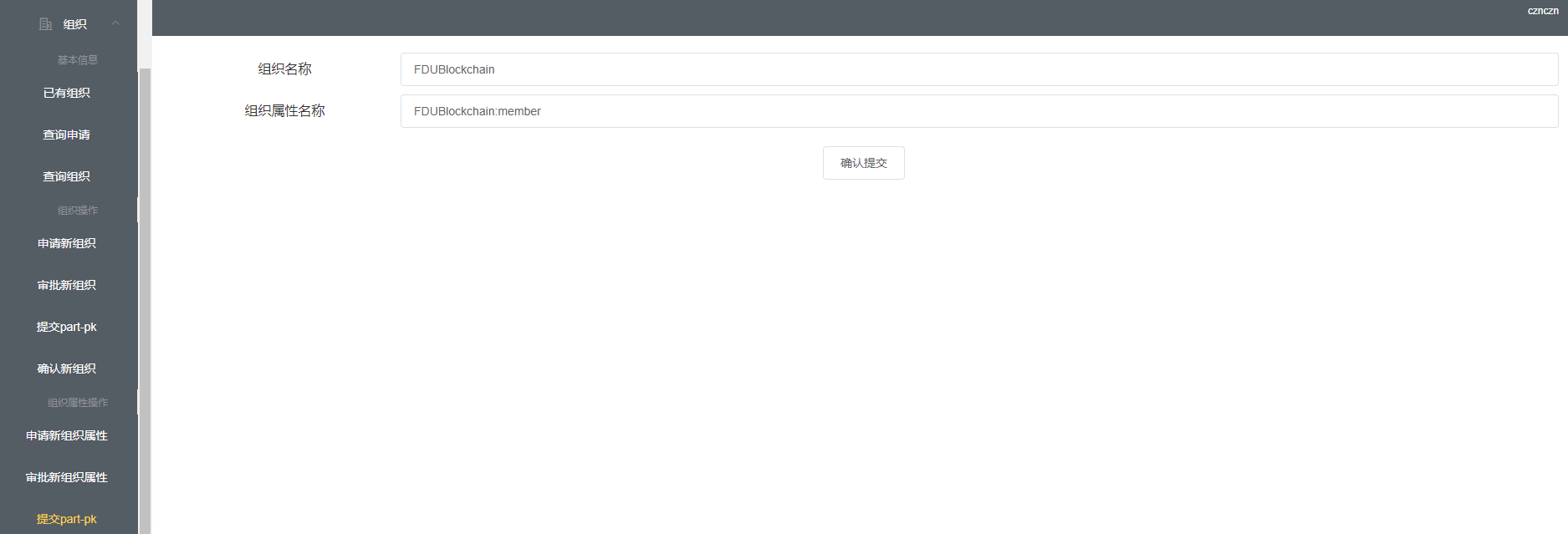


图 6-8 新组织属性的申请、审批与查询



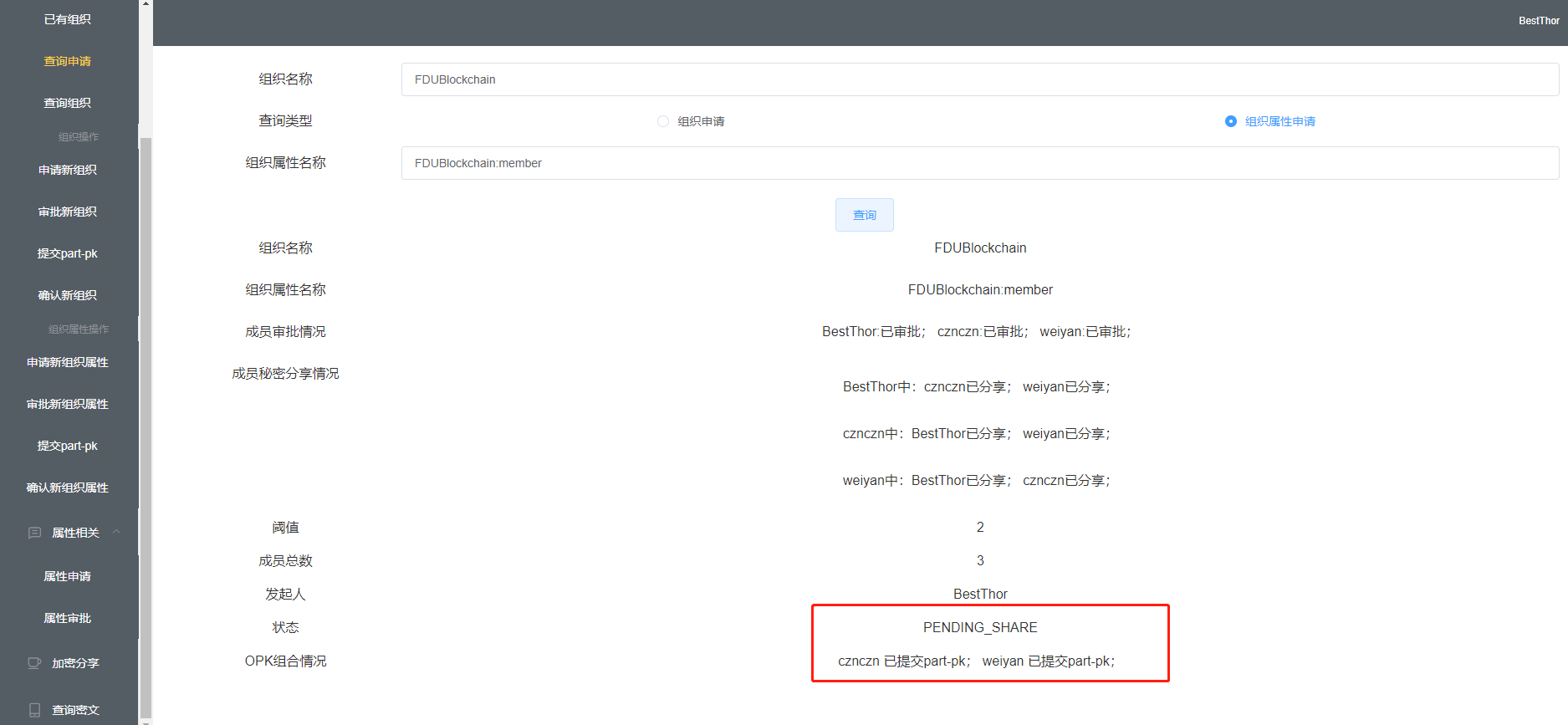
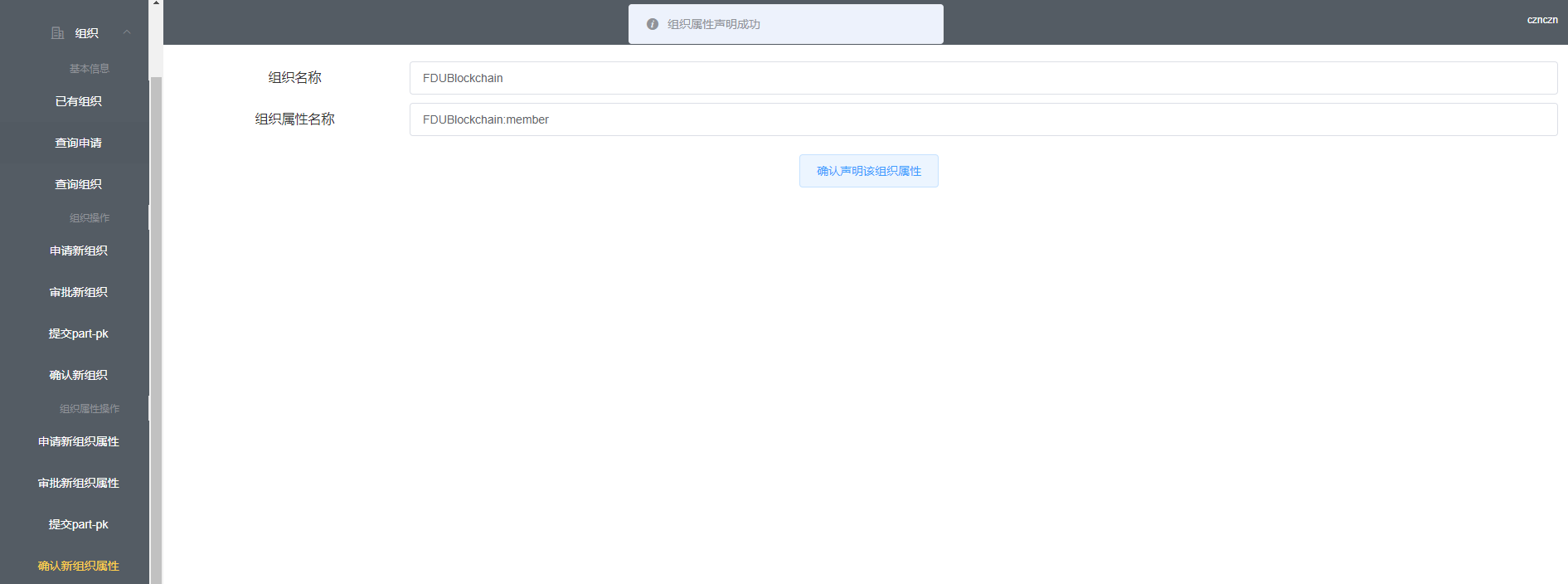


图 6-9 新组织属性提交part-pk与查询



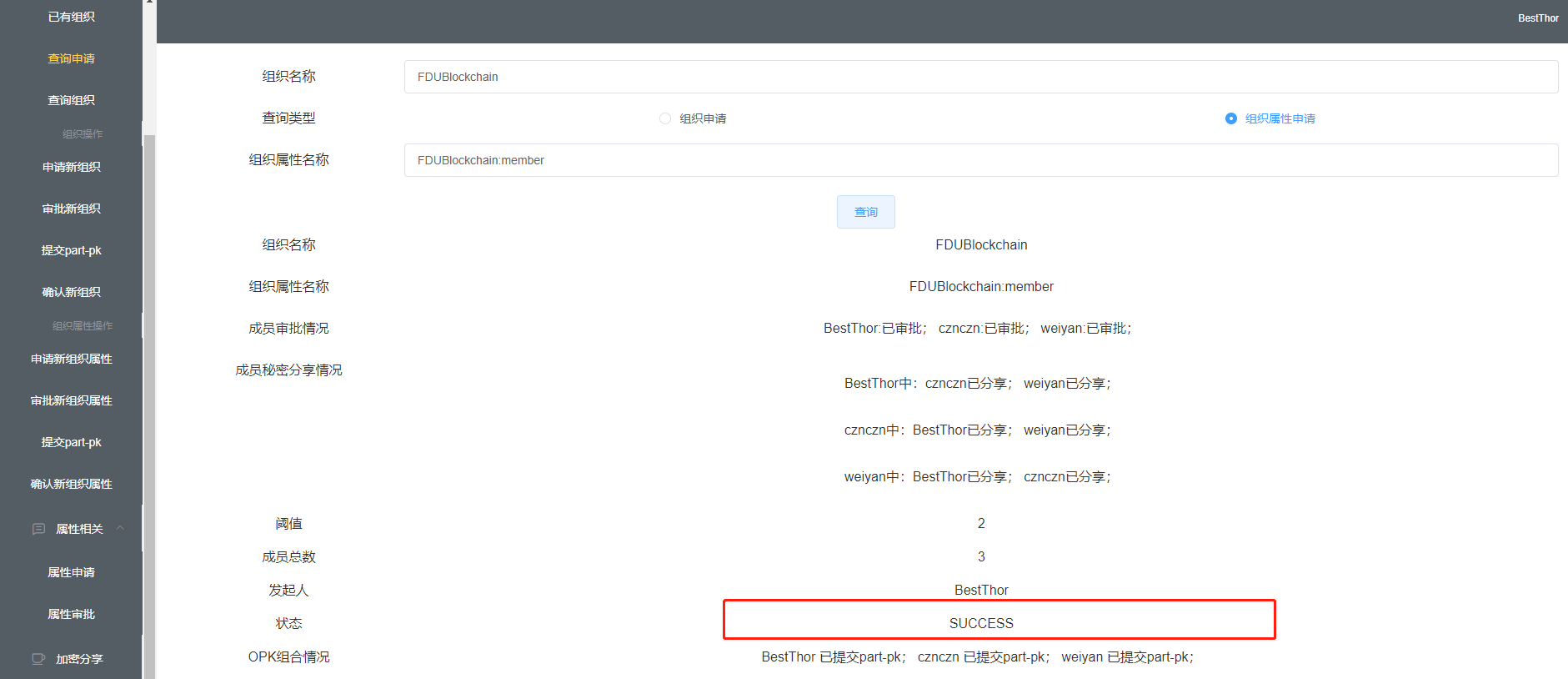


图 6-10 确认新组织属性与查询

### 用户申请属性

用户需要对自己想要的属性提出申请，被申请人需要处理申请信息，正确适当地授予自身属性。下面我们展示用户BestThor对weiyan:friend属性和FDUBlockchain:member属性的公开申请，如图6-11所示：



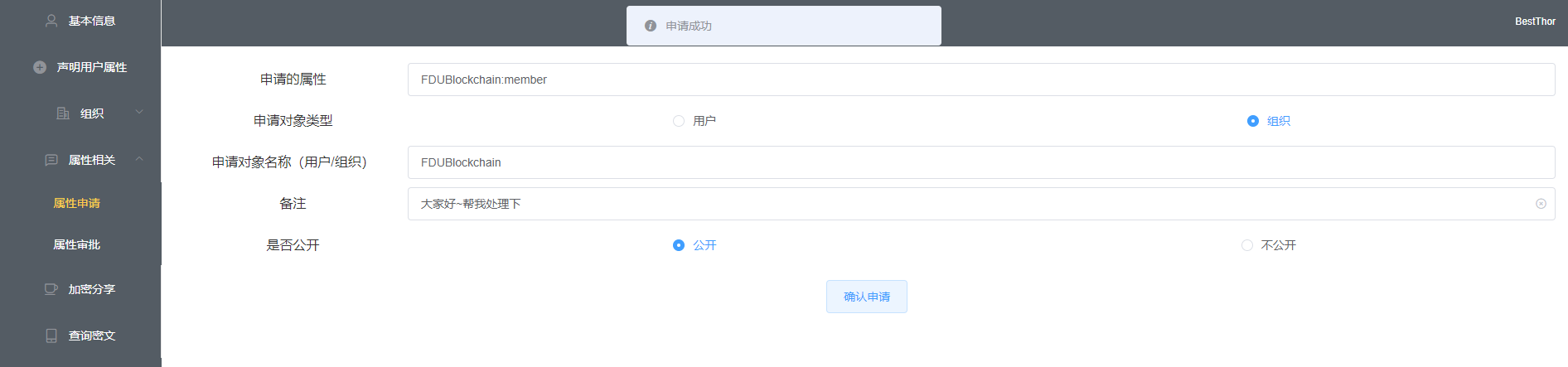
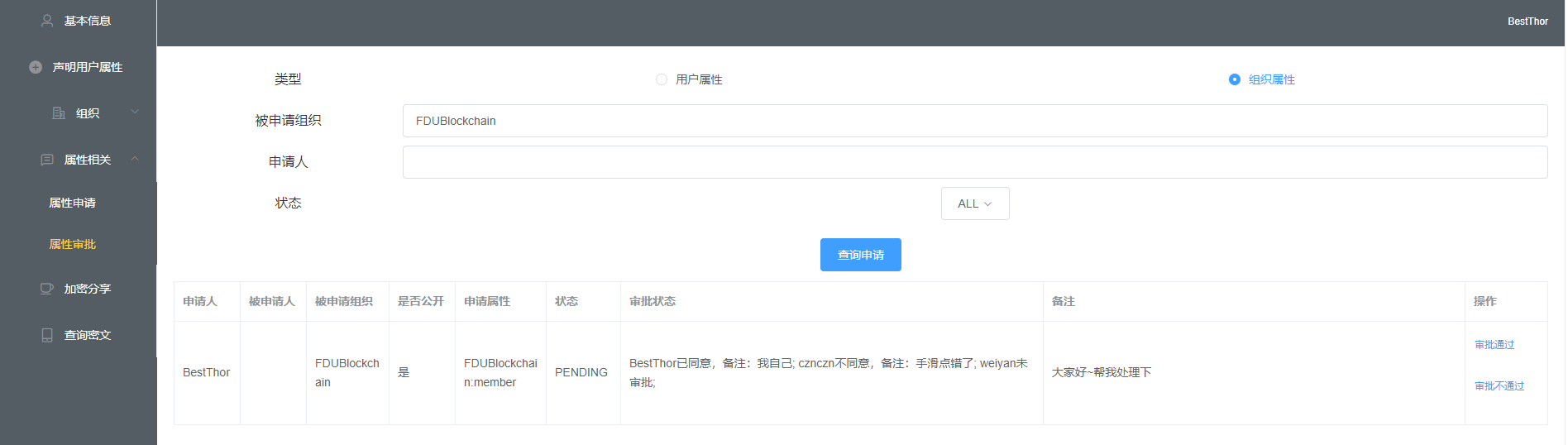


图 6-11 BestThor申请属性

### 用户审批属性

针对用户属性的申请，只需要被申请用户的同意即可，而对于组织属性的申请，则需要t个组织成员的同意（自身申请的话自己也算）。图6-12展示了用户BestThor和weiyan对属性的审批过程，可以看到在cznczn“手滑”点击了审批不通过后，BestThor和weiyan审批通过后也能成功申请到属性：





**图 6-12 用户审批属性界面**

### 组织属性的同步

因为属性审批的过程存在异步的处理（组织），因此需要用户手动地将组织属性审批中包含的部分属性私钥集合成可用的完整属性私钥。见图6-13，在用户基本信息页面最下方有一个同步属性按钮，点击同步属性即可在整条区块链中搜索和自身相关的已经成功的属性申请，并请求合约DABE集成完整属性私钥（用户属性不需要）并存储。

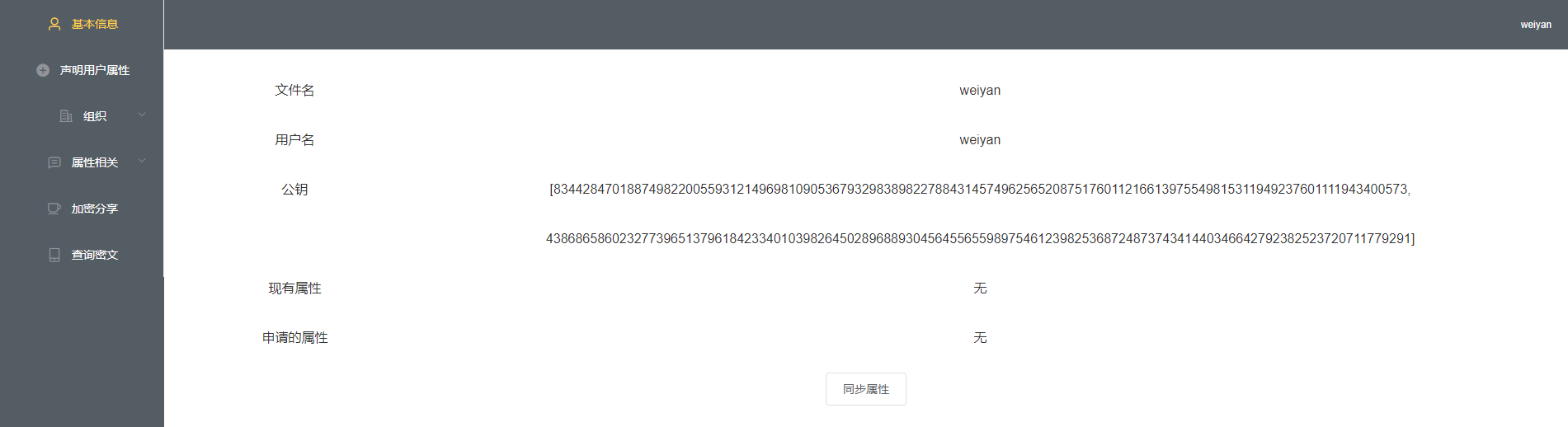


图 6-13 用户基本信息

## 文件共享管理

负责对要共享的文件使用属性密码算法进行加解密。

### 加密分享

用户可以通过标签来分享加密后的信息，并通过灵活的访问控制策略来控制信息分享的人群。以图6-14为例，用户cznczn发布了一条消息，指定自己的朋友或者weiyan朋友中拥有组织属性FDUBlockchain:member的人可以查看：



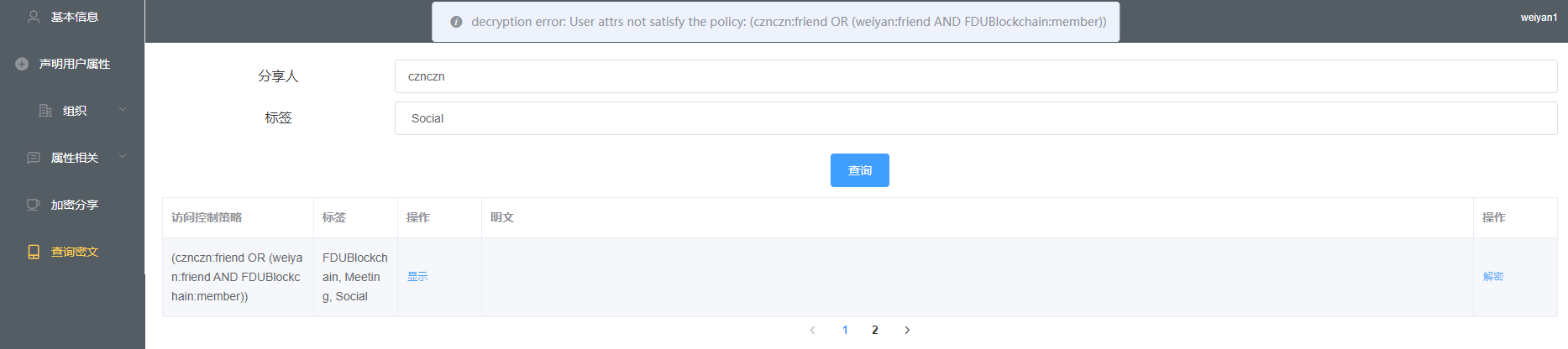
**图 6-14 加密分享**

### 查询解密

以图6-14设置的访问控制策略为例，weiyan和BestThor分别作为两个条件的满足者都可以查看该消息，而用户weiyan1不能看到该消息，如图6-15所示。







**图 6-15 三个不同用户的查询与解密**