StreamNet 用户注册机制的设计与实现

## 一、需求分析

StreamNet 是基于DAG的实现方案IOTA而改造出来的。IOTA本身是条公链，或者说是“无需许可链”，任何用户都可以往网络中发送交易信息。而在我们的生产环境中，需要增加访问控制权限，不是任何人都可以发送交易的，需要经过授权之后才可以发送交易和查询信息。

## 二、方案设计

我们在IOTA的基础上用户访问权限控制。方法是采用“证书”的方式来控制用户。注册中心负责证书的发放，用户需要先注册，得到自己的用户信息和证书，然后在发送交易时附带这些信息；DAG节点能够检查证书的合法性，只有当用户的证书合法时，才会接受用户发送的交易，否则会将交易丢弃掉。

总的流程如下。

1）用户首先需要在注册中心中注册，得到自己的用户信息，如图一所示。



图一 用户注册

2）DAG节点的客户端在启动时需要注册中心的部分信息来验证证书。

用户在发送交易之前需要获取DAG和节点信息。



图二 DAG 结构和节点

3）向某个节点发送请求，节点内会进行权限检查，如何符合要求，才会发送到DAG网络中。



图三 发送交易

## 三、详细设计

1. 用户注册

注册中心的工作机制如下所示：



图四 用户与注册中心

首先，注册中心内部有一对预先生成好的非对称加密的公私钥对：PK 和 SK；

然后，当新用户注册时，注册中心能够生成公私钥对，作为用户的标志返回给用户：PK1 和 SK1；

最后，注册中心还能够给用户的公私钥对做认证：SK(PK1)。这就相当于给用户的证书。

2. DAG 节点的客户端在启动时需要获得注册中心的部分信息来验证用户的证书，具体来说就是注册中心的公钥PK。

3.用户发送交易

用户在发送交易时需要携带自己的证书，并且用分配给自己的公私钥做签名。DAG节点的客户端会对证书进行验证，如果合法，才会继续进行下去。

## 四、接口说明

下面是发送交易和查询信息时具体的接口形式。其中PrivateKey 和 Address 即是由注册中心在用户注册时为用户分配的，Sign就是发放的证书。

1. 发送交易

请求地址：http://127.0.0.1:8000/AddNode

{

"Attester": "192.168.130.2", //证实节点

"Attestee": "192.168.130.3", //被证实节点

"Score": "1", //打分

"PrivateKey": "", //节点私钥, 注册后由权限客户端分配

"Address": "123456", //节点地址，注册后由权限客户端分配

"Sign": "EoZ1BUTx" //权限客户端使用服务器私钥对给注册用户分配的Address做签名

}

2. 查询请求地址：http://127.0.0.1:8000/QueryNodes

{

"period": 1,

"numRank": 100,

"Address": "123456", //节点地址，注册后由权限客户端分配

"Sign": "EoZ1BUTx" //权限客户端使用服务器私钥对给注册用户分配的Address做签名

}