鉴权服务-需求调研文档

# 背景

无

# 需求

应面试官要求, 需要保密, 放github可能泄露, 故而省略不写

# 鉴权方案调研

## Cookie

用户身份信息存储在客户端侧cookies中, 发起请求时由浏览器自动带上以便识别身份

* 优点: 简单快捷易实现
* 缺点:
  + 容易泄露, 且难以防范伪造, 即便加密防不了CSRF
  + 跨域限制

## Session

用户身份信息存在服务端, 客户端只保存session\_id, 一般存在cookies里, 也可以放其他地方, 比如url里, 请求时携带session\_id, 服务端以此查出用户身份

* 优点
  + Session\_id是由服务端随机生成的, 防碰撞, 难以伪造
  + 不受cookie 4kb限制和字符串类型限制, 可以存储更多身份信息和更多类型
  + 如果session\_id不存在cookie里, 也不会受到跨域限制
* 缺点
  + 服务端要付出存储成本和查询成本
  + 如果session\_id放在cookies里, 也还是无法防范CSRF
  + 集群部署时需要保证一致性

## Token

Token = 用户身份信息 + 签名

账号密码登陆后, 服务端签发token给客户端, 和session\_id一样, 客户端可以存在cookies或url等其他地方, 请求时带上

和session关键的不同点在于, token可以解码出原始的用户身份信息, 而不是session那样去查询

* 优点
  + Session的优点token也都有
  + 服务端不需要存数据了, 但需要解码, 属于以计算时间换存储空间
  + Token的信息是自包含的, 因此没有session的集群一致性问题
* 缺点: 签发token后不可回收, 需要依靠超时, 黑名单等额外处理来控制生命周期

## JWT

一套token的具体实现规范

将token规范为header、payload、signature三个部分, 各自有相应的要求

能力不变, 都是些具体细则, 不做赘述

## OAuth2.0

Oauth是授权协议, 是以A网站身份授权B网站登录, 比如豆瓣网可以通过QQ登录

# 鉴权方案选型

题目限定了要实现Token方案, 我们采取JWT规范来实现