系统二期设计

因为银行账户余额的变动是一个非常关键的操作，在真实的环境中还需要加强一下设计和考虑：

1. 确保转账操作的所有步骤（如扣款、入账、日志记录）都有持久化存储，目前系统的日志记录还不够详细，可以进一步详细，并且及时入库，避免因系统故障导致的中间状态丢失。也可以提供问题的追踪查询。而且可以基于日志对所有转账操作进行审计。
2. 数据的加密：
   1. 即时内网的通信中，也需要使用https协议。
   2. 数据库中账户，姓名，余额等信息需要加密进行存储。
   3. Redis的缓存信息需要加密存储，防止被篡改。
3. 身份认证与权限管理
   1. 多因素认证（MFA）：确保账户操作是由真实用户发起的。
   2. 权限控制：只允许合法用户访问特定账户的数据和功能。
   3. 防止权限越界：一些交易的发起需要有合理的权限。
4. 真实转账过程可能非常复杂，需要涉及多个业务的操作，对数据库操作依赖很大，如果突然流量过大，可能击穿数据库，造成数据库不堪重负，从而影响整套交易系统。可以针对一套机制，当流量超过一定的限额时，可以开启限流机制，而且适当将一些流量采用消息队列（如 Kafka、RabbitMQ）的形式进行处理，逐步的释放给数据库，降低数据库的压力。
5. 如果账户数量巨大，单库已经放不下，需要考虑根据账户特性进行分库分表，提升数据库的并发能力。例如根据省来分库。
6. 有的转账业务是不是可以考虑异步通知用户转账结果，减少操作延迟。配合交易结果查询。