Message结构

User结构

AES中的SecretKey

RAS中的KeyPair

客户端生成的Message中的数据和服务器端的KeyPair中的公钥是公共数据，不需要加密保护

User中的数据和和服务器端的KeyPair中的私钥是隐私数据，需要进行加密保护

服务器端公开自己的加密公钥，客户端公开签名公钥。

数字签名内容是：用户名：信息摘要

客户端加密流程：（客户端已经获得服务器端的加密公钥）

1. 数据分类，数据分为隐私数据PrivateMessage、公共数据PublicMessage、AES密钥。
2. 隐私数据PrivateMessage（User中的信息）转化为json字符串PrivateString，对PrivateString进行AES加密获得加密字符串CiperPrivateString（串1）
3. 使用服务器端的加密公钥对客户端的AES密钥进行加密，得到加密后的AES密钥字符串 CipherSecretKey（串2）
4. 把公共数据PublicMessage（TraficMessage中的信息）转为json字符串PublicString（串3）。
5. 使用签名密钥生成一个签名（串4）
6. 串1、串2、串3、串4结构化为一个数据包messageTranfer发送给服务器端。

服务器端解密流程（服务器端已经获得客户端的签名公钥）

1. 把messagePackage解析成四个对应的串（串1、2、3、4）
2. 使用签名公钥进行认证，认证成功进入3，否则丢弃数据（并进行恶意节点判定）
3. 把PublicString转化为jsonObject，获得公共数据。
4. 使用服务器端解密公钥对加密后的AES密钥字符串CipherSecretKey进行解密，获得AES密钥。
5. 使用AES密钥对加密后的隐私数据进行解密，获得隐私数据
6. 数据整合为一个完整的数据包，然后进行后续的数据处理

分类的过程：User类（隐私的），Message类（公共的）

MessageCollect类中包含User user和一个ArrayList<Message> messages，和AES密钥。采集到的数据全部放入ArrayList中。

ClientMessageProcessor对MessageCollect进行解析（例如一段上的均速，采集时采集通过两个点的时间，有两个message，经过解析后，转换为取两个路口中间的中点代替这个路段，时间差解析为速度形成一个message）解析完成后，将MessageCollect数据分为三类，最后加上签名生成messageTranfer。