# 应用层加密解密方案

## 加密解密流程

1. 客户端到认证服务器获取认证（AccessToken）
2. 客户端生成ClientToken，并向服务端申请非对称加密公钥（PublicKey）
3. 服务端通过认证服务器认证客户端的AccessToken
4. 服务端生成非对称加密密钥对（KeyPair），以ClientToken作为Key缓存非对称加密密钥对，返回客户端公钥（PublicKey）
5. 客户端生成对称加密密钥（SecureKey），使用PublicKey加密，将加密后的SecureKey和ClientToken发送到服务端
6. 服务端通过认证服务器认证客户端的AccessToken
7. 服务端使用ClientToken获取密钥对（KeyPair）,使用密钥（PrivateKey）解密被加密的SecureKey，以ClientToken为Key存储SecureKey

#### 名词解释：

1. Client Token：客户端与服务端相处传递时的标识，用于隐藏真正的密钥，以及标识请求。
2. Secure Key：对称加密密钥，由客户端生成，用于客户端与服务端请求的加密解密。
3. Access Token：认证服务器办法的认证令牌
4. KeyPair：非对称加密的密钥对，可以产生公钥（Public Key）和私钥（Private Key）

## 优缺点

#### 缺点

* 应用层做加密解密，对请求速度有影响，延迟程度取决于需要加密请求的数据大小（待测试）
* 框架复杂度变高，客户端框架需要实现对应的加密解密算法和流程
* 测试复杂度变高，服务端接收的时加密数据，解密需要对应的SecureKey，测试时需要构建这些前提条件

#### 优点

* 更加安全，在每个客户端一个密钥的基础上还可以做到每个用户一个密钥，所以即使某个客户端或者某个客户被攻破，不会影响到其他客户端或者用户。
* 算法替换更加灵活，加密解密由框架控制，不受证书影响

## 时序图



## 默认加密解密算法

##### 对称加密

AES-256

##### 非对称加密

RSA-1024

##### 签名算法

HMAC-SHA256 加密

RSA-512 数字签名