swarm-client 包结构设计文档

一 目的：

1：为用户提供可靠的云边消息安全传输

2：高度封装，屏蔽底层机制，用户易用。

3：封装系统功能（用户信息监控等）。

4：提供一套代码框架，用户只需实现业务逻辑即可。

二 用户范围：

得到证书授权的用户

三 实现方案

程序设计：

1 : 实例化(解析证书，实例化结构体)

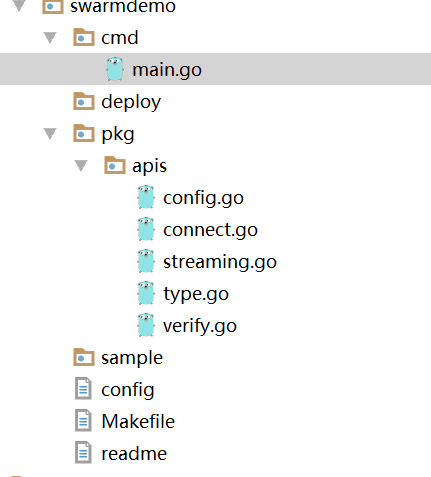
2: connection(连接swarm系统，并返回swarm 连接句柄)

3：调用方法（pub,sub）

四：伪代码演示

1：将用户证书配置文件放置于/etc/swarm/目录下

2：代码结构如下：



pkg包中的apis目录下的type.go存放底层结构

封装了nats提供的消息传输协议与streaming的传输协议。方便后续扩展。

pkg包中的apis目录中的connection存放底层连接swarm 消息系统的逻辑，如消息重连。

pkg包中的apis目录中nats和streaming每个文件都对对相应的系统的协议重新进行封装并添加系统功能.

之所以将nats和streaming进行单独分装是因为它们具备不同的消息属性。如requests-reply模式，只有nats可以提供。而对消息状态的记录只有streaming才有。当然这些属于功能扩展。对用户屏蔽。

3：\*\*\*使用说明

1：swarmclient 函数命名规则

如函数LocalCommonPub与CloudCommonPub

local代表在本节点传递消息

Cloud代表云边消息传递与边边消息传递

Common代表普通消息不会持久化（没有该标识的方法会持久化）

pub代表功能

2：客户端启用必须指定clientid，该标志为客户端唯一标识。

在sub中启动DurableName("my-durable")参数时会与my-durable组成存储的唯一标识。

3：swarmclient 默认config的存储路径为/etc/swarm/config

如用户指定路径可以使用apis.ConfigPathOption(configPath)。

4：pub的输入与输出

输入：1.subject(主题) 2.message（传递的消息）

输出：error 传递成功与否的标志

5：sub的输入与输出

输入：1.context（中断sub与解除订阅）

2.subject（主题）

3：func([]byte) 消息处理函数

4：接收消息规则（可选，不填默认使用durablename(my-durable)）

输出 error

6：unsubscribe （取消订阅，不建议使用，如果持久化存储会丢失数据）

7：close（关闭消息系统连接）