1 相关依赖

1. paho.mqtt.c

代码地址<https://github.com/eclipse/paho.mqtt.c>

Version：1.3.10+

1. paho.mqtt.cpp

代码地址 https://github.com/eclipse/paho.mqtt.cpp

Version：1.2.0

1. openssl

2 安装步骤

1 安装paho.mqtt.c，下载代码进入代码目录

使用cmake安装

1）cmake -Bbuild -H. -DPAHO\_ENABLE\_TESTING=OFF -DPAHO\_BUILD\_STATIC=ON -DPAHO\_WITH\_SSL=ON -DPAHO\_HIGH\_PERFORMANCE=ON

2）cmake --build build/ --target install

3）ldconfig

2 安装paho.mqtt.cpp，下载代码进入代码目录

1）cmake -Bbuild -H. -DPAHO\_BUILD\_STATIC=ON -DPAHO\_BUILD\_DOCUMENTATION=TRUE -DPAHO\_BUILD\_SAMPLES=TRUE

2）cmake --build build/ --target install

3）ldconfig

3 编译MqttConn=

将paho.mqtt.cpp的src下的mqtt文件夹拷贝到/MqttConn/include目录

修改CmakeList.txt指定引入的依赖so文件

target\_link\_libraries(${PROJECT\_NAME} libpaho-mqttpp3.so libpaho-mqtt3cs.so libpaho-mqtt3as.so)

修改CmakeList.txt指定so名称

add\_library(xxx SHARED MQTT\_CONN.cpp)

命令：

1) cmake .

2) make

3 接口说明

1. 创建客户端接口

mqtt::async\_client\* MqqClientConn(std::string serverAddress, std::string port, std::string clientId, std::string keyPath [], void (\*func)(std::string, std::string));

入参：

serverAddress：连接的IP地址

port：连接端口

clientId：客户端ID

keyPath：长度为三的数组，分别存根证书路径、用户证书路径以及密钥

func: 回调函数指针，接收消息后处理的函数

返回值：返回mqtt客户端，可以使用客户端进行消息发送和订阅

1. 消息发送接口

int MqttClientSendMsg(mqtt::async\_client\* client, const std::string& topic, const std::string& msg);

入参：

client： 客户端

topic： 消息发送的topic

msg： 消息发送的内容

返回值 int 0表示成功 1表示失败

1. 消息订阅接口

int MqttClientSubscribeMsg(mqtt::async\_client\* client, const std::string& topic);

入参：c

client：客户端

topic： 要订阅的Topic

返回值 int 0 表示成功 1表示失败

1. 消息取消订阅接口

int MqttClientUnSubscribeMsg(mqtt::async\_client\* client, const std::string& topic);

入参：c

client：客户端

topic： 要取消订阅的Topic

返回值 int 0 表示成功 1表示失败

5．客户端断开接口

int MqttClientDisconnect(mqtt::async\_client \* client);

入参：

client： 要断开连接的客户端

4 其他说明

1. demo代码中的Logger类是一个简单的日志记录类，只有简单的日志打印和记录到文本的功能，缺少日志的管理（日志的大小限制、多大打包存储等功能），所以请使用自己的日志框架进行替换以达到日志的管理功能。
2. demo代码对于错误码的管理只有成功返回0错误返回1，缺少详细的错误码定义，可以使用自己的错误处理机制进行处理，以记录成功或者失败的细节。