

售货柜 MQTT 通讯协议

1. 通讯协议

后台硬件服务器采用 MQTT 服务器。消息封装采用 JSON 格式。

服务器建议安装 mosquitto 1.4.9 或 Apache ActiveMQ 5.x.x

2. 消息主题定义

规则：MQTT 消息主题定义：厂家编码/设备 ID。

厂家编码为 CQAZ，服务器的 ID 编码为 CQAZ01，设备的 ID 采用 IMEI 号。

例如：重庆厂家编码=CQAZ，服务器的 ID= CQAZ01，设备 ID=231253958M25，则

设备的消息主题为 CQAZ/CQAZ01。后台服务器的主题为 CQAZ/231253958M25。

3. 消息格式定义

名称	说明	值对	说明
c	【code】简写，通讯指令	int32	1-4 位数字字符
f	【from】简写，发送方 ID	string	发布方的 ID
t	【to】简写，接收送方 ID	string	接收方的 ID
m	【message】简写，消息内容	string	由 code 码来解析 message
mi	【message id】简写，消息 ID 号，需要应答的消息携带此 id	int32	标识每条消息自动+1

指令 c

指令	说明	消息	说明
c=22	s->c，服务器下发提货指令	m=C&SI	C:货道号 (1--N) SI:销售单号，不能重复
c=23	c->s，设备应答服务器出货执行结果	m= C&SI& R	C:货道号 SI:销售单号，不能重复 R:执行结果状态 (0: 出货成功)

			1: 缺货 2: 电机超时 3: 未检测到掉货 4: 电机短路 5: 销售订单号重复
c=24	s->c, 服务器下发查询货道电机状态指令	m=0	无参数, 设置为 "0"
c=25	c->s, 设备应答服务器查询结果	m=E0,E1,E2,E3,E4,E5,E6...	可以主动上传 E: 故障代码 (0: 无故障 1: 缺货 2: 电机超时 3: 未检测到掉货 4: 电机短路)
c=26	s->c, 服务器下发查询已出货数指令	m=0	无参数, 设置为 "0"
c=27	c->s, 设备应答服务器查询结果	m=N0,N1,N2,N3,N4,N5,N6...	E: 每个通道已出货数量 (0-255)
c=28	s->c, 服务器下发补货复位指令	m=0	参数, 设置为 "0" 下发指令进行补货复位
c=29	c->s, 设备发送补货库存已恢复默认	m=0	补货工人到达现场, 通过按钮操作主动上传。全部通道计数清 0
c=30	s->c, 服务器下查询温度指令	m=0	参数, 设置为 "0"
c=31	c->s, 设备发送温度报告	m=T	查询或每隔 10 分钟上传一次服务器 T: 温度值
c=32	s->c, 服务器下发查询网络指令	m=0	参数, 设置为 "0"
c=33	c->s, 设备应答服务器查询结果	m=rssi&ICCID& CLBS	信号强度 (0—31,99 没信号), -113dBm+ (rssi*2) SIM 卡唯一标识 ICCID GPRS 基站定位
c=35	c->s, 设备心跳包	m=N	心跳累加计数, 30 秒发送一次

4. 测试 json 数据

```
{"c":22,"f":" CQAZ01","t":"866710036447339","m":"1&12843893","mi":2}
```

```
{"c": 23,"f":"866710036447339","t":" CQAZ01","m":"1&12843893&0&0","mi":2}
```

```
{"c":24,"f":" CQAZ01","t":"866710036447339","m":"0","mi":6}
```

```
{"c": 25,"f":"866710036447339","t":" CQAZ01","m":"0,0,0,0,0,0,0,0","mi":6}
```

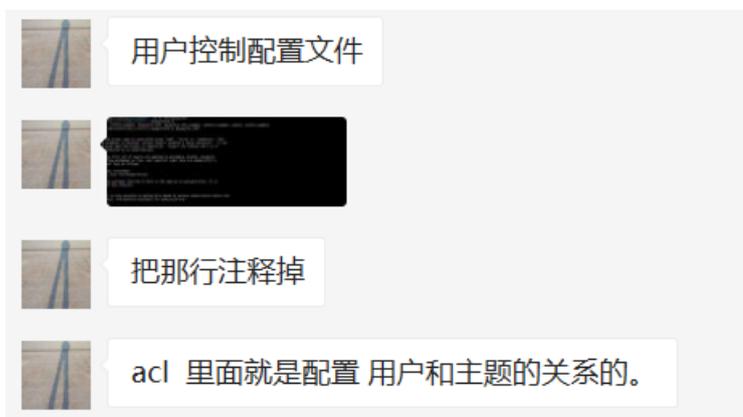
```
{"c": 27,"f":"866710036447339","t":" CQAZ01","m":"0","mi":7}
```

服务器安装注意事项:

mosquitto 1.4.9

配置文件里面。 有个 `acl_file`，解除固定主题

```
~]# cd /etc/mosquitto/
[~/etc/mosquitto]# ls
acl aclfile.example mosquitto.conf mosquitto.conf.example pskfile.example pwfile pwfile.example
[~/etc/mosquitto]# vi mosquitto.conf
#
# The access type is controlled using "read", "write" or "readwrite". This
# parameter is optional (unless <topic> contains a space character) - if not
# given then the access is read/write. <topic> can contain the + or #
# wildcards as in subscriptions.
#
# The first set of topics are applied to anonymous clients, assuming
# allow_anonymous is true. User specific topic ACLs are added after a
# user line as follows:
#
# user <username>
acl_file /etc/mosquitto/acl
#
# The username referred to here is the same as in password_file. It is
# not the clientid.
#
# If it is also possible to define ACLs based on pattern substitution within the
# topic. The patterns available for substitution are:
#
```



Php mqtt 调用

<https://github.com/bluerhinos/phpMQTT?files=1>

V1.02	增加订单重复处理	2018-09-02