

MQTT客户端报文说明

订阅

在星纵LoRaWAN®网关集成的MQTT客户端将它从设备接收的所有数据发布为使用MQTT传输的JSON。因此，如果要从MQTT服务器或其他客户端接收数据，你需要订阅其MQTT主题。

主题定义中可使用+作为单层路径掩码，#作为多层路径掩码。

上行/Uplink data

包含上行链路应用程序有效负载的数据和元数据。

示例：

```
{
  "applicationID": "1",           // 应用ID
  "applicationName": "cloud",    // 应用名称
  "deviceName": "24e1641092176759", // 设备名称
  "devEUI": "24e1641092176759", // 设备EUI
  "rxInfo":
  [{
    "mac": "24e124ffffef021be", // 网关ID
    "rssi": -57,                // 信号强度 (dBm)
    "loRaSNR": 10,              // 信噪比
    "name": "local_gateway",    // 网关名称
    "latitude": 0,              // 网关经度
    "longitude": 0,            // 网关纬度
    "altitude": 0               // 网关海拔
  }],
  "txInfo":                      // 节点信息
  {
    "frequency": 868300000,     // 使用频率
    "dataRate":
    {
      "modulation": "LORA",     // LORA调制
      "bandwidth": 125,        // 带宽
      "spreadFactor": 7        // 扩频因子
    },
    "adr": false,              // 设备ADR状态
    "codeRate": "4/5"         // 编码率
  },
  "fCnt": 0,                   // 帧计数
  "fPort": 85,                 // 应用端口
  "data": "AWcAAAJoAA==",     // base64编码 (已解密)
}
```

下行/Downlink data

通过MQTT服务端向单节点下行数据时网关接收的下行主题。

示例：

```
{
  "confirmed": true, // 下行数据包是否为确认包
  "fPort": 10, // 使用的应用端口 (必须大于0)
  "data": "...", // base64编码
}
```

组播下行/Multicast downlink data

通过MQTT服务端向组播下行数据时网关接收的下行主题。

示例：

```
{
  "multicastName": "222222", // 组播的组名
  "fPort": 85, // 使用的应用端口 (必须大于0)
```

```
"data": "... " // base64编码
}
```

注:更详细的组播操作步骤可参考: [星纵网关MQTT对接第三方平台。](#)

入网/Join notification

设备加入网络时发布的事件。请注意,这是在第一个收到的上行链路(数据)帧之后发送的。示例:

```
{
  "applicationID": "1",           // 应用ID
  "applicationName": "cloud",     // 应用名称
  "deviceName": "24e1641092176759", // 设备名称
  "devEUI": "24e1641092176759",  // 设备EUI
  "devAddr": "06df7989"         // 分配的设备地址
}
```

确认/Ack notification

事件在下行链路帧得到节点设备确认后发布。示例:

```
{
  "applicationID": "1",           // 应用ID
  "applicationName": "cloud",     // 应用名称
  "deviceName": "24e1641092176759", // 设备名称
  "devEUI": "24e1641092176759",  // 设备EUI
  "acknowledged": true,          // 此数据帧是否需要确认 (e.g. 超时)
  "fCnt": 1,                    // 下行帧计数
}
```

错误/Error notification

在与有效负载调度或处理相关的错误的情况下发布的事件。例如,当有效载荷超过最大有效载荷大小而无法调度时。示例:

```
{
  "applicationID": "1",           // 应用ID
  "applicationName": "cloud",     // 应用名称
  "deviceName": "24e1641092176759", // 设备名称
  "devEUI": "24e1641092176759",  // 设备EUI
  "error": "...",                // 错误信息
  "fCnt": 1,                    // 出错帧的对应帧计数 (若有)
}
```

发布/Request data

发布的主题:用于平台端发起请求。网关根据该主题接收到的推送内容进行相关配置处理。

示例:

```
{
  "type": "ns-api",              //默认为ns-api用于内部标识请求类型(该选项必填)
  "id": "123",                  //随机码,用于标识eventReponse(该选项必填)
  "method": "GET",              //method方法,不同的接口和方法均不同详见各接口详细说明(该选项必填)
  "url": "/api/urdevices",      //url链接地址,不同的接口和方法均不同详见各接口详细说明(该选项必填)
  "gatewayEUI": "",             //当需要指定网关响应该请求时才需要
  "body": {
    //请求数据部分,当method为"POST"或"PUT"时才需要
    //不同的接口和方法均不同见各接口详细说明
  }
}
```

订阅/Response data

订阅的主题 · 用于平台端接收响应订阅的主题 · 网关收到指令后的回复包将会以该主题进行推送。

示例：

```
{
  "id": "123", //同对应的请求随机码
  "gatewayEUI": "24E124FFFEF5A6A8", //该网关的eui, 用于对不同网关的回复进行区分
  "method": "GET", //同对应的请求方法
  "gatewayEUI": "",
  "url": "/api/urdevices", //同对应的请求url
  "body": {
    //回复数据部分, 不同的接口和方法均不同见各接口详细说明
  }
}
```