lierda

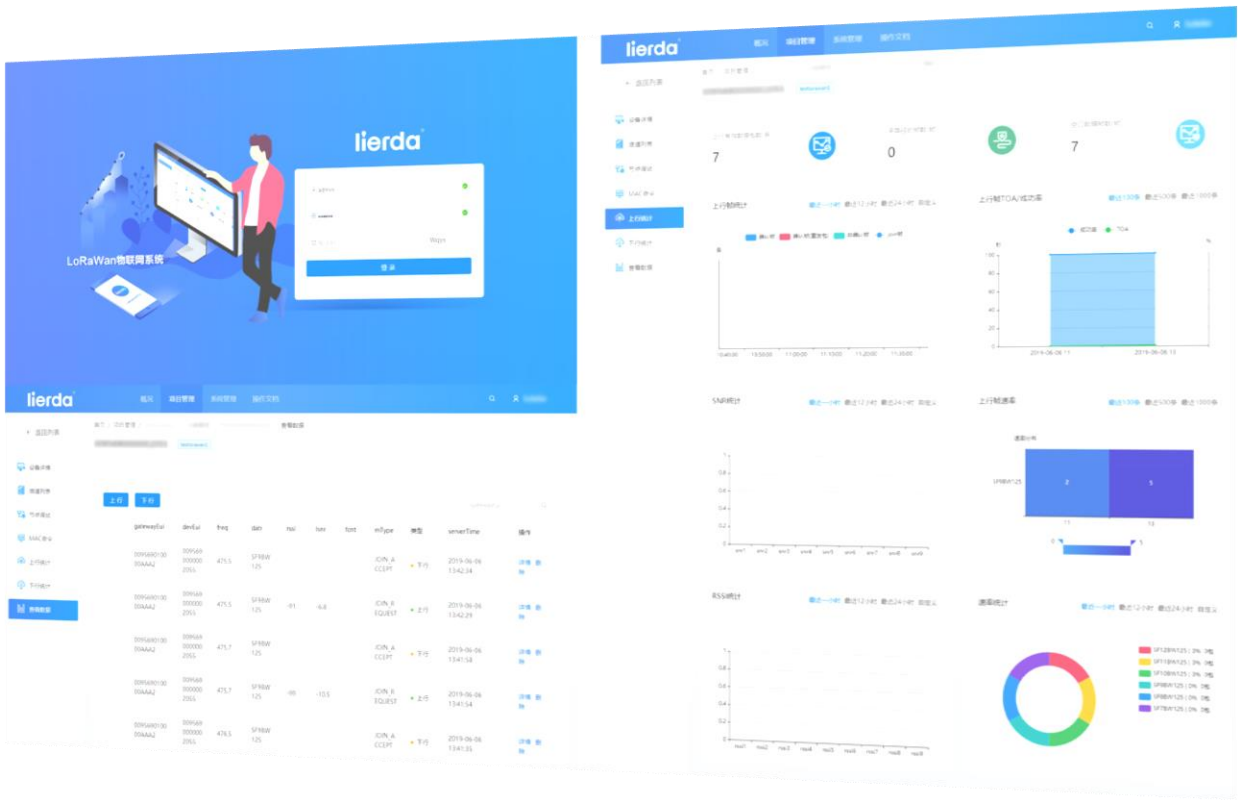
物联网嵌入式解决方案领导者

LoRaWAN NS 3.0

开发者文档

LoRaWAN NS 3.0 是利尔达全新架构的 LoRaWAN Network Server，相比于前代版本，3.0 新增了大量特性，优化了大量网络相关功能，并且修复了 2.0 中的很多 bug。同时，LoRaWAN NS 3.0 支持云部署和本地私有化部署，支持多种架构、负载均衡，用户可以自由选择所需配置。

本文档给出 190613 版本云平台的开发者文档。



前言 浙江利尔达物联网技术有限公司提供该文档内容用以支持其客户的产品设计。客户须按照文档中提供的规范，参数来设计其产品。由于客户操作不当而造成的人身伤害或财产损失，本公司不承担任何责任。在未声明前，利尔达公司有权对该文档进行更新。

文件修订历史

版本	日期	作者	变更描述
1.0.0	2019-06-14	Cokin	创建文档

目录

1 前言

1.1 目的

本文档介绍了应用平台与 LoRaWAN 核心网对接的各部分细节。

1.2 文档内容

本文档包括 LoRaWAN 核心网给应用平台推送数据的内容和方式，应用平台给 LoRaWAN 和核心网下发数据的内容与方式等。

2 推送

2.1 编码方式

LoRaWAN 核心网给应用平台推送数据时采用的数据编码方式为 JSON，字符编码方式为 UTF-8。

2.2 推送方式

LoRaWAN 核心网提供两种数据推送方式，即 HTTP 方式与 MQTT 方式。应用平台可选任何一种，均可收到 LoRaWAN 核心网的推送。**（优先推荐使用 HTTP 方式）**

2.2.1 HTTP 推送

HTTP 推送采用 POST 的方式提交，数据存放到 Body 中，使用 JSON 编码，用 CURL 模拟推送的代码如下：

```
curl -X POST '应用平台提供的用于接受推送的 HTTP 接口地址' -H 'Content-Type: application/json' -d '数据内容（JSON 格式）'
```

2.2.2 MQTT 推送

MQTT 的推送方式要求用户应用平台提供一个支持 MQTT 的消息代理服务器，如 Eclipse Mosquitto、HiveMQ 等，用于中转 LoRaWAN 核心网提供的数据。这样应用平台就可以根据需要去订阅相关的主题，从而收到数据推送。

2.3 推送类型

LoRaWAN 核心网会向应用平台推送 7 种类型的数据，即 **节点（模块）上行数据、网关上线通知、网关离线通知、网关配置通知、网关更新通知、读取网关配置通知、设备下行数据超长包** 通知。

数据格式：LoRaWAN 核心网以 JSON 格式编码推送数据。其中，各字段定义如下：

字段	类型	必填	说明
type	Number	是	指定推送类型 7001：节点上行数据，7004：网关上线通知，7005：网关离线通知，7006：网关配置通知，7007：网关更新通知，7008：读入网

			关配置通知，7009：设备下行超长包通知
content	Object	是	字段定义与 type 相关
projectId	String	是	项目 ID
serverTime	Number	是	事件发生时的 UNIX 时间戳（单位 ms）

2.3.1 类型一：节点上行数据

节点上行数据承载了 LoRa 节点发送的 LoRaWAN 数据包经过 LoRaWAN 核心网的解包与处理后，最终的 FramePayload 数据，即应用数据。节点上行数据不包含 LoRaWAN 相关的协议细节，用户只需要专注应用数据的处理。

2.3.1.1 数据格式

字段	类型	必填	说明
devEui	String	是	节点的唯一标识 devEui
data	String	是	节点上报的应用数据（已经经过数字签名且加密）
timestamp	Number	是	LoRaWAN 核心网推送数据时的 UNIX 时间戳（单位 ms），此值参与上述数字签名运算，目的是与防止重放攻击。

2.3.1.2 推送示例

```
{
    "type": 7001,
    "content": {
        "devEui": "004A77006600173D",
        "data":"t6ZQNlHLOJyEo5ooDxthqpuF3wcjONeNlUtIdrzKCBIuTXh1O7fOxBXZanNJyZkF",
        "timestamp": 1541486638214,
        "fPort": 10
    },
    "projectId": "5b8400ec24583b6eb04c628a",
    "serverTime": 1541486638145
}
```

2.3.2 类型二：网关上线通知

当 LoRaWAN 核心网检测到网关上线时，会向应用平台推送一条网关上线的消息。

2.3.2.1 数据格式

字段	类型	必填	说明
gatewayEui	String	是	网关的唯一标识 gatewayEui

2.3.2.2 推送示例

```
{
    "type": 7004,
    "content": {
        "gatewayEui": "FFFE6CECEBD7E077"
    },
    "projectId": "5b8400ec24583b6eb04c628a",
    "serverTime": 1541486795717
}
```

2.3.3 类型三：网关离线通知

当 LoRaWAN 核心网检测到网关离线时，会向应用平台推送一条网关离线的消息。注意：应用平台收到网关离线通知的时间相对网关实际离线的时间可能会稍有延迟（若干分钟），这一般是由于网关异常关闭与 LoRaWAN 核心网的 TCP 连接造成的，LoRaWAN 核心网需要一段时间去检测网关的异常断开。

2.3.3.1 数据格式

字段	类型	必填	说明
gatewayEui	String	是	网关的唯一标识 gatewayEui

2.3.3.2 推送示例

```
{
    "type": 7004,
    "content": {
        "gatewayEui": "FFFE6CECEBD7E077"
    },
    "projectId": "5b8400ec24583b6eb04c628a",
    "serverTime": 1541486795717
}
```

2.3.4 类型四：网关配置通知

网关进行了配置操作后，会向应用平台推送一条网关配置的消息。

2.3.4.1 数据格式

字段	类型	必填	说明
gatewayEui	String	是	网关的唯一标识 gatewayEui
succeed	Boolean	是	请求是否成功
code	Number	是	请求成功的状态码

2.3.4.2 推送示例

```
{
    "type": 7006,
    "content": {
        "gatewayEui": "FFFE6CECEBD7E077",
        "succeed": true,
        "code": 0
    },
    "projectId": "5b8400ec24583b6eb04c628a",
    "serverTime": 1541486795714
}
```

2.3.5 类型五：网关更新通知

网关进行了更新操作后，会向应用平台推送一条网关更新的消息。

2.3.5.1 数据格式

字段	类型	必填	说明
gatewayEui	String	是	网关的唯一标识 gatewayEui
oldVersion	String	是	固件升级的旧版本
newVersion	String	是	固件升级的新版本
succeed	Boolean	是	请求是否成功
code	Number	是	请求成功的状态码

2.3.5.2 推送示例


```
{
    "type": 7007,
    "content": {
        "gatewayEui": "FFFE6CECEBD7E077",
        "oldVersion": "gw1.1:2.0.05",
        "newVersion": "gw1.1:2.0.06",
        "succeed": true,
        "code": 0
    },
    "projectId": "5b8400ec24583b6eb04c628a",
    "serverTime": 1541486795714
}
```


2.3.6 类型六：网关读入配置通知

网关进行了更新操作后，会向应用平台推送一条网关更新的消息。

2.3.6.1 数据格式

字段	类型	必填	说明
gatewayEui	String	是	网关的唯一标识 gatewayEui
succeed	Boolean	是	请求是否成功
code	Number	是	请求成功的状态码
sx1301Config	Map<String,Object>	是	网关的 SX1301 配置
sx1301Configh	Map<String,Object>	否	网关的 SX1301 配置（16 通道网关）
gatewayConfig	Map<String,Object>	是	网关配置
gatewayFeatures	Map<String,Obejct>	是	网关特性信息

2.3.7 类型七：设备下行数据超长包通知

应用平台在下行的时候如果下发的数据包超长，则 NS 会向应用平台推送一条数据超长不能处理的错误提示信息。

2.3.7.1 数据格式

字段	类型	必填	说明
devEui	String	是	节点的唯一标识 devEui
data	String	是	推送上去的数据（已经经过数字签名且加密）
errorMsg	String	是	下行数据过长，本条数据包未处理

2.3.7.2 推送示例

```
{
    "type": 7009,
    "content": {
        "devEui": "004A77006600173D",
       "data":  "t6ZQNlHLOJyEo5ooDxthqpuF3wcjONeNlUtIdrzKCBIuTXh1O7fOxBXZanNJyZkF",
        "errorMsg": "下行数据过长，本条数据包未处理"
    },
    "projectId": "5b8400ec24583b6eb04c628a",
    "serverTime": 1541486638145
}
```

2.4 校验和解密

对于 HTTP 推送方式中的 7001 推送类型（即节点上行数据）和 7009 推送类型（即下行数据超长包通知），LoRaWAN 核心网会将 content 中承载的应用数据（即 data 字段）进行数字签名且加密。应用平台在收到数据推送后，应按照以下步骤完成应用数据的校验和解密工作。

2.4.1 解密

1. 解析收到的顶层 JSON 对象，获取 Number 类型的 timestamp 字段，并将其与收到数据推送时的 UNIX 时间戳（单位 ms）进行对比，若两者差值太大（如大于 30s），则应丢弃此包，因为这可能是重放攻击产生的数据包。
2. 获取 String 类型的 data 字段，对 data 进行 Base64 解码，得到字节数组 A。
3. 使用"AES/ECB/PKCS5Padding"算法的解密模式，对字节数组 A 进行解密（其中，解密密钥使用对应应用接口的 secretKey），得到字节数组 B。
4. 其中，字节数组 B 的结构如下：

范围	命名	内容

B[0] – B[31]	字节数组 C	数字签名
B[32] – B[B.length]	字节数组 D	应用数据

2.4.2 签名

我们强烈建议在使用最终的应用数据（即字节数组 D）时，应重新计算应用数据的数字签名并与 C 做相等匹配。对于数字签名校验不通过的数据，作丢弃处理，因为这很可能是伪造的数据包。

数字签名的计算方法如下:

1. 将 timeStamp 转化为字节表示，得到字节数组 E。
2. 将 D 与 E 前后拼接，得到字节数组 F,即 F = D | E。
3. 使用“HmacSHA256”算法，对 F 做哈希摘要（其中，签名密钥使用对应接口的 signToken），即得到最终的数字签名。

2.4.3 示例代码

以 java 语言为例子，校验和解密的核心代码如下，此代码依赖于 Apache Commons Codec1.11 或以上版本。

代码见附件包

涉及到字节顺序的均采用 Big-Endian 即大端序，涉及到字节编码均采用 UTF-8，7009 类型的推送只有在下行数据包超长的时候才会推送。

3 下发

3.1 编码方式

应用平台给 LoRaWAN 核心网下发数据时采用的数据编码方式为 JSON，字符编码方式为 UTF-8。

3.2 下发方式

应用平台可以使用两种方式给 LoRaWAN 核心网的节点（也即模块，以下统称节点）下发数据，即被动下发与主动下发。

被动下发是指应用平台在收到 LoRaWAN 核心网的 **HTTP 推送** 时，采用 HTTP Response 的方式给 LoRaWAN 核心网下发数据;

主动下发是指应用平台通过 LoRaWAN 核心网提供的接口给节点（模块）下发数据。

上述两种下发方式，应用平台可根据实际需求任选一种。

3.2.1 方式一：被动下发

被动下发的方式要求应用平台必须以 HTTP 推送的方式接受 LoRaWAN 核心网推送的节点上行数据。当收到 LoRaWAN 核心网的 HTTP 推送后，应用服务器一般以同步的方式处理推送的数据，并根据业务需要决定是否给节点（模块）回复应用数据，回复的数据通过 HTTP Response 的方式下发。

在收到应用平台回复的数据后，LoRaWAN 核心网会根据节点属性中下发的默认选择下发，然后 LoRaWAN 核心网将判断当前时刻是否处于节点（模块）的 RX1 窗口内或者 RX2 窗口内。若是，则最近一包应用数据（由于可能存在并发的数据下发操作，如应用平台在进行被动下发的同时也进行主动下发或节点调试，则多包数据的顺序不能保证。因此 LoRaWAN 核心网采用队列来缓存所有要下发的应用数据，并以先进先出的顺序依次发送给节点）将通过 LoRaWAN 协议层加密和打包，最终发给节点。若 LoRaWAN 核心网收到应用平台回复数据的时间超出节点当前 RX1 窗口的时间或者 RX2 窗口的时间，则数据将追加到 LoRaWAN 核心网内部的先进先出队列中，供下次节点的 RX1 窗口或者 RX2 窗口打开时使用。

上述过程即 LoRaWAN™ Specification 中定义的 Class A 设备的通信时序，主要适用于由节点端发起的一应一答的应用场景，如时间同步，通过应用服务器查询外部信息等。

3.2.2 方式二：主动下发

通过 LoRaWAN 核心网提供的接口，应用平台可以主动给节点下发数据。使用节点默认的通信时序（若

节点为 Class A 设备，则使用 Class A 协议；若节点为 Class C 设备，则使用 Class C 协议）下发。

ClassA 设备在主动下发的时候可以选择三个模式去下发，自动（DEFAULT_MODE）、RX1(RX1_MODE)、RX2(RX2_MODE)，当选择自动的时候，会用默认的下发模式去下发，选择 RX1 则用 RX1 模式下发，选择 RX2 则用 RX2 模式下发。 如果在超出对应的时间，则数据将存入 LoRaWAN 核心网的先进先出队列中，供下次节点的对应窗口打开时使用；

ClassC 设备在主动下发的时候也可以选择三个模式去下发，自动（DEFAULT_MODE）、ClassA 的 RX1(RX1_MODE)、ClassA 的 RX2(RX2_MODE)，如果选择自动的时候，则会立即下发，如果选择 RX1 或者 RX2 模式下发的时候，则会将数据包存到 LoRaWAN 核心网先进先出队列中，等待下一包数据上来之后进行下发。

3.2.2.1 主动下发接口

HTTP 方式	POST
URL	http://云平台 base 地址/api2/v1/lorawan/downlink

data	{
	"data": "string",
	"devEui": "string",
	"fPort": 0,
	"modeEnum": "DEFAULT_MODE",
	"priority": true,
	"timestamp": 0,
	"useClassA": true
	}
	详见下方章节

Response	{
	"code": 0,
	"msg": "string"
	}

3.3 数据格式

3.3.1 被动下发的数据格式

应用平台以 JSON 格式编码要下发的数据。被动下发，HTTP 请求中字段的定义如下：

字段	类型	必填	说明
data	String	否	要下发给节点的应用数据（已经过数字签名且加密）。若为 null，代表

			应用平台不回复（下发）节点，LoRaWAN 核心网将自动处理后续的逻辑（如：当节点的上行数据帧类型为 CONFIRMED_DATA_UP 时，LoRaWAN 核心网将自动给节点回复空包）
timestamp	Number	否	应用平台下发数据时的 UNIX 时间戳（单位 ms），此值参与上述数字签名运算，目的是防止重放攻击。当 data 字段存在时，此值必填。否则 LoRaWAN 核心网不处理此包数据
priority	Boolean	否	默认为 false。若设置为 true，下发的数据将插队到 LoRaWAN 核心网先进先出队列的最前端，使当前数据优先发给节点。这主要是为了应对队列中有数据堆积的情况。在严格的一应一答应用场景中，建议设置此值为 true

3.3.2 主动下发的数据格式

应用平台以 JSON 格式编码要下发的数据。主动下发，HTTP 请求中字段的定义如下：

字段	类型	必填	说明
devEui	String	是	节点的唯一标识 devEui
data	String	是	要下发给节点的应用数据（已经过数字签名且加密）
fPort	Number	是	LoRaWAN™ Specification 中定义的 FPort 字段（现默认是 10）
timestamp	Number	是	应用平台下发数据时的 UNIX 时间戳（单位 ms），此值参与上述数字签名运算，目的是防止重放攻击
useClassA	Boolean	否	默认为 false。若设置为 true，LoRaWAN 核心网将采用 Class A 协议下发此包数据，即数据会追加到 LoRaWAN 核心网内部的先进先出队列中，等到 RX1 窗口打开时才依次发送给节点
priority	Boolean	否	默认为 false。若设置为 true，且 useClassA 为 true，下发的数据将插队到 LoRaWAN 核心网先进先出队列的最前端，使当前数据优先发给节点。这主要是为了应对队列中有数据堆积的情况
modeEnum	String	否	下发的模式，可选择（DEFAULT_MODE、RX1_MODE、RX2_MODE）

3.4 签名和加密

应用平台在下发应用数据之前，应该对应用数据的载荷部分进行数字签名并加密，以防止通信过程中数据被伪造或窃听。具体的操作步骤如下:

1. 假设应用平台需要下发应用数据的载荷内容为字节数组 D。首先，获取当前的 UNIX 时间戳（单位 ms）timestamp，置于 timestamp 字段中。并将 timestamp 转化为字节表示，得到字节数组 E。并将 D 与 E 前后拼接，得到字节数组 F，即 F = D | E。
2. 使用"HmacSHA256"算法，对 F 做哈希摘要（其中，签名密钥使用对应**项目**的 signToken），得到数字签名数组 C。
3. 将 C 与 D 前后拼接，得到字节数组 B，即 B = C | D。
4. 使用"AES/ECB/PKCS5Padding"算法的**加密**模式，对字节数组 B 进行加密（其中，加密密钥使用对应**项目**的 secretKey），得到字节数组 A。若项目选择不加密，则直接将字节数组 B 赋值给字节数组 A。
5. 对字节数组 A 进行 Base64 编码，并置于 data 字段中即可。

3.5 示例代码

以 Java 语言为例，签名和加密的核心代码如下，此代码依赖于 Apache Commons Codec 1.11 或以上版本。

代码见附件包

涉及到字节顺序的均采用 Big-Endian 即大端序，涉及到字节编码均采用 UTF-8。

4 组播

4.1 编码方式

应用平台给 LoRaWAN 核心网组播数据时采用的数据编码方式为 JSON，字符编码方式为 UTF-8。

4.2 下发方式

应用平台可以使用主动下发的方式给 LoRaWAN 核心网的节点（即模块，以下统称节点）下发数据。主动下发是指应用平台用过 LoRaWAN 核心网提供的接口给节点下发数据。

4.2.1 下发接口

HTTP 方式	POST
URL	http://云平台 base 地址/api2/v1/lorawan/multicast
data	{ "addr": "string", "data": "string", "timestamp": 0 } 详见下方章节
Response	{ "code": 0, "msg": "string" }

e.g.

```
{
    "addr":"CE89722B",
    "data": "t6ZQNlHLOJyEo5ooDxthqpuF3wcjONeNlUtIdrzKCBIuTXh1O7fOxBXZanNJyZkF",
    "timestamp": 1541486638214
}
```

4.3 数据格式

应用平台以 JSON 格式编码要下发的数据。主动下发，HTTP 请求中字段的定义如下：

字段	类型	必填	说明

addr	String	是	组播的 addr
data	String	是	要下发给节点的应用数据（已经过数字签名且加密）
timestamp	Number	是	应用平台下发数据时的 UNIX 时间戳（单位 ms），此值将参与上述数字签名运算，目的是防止重放攻击

4.4 签名和加密

应用平台在下发应用数据之前，应该对应用数据的载荷部分进行数字签名并加密，以防止通信过程中数据被伪造或窃听。具体的操作步骤如下：

1. 假设应用平台需要下发应用数据的载荷内容为字节数组 D。首先，获取当前的 UNIX 时间戳（单位 ms）timestamp，置于 timestamp 字段中。并将 timestamp 转化为字节表示，得到字节数组 E。并将 D 与 E 前后拼接，得到字节数组 F，即 F = D | E。
2. 使用"HmacSHA256"算法，对 F 做哈希摘要（其中，签名密钥使用对应项目的 signToken），得到数字签名数组 C。
3. 将 C 与 D 前后拼接，得到字节数组 B，即 B = C | D。
4. 使用"AES/ECB/PKCS5Padding"算法的加密模式，对字节数组 B 进行加密（其中，加密密钥使用对应项目的 secretKey），得到字节数组 A。若项目选择不加密，则直接将字节数组 B 赋值给字节数组 A。
5. 对字节数组 A 进行 Base64 编码，并置于 data 字段中即可。

4.5 示例代码

以 Java 语言为例，签名和加密的核心代码如下，此代码依赖于 Apache Commons Codec 1.11 或以上版本。

代码见附件包

涉及到字节顺序的均采用 Big-Endian 即大端序，涉及到字节编码均采用 UTF-8。