## LoRaWAN 系统调试安装说明-chirpstack

**LoRaWAN是基于Semtech公司开发的一种低功耗局域网无线标准，其目的是为了解决功耗与传输难覆盖距离的矛盾问题。一般情况下，低功耗则[传输距离](https://baike.baidu.com/item/%E4%BC%A0%E8%BE%93%E8%B7%9D%E7%A6%BB/1597413?fromModule=lemma_inlink" \t "_blank)近，高功耗则传输距离远，通过开发出LoRaWAN技术，解决了在同样的功耗条件下比其他无线方式传播的距离更远的技术问题，实现了低功耗和远距离的统一。**

**因此 LoRaWAN被广泛运用于低功耗及私网部署。一个完整的LoRaWAN系统 含以下几个部分 ：**

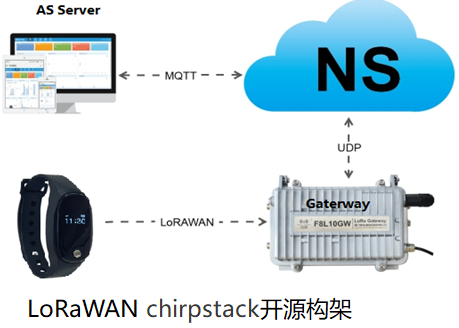
**1: 带LoRaWAN模组的终端设备**

**2：LoRaWAN网关**

**3：LoRaWAN运行平台**

**4：LoRaWAN平台和自身管理平台的集合**

**至少拥有前三部分才能构成一个独立的LoRaWAN系统 。由于世界各国对LoRaWAN的前景看好，更多的企业纷纷加入LoRaWAN设备的开发同时 更多的关于LoRaWAN的 开源架构也纷纷被开发 。这样 大大地降低了人们使用LoRaWAN的门槛和成本。**



**这份试行文档 将对整个系统的部署 实施 及故障的排除做一些简要的说明。目的就是 希望大家少走弯路 。更快更方便的部署LoRaWAN系统。**

**文档将对 网关安装及配置 ，LoRaWAN平台的安装及配置 。LoRaWAN平台和自身管理系统的对接。故障排除 做一个简短说明。**

目录

[LoRaWAN 系统调试安装说明（试行文档） 1](#_Toc139271853)

[一、网关的安装 2](#_Toc139271854)

[二、LoRaWAN平台的安装 4](#_Toc139271855)

[三、添加网关 6](#_Toc139271856)

[四、添加设备 7](#_Toc139271857)

[五、故障排除 10](#_Toc139271858)

[六、LoRaWAN平台和管理系统的对接 11](#_Toc139271859)

### **一、网关的安装**

LoRaWAN网关的安装 比较简单。

我们这里以rhf2s024网关为例。网关拿到手，有以下几个值是我们要关心记下来的（网关ID、SN 、频点 ）其中网关Id，频点都是要和LoRaWAN平台对应的。SN是厂家跟踪使用的。我们如下操作即可：

1：网关二维码旁边的网关类型 ，还有支持的频点是我们所要关心的 根据LoRaWAN国际组织的分配，国内这里是470，香港频点923。另外要记下的是网关ID 即网关的mac地址。

2：网关正常含电源口一个，网口一个以及usb插座1，reset 1，我们只要把网线装上去，电源接上。给网关加电。

3：如何识别接电后的网关IP地址 （私网分配的网址）。这里提供两种办法：

3.1：如果是通过网线接到内网里，你的内网是通过路由器分配的。这个时候可以通过路由器的web界面查看目前在线的设备，这种设备会有rising的标注。同时你也可以通过登录时间长短来判断出网关分配到的IP地址。

3.2：如果不是通过路由器分配的。你可以用一台工作电脑。把工作电脑的IP设置到和网关在同一个网段。然后连接电脑和网关。

通过3.1登录的 ，打开内网分配的IP地址。会跳转到一个认证要求的界面



图示以rhf2s024为例。

进去后，不管型号如何，界面如何不一样。我们在网关关注和修改的地方：

1通信方式， 都是选用packetforwarder。更新的类型也可以选LoRaWANserver。

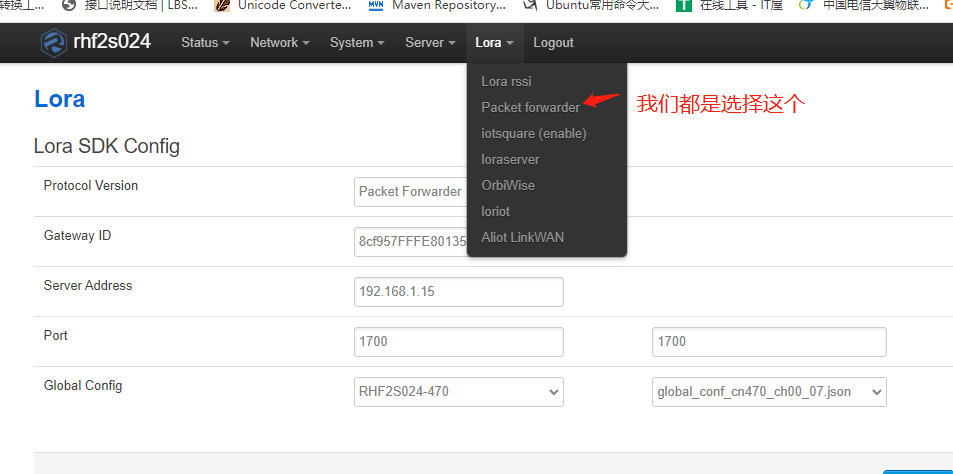
2：packetforwardwe进去后设置 ServerAddress：输入LoRaWAN平台在内网里的地址。Port是 UDP 1700，

Globalconfig选择和网关包装相同的频点和信道设置。（信道设置这部分要和LoRaWAN设备 配对。不了解时 ，可以和我们设备端开发技术协商）。

然后重启网关。这些设置好了。网关这部分的操作就结束了。

题外的话。现在已经有部分网关内嵌NSserver，这是nsserver前推至网关，然后和LoRaWAN平台间通过Mqtt通信。这部分配置也要和LoRaWAN平台端同样配置。同时这样做，也是打破网关只能在私网里部署的约束。扩大部署范围。

在海外安装，要查一下当地设置，如在香港选择915，后面选择as923等。当然这种情况也要引起安装的人员注意。



### **二、**LoRaWAN平台的安装

平台的安装我们推荐使用的的是开源架构chirpstack框架。具体可以参见 https://www.chirpstack.io/ .在这个下面有比较详细的介绍。我们在这一版推荐 V4。

V3和V4版本虽然架构一样，但在新发布的v4版本中。我们发现 V4集成的更好。配置的也更科学。安装时间短。能大大节省开发成本和项目实施周期。

根据我们在以往的检验和客户安装中发现的问题。chirpstack首推使用的是Docker版本,使用docker的好处：

1可以在Windows，Linux和Mac自由安装

2设置参数的地方少。

chirpstack安装的版本的链接见（下载压缩包）:<https://github.com/chirpstack/chirpstack-docker>。

如果没使用过docker版本 ，

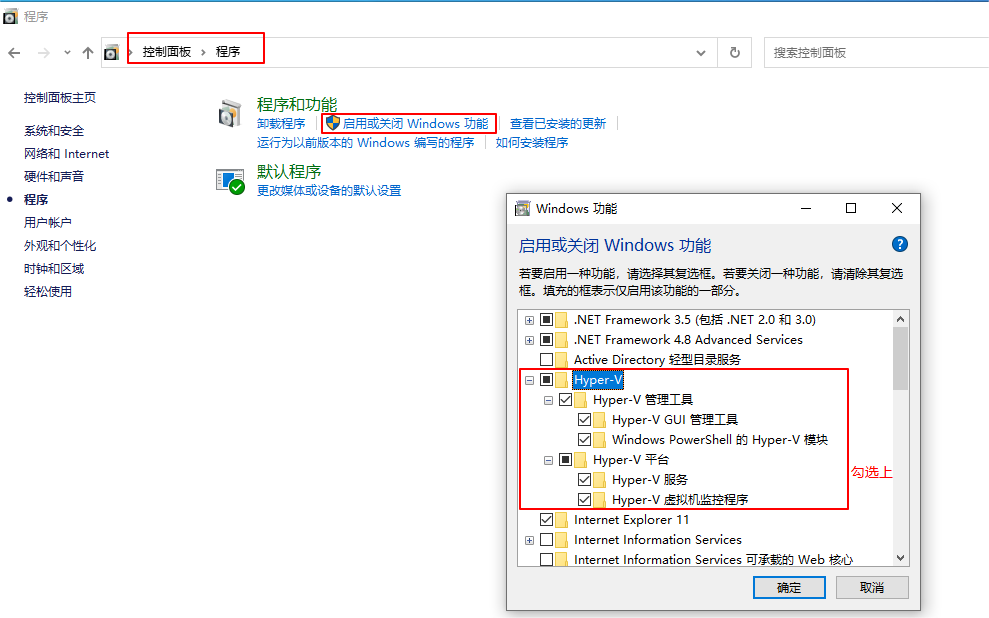
熟悉安装docker和 compose，则可以先参见 <https://docs.docker.com/compose/>。

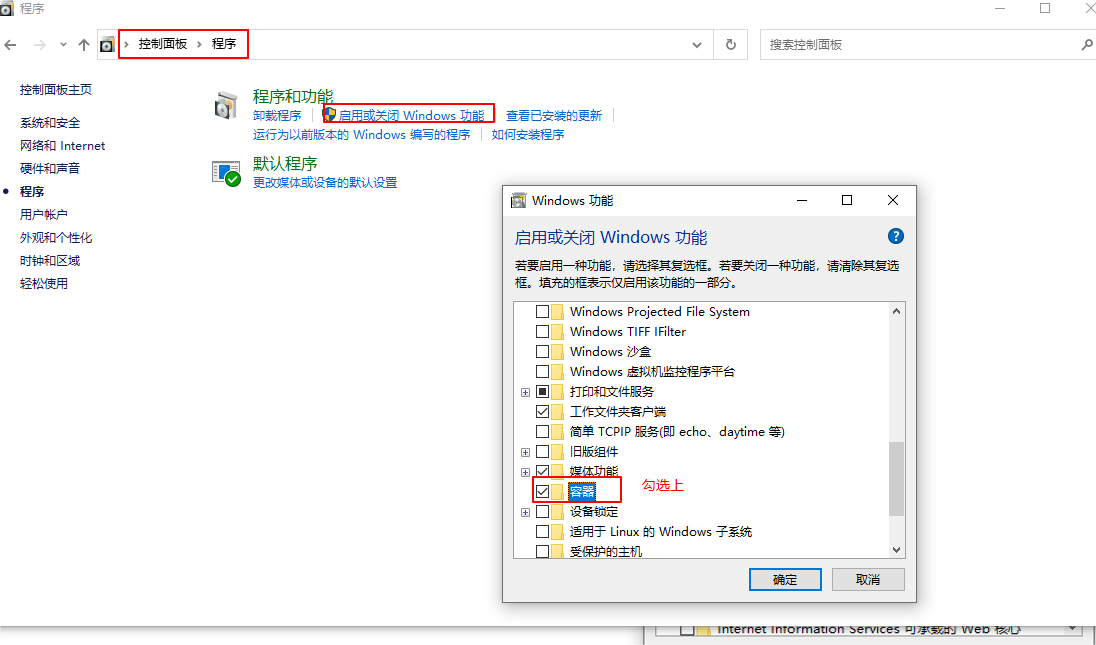
**1.windows安装**

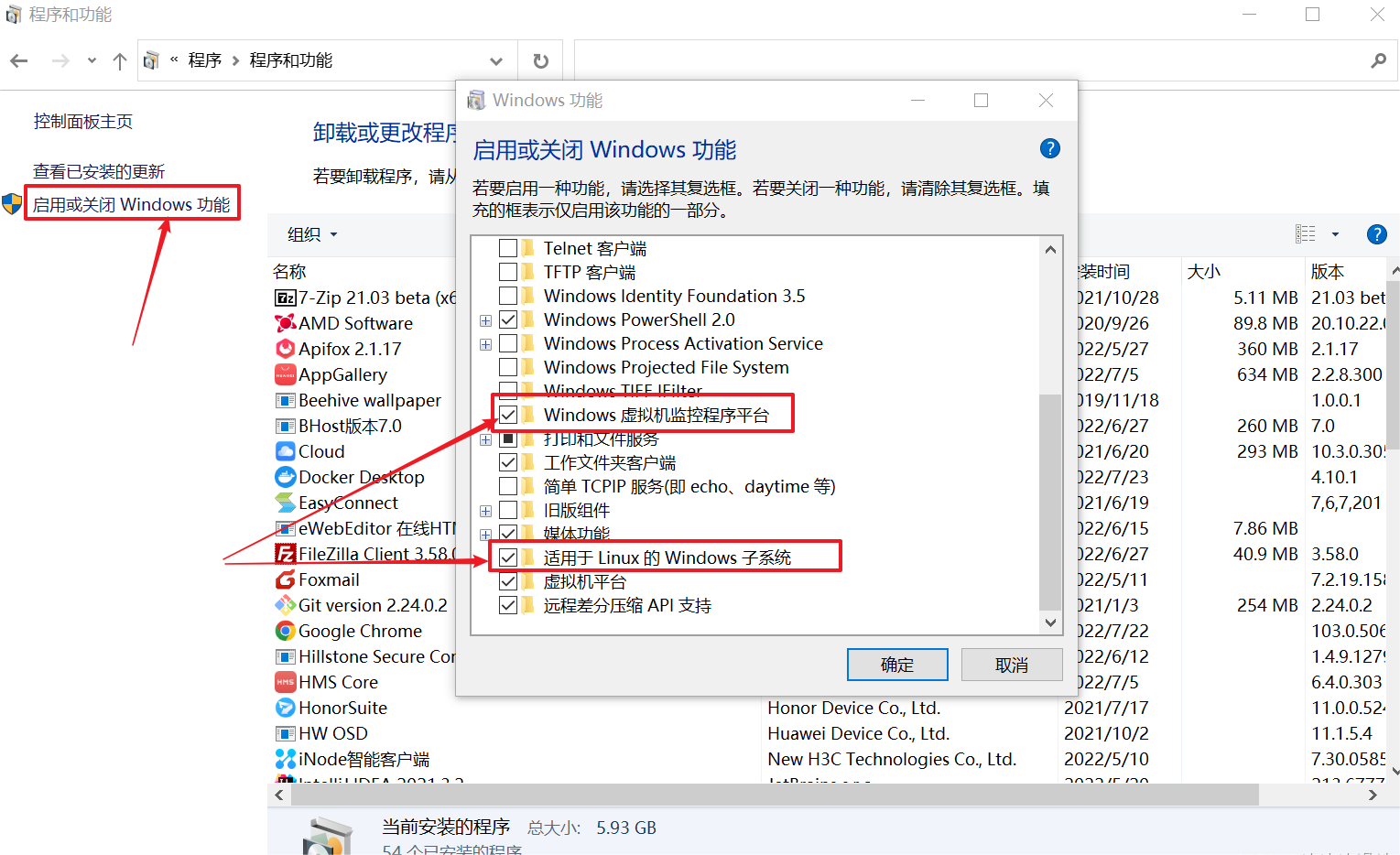
**（1）** 在该地址下<https://docs.docker.com/compose/> ，找到Intall Compose，点击进入安装界面，安装页面在Install下选择对应的操作系统。（操作系统家庭版对docker不适用,操作系统不能是虚拟机），先安装好docker再进行chirpastack安装

**（2）**安装好docker和下载好chirpstackV4版本后，cmd打开chirpstack地址，如下载后chirpstack-docker-master解压的地址为：D:\Lora\chirpstack-docker\chirpstack-docker-master

注意安装docker，操作系统要开启以下图片框选内容







**（3）**安装前，Chirpstack docker这个版本下面的config里有两个要修改的toml文件

1. chirpstack-gateway-bridge.toml:

要修改的地方是节点(直接复制以下代码（需修改）粘贴)：integration.mqtt

event\_topic\_template="eu868/gateway/{{.GatewayID}}/event/{{.EventType}}"

state\_topic\_template="eu868/gateway/{{.GatewayID}}/state/{{.StateType}}"

command\_topic\_template="eu868/gateway/{{.GatewayID}}/command/#"

这个示例是程序中自带的eu868，根据手环手表设备频段改成对应的频点，一般情况下国内只要改成cn470，香港那边是as923.

注意AS923，频点默认不完整，需要在region\_as923.toml里面加入频点，修改后需要重新启动chirpstack

复制以下内容到region\_as923.toml文档结尾：

[[regions.network.extra\_channels]]

frequency=923600000

min\_dr=0

max\_dr=5

[[regions.network.extra\_channels]]

frequency=923800000

min\_dr=0

max\_dr=5

[[regions.network.extra\_channels]]

frequency=924000000

min\_dr=0

max\_dr=5

[[regions.network.extra\_channels]]

frequency=924200000

min\_dr=0

max\_dr=5

[[regions.network.extra\_channels]]

frequency=924400000

min\_dr=0

max\_dr=5

[[regions.network.extra\_channels]]

frequency=924600000

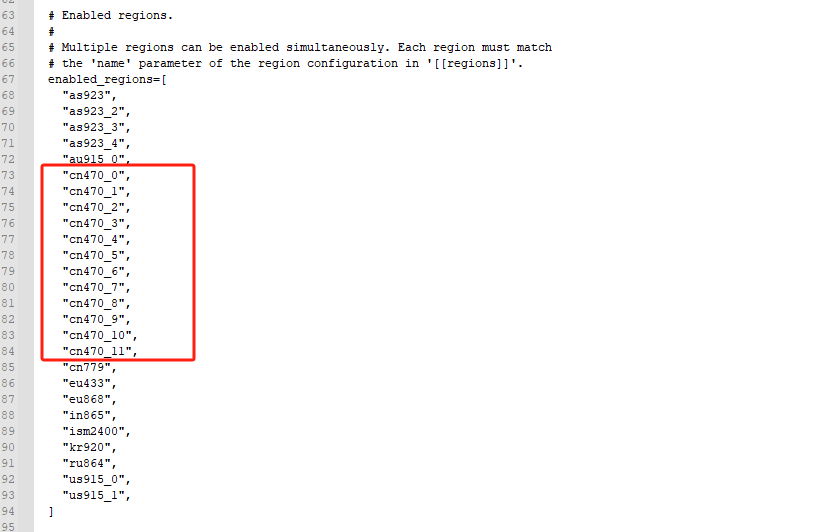
min\_dr=0

max\_dr=5

1. chirpstack.toml:

在 节点 enabled\_regions 下：只要保留本地使用的对应频点即可。其余可删除。

cn470的设备则需要加入红框内容，修改后需要重新启动chirpstack：



至此，在命令行里 启动 docker-compose up 即可。Docker版本就算安装好了。

这个命令行是指下载的文件夹路径那里执行。不是docker的安装目录

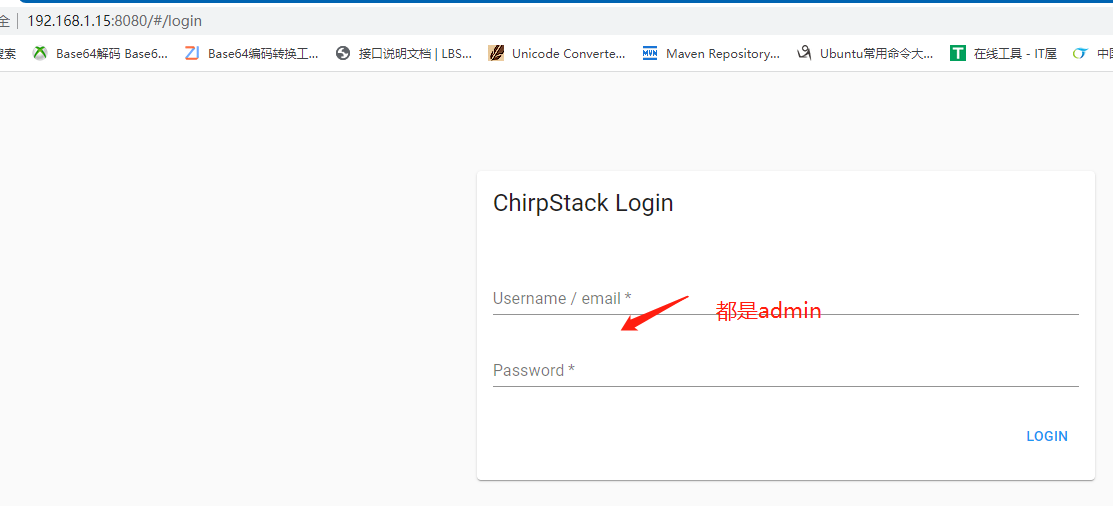
**（4）**在该地址下输入：docker-compose up 开始安装，第一次启动时间较久



**（5）**安装redis 。则需要保证 版本至少是redis5.0以上。

**（6）**假设在前面的操作（包括”一.网关的安装”）都已完成。，你就可以在浏览框里进入 http://localhost:8080

登录用户名admin、 密码 admin。你就可以进去了.



一切正常的话。这个时候就可以在设备的相关devicedata看到数据了，如果没有数据。则要进行故障排查。转第五节 故障排查。

**2.linux安装**

在linux安装chirpstack的其它版本 。

则要保证 系统是 Ubuntu 22.04 LTS 以上或者 Debian 11 (bullseye)

同时要确保下列已经预安装。

sudo apt install \

mosquitto \

mosquitto-clients \

redis-server \

redis-tools \

postgresql

然后在postgresql安装好设置好。这部分具体操作参见 <https://www.chirpstack.io/docs/getting-started/debian-ubuntu.html>

操作postgresql时 还需要对表里相关地方进行设置：

sudo -u postgres psql

-- create role for authentication

create role chirpstack with login password 'chirpstack';

-- create database

create database chirpstack with owner chirpstack;

-- change to chirpstack database

\c chirpstack

-- create pg\_trgm extension

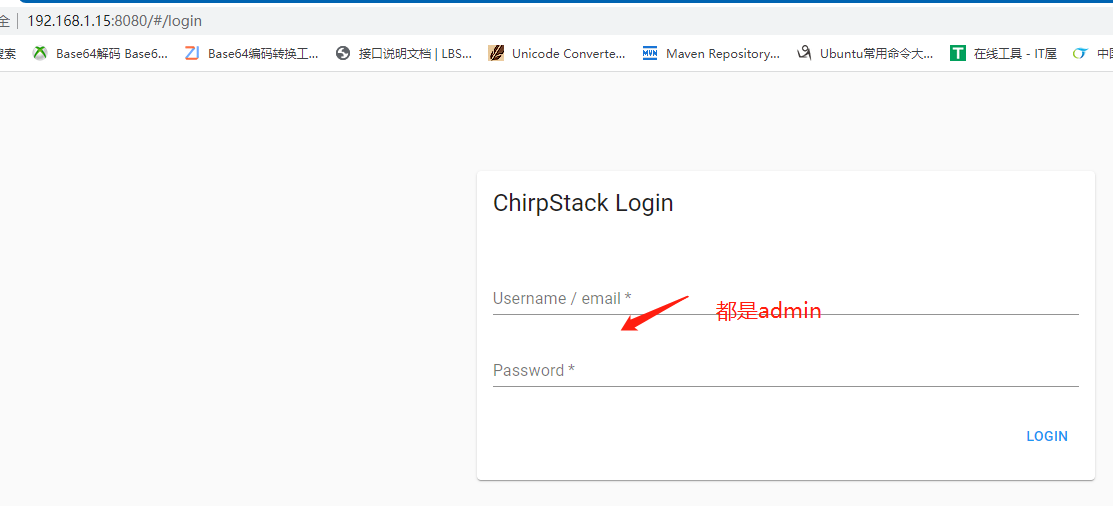
create extension pg\_trgm;

-- exit psql

\q

假设在前面的操作都已完成。，你就可以在浏览框里进入 http://localhost:8080

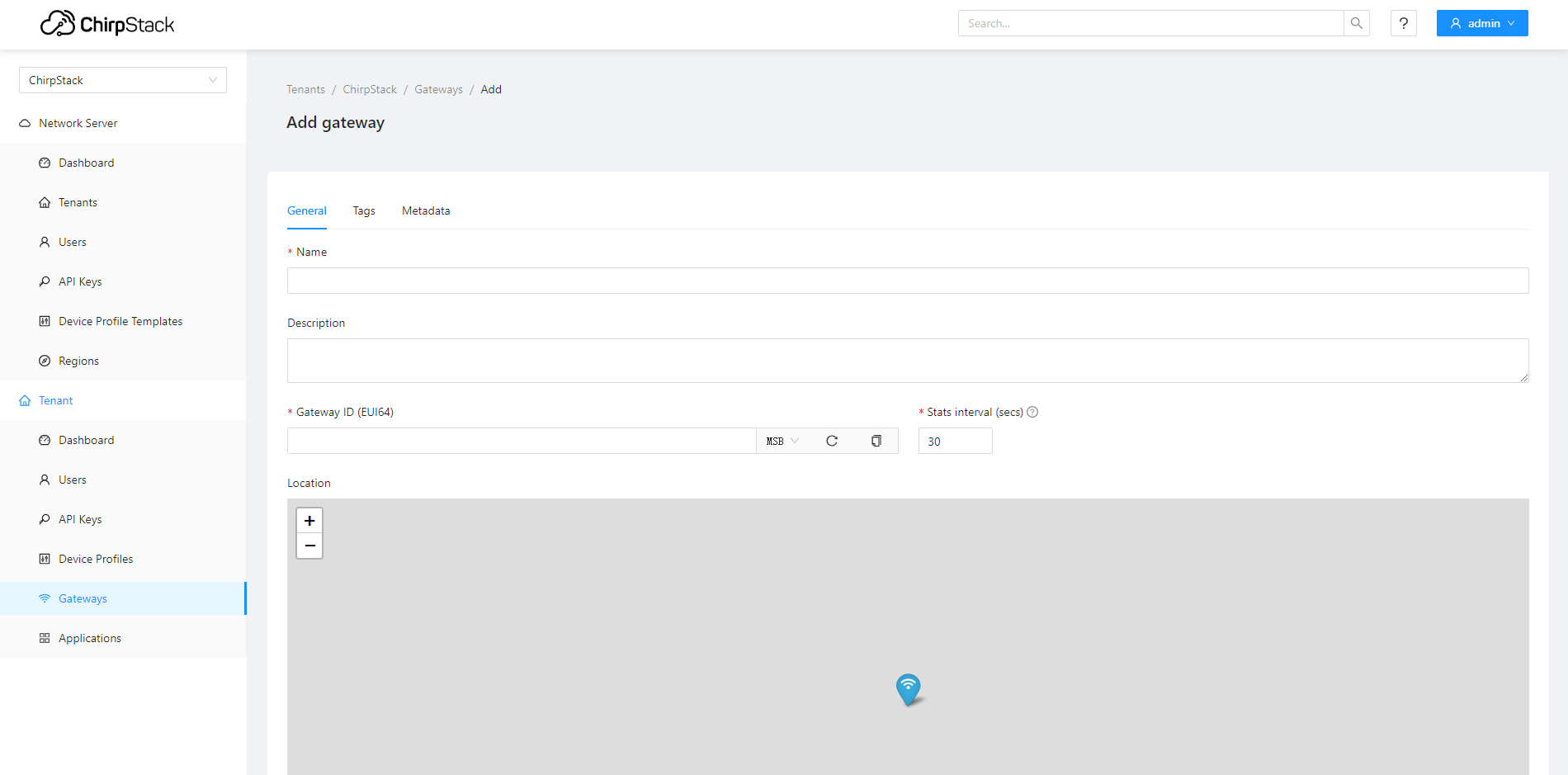
登录用户名admin、 密码 admin。你就可以进去了.



一切正常的话。这个时候就可以在设备的相关devicedata看到数据了，如果没有数据。则要进行故障排查。转第五节 故障排查。

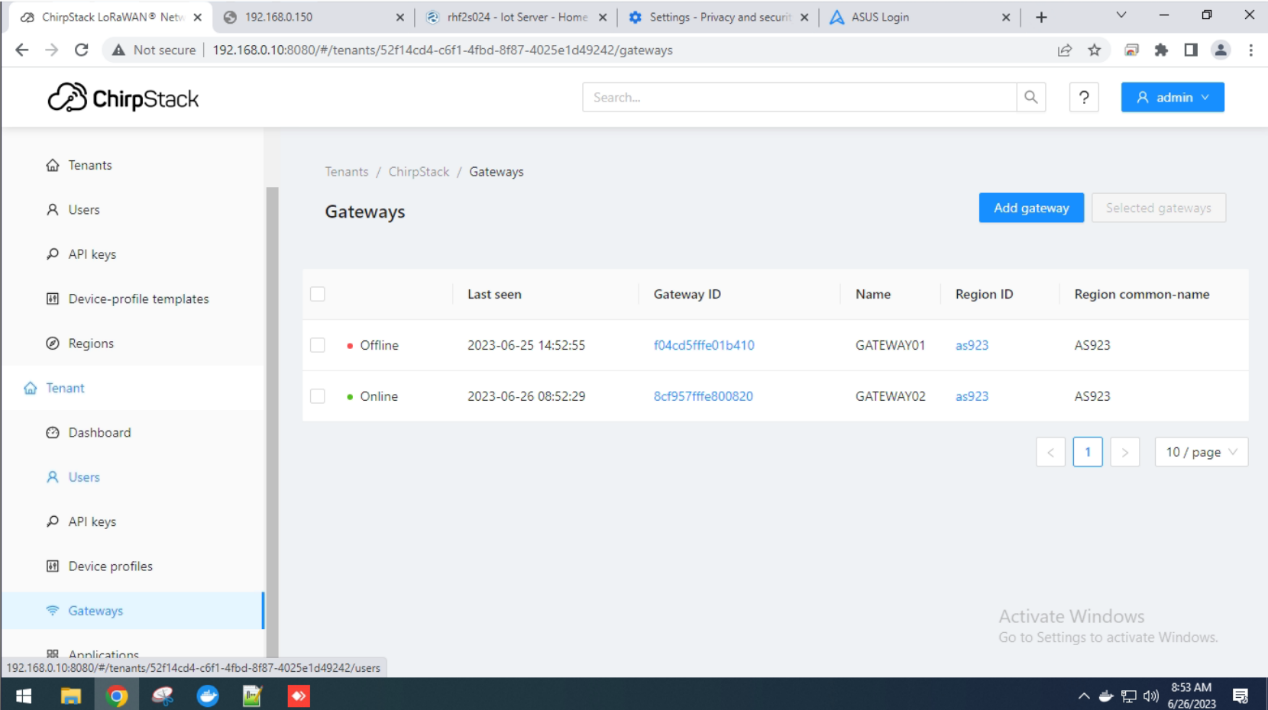
### 三、添加网关

在LoRaWAN平台好了后 我们就可以减去操作添加网关和设备了。在左边选择gateways 进去后选择 add gateway的按钮。



Name自己根据需要命名一下

这里gatewayID就是mac地址中间（6位后）插入FFFE（rising网关如此），添加好后，如果配置正确 后面regionID应该显示 和网关对应的region。如国内是cn470。



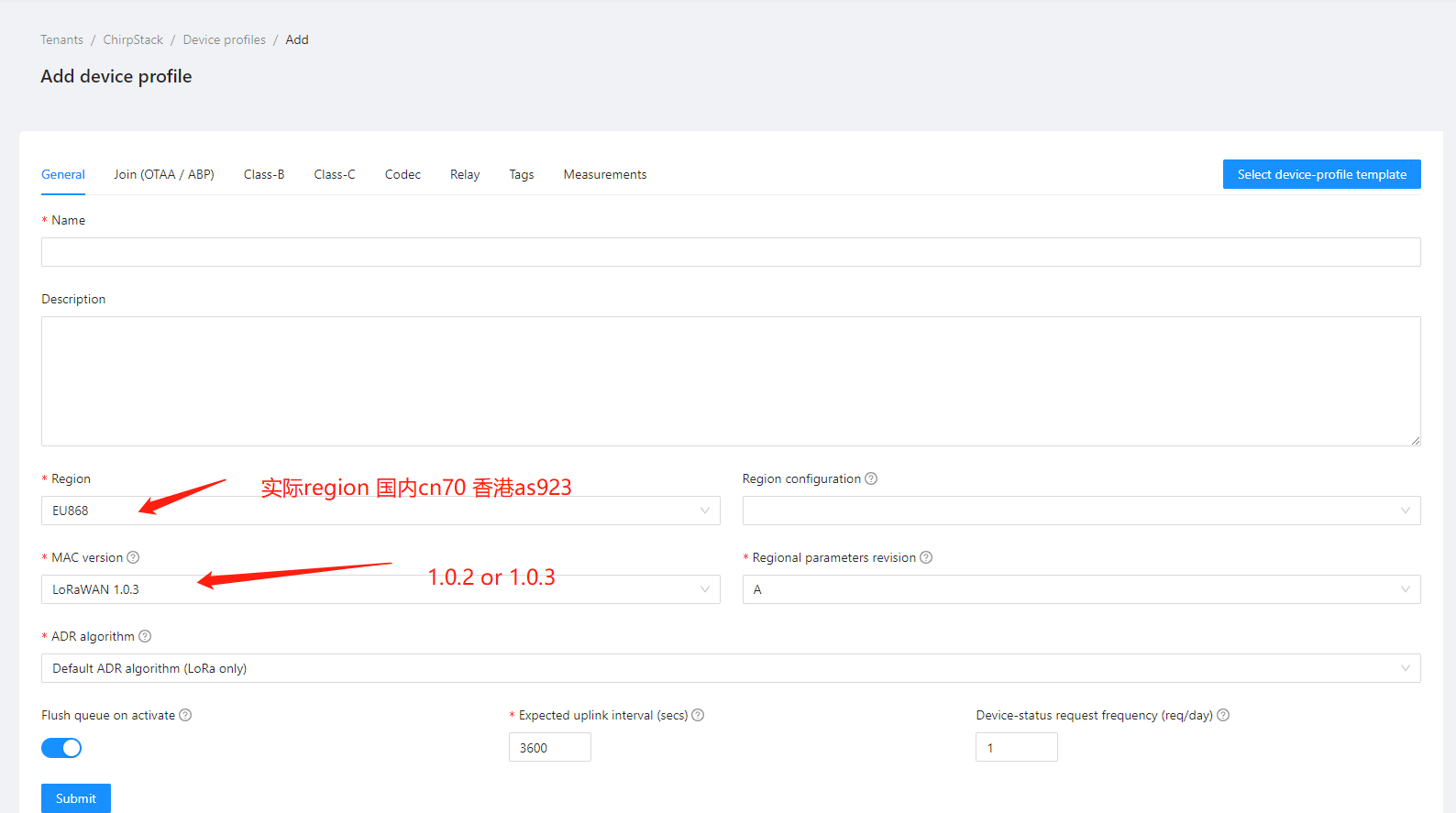
### 四、添加设备

添加设备前，可能需要对设备进行一个分组，分组的原因也是为了设备时更好操作。这部分操作在 Device Profiles 里面操作。

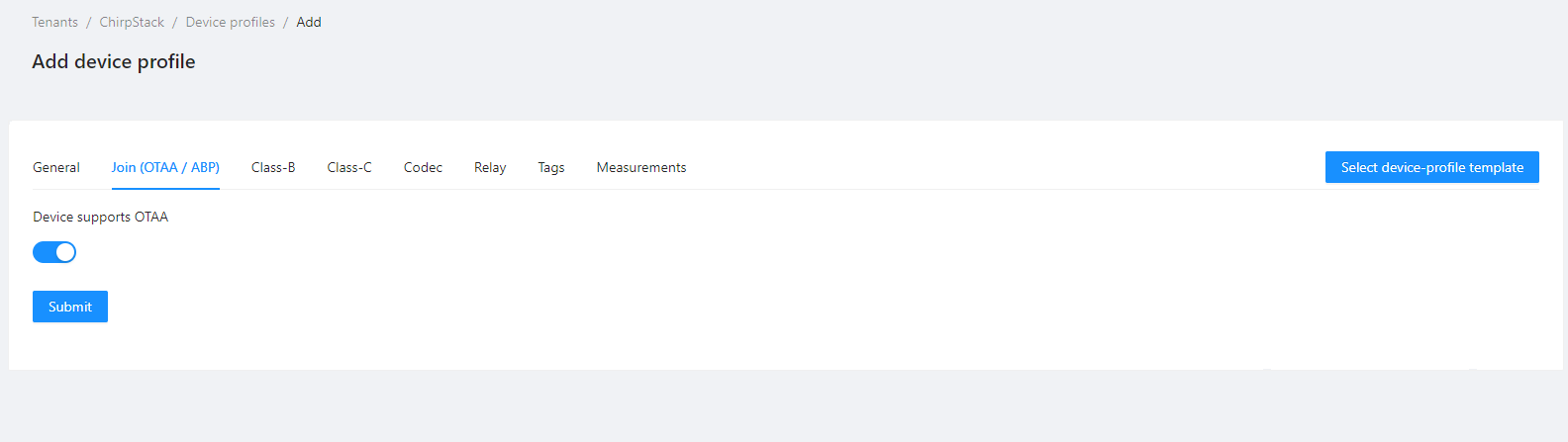
这里主要是上报方式(otaa还是abp)希望详细了解的，可参见 https://www.chirpstack.io/docs/chirpstack/use/device-profiles.html 。

我们正常需要操作的是，如果是otaa 在对应的support Otaa 选中。如果是ABP 就要disable这个功能。然后保存。可以添加许多这种configuration。

点击后 进入如下界面：（修改地方 用红字标注）



然后点击 join(otaa/abp)



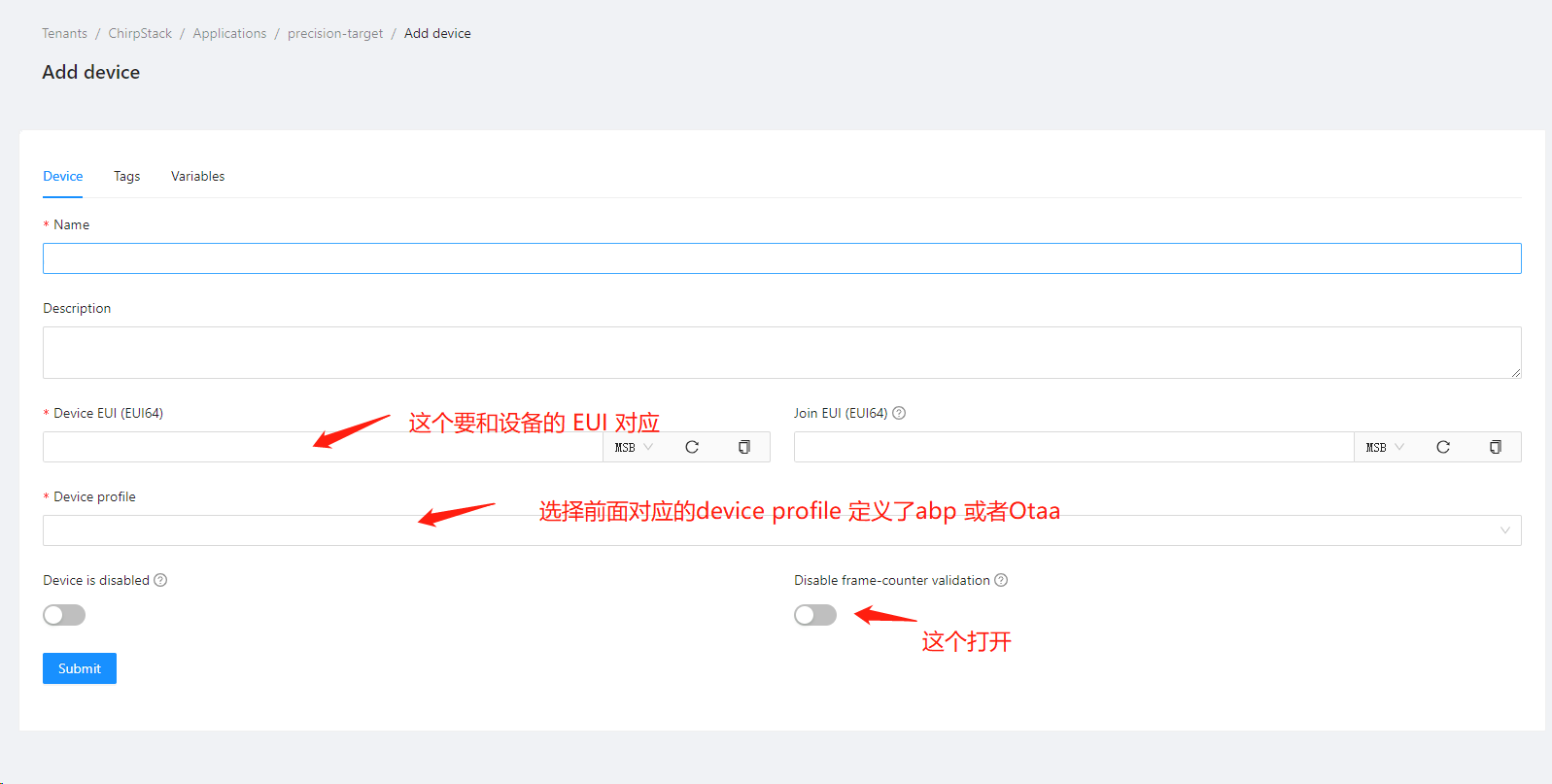
如果是otaa 保持不变 。abp 则disable。

在完成了这部分操作后.我们就可以进入添加设备这个环节了。

我们在界面上是直接看不到设备管理。因此我们需要找到Applications

点击add application ，这里对整个程序影响不大，就不添加图示了。

添加好应用后，在列表中就多出来一个新的应用。双击新添的应用后，进入后 点击右上角的add device 进入下图：



这个里面 name是用于识别的，可以根据自己的特点命名。

Description 是用于说明的。Device Eui是整个系统唯一标识的。

Device profile 是前面定义好的。

这里Submit添加后，在applications设备列表里-点击设备进去，在activation 里：

Otaa 则要输入 appkey

abp 则要输入 network session key 和 application session key

手表加网方式（模组：在那）（wristband register network）：

1：Activation by Personalization (激活方式 OTAA-CLASSA)

Default APPSKEY： 2B7E151628AED2A6ABF7158809CF4F3C

If you need special KEY, please contact with Oviphone

2：另一种ABP入网模式

CAPPSKEY - EF6D6E2503F57AE2FA151CDA87455F18

CNWKSKEY - 2E8C8650B4041672BBB9A399F2DEB427

然后点击active device。

到此，一个LoRaWAN自封闭的系统就算完成了。

下面就可以到设备端的device data 或者LoRaWANwan frames 里看数据了。

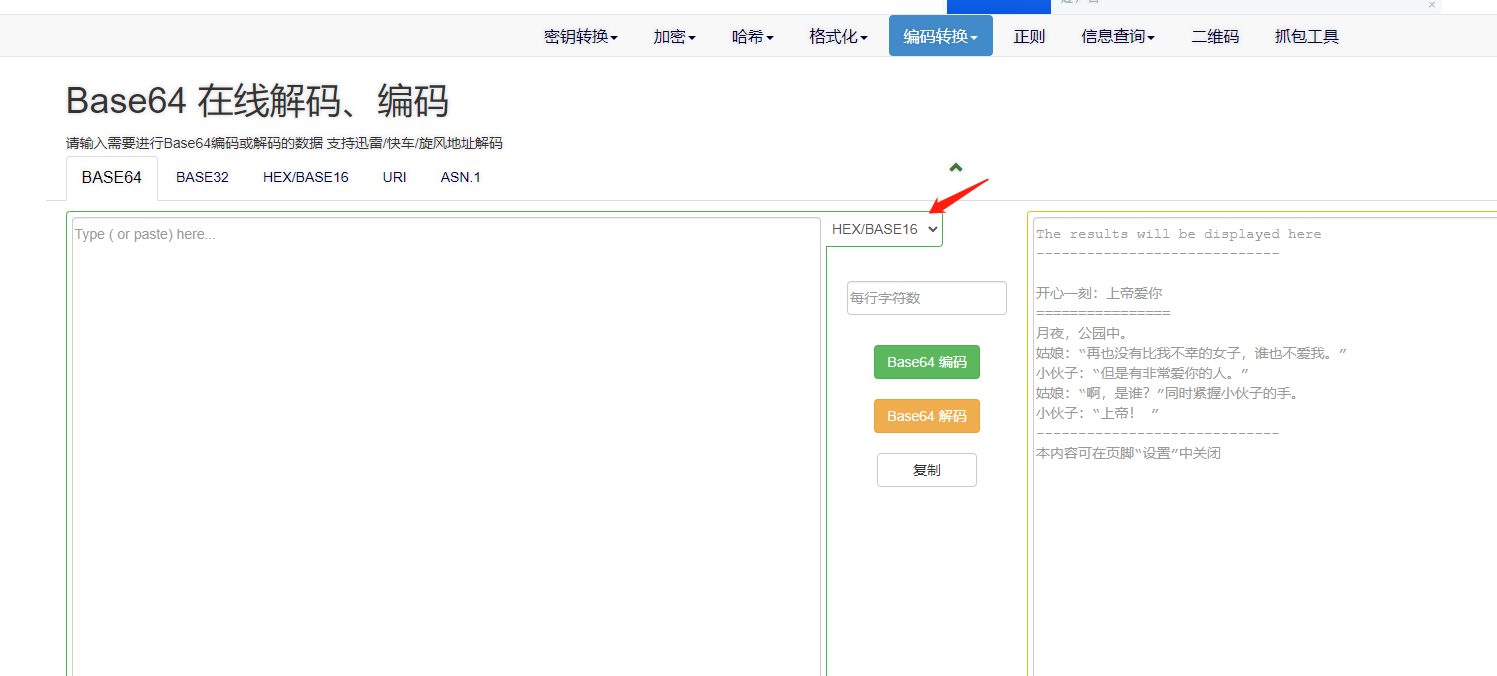
还有一个问题未能讲到。就是device的命令下行检测

我们可以在devices 的detail 下面找到这个地方：



这里下行fport 正常会选择7-10 测试，下行的指令根据我们提供的协议自己手动编写一个（校时指令，下发时间间隔指令，下发通知等）然后用base64编码工具记性编码。然后把编码放在这里下发。选择了confirmed downlink为true。点击右下角按钮。在下面的列表就会显示出来，记住前面的编号。后面可在上报的报文里做跟踪。也可以在设备端直接观察下发报文的效果。

工具链接：<https://the-x.cn/encodings/Base64.aspx>



### 故障排除

注意：操作系统不要装在虚拟机上

1设备没信号：（现象如此）

设备没信号的情况有很多因素。下面罗列几种工现场诊断。

a.网关端的频点和信道的选择和设备端频点和信道不一致所致。

b.网关端和设备端通信未能正确地选择packet forward 的通信（一般情况下 默认选择1700）.也有可能是其他LoRaWAN server。但这时需要确认服务器的该端口正在工作中。

c.网关所处环境不好，这时网关所测底噪往往-120以下。

d.部分模组需要下发校时指令才能注网。早期部分设备存在这种现象。

e.设备端未能及时收到网关下发的响应指令所致。

f．添加网关时 后面的region 和设备预设的不一致：

这个问题是由于平台安装时，未正确配置gateway-bridge.toml的文件引起的，详细参见前面第二节的2.1

2：设备前面通信还正常。后面重启了一下或者断线一下就不正常：

这个是由于添加设备时。未将 frame-count disable 所致。造成两边计数不一致所致。

3: 关于无效mic的问题：

这个在gateway的LoRaWANwan frames的报文中可以见到，多以error报出。反映出的问题是添加设备时的两种情况

a.未能搞清楚设备的上报方式（即Otaa与abp）没分清，造成appkey与appskey搞错了。

b.设备方提供的key值有误或输入有误，比如nskey和appskey输入反了。

4:LoRaWAN 平台界面无数据

其中验证组件问题的检测同设备无信号是同一回事，换句话说，确保前面的每个环节都正常的话。

如果您已经验证了上述组件，那么您应该能够看到发布到MQTT代理的消息。为此，请重复MQTT验证步骤。

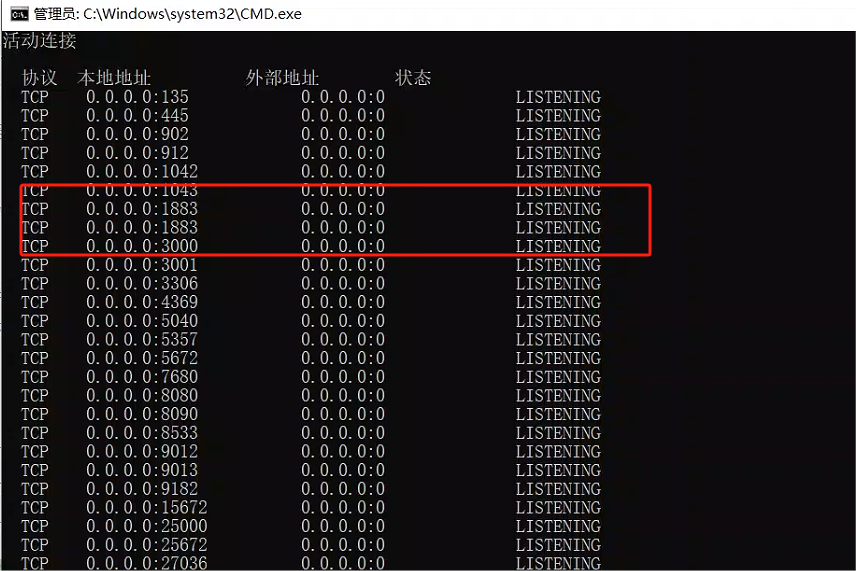
如果ChirpStack网关桥确实记录了它正在发布事件，但您无法从MQTT代理接收数据，则验证您是否配置了阻止ChirpStacck网关桥发布事件和/或阻止MQTT客户端接收事件的ACL。

如果ChirpStack网关桥确实记录了它正在发布事件，但您无法从MQTT代理接收数据，则验证您是否配置了阻止ChirpStacck网关桥发布事件和/或阻止MQTT客户端接收事件的ACL。如果您已验证（上行链路）事件已发布到MQTT代理，您可以验证这些是由ChirpStack接收的。当ChirpStack通过MQTT代理从您的一个网关接收到上行链路帧时，它将打印类似于：

gateway/mqtt: uplink frame received gateway\_id=00800000a00016b6 uplink\_id=be864b60-1392-47a1-98ef-8cd1115b6f05

没有这些，你就要去确认是否配置MQTT ACLs，ChirpStack可以访问已配置的MQTT Topic。还有region是否配齐。

此外，可以检查有多个1883端口开启，则要注意是否有其他应用服务占用，可以将多出这个服务停止即可



如果数据还未能正确显示。你就要查看你硬件是否有其它问题了。

根据网关型号的不同，检索数据包转发器日志的命令可能会有所不同。请参阅ChirpStack网关桥接器的安装，了解特定型号的说明。

5:LoRaWAN 平台界面设备数据帧页面提示连接websocket失败问题

这个是在我们合作伙伴发现的。 其中一个原因及解决：chirpstack官网显示redis2.6可正常使用，但查看后台发现有相关报错，替换成redis7.0后，后台报错解决，websocket连接失败错误解决，手环数据上报正常。

6：设备只有重启后才能收到之前的消息,开机之后再发送就不到是什么原因？

产生这种情况原因：1 设备的信号 2：设备版本是上报优先还是下行优先，这个直接关系到设备通信的种类 就是devices-profile里。设备是支持ClassA 还是ClassC 。这个可以提供设备版本，和我们技术支持核实一下。

7：有些设备可能开机时发出几条信息，然后 后面就不发了。这个时候要了解一下产品应用场景。

### 六、LoRaWAN平台和管理系统的对接

LoRaWAN 平台和管理平台如何对接数据。Chirpstack系统集成了好多处理方式。

MQTT和HTTP是其中两种应应用的比较广泛的两种。主要也是因为它们接口开发比较简单。

MQTT：（首推）

做一个mqtt 客户端 。通过订阅

mosquitto\_sub -t "application/#" –v 就可以获取设备上报的报文。然后按照我们提供的协议解析入库。

通过发布 application/APPLICATION\_ID/device/DEV\_EUI/command/down

mosquitto\_pub -t "application/{APPLICATION\_ID}/device/{dev\_EUI}/command/down"下发

{APPLICATION\_ID}在以前的版本里是数字，新的版本是一串32位字符和数字混合的ID，

{dev\_EUI} 就是device的eui编码。

下发的结构体如下：

{

"devEui": "0102030405060708",

"confirmed": true,

"fPort": 10,

"data": "...." // base64 编码

}

可以下载mqttfx调试工具调试复现一下

HTTP：

http请求都是以Post的方式提供。可以根据自己的需要设置多个url（用逗号隔开）去访问。如果您在与ChirpStack相同的主机上运行此示例，则HTTP集成的端点为<http://localhost:8090>.

具体可以参见 https://www.chirpstack.io/docs/chirpstack/integrations/http.html