

# LPKT001 用户手册

## Lora\_server 安装部署说明

版本 1.00

本文档可以不经通知自行调整

### 文档信息

#### 免责声明

领派智能在不另行通知情况下有权改变产品规格。改变的功能和规格将尽可能在产品特定的勘误表或新版本的文档中发布。建议客户检查领派智能最近更新的这种产品的说明书。

版权©2017 领派智能（无锡）有限公司

# 目 录

1 概述.....	3
2 WIN10 系统下开启 Hyper-V.....	3
3 Hyper-V 下 Ubuntu 系统安装.....	5
4 Ubuntu 系统 docker 环境配置.....	13
5 Lora_server 安装.....	13
6 Lora_server 服务配置.....	16
6.1 将 lorabridge 服务停止.....	16
6.2 网关数据指向配置.....	17
6.3 启动 loraserver.....	17
6.4 配置 lorawanserver.....	18
a)配置网络服务器.....	18
b)添加服务描述.....	19
c)添加设备描述.....	20
d)添加网关.....	21
e)添加应用.....	22
f)添加终端节点设备.....	23
g)通信测试.....	25
7 关于领派智能.....	26

## 1 概述

本文档安装步骤是基于 Ubuntu16.04 系统下的 docker 安装，所以先从 WIN10 系统下 Hyper-V 配置开始，其他虚拟机环境操作类似，请自行选择安装，不再赘述。

## 2 WIN10 系统下开启 Hyper-V

Hyper-V 是微软的一款虚拟化产品，是微软第一个采用类似 VMware 和 Citrix 开源 Xen 一样的基于 hypervisor 的技术。这也意味着微软会更加直接地与市场先行者 VMware 展开竞争。Hyper-V 作为 Windows Server 2008 中最耀眼的部分，自问世以来一直吸引了太多的关注。

Hyper-v 要求:

1. Intel 或者 AMD64 位处理器。
2. Windows Server 2008 R2 及以上(服务器操作系统); Windows 7/8 及以上(桌面操作系统)。
3. 硬件辅助虚拟化。这是在现有的处理器，包括一个虚拟化的微软虚拟化构架。
4. CPU 必须具备硬件的数据执行保护 ( DEP ) 功能，而且该功能必须启动。
5. 内存最低限度为 2GB

win10 自带的 Hyper-V 产品，对 win10 用户来说，即为方便。用户们不在需要多个物理机，而是在一个物理机上实现多个操作系统的运行，极大的提高的物理机的使用效能。

在 win10 系统中 hyper-v 工具不是直接开启的，需要用户手动开启，操作如下：

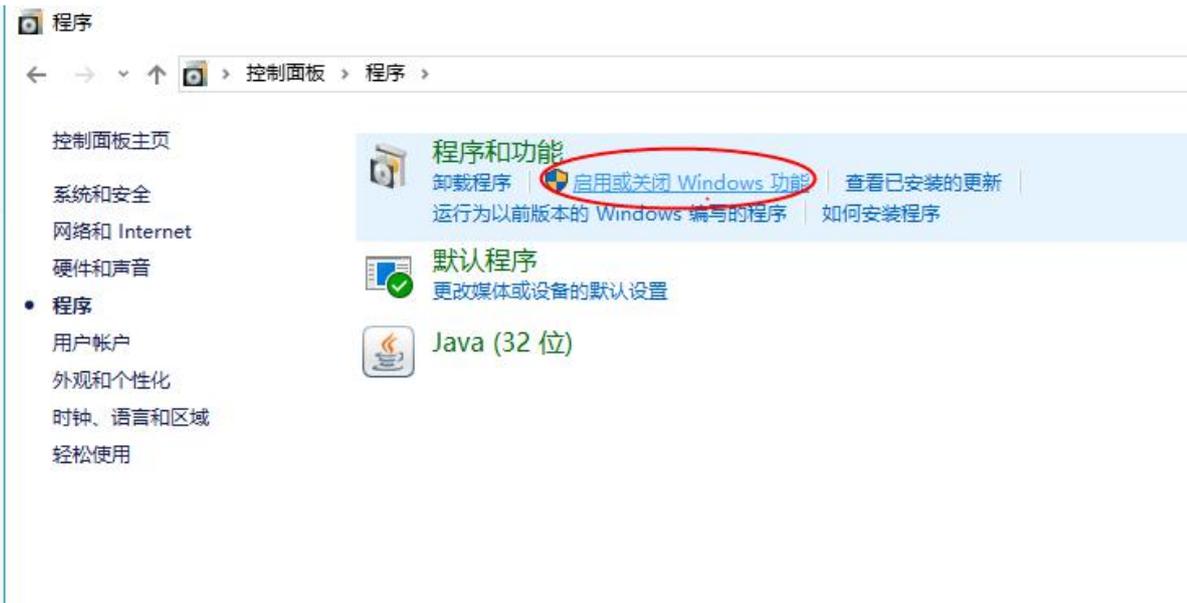
打开控制面板。在 win10 左下角处右键，弹出快捷菜单。

在快捷菜单中选择控制面板，点击后进入控制面板菜单栏。

在控制面板菜单栏中，选择程序。

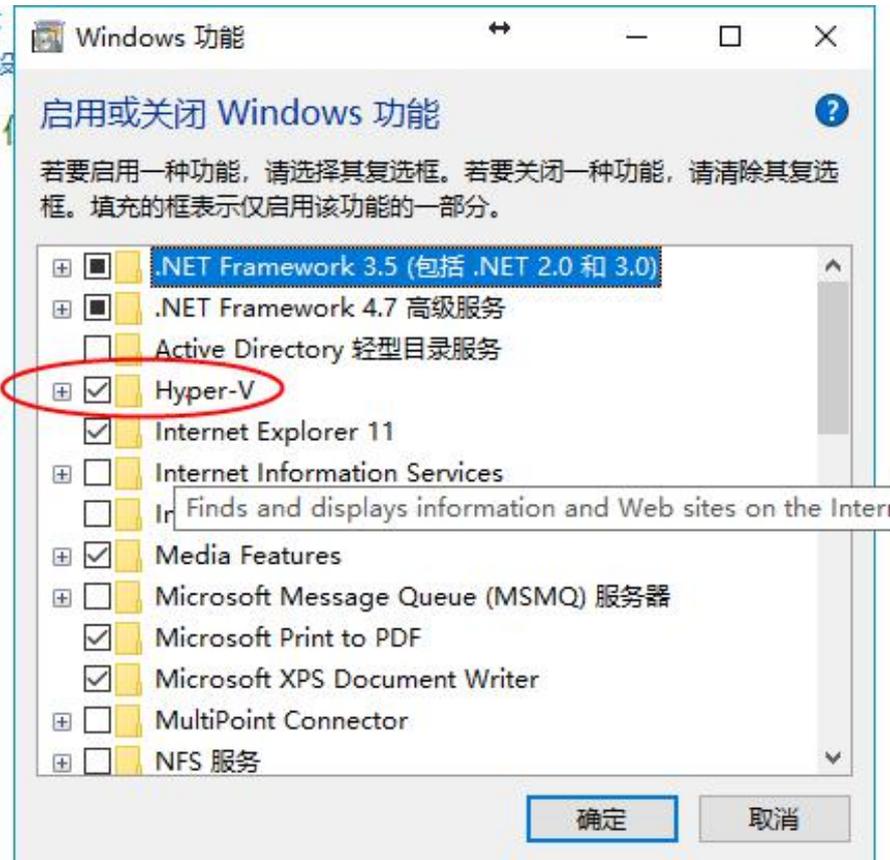


在程序菜单中，选择程序和功能下的启用或关闭 windows 功能



勾选 Hyper-V，点击应用后确定。

应用完成后需要重启电脑才能完成更新安装。



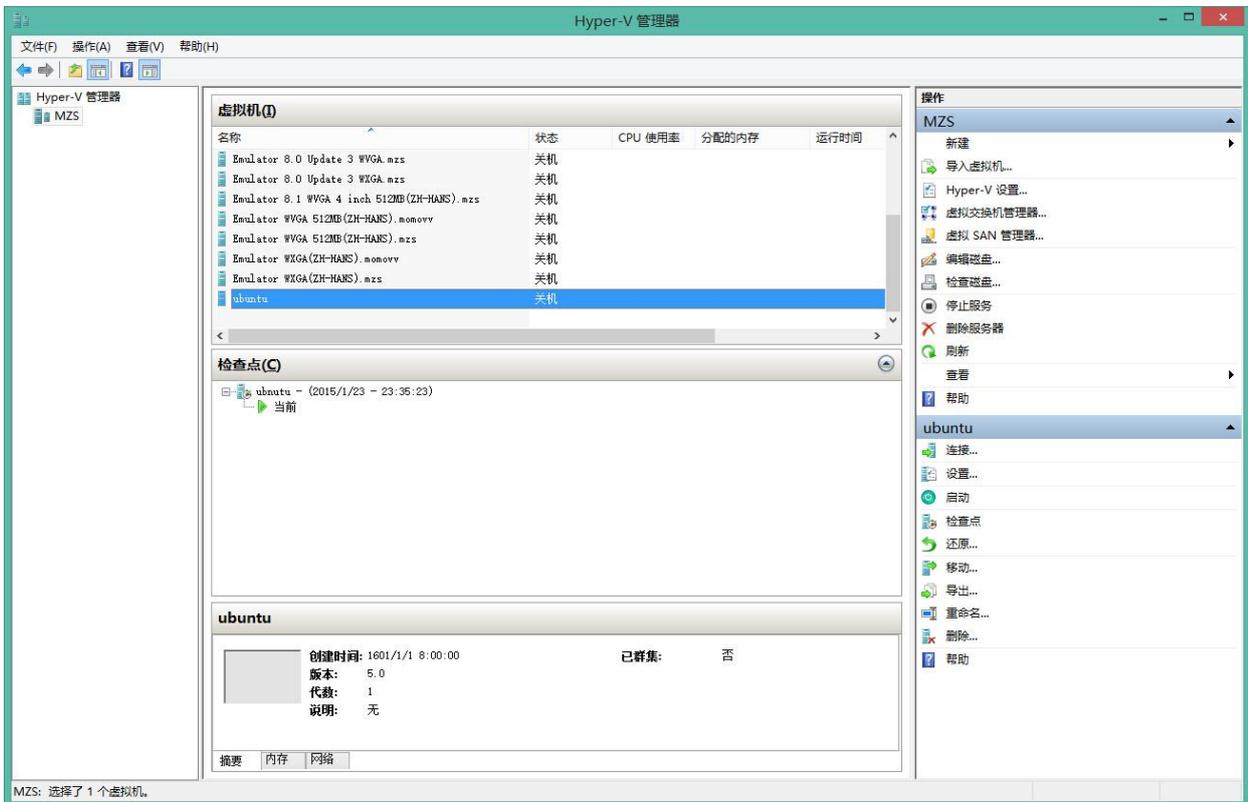
安装完成后，系统会新增该功能。



### 3 Hyper-V 下 Ubuntu 系统安装

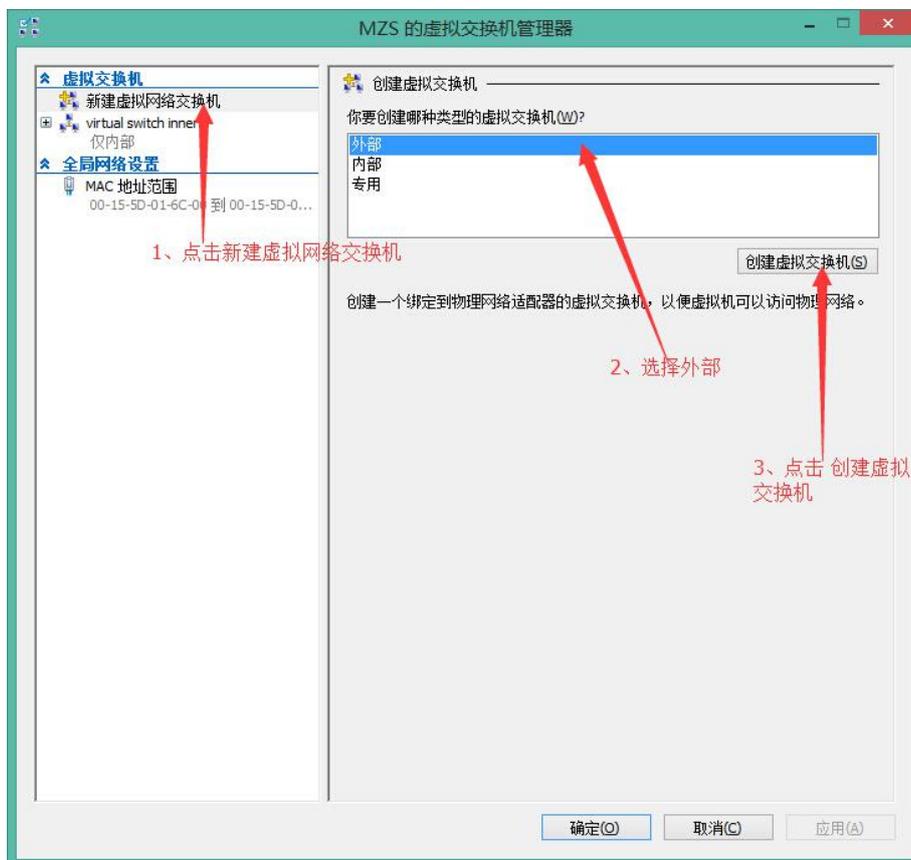
#### 一、安装 ubuntu

首先打开我们的 Hyper-V 管理器，是这个样子的：

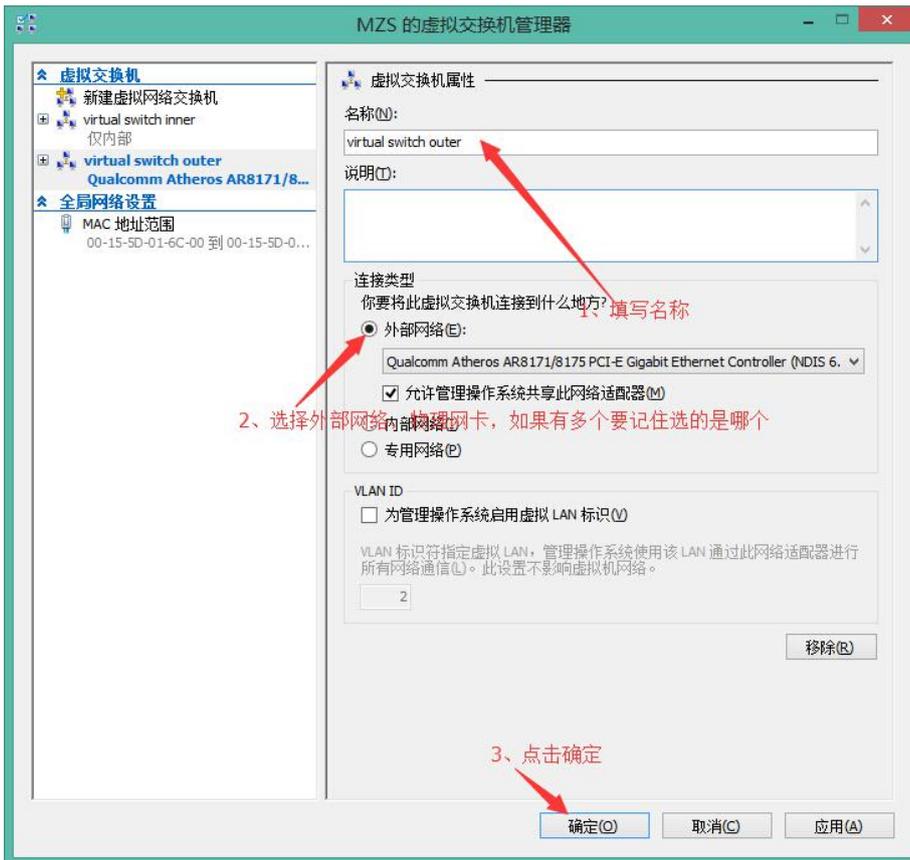


可以看到里面已经装好了好多 windowsphone 各种版本的虚拟机了。我们的任务是要装 ubuntu，但是，在这之前我们要先新建一个网络交换机：

点击右侧“虚拟机管理器”出现下图



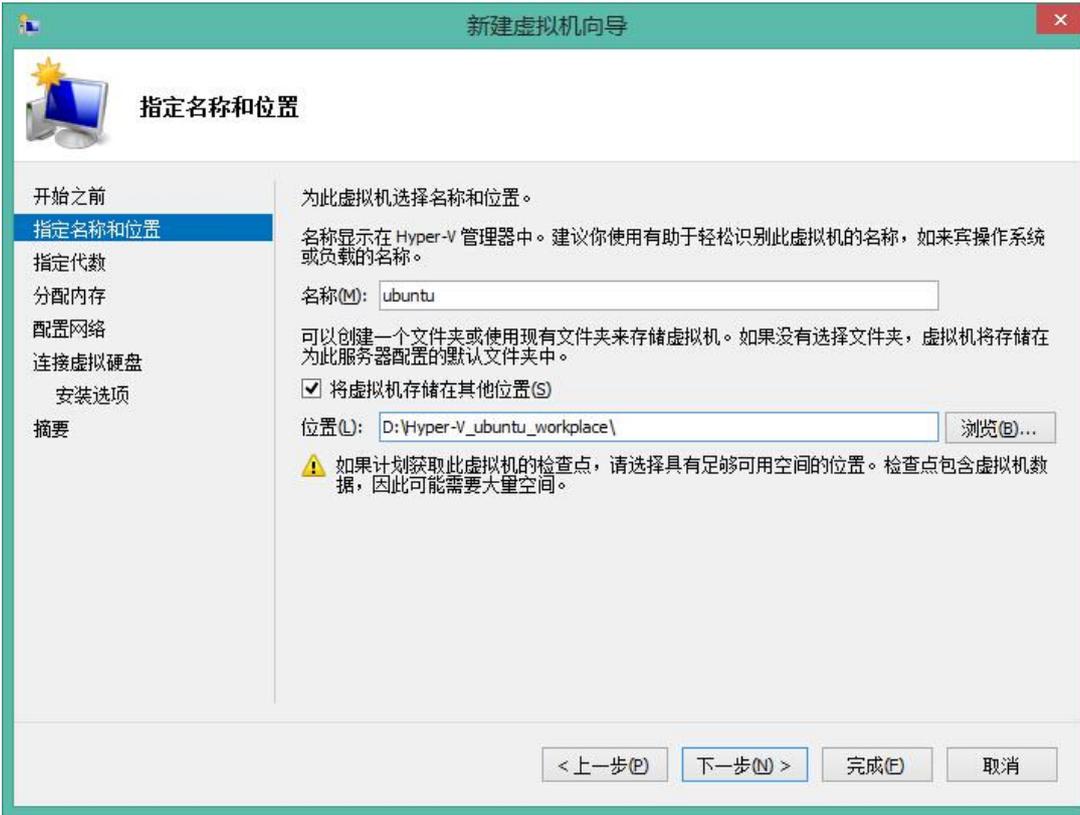
点击创建之后，出现下图，点击确定。



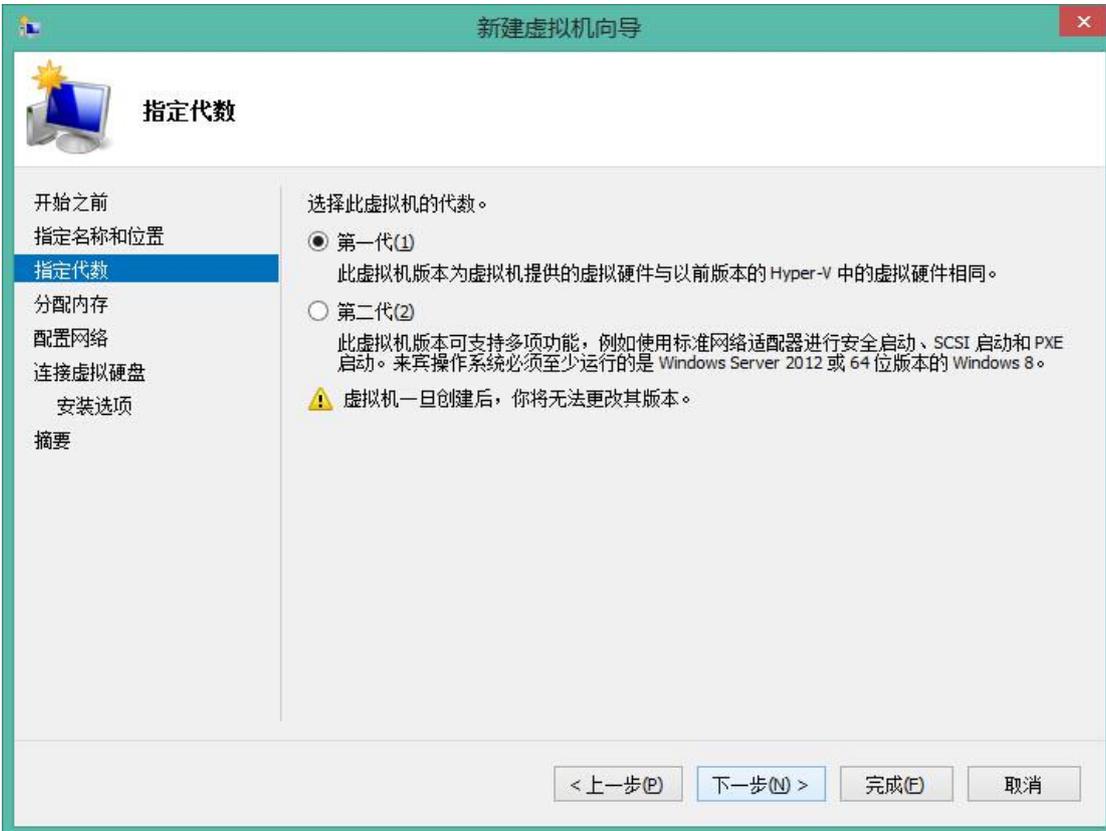
点击“新建”->“虚拟机”



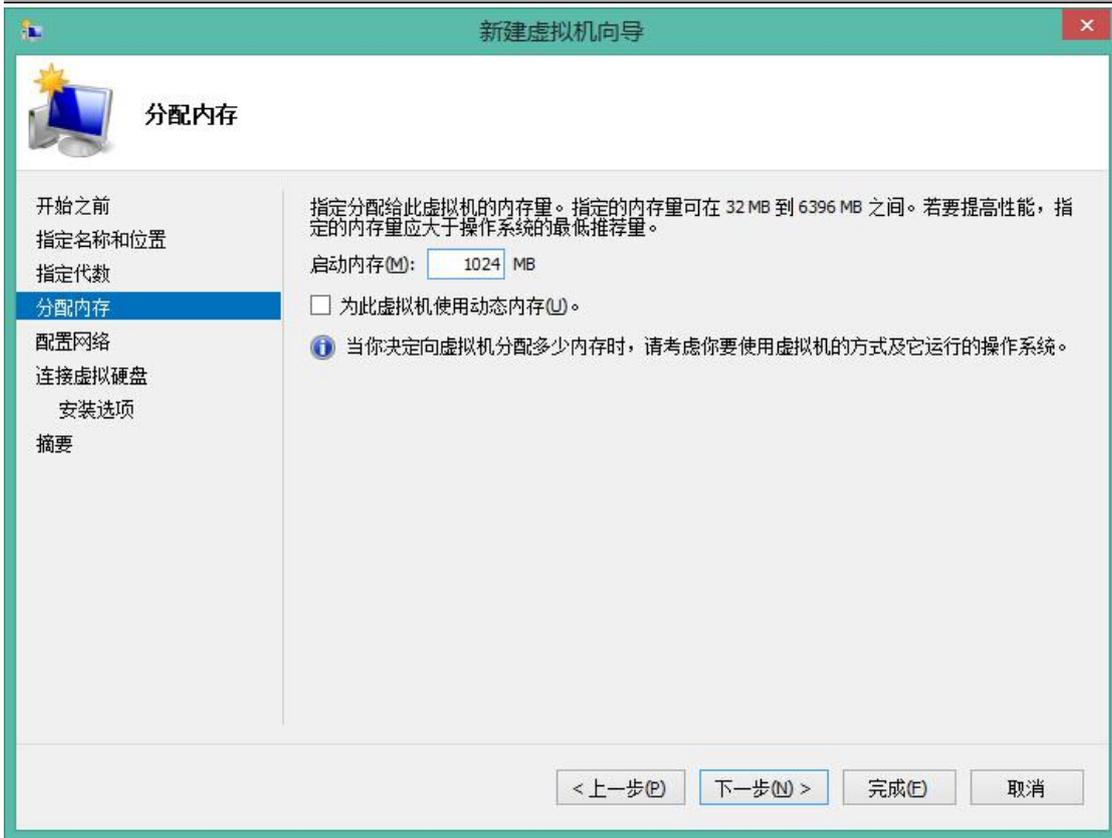
如下图操作



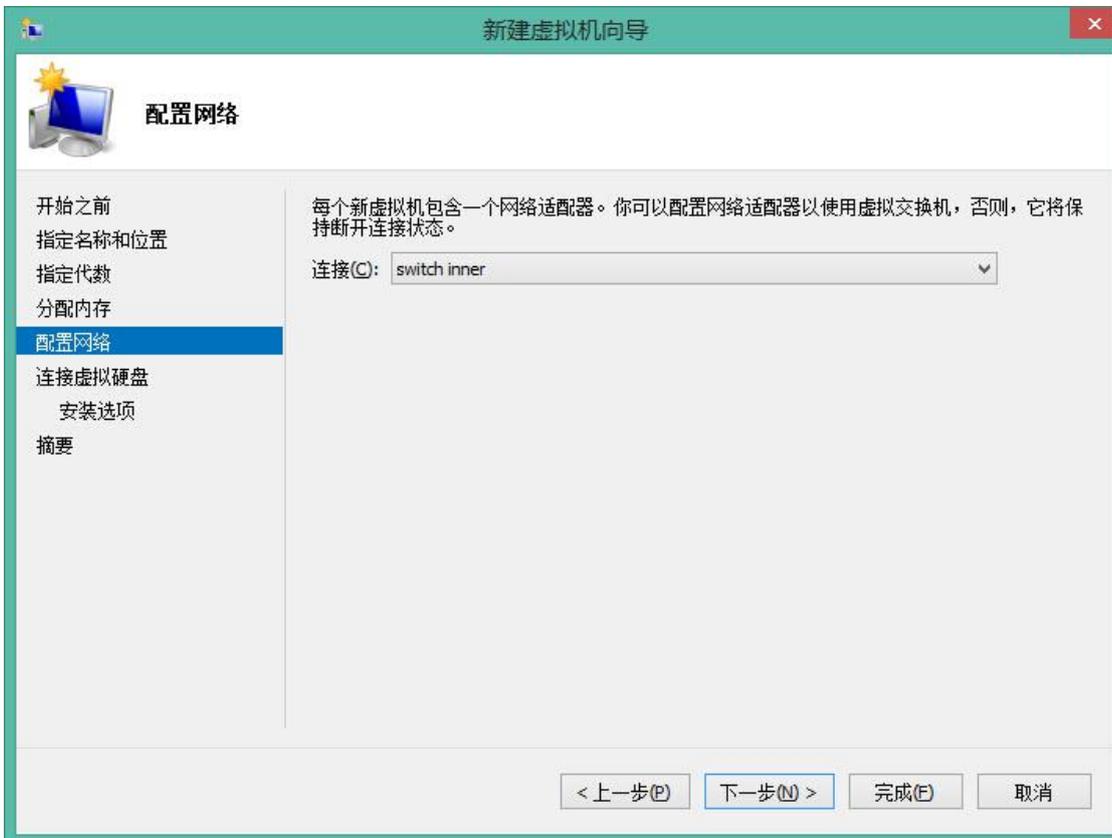
名称自己起一个，默认是把虚拟机保存在 c 盘，勾选“将虚拟机存储在其他位置”，选择你要存储的位置。选择第一代。。。



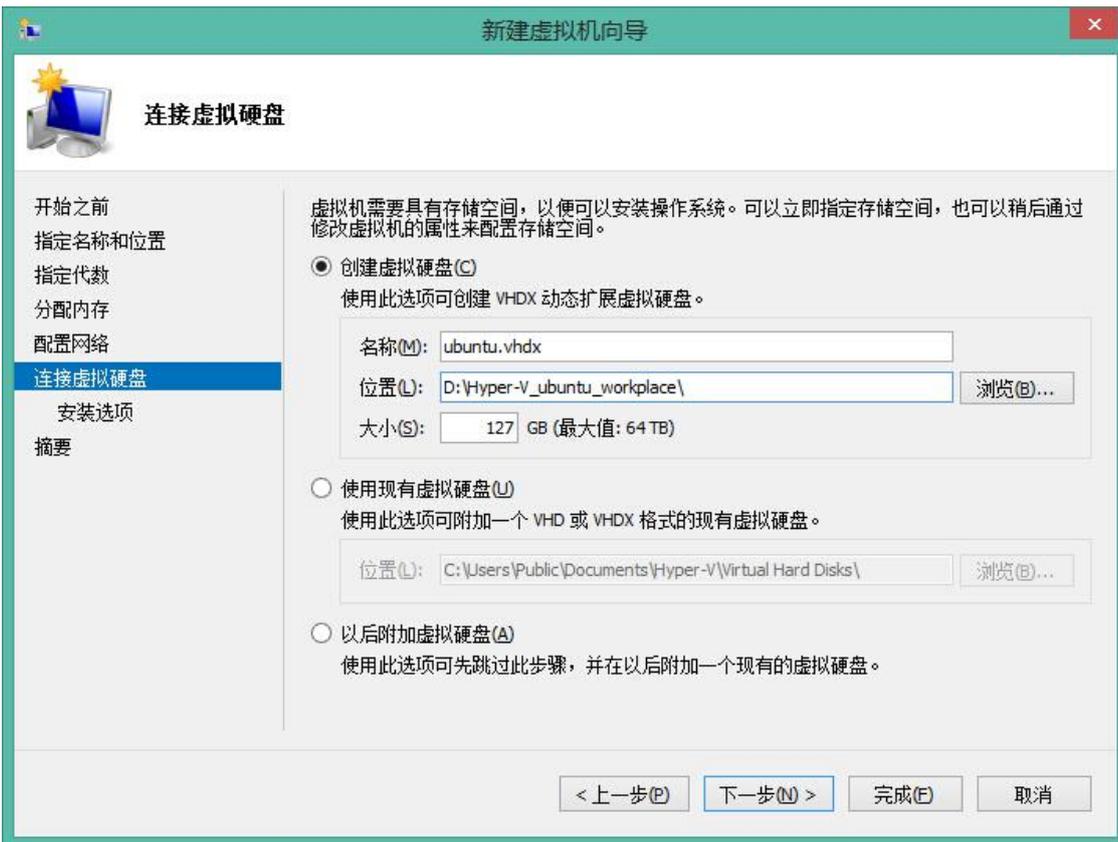
分配内存



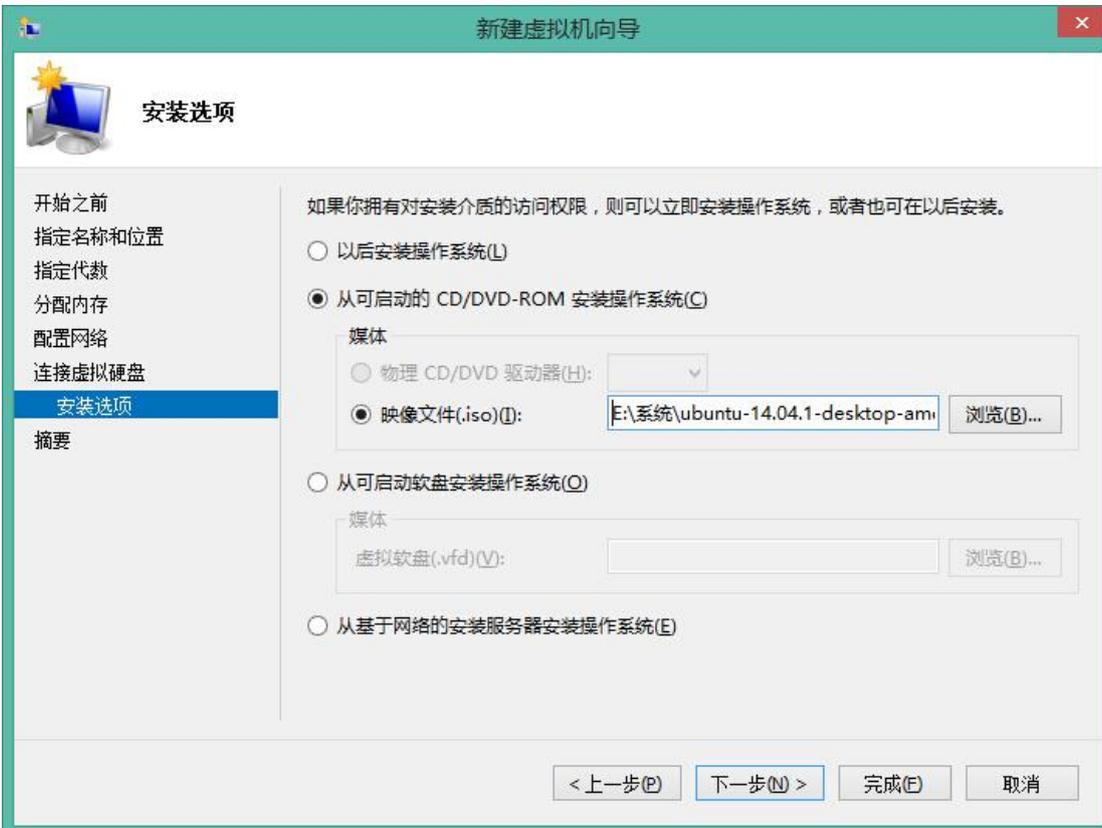
接下来，选择刚才创建的虚拟交换机



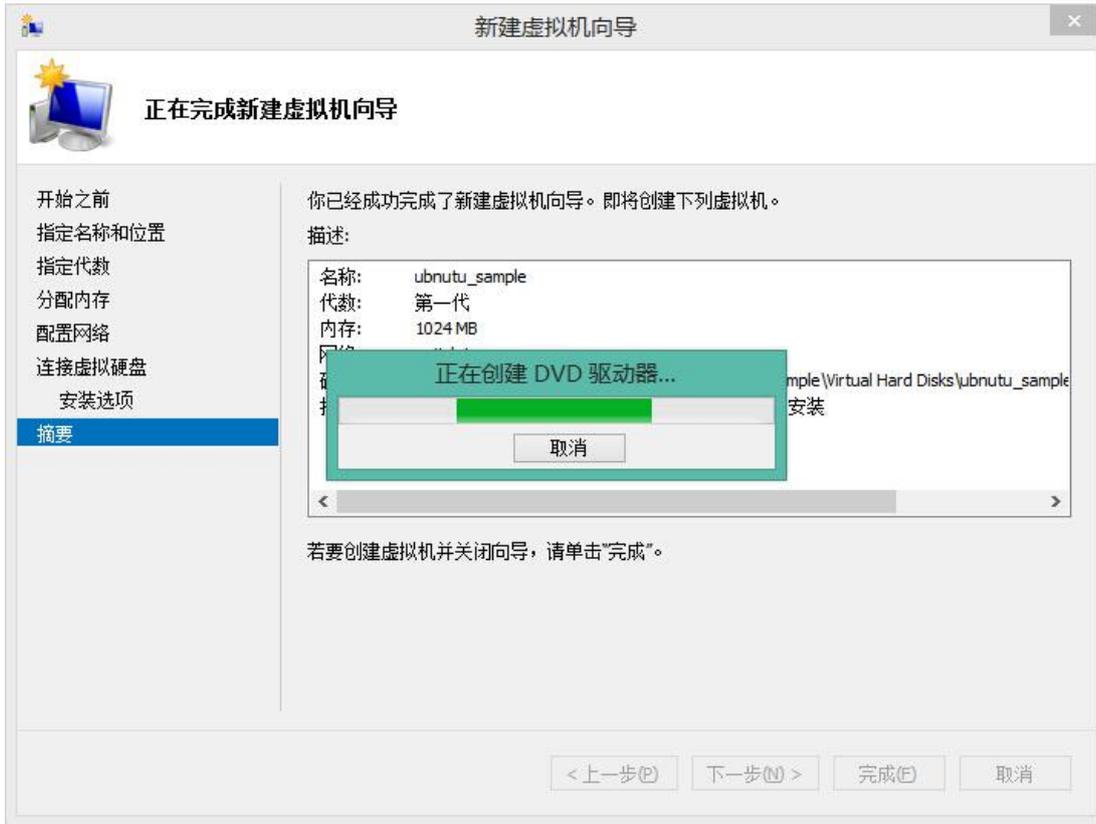
下一步，设置好硬盘大小



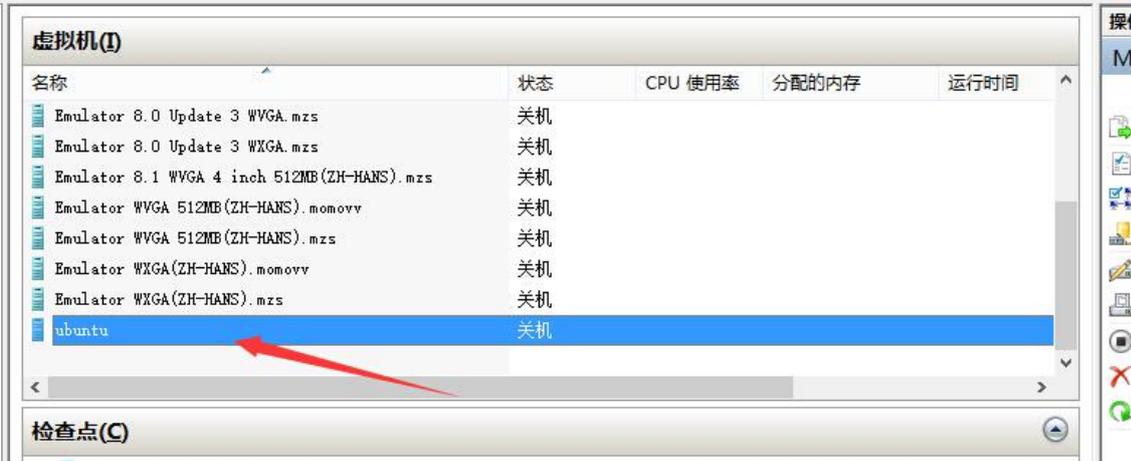
下一步，选择安装系统的 iso 文件。（Ubuntu 的 ISO 安装镜像请自行下载）



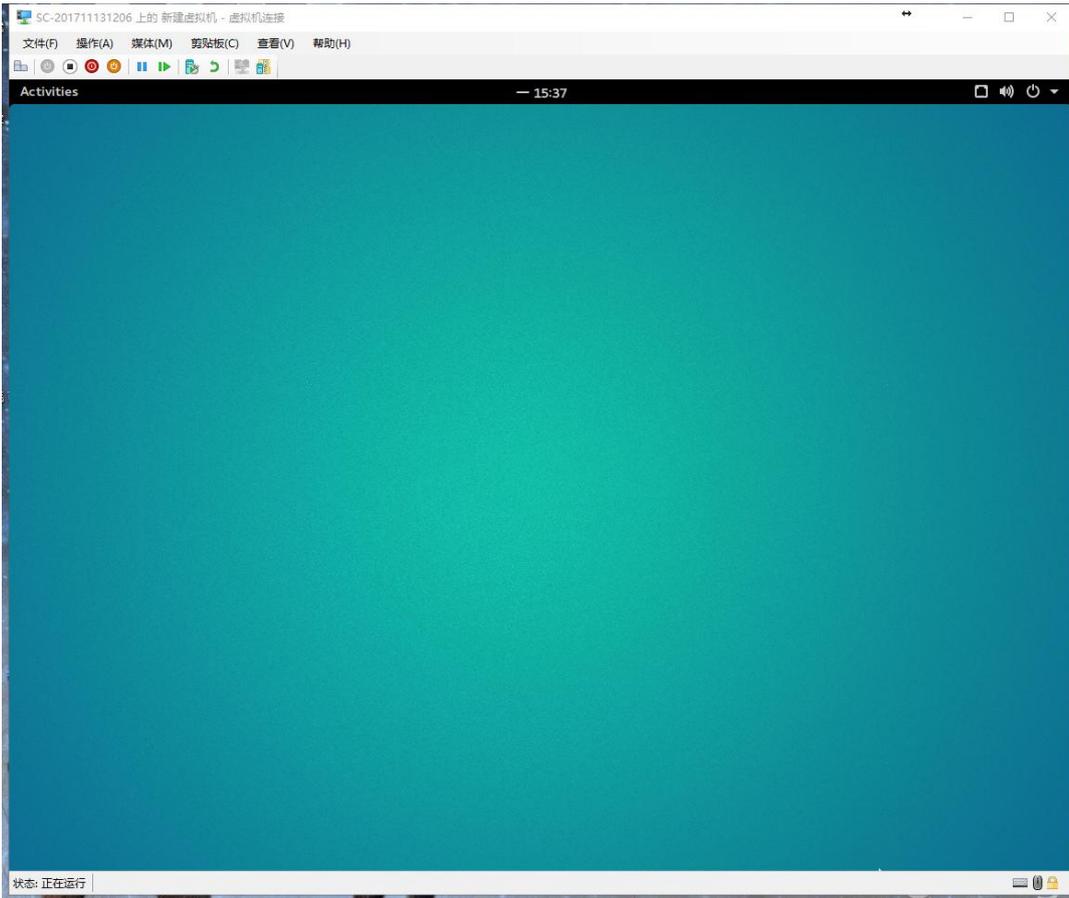
下一步，最后的配置信息，配置无误点击完成。



稍等待安装完毕！回到主页面



点击连接。。。  
 点击，启动安装开始。。。  
 安装过程，我就不赘述了。。。  
 下图为安装完成的 Ubuntu 系统界面。



Ubuntu 设置 ssh 服务:

第一步: ubuntu 开启 ssh

为了方便, 切换 ubuntu 管理员账户

- 1、 打开 Terminal (ctrl+alt+t)
- 2、 输入 sudo passwdroot
- 3、 输入当前密码
- 4、 输入 root 密码
- 5、 重复输入 root 密码
- 6、 切换到 root , su root
- 7、 输入 root 密码

第二步: 安装 ssh

输入命令: apt-get install openssh-server 安装 ssh

安装完成后, 开启服务

```
/etc/init.d/ssh start
```

之后使用如下命令查看是否正确启动

```
ps -e | grep ssh
```

出现如下图, sshd 和 ssh-agent 才算启动成功

```
root@lookingfor-pc:/home/lookingfor# ps -e | grep ssh
1020 ?          00:00:00 sshd
1732 ?          00:00:00 ssh-agent
```

第三步：关闭防火墙

```
ufw disable
```

查看防火墙状态

```
ufw status
```

如果关闭的话是 inactive

现在可以远程登陆了。

设置 ssh IP、用户名、密码，连接成功，如下图：

```
lora@lora-Virtual-Machine: ~
login as: lora
lora@192.168.1.120's password:
Welcome to Ubuntu 16.04 LTS (GNU/Linux 4.4.0-21-generic x86_64)

 * Documentation:  https://help.ubuntu.com/

562 packages can be updated.
262 updates are security updates.

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

lora@lora-Virtual-Machine:~$ █
```

## 4 Ubuntu 系统 docker 环境配置

参照 docker 官网文档，完成 Docker CE 安装，链接如下：

<https://docs.docker.com/engine/installation/linux/docker-ce/ubuntu/#supported-storage-drivers>

参照下文完成 docker-compose 安装：

<https://docs.docker.com/compose/install/#install-compose>

## 5 Lora\_server 安装

1、为该项目创建一个目录：

```
mkdir loraserver_test
```

```
cd loraserver_test
```

2、复制下文内容，在目录下创建 docker-compose.yml 文件：

```
version: "2"
services:
  loraserver:
    image: loraserver/loraserver
    environment:
      - DB_AUTOMIGRATE=true
      - LOG_NODE_FRAMES=true
      - NET_ID=010203
      - BAND=EU_863_870
      - REDIS_URL=redis://redis:6379
      - GW_MQTT_SERVER=tcp://mosquitto:1883
      - GW_SERVER_JWT_SECRET=verysecret
      -
    POSTGRES_DSN=postgres://loraserver_ns:loraserver_ns@postgresql_ns/loraserver_ns?sslmode=disable
      - JS_SERVER=http://appserver:8003
  appserver:
    image: loraserver/lora-app-server
    ports:
      - 8080:8080
    environment:
      - DB_AUTOMIGRATE=true
      - REDIS_URL=redis://redis:6379
      -
    POSTGRES_DSN=postgres://loraserver_as:loraserver_as@postgresql_as/loraserver_as?sslmode=disable
      - MQTT_SERVER=tcp://mosquitto:1883
      - JWT_SECRET=verysecret
      - HTTP_TLS_CERT=/etc/lora-app-server/certs/http.pem
      - HTTP_TLS_KEY=/etc/lora-app-server/certs/http-key.pem
      - AS_PUBLIC_SERVER=appserver:8001
  gatewaybridge:
    ports:
      - 1700:1700/udp
    image: loraserver/lora-gateway-bridge
    environment:
      - MQTT_SERVER=tcp://mosquitto:1883
  postgresql_ns:
    image: postgres:9.6-alpine
    ports:
      - 5432
    environment:
      - POSTGRES_PASSWORD=loraserver_ns
      - POSTGRES_USER=loraserver_ns
      - POSTGRES_DB=loraserver_ns
```

```

postgresql_as:
  image: postgres:9.6-alpine
  ports:
    - 5432
  environment:
    - POSTGRES_PASSWORD=loraserver_as
    - POSTGRES_USER=loraserver_as
    - POSTGRES_DB=loraserver_as
redis:
  ports:
    - 6379
  image: redis:4-alpine
mosquitto:
  ports:
    - 1883:1883
  image: eclipse-mosquitto
  
```

使用 `sudo docker-compose up` 命令启动 `lora_server` 服务, 在此过程中会自动从 `dockerHUB` 上拉取相应得文件进行编译, 之后运行。

```

compose
% Total      % Received % Xferd  Average Speed   Time    Time       Time  Current
           Dload  Upload   Total     Spent    Left     Speed
100 617      0 617    0    0    235      0 --:--:--   0:00:02 --:--:--   235
100 8280k 100 8280k    0    0  586k      0 0:00:14 0:00:14 --:--:--  690k
root@lora-Virtual-Machine:/home/lora# sudo chmod +x /usr/local/bin/docker-compose
root@lora-Virtual-Machine:/home/lora# docker-compose --version
docker-compose version 1.18.0, build 8dd22a9
root@lora-Virtual-Machine:/home/lora# mkdir loraserver_test
root@lora-Virtual-Machine:/home/lora# cd loraserver_test
root@lora-Virtual-Machine:/home/lora/loraserver_test# nano docker-compose.yml
root@lora-Virtual-Machine:/home/lora/loraserver_test#
root@lora-Virtual-Machine:/home/lora/loraserver_test#
root@lora-Virtual-Machine:/home/lora/loraserver_test#
root@lora-Virtual-Machine:/home/lora/loraserver_test#
root@lora-Virtual-Machine:/home/lora/loraserver_test# ls
docker-compose.yml
root@lora-Virtual-Machine:/home/lora/loraserver_test# sudo docker-compose up
Creating network "loraservertest_default" with the default driver
Pulling postgresql_as (postgres:9.6-alpine)...
9.6-alpine: Pulling from library/postgres
550felbea624: Pull complete
04bf519c70df: Pull complete
2af56c798cc1: Pull complete
9ad5d4394eec: Pull complete
df51d99b22c1: Pull complete
b8d2552ec456: Pull complete
f8ef74d64f40: Pull complete
dbf7954a040a: Pull complete
3250b956dc72: Pull complete
Digest: sha256:bf87ee22821e1bc5cedd5da2def1700685a9e3828605b31162d8f04e16c06385
Status: Downloaded newer image for postgres:9.6-alpine
Pulling loraserver (loraserver/loraserver:latest)...
latest: Pulling from loraserver/loraserver
ff3a5c916c92: Downloading [=====>] 1.179MB/2.066MB
df0d49b750d4: Download complete
ad303d98fe40: Downloading [=====>] 1.294MB/5.221MB
  
```

```

sun@sun-Virtual-Machine:~/composetest$ sudo docker-compose up
Starting composetest_web_1 ...
Starting composetest_redis_1 ... done
Attaching to composetest_web_1, composetest_redis_1
redis_1 | 1:C 22 Jan 08:11:17.890 # o000o000o000o Redis is starting o000o000o000o
redis_1 | 1:C 22 Jan 08:11:17.890 # Redis version=4.0.6, bits=64, commit=00000000, modified=0, pid=1, just started
redis_1 | 1:C 22 Jan 08:11:17.890 # Warning: no config file specified, using the default config. In order to specify a config fi
redis_1 | 1:M 22 Jan 08:11:17.893 * Running mode=standalone, port=6379.
redis_1 | 1:M 22 Jan 08:11:17.894 # WARNING: The TCP backlog setting of 511 cannot be enforced because /proc/sys/net/core/somaxc
redis_1 | 1:M 22 Jan 08:11:17.894 # Server initialized
redis_1 | 1:M 22 Jan 08:11:17.894 # WARNING overcommit_memory is set to 0! Background save may fail under low memory condition.
his to take effect.
redis_1 | 1:M 22 Jan 08:11:17.894 # WARNING you have Transparent Huge Pages (THP) support enabled in your kernel. This will crea
s root, and add it to your /etc/rc.local in order to retain the setting after a reboot. Redis must be restarted after THP is disa
redis_1 | 1:M 22 Jan 08:11:17.895 * DB loaded from disk: 0.001 seconds
redis_1 | 1:M 22 Jan 08:11:17.895 * Ready to accept connections
web_1 | * Running on http://0.0.0.0:5000/ (Press CTRL+C to quit)
web_1 | * Restarting with stat
web_1 | * Debugger is active!
web_1 | * Debugger PIN: 698-510-241
^CGracefully stopping... (press Ctrl+C again to force)
Stopping composetest_web_1 ... done
Stopping composetest_redis_1 ... done
sun@sun-Virtual-Machine:~/composetest$ cd ..
sun@sun-Virtual-Machine:~$ cd loraserver_test/
sun@sun-Virtual-Machine:~/loraserver_test$ ls
docker-compose.yml docker-compose.yml.save docker-compose.yml.save.1
sun@sun-Virtual-Machine:~/loraserver_test$ sudo docker-compose up
Starting loraserver_test_postgresql_ns_1 ...
Starting loraserver_test_mosquitto_1 ...
Starting loraserver_test_loraserver_1 ...
Starting loraserver_test_gatewaybridge_1 ...
Starting loraserver_test_redis_1 ...
Starting loraserver_test_postgresql_as_1 ...
Starting loraserver_test_appserver_1 ... done
Attaching to loraserver_test_postgresql_ns_1, loraserver_test_postgresql_as_1, loraserver_test_redis_1, loraserver_test_gatewaybridge
postgresql_as_1 | LOG: database system was shut down at 2018-01-20 07:57:36 UTC
postgresql_as_1 | LOG: MultiXact member wraparound protections are now enabled
postgresql_as_1 | LOG: database system is ready to accept connections
postgresql_as_1 | LOG: autovacuum launcher started
postgresql_ns_1 | LOG: database system was shut down at 2018-01-20 07:57:36 UTC
postgresql_ns_1 | LOG: MultiXact member wraparound protections are now enabled
    
```

## 6 Lora\_server 服务配置

### 6.1 将 lorabridge 服务停止

```

sudo systemctl stop lorabridge
sudo systemctl disable lorabridge
sudo systemctl status lorabridge
    
```

```

pi@raspberrypi:~$ sudo systemctl stop lorabridge
pi@raspberrypi:~$ sudo systemctl disable lorabridge
Removed symlink /etc/systemd/system/multi-user.target.wants/lorabridge.service.
pi@raspberrypi:~$ sudo systemctl status lorabridge.service
● lorabridge.service - senzflow lorabridge
   Loaded: loaded (/lib/systemd/system/lorabridge.service; disabled)
   Active: inactive (dead)

Jan 23 06:17:03 raspberrypi systemd[1]: Starting senzflow lorabridge...
Jan 23 06:17:03 raspberrypi systemd[1]: Started senzflow lorabridge.
Jan 24 02:23:21 raspberrypi systemd[1]: Stopping senzflow lorabridge...
Jan 24 02:23:21 raspberrypi systemd[1]: Stopped senzflow lorabridge.
pi@raspberrypi:~$
    
```

如要回复进行以下操作:

```

sudo systemctl start lorabridge
sudo systemctl enable lorabridge
sudo systemctl status lorabridge
    
```

## 6.2 网关数据指向配置

```
cd lora/packet_forwarder/gps_pkt_fwd/
nano global_conf.json
```

```
"gateway_conf": {
  "gateway_ID": "AA555A0000000000",
  /* change with default server address/ports, or overwri
  "server_address": "192.168.1.120",
  "serv_port_up": 1700,
  "serv port down": 1700,
  /* adjust the following parameters for your network */
  "keepalive_interval": 10,
  "stat_interval": 30,
  "push_timeout_ms": 100,
  /* forward only valid packets */
  "forward_crc_valid": true,
  "forward_crc_error": false,
  "forward_crc_disabled": false,
  /* GPS configuration */
  "gps_tty_path": "/dev/nmea"
}
```

修改好之后保存退出,执行下述命令重启 lrgateway 服务

```
sudo systemctl restart lrgateway.service
```

## 6.3 启动 loraserver

在启动 loraserver 前修改 docker-compose.yml 中得两个地方, 使之与节点配套

```
sudo nano docker-compose.yml
```

```
- NET_ID=666888
- BAND=CN 470 510
```

其中 band 支持以下频段: AS\_923, AU\_915\_928, CN\_470\_510, CN\_779\_787, EU\_433, EU\_863\_870, IN\_865\_867, KR\_920\_923, US\_902\_928

修改好之后保存退出。

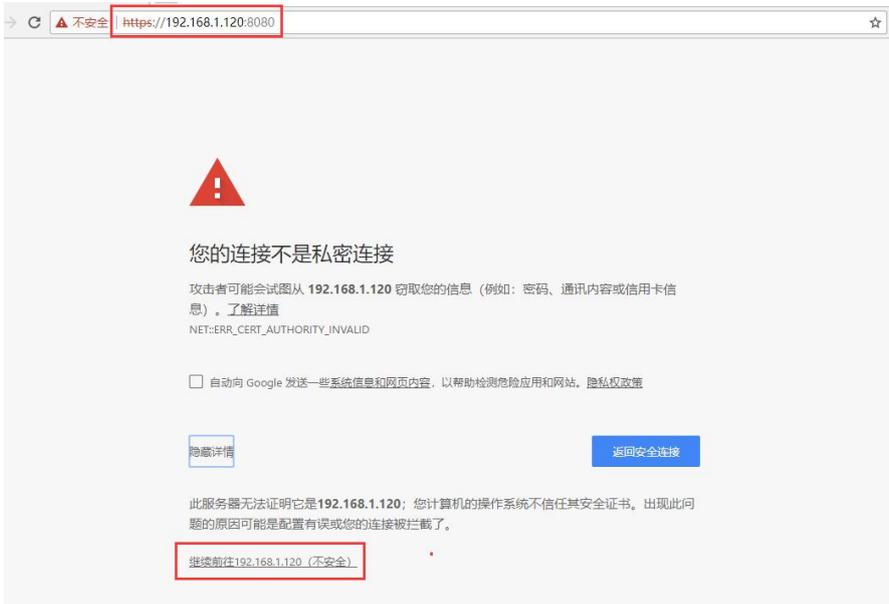
启动上文安装好的虚拟机,启动 docker

```
cd loraserver_test
sudo docker-compose up
```

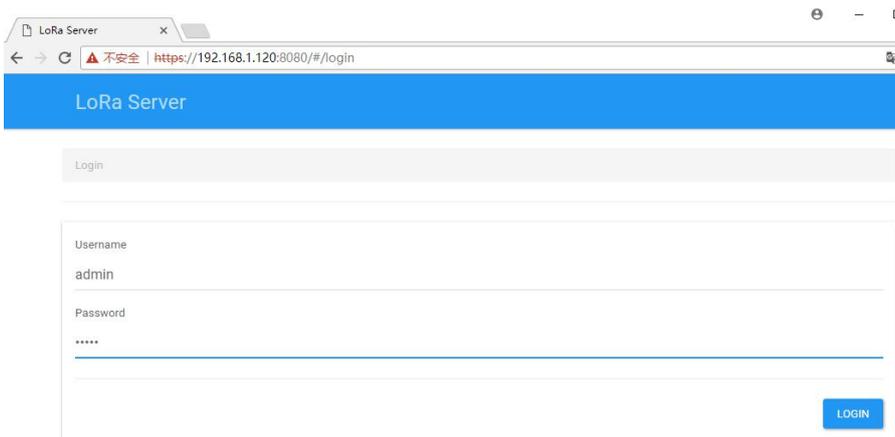
```
Last login: Tue Jan 23 13:17:18 2018 from 192.168.1.101
lora@lora-Virtual-Machine:~$ ls
Desktop  Downloads  Music      Public     Videos
Documents  loraserver_test  Pictures  Templates
lora@lora-Virtual-Machine:~$ cd loraserver test/
lora@lora-Virtual-Machine:~/loraserver_test$ ls
docker-compose.yml
lora@lora-Virtual-Machine:~/loraserver_test$ sudo docker-compose up
[sudo] password for lora:
Starting loraserververtest_appserver_1 ...
Starting loraserververtest_postgresql_ns_1 ...
Starting loraserververtest_mosquitto_1 ...
Starting loraserververtest_postgresql_as_1 ...
Starting loraserververtest_loraserver_1 ...
Starting loraserververtest_gatewaybridge_1 ...
Starting loraserververtest_postgresql_ns_1 ... done
Attaching to loraserververtest_loraserver_1, loraserververtest_postgresql_as_1, lorase
rvertest_gatewaybridge_1, loraserververtest_redis_1, loraserververtest_mosquitto_1, lo
raserververtest_appserver_1, loraserververtest_postgresql_ns_1
loraserver_1 | time="2018-01-24T02:52:58Z" level=info msg="starting LoRa Ser
ver" band=EU_863_870 docs="https://docs.loraserver.io/" net_id=010203 version=ec
2d6fd
loraserver_1 | time="2018-01-24T02:52:58Z" level=info msg="setup redis conne
ction pool" url="redis://redis:6379"
```

## 6.4 配置 lorawanserver

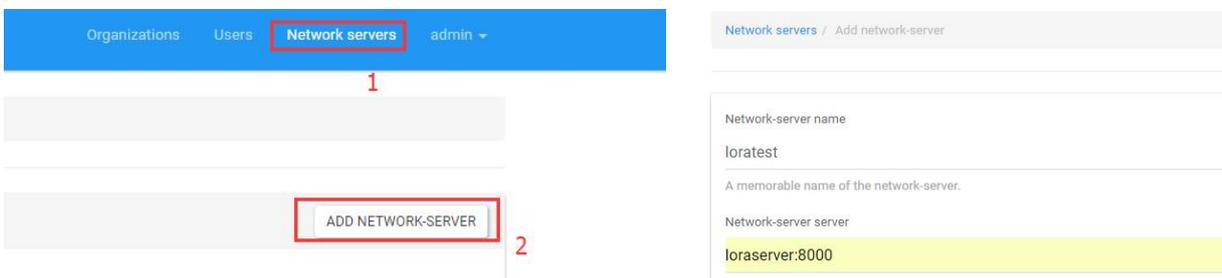
进入 loraserver 得 webUI 进行配置，注意使用 **https://**



用户名:admin 密码:admin



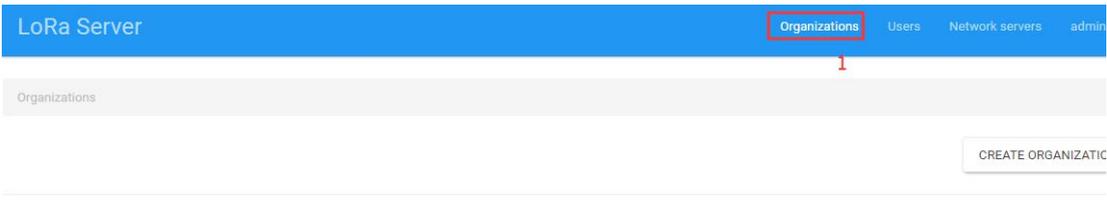
### a)配置网络服务器



其他保持默认

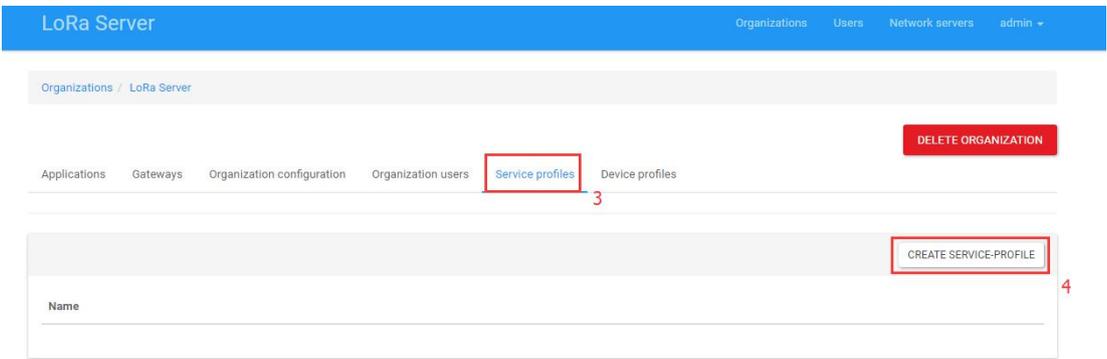
Name	Server
loratest	loraserver:8000

b)添加服务描述



The screenshot shows the 'Organizations' menu in the LoRa Server interface. The 'Organizations' menu item is highlighted with a red box and labeled with a red '1'.

ID	Name	Display name	Can have gateways
1	loraserver	LoRa Server	✓



The screenshot shows the 'Service profiles' menu in the LoRa Server interface. The 'Service profiles' menu item is highlighted with a red box and labeled with a red '3'. A 'CREATE SERVICE-PROFILE' button is highlighted with a red box and labeled with a red '4'.

Create service-profile

Service-profile name

loratest

A memorable name for the service-profile.

Network-server

Irtest

Irtest

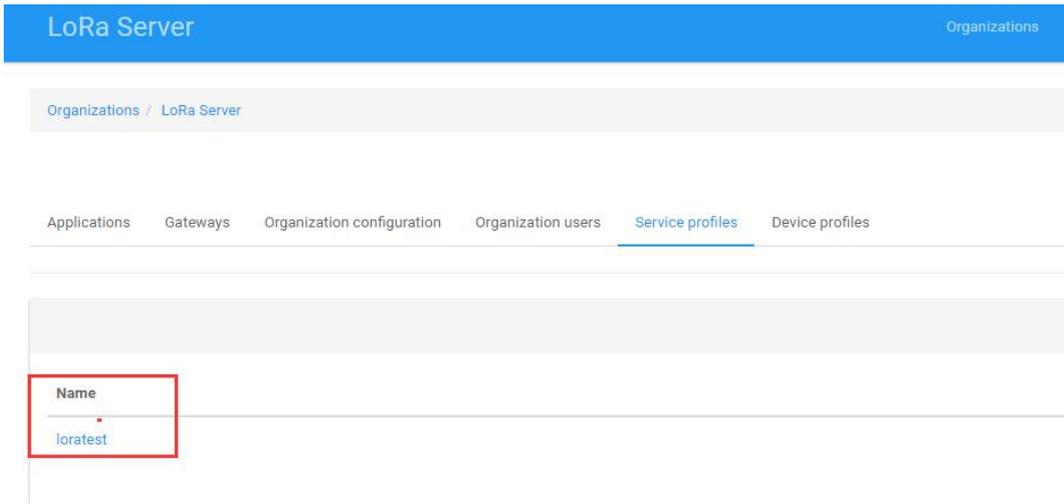
Add gateway meta-data

Add gateway meta-data

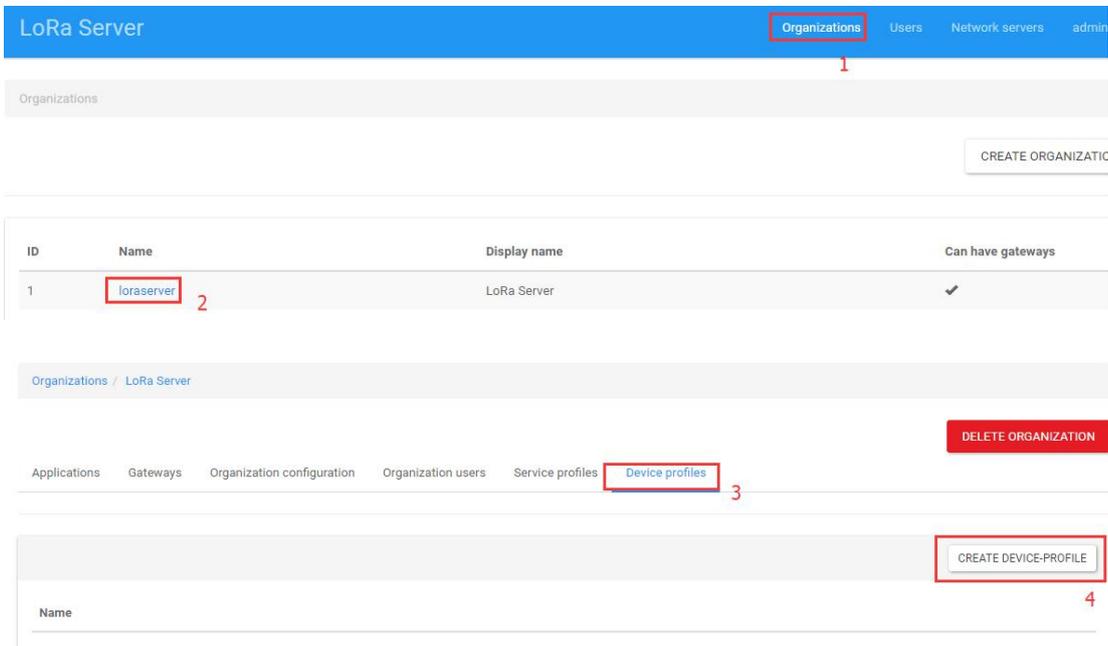
GW metadata (RSSI, SNR, GW geoloc., etc.) are added to the packet sent to the application-server.

此处为勾选项，为之前配置得网络服务器

创建好得服务描述如下：



c)添加设备描述



Create device-profile

General Join (OTAA / ABP) Class-C

Device-profile name  
  
 A memorable name for the device-profile.

Network-server  
  
 The network-server on which this device-profile will be provisioned. After creating the device-pr

LoRaWAN MAC version  
  
 Version of the LoRaWAN supported by the End-Device.

LoRaWAN Regional Parameters revision  
  
 Revision of the Regional Parameters document supported by the End-Device.

Max EIRP

创建好得设备描述如下:

Applications Gateways Organization configuration Organization users Service profiles **Device profiles**

---

Name

d)添加网关

在虚拟机后台中查看有数据过来得网关 MAC:b827ebfffe4c6a9c

```

loraserver_1 | time="2018-01-24T03:39:16Z" level=info msg="gateway: stats packet received" addr="192.168.1.119:35638" mac=b827ebfffe4c6a9c
loraserver_1 | time="2018-01-24T03:39:16Z" level=error msg="handle stats packet error: get gateway error: gateway does not exist"
gatewaybridge_1 | time="2018-01-24T03:39:16Z" level=info msg="backend: publishing packet" topic=gateway/b827ebfffe4c6a9c/stats
gatewaybridge_1 | time="2018-01-24T03:39:16Z" level=info msg="gateway: sending udp packet to gateway" addr="192.168.1.119:35638" protocol=
gatewaybridge_1 | time="2018-01-24T03:39:18Z" level=info msg="gateway: received udp packet from gateway" addr="172.18.0.1:46219" protocol=
gatewaybridge_1 | time="2018-01-24T03:39:18Z" level=info msg="gateway: sending udp packet to gateway" addr="172.18.0.1:46219" protocol=
    
```

LoRa Server **Organizations** Users Network servers admin

Organizations

CREATE ORGANIZATIC

ID	Name	Display name	Can have gateways
1	<input type="text" value="loraserver"/>	LoRa Server	✓

Organizations / LoRa Server

**DELETE ORGANIZATION**

Applications **Gateways** Organization configuration Organization users Service profiles Device profiles

3

Name	MAC	Gateway activity (30d)
		4

**CREATE GATEWAY**

### LoRa Server

Applications **Gateways** Organization configuration Organization users Service profiles

**Create gateway**

Gateway name  
loratestgw  
The name may only contain words, numbers and dashes.

Gateway description  
estgw

MAC address  
b827ebfffe4c6a9c  
Enter the gateway MAC address as configured in the packet-forwarder configuration on the gateway.

Network-server  
Irtest  
Select the network-server to which the gateway will connect. When no network-servers are available in the...

Channel-configuration  
Select...

其他保持默认，添加好的网关如下所示：

Applications **Gateways** Organization configuration Organization users Service profiles Device profiles

Name	MAC	Gateway activity (30d)
loratestgw	b827ebfffe4c6a9c	

**CREATE GATEWAY**

e)添加应用

Organizations

CREATE ORGANIZATION

ID	Name	Display name	Can have gateways
1	loraserver	LoRa Server	✓

CREATE APPLICATION

Create application

Application name  
loratestnode  
The name may only contain words, numbers and dashes.

Application description  
testndoe

Service-profile  
loratest  
The service-profile to which this application will be attached. Note that

Payload codec  
None  
By defining a payload codec, LoRa App Server can encode and decode

其他保持默认，添加好得应用如下：

Applications Gateways Organization configuration Organization users Service profiles Device profiles

CREATE APPLICATION

ID	Name	Service-profile	Description
1	loratestnode	loratest	testndoe

f)添加终端节点设备



添加好得节点设备如下所示:

Device name	Device EUI	Device-profile	Device description
lpmptest	bbbbbbbbbbbbbb	loratest	test

g)通信测试

更多使用开发细节请参考:

<https://docs.loraserver.io/loraserver/overview/>

## 7 关于领派智能

### ● 核心业务：

专注为物联网（IoT）提供“云、网、端”的服务方案提供商，为客户快速实现“端到端”自主网络的全产业链体系,以及云化服务平台，城市级网关路由器，设备通讯模块及应用产品，完全满足客户的各行各业的大规模自组网应用；实现设备数据采集、远程通讯，数据转发及垂直应用。

### ● 技术历程：

团队初创于 2008 年，远距离通讯技术，面向智慧城市、智慧停车、智慧能源、智慧农业、智慧畜牧、资产监管、物联网金融等行业，提供全面的 IoT 解决方案和服务。

- 2008 年-以 433M 射频 zigbee 无线传感网起家
- 2010 年-顺应物联网潮流进军物联网应用市场
- 2013 年-陆续推出智能家居、农业等实用系统
- 2015 年-抢先研发 lorawan 为代表 LPWAN 系统
- 2017 年-lorawan 基站 NS 私有云服务全面商用

### 领派智能（无锡）（学鑫科技）：

售前技术支持：0510-85386543-801 王工 wangxuedan@lplinkpi.com

售后技术支持：0510-85386543-807 孙工 sunlifang@lplinkpi.com

### 领派智能（深圳）：

售前技术支持：

陈工 chengong@lplinkpi.com

国内外销售：sales@lplinkpi.com

技术售后：support@lplinkpi.com

网站(论坛)：<http://www.lplinkpi.com>