



## SPN v3.0(a.k.a IoTSec) - LoRa Security Project (Part III)

08.21.2019 ~

Doc. Revision: 1.4

We Secure the Internet of Things with 2STON SPN.

---

Chunghan.Yi(michael@2ipco.com)

R&D Center

2ip Inc.

# Table of Contents

## Part I.

1. LoRa 개요
2. RAKWireless RAK831 LoRaWAN Kit
3. Dragino IoT Kit v2
4. Dragino LG308 LoRa Gateway
5. RAKWireless **RAK7258** LoRa Gateway
6. RAKWireless **RAK7249** Outdoor LoRa Gateway
7. MatchX **MX1702** LoRa Gateway(**LBT 지원 모델**)

## Part II.

8. MultiTech **MultiConnect Conduit** IP67 LoRa Gateway
9. LoRa Gateway에 **SPN S/W Porting**하기
10. **LoRaServer** Project 1 - **설치 & 운용**
11. **LoRaServer** Project 2 - **External Interface**
12. **ThingsBoard** Integration - **대박** :)
13. Our LoRa Viewer: **OLoRa( = ThingsBoard)**

## Part III.

14. Outdoor LoRa Node - **Libelium**
15. **UrsaLink** LoRa Products
16. NetVox PM2.5 Sensor
17. LoRaWAN Stack

## Part III. Chapter 14 ~ Chapter 17.

Part III에서는 주로 LoRa Node에 관한 얘기를 다루고자 한다. LoRa Gateway/Server/Application Server(or Viewer) 까지 검토하고 나니, 남은 문제는 역시 LoRa Node를 어떻게 확보해야 하는 문제로 귀결되는 느낌이다. LoRa Node는 실제 사업화 방향과도 맞닿아 있는 부분인 것 같다.

지금까지 직/간접적으로 검토한 LoRa Node를 정리해 보면 다음과 같다.

- 1) RAK831
  - 아두이노 환경
- 2) Dragino Kit
  - 아두이노 환경
- 3) Dragino LGT-92 GPS tracker : STM32L072CZT6 MCU
  - ST-Link V2 사용해야 함. source 확보 가능
- 4) RAK7200 GPS tracker: STM32L073
  - flash programmer는 자체 version 존재하나, source code 없음.
- 5) RAK7204 온습도 센서
  - flash programmer는 자체 version 존재하나, source code 없음.
- 6) MatchX Dev Kit
  - 애플은 source code 제공 안함. CPU: Dailog사 제품(ARM Cortex-M0)

## 14. Outdoor LoRa Node - Libelium

지금까지는 LoRa Gateway가 Outdoor인 경우만을 고민했던 것 같다. 다시 말해 은연중에 LoRa Node는 사람이 가지고 다니던지, 실내에 있는 경우만을 생각했던 것이다. 하지만, 이게 얼마나 웃기는 얘기인가? LoRa Device(Node)가 외부에 고정된 형태로 있어야 하는 경우는 아주 쉽게

생각해 볼 수 있다. 지금부터는 **Libelium**이라는 스페인 회사에서 개발하여 판매중인 **Waspote** 제품을 상세히 분석해 보고, 우리 제품에 어떻게 녹일 수 있을지 고민해 보도록 하겠다.



[그림 14.1] Libelium LoRaWAN network 구성

<https://www.the-iot-marketplace.com/libelium-loriot-development-lorawan-kit>



[그림 14.2] Libelium-Multitech-Loriot Development Kit(5500 유로)

### 1) Libelium Wasmote Plug & Sense



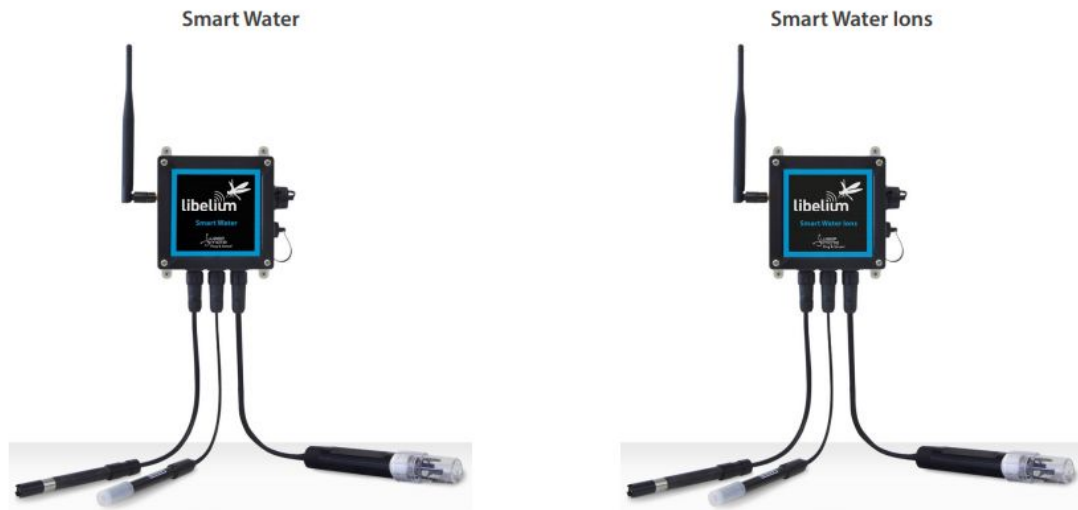
[그림 14.3] Libelium Waspote Plug & Sense !(1)



[그림 14.4] Libelium Waspote Plug & Sense !(2)



[그림 14.5] Libelium Waspote Plug & Sense !(3)



[그림 14.6] Libelium Wasp mote Plug & Sense !(4)



[그림 14.7] Libelium Wasp mote Plug & Sense !(5)

Smart Parking



Radiation Control



[그림 14.8] Libelium Wasmote Plug &amp; Sense !(6)

<TBD> 너무 비싸다. 구매 자체는 나중에 다시 검토하기로 하자 :( 하지만, Libelium 제품이 앞으로 우리의 개발 방향과 일치하는 것은 매우 중대한 사실이다.

## 2) Libelium Wasmote Dev Kit

<TBD> 이걸 하나 구매해야 겠다.



[그림 14.9] Libelium Wasmote Dev Kit

아래 site에 Wasmote Dev Kit를 가지고 다양한 programming을 하는 방법이 소개되어 있다.

<http://www.libelium.com/development/wasmote/documentation/?cat=programming>



<Libelium development web page>

<http://www.libelium.com/v12/development/>

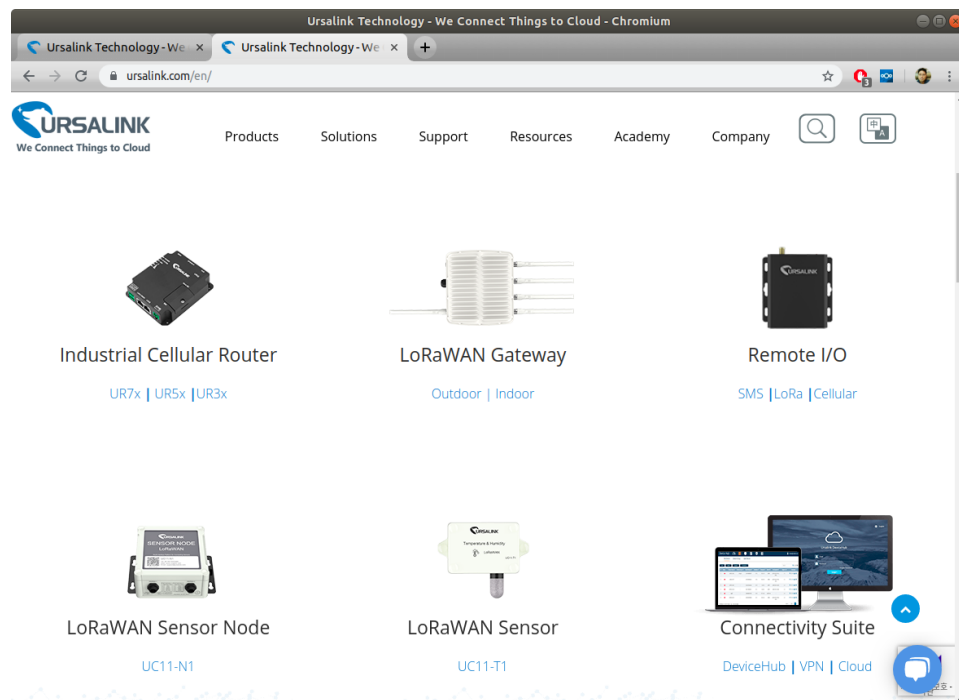
<waspmoteapi github>

<https://github.com/Libelium/waspmoteapi>

Libelium 제품은 LoRa(단점: KR920을 지원 안함, 너무 비쌌)가 아니더라도 다른 무선 or 유선 환경에서 앞으로 활용도가 높은 제품으로 보인다.

## 15. UrsaLink LoRa Products

이번 장에서는 "We Connect Things to Cloud"라는 slogan을 전면에 내세운 **UrsaLink LoRa 제품(industrial router, LoRa 제품 보유)**을 소개해 보고자 한다. 참으로 배울 것이 많은 회사 같다. 우리 회사의 방향성을 이곳에서 찾을 수 있지 않을까 생각해 본다(우리에게 제 2의 GL.iNet이 될 지 검토해 보기로 하자).



[그림 15.1] URSALINK Home Page



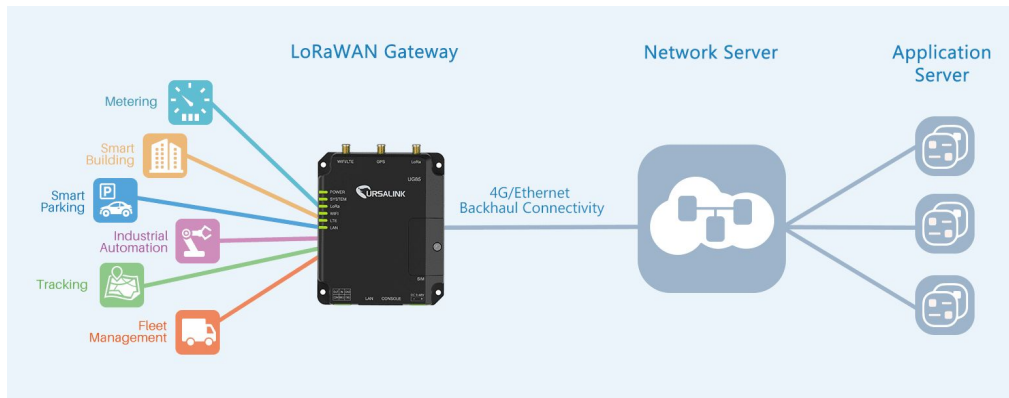
[그림 15.2] URSALINK LoRa Gateways

### 1) UG85 Indoor LoRa Gateway

<TBD> 우선 아래 제품을 하나 구매하여 내부를 분석해 보아야 겠다.

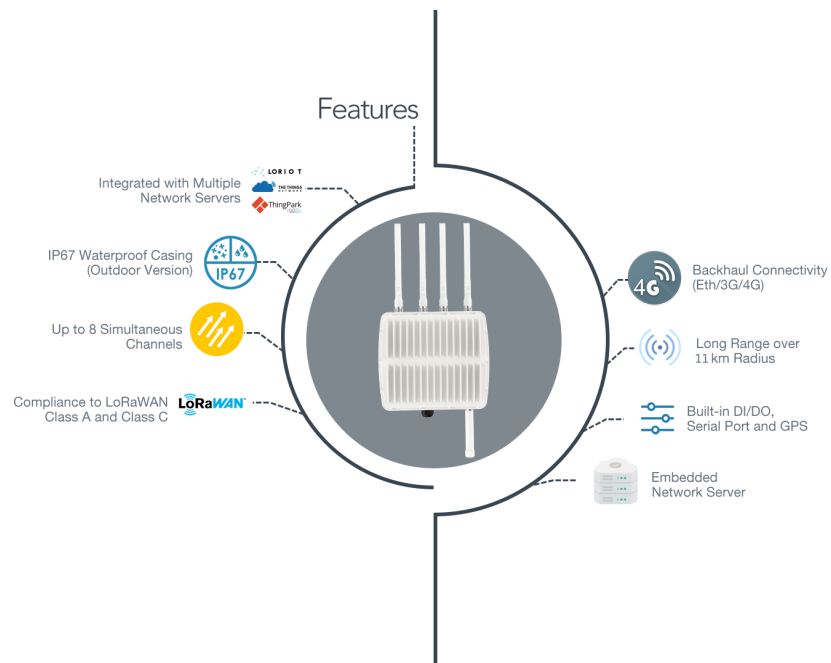


[그림 15.3] URSALINK UG85 Indoor LoRa Gateway(1)

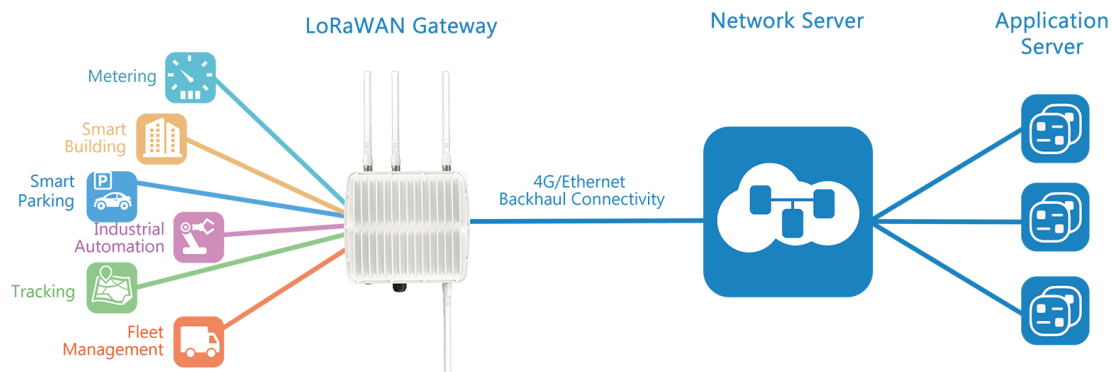


[그림 15.4] URSALINK UG85 Indoor LoRa Gateway(2)

## 2) UG87 Outdoor LoRa Gateway

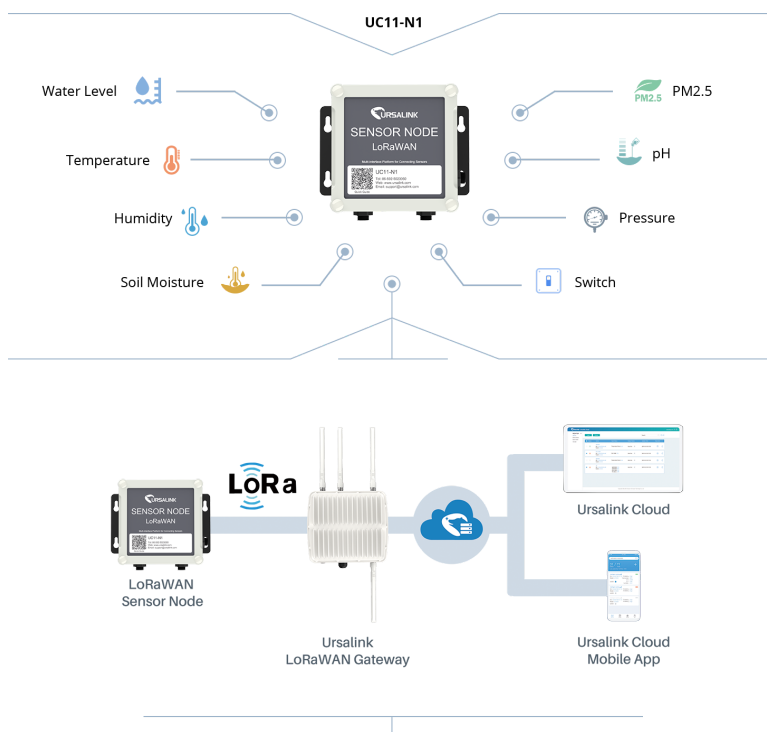


[그림 15.5] URSALINK UG87 Outdoor LoRa Gateway(1)



[그림 15.6] URSALINK UG87 Outdoor LoRa Gateway(2)

### 3) LoRaWAN Sensors



[그림 15.7] URSALINK UC11-N1 LoRaWan Sensor



Embedded with a range of trigger conditions and actions, Ursalink Remote I/O series enable users to create various combinations of commands accordingly. With rich functionalities and small form-factor design, Ursalink Remote I/O series are able to be attached to many electronic devices at homes, in offices, or on any other sites.



SMS Remote I/O

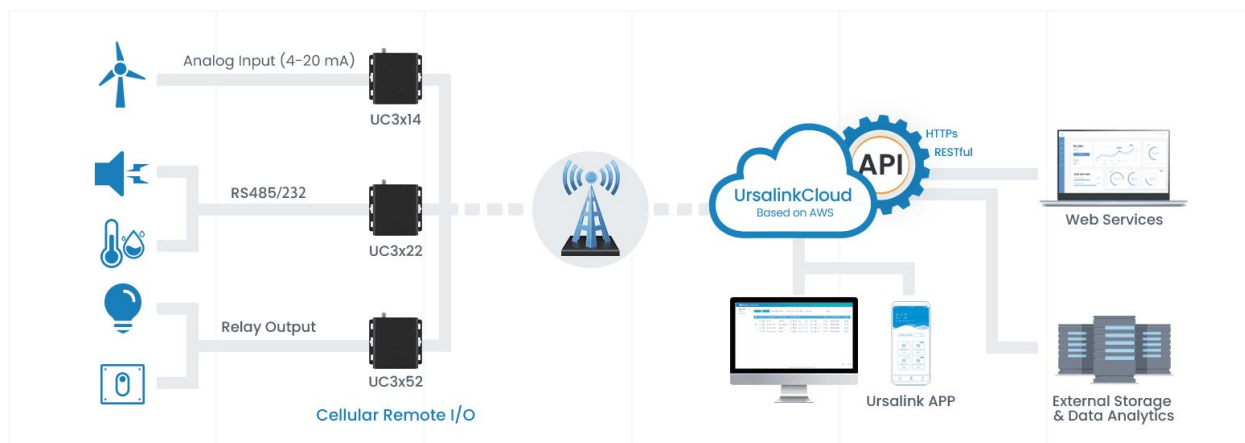


LoRa Remote I/O



Cellular Remote I/O

[그림 15.8] URSALINK Cellular Remote I/O(1)



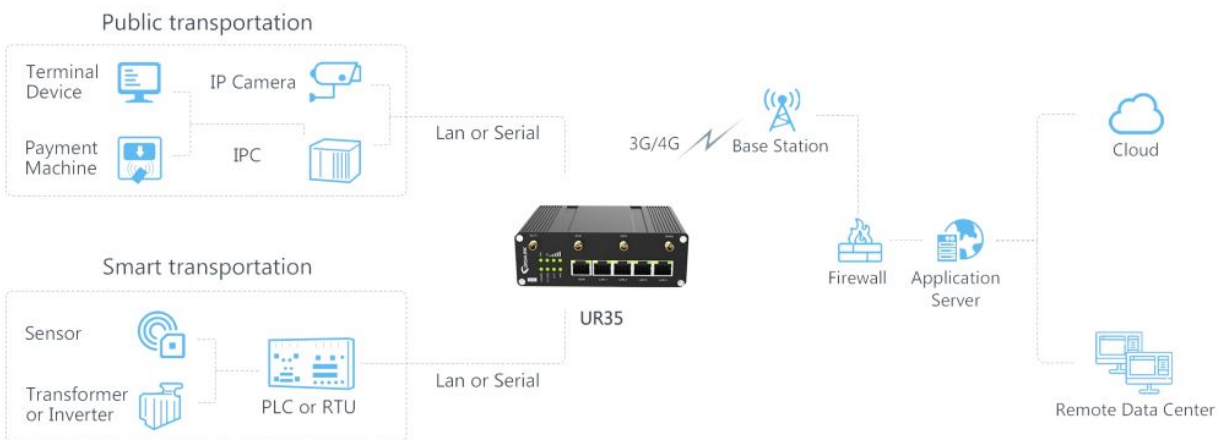
[그림 15.9] URSALINK Cellular Remote I/O(2)

#### 4) UG75 Industrial Cellular Router

얘는 LoRa Gateway는 아니지만, SPNBox의 모델 후보로 가능할 듯 보여 여기서 잠시 소개하고자 한다.



[그림 15.10] URSALINK UR7X Cellular Router



[그림 15.11] URSALINK UR7X Cellular Router 네트워크 구성

## 16. Netvox PM2.5 Sensor

이번 장에서는 Netvox PM2.5 Sensor를 LoRa Gateway & LoRaServer에 붙여 보도록 하겠다.

### 미세먼지상식 미세먼지 개념

#### 1. 정의

- 우리는 일반적으로 먼지를 미세먼지와 초미세먼지로 구분하여 왔으나 실제 법에서 정의된 용어는 다르다는 사실.
- 대기환경 뿐만 아니라 환경 보전을 목적으로 하는 환경정책기본법에서 미세먼지(PM-10)은 입자의 크기가  $10\mu\text{m}$  이하인 먼지, 미세먼지(PM-2.5)는 입자의 크기가  $2.5\mu\text{m}$  이하인 먼지로 정의되어 있음.
- 즉, '미세먼지'라는 용어는 동일하게 사용하되, ( )안의 숫자가 먼지의 크기를 의미하며, PM은 Particulate Matter의 약자임.

#### 2. 크기

- 미세먼지의 크기 단위인  $\mu\text{m}$ 는 1m보다 1백만분의 1만큼 작다는 의미임.
- 사람의 머리카락과 비교할 때 미세먼지(PM-10)는 1/6, 미세먼지(PM-2.5)는 1/20~1/30의 수준임.
- ※ 미세먼지는 우리 눈에 잘 보이지 않지만 빛의 산란에 의해 대기 중에 미세먼지가 많은 날은 뿌옇게 흐린 상태가 됨

#### 3. 왜 미세먼지가 논란이 되고 있나?

- 미세먼지(PM-10, PM-2.5)는 눈에 보이지 않을 정도로 매우 작기 때문에 공기 중에 머물러 있다 우리가 호흡할 때 호흡기를 거쳐 폐 등에 침투하거나 혈관을 따라 체내로 이동해 들어감으로써 건강에 나쁜 영향을 미침.
- 일반적으로 미세먼지(PM-2.5)가 더 유해하여 미세먼지(PM-10)가 호흡기 질환을 유발하는 반면 미세먼지(PM-2.5)는 심혈관 질환 및 뇌 질환까지 관계가 있는 것으로 보고되고 있음.
- 세계보건기구(who)는 미세먼지(PM-10, PM-2.5)에 대한 대기질 가이드라인을 1987년부터 제시해 왔고, 2013년에는 WHO 산하 국제암연구소(IARC, International Agency for Research on Cancer)에서 미세먼지를 사람에게 발암이 확인된 1군 발암물질(Group 1)로 지정하였음.



#### $\mu\text{g}/\text{m}^3$ (대기오염의 측정단위)

공기 1세제곱미터에 오염물질이 1그램의 100만분의 1만큼 들어 있다는 것을 의미한다. 예를 들어 미세먼지 농도가  $30\mu\text{g}/\text{m}^3$ 이라면, 1세제곱미터의 공기 속에 들어 있는 분자의 무게가 100만분의 30인 셈이다. 대기오염 연구는 이처럼 극미량의 세계를 다루기 때문에 정확한 측정이 중요하다.

#### PM10(Particulate Matter 10)

공기 중에 있는 먼지를 측정하기 위해서는 깔때기처럼 생긴 공기흡입구에 필터를 설치하고 그 아래 연결한 전기모터를 가동시켜 일정 시간 공기를 빨아들이게 한다. 그렇게 하면 필터에 먼지가 모이게 된다. 이후 먼지의 무게를 잰 다음 흡입된 공기의 양을 환산해서 공기 1 세제곱미터당 얼마나 되는 먼지가 필터에 모였는지 측정한다.

한편 대기오염 측정기에는 지름  $10\mu\text{g}$ 이하의 먼지는 통과하도록 장치를 만들어 놓고, 지름이  $10\mu\text{g}$ 이상 되는 먼지 농도만 측정한다. 그 결과  $10\mu\text{g}$ 이상 먼지 농도를 제외한 남은 먼지, 즉  $10\mu\text{g}$ 이하의 먼지농도만 남게 되는데, 이를 PM10이라고 부른다. 직경  $2.50\mu\text{g}$ 의 필터를 사용하면 마찬가지로 방식으로 PM2.5 농도를 측정할 수 있다.

[그림 16.1] 미세먼지 개념 정의



Index Values	Category	Cautionary Statements	PM <sub>2.5</sub> ( $\mu\text{g m}^{-3}$ )	PM <sub>10</sub> ( $\mu\text{g m}^{-3}$ )
0-50	Good	None	0-15.4	0-54
51-100	Moderate	Unusually sensitive people should consider reducing prolonged or heavy exertion	15.5-40.4	55-154
101-150	Unhealthy for sensitive groups	Sensitive groups should reduce prolonged or heavy exertion	40.5-65.4	155-254
151-200	Unhealthy	Sensitive groups should avoid prolonged or heavy exertion; everyone else should reduce prolonged or heavy exertion	65.5-150.4	255-354
201-300	Very unhealthy	Sensitive groups should avoid all physical activity outdoors; everyone else should avoid prolonged or heavy exertion	150.5-250.4	355-424

Source: US EPA, 1997

[그림 16.2] 미세먼지 테이블

## 1) Netvox RA0716 PM2.5 센서

대만산 Netvox RA0176 PM2.5 미세먼지 센서(PM2.5, 온/습도 센서)를 하나 구입했다. 지금 부터는 이 센서를 2ip LoRaWAN 망에 붙여 보도록 하겠다.



[그림 16.3] Netvox RA0176 PM2.5 센서 - Indoor type

## 2) LoRaServer와 연동

센서를 LoRaServer에 연결(OTAA)하기 위해서는 DEV EUI와 App Key가 반드시 필요하다.

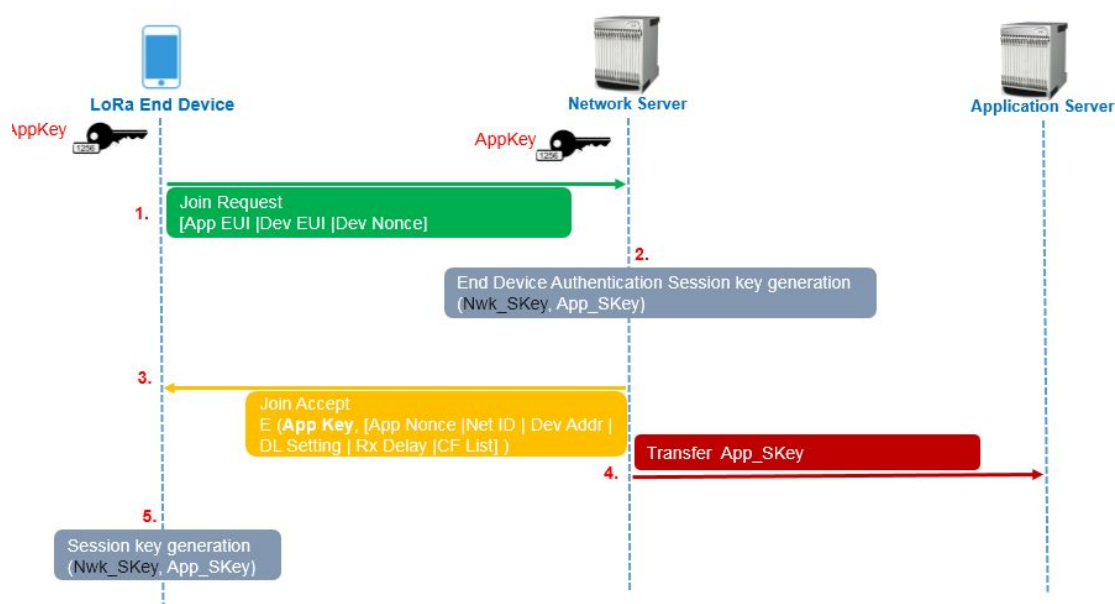


DEV EUI: **00137a1000000ad1**

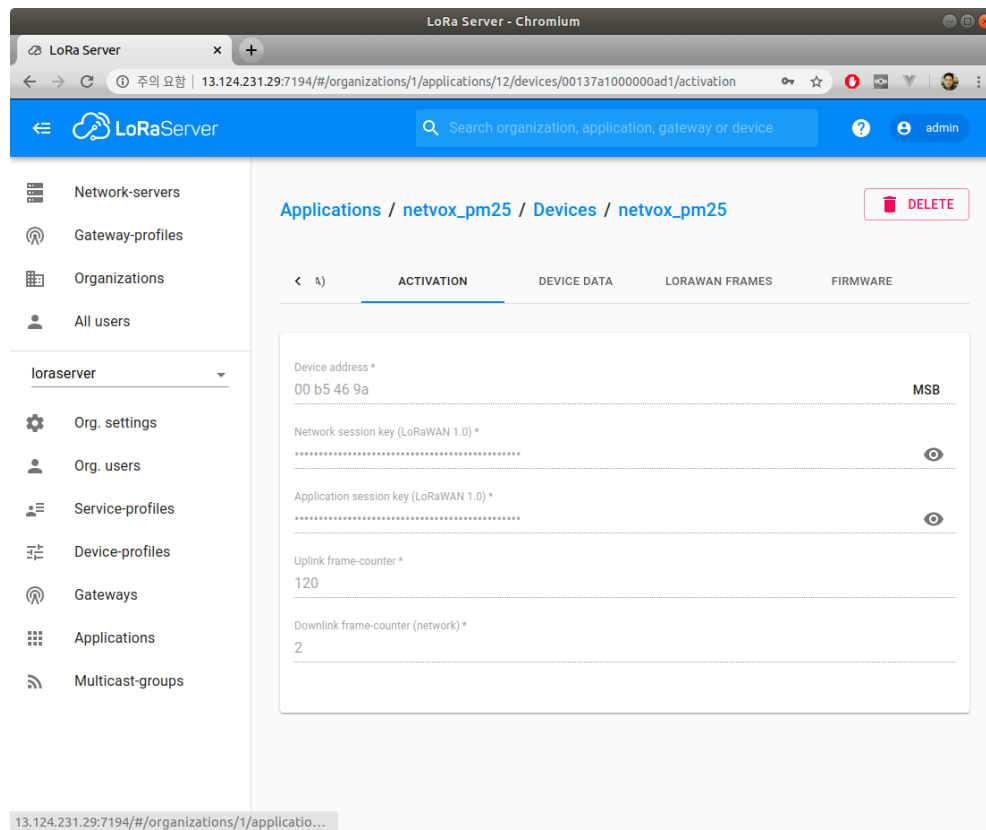
APP KEY: **5A-69-67-42-65-65-41-6C-6C-69-61-6E-63-65-30-39**

APP EUI: **00-13-7A-10-00-00-00-00**

참고: Netvox 애들이 APP Key 정보를 문서에서 제공하지 않아 초기 연결에 애를 먹었다.  
Netvox는 모든 LoRaWAN 제품에 동일한 App Key와 App EUI를 사용한단다 :(



[그림 16.4] OTTA procedure



[그림 16.5] Netvox PM2.5 센서 & LoRaServer 연동 모습

### 3) LoRaServer Codec 설정

Netvox RA0176에서 LoRaWAN을 통해 보내는 packet의 format을 알아야 LoRaServer와 ThingsBoard를 연결할 수가 있다.

[https://cdn.shopify.com/s/files/1/1386/3791/files/Netvox\\_LoRaWAN\\_Application\\_Command\\_V1.9.pdf](https://cdn.shopify.com/s/files/1/1386/3791/files/Netvox_LoRaWAN_Application_Command_V1.9.pdf)

## 1、 ReportDataCmd(UpDirection)

FPort: 0x06

Bytes	1	1	1	Var(Fix=8 Bytes)
	Version	DeviceType	ReportType	NetvoxPayLoadData

**Version**— 1 bytes —0x01—the Version of NetvoxLoRaWAN Application Command Version

**DeviceType**— 1 byte — Device Type of Device

The devicetype is listed in [Netvox LoRaWAN Application Devicetype V1.9.doc](#)

**ReportType** — 1 byte —the Presentation of the NetvoxPayLoadData, according the devicetype

**NetvoxPayLoadData**— Fixed bytes (Fixed =8bytes)—

RA0716/R72616/R72716(alias RA0716Y)/R718SA/R728SA/ R729SA/RA0716A	0x35/0x36/x037/0 x5D/0x5E/0x05F/0 x6A	0x01	Battery(1Byte, unit:0.1V)	Temperature(Signed 2Bytes,unit:0.01° C)	Humidity(2Bytes,u nit:0.01%)	PM2.5(2B yte 1ug/m3)	Reserved(1Bytes,fixe d 0x00)
---	---	------	------------------------------	---	---------------------------------	----------------------------	---------------------------------

[그림 16.6] Netvox RA0176 PM2.5 센서 - payload format

위의 내용을 정리해 보면 다음과 같다.

Version	DeviceType	ReportType	Battery	온도	습도	<b>PM2.5</b>	Reserved
[1 byte]	[1 byte]	[1 byte]	[1 byte]	[2 bytes]	[2 bytes]	[2 bytes]	[1 byte]

PM2.5 센서의 packet format이 확인되었으니, 이를 ThingsBoard로 전달하기 위해 Codec(decoding) 부분에 javascript code를 추가할 차례이다.

Payload codec

## Custom JavaScript codec functions

By defining a payload codec, LoRa App Server can encode and decode the binary device payload for you.

```

1 // Decode decodes an array of bytes into an object.
2 // - fPort contains the LoRaWAN fPort number
3 // - bytes is an array of bytes, e.g. [225, 230, 255, 0]
4 // The function must return an object, e.g. {"temperature": 22.5}
5 function Decode(fPort, bytes) {
6
7   var BatteryVolts = bytes[3] / 10; //TBD
8   var Temperature = ((bytes[4] << 8) | bytes[5]) / 100; //0.01(Celcius)
9   var Humidity = ((bytes[6] << 8) | bytes[7]) / 100; //0.01%
10  var PM25 = (bytes[8] << 8) | bytes[9]; //1ug/m3
11
12  return {
13    "battery": BatteryVolts,
14    "temperature": Temperature,
15    "humidity": Humidity,
16    "pm25": PM25
17  };
18 }

```

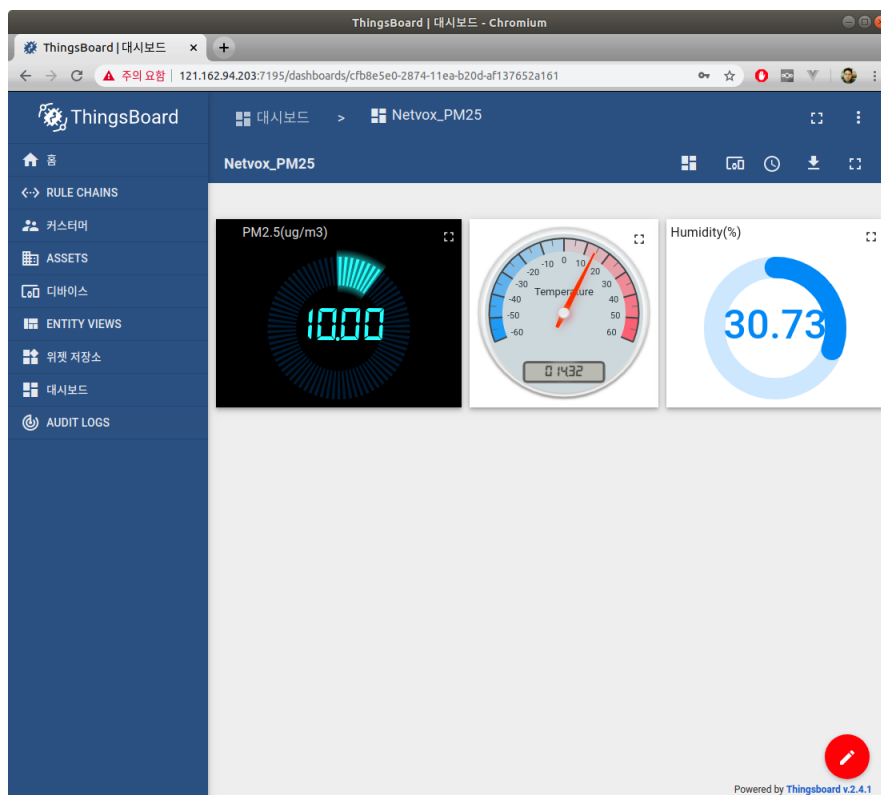
The function must have the signature **function Decode(fPort, bytes)** and must return an object. LoRa App Server will convert this object to JSON.

```

1 // Encode encodes the given object into an array of bytes.
2 // - fPort contains the LoRaWAN fPort number
3 // - obj is an object, e.g. {"temperature": 22.5}
4 // The function must return an array of bytes, e.g. [225, 230, 255, 0]
5 function Encode(fPort, obj) {
6   return [];
7 }

```

[그림 16.7] Netvox RA0176 PM2.5 센서 - Decoding javascript codes



[그림 16.8] Thingsboard Netvox RA0176 PM2.5 센서 연동 모습

참고: PM 측정 단위는 ug/m3(마이크로그램 퍼 세제곱미터)이다. 즉 센서를 통해 10이라는 값을 전달 받았다는 얘기는 1 세제곱 미터(정육면체) 내의 공간에 존재하는 미세먼지의 무게가 100만분의 10 gram 이라는 뜻이 된다.

## 17. LoRaWAN Stack

이번 장에서는 그 동안 미뤄 두었던 LoRaWAN protocol에 대한 얘기를 하고자 한다. 그 동안 얘를 미뤄왔던 이유는 아래와 같은 2가지 이유에서 였다.

1. 과연 우리(2ip)가 정말로 LoRa 관련 사업을 할 수 있을까?
2. 할 수 있다면, 과연 어느 부분까지를 우리가 cover 할 수 있을까?

지금까지 우리는 이 두가지 의문에 대한 답을 찾기 위해 노력해 왔다. 그 결과 아래 내용을 통해 정상적으로 LoRa 통신이 이루어짐을 확인해 보았고, 어느 정도 자신감을 가지게 되었다. 물론, 아직도 KR920/LBT 환경에 적합한 LoRa Gateway를 추가로 검토해야 하는 숙제는 남아 있다.

<지금까지 확인한 내용>

**LoRa Node(상용 제품) ⇒ LoRa Gateway(상용 제품) ⇒ LoRa Server(Open Source) ⇒ ThingsBoard(Open Source)**

흑자는 그렇게 말할지도 모른다. 2ip가 무슨 LoRa 관련 기술이 있느냐고 ... 남들이 만들어 놓은 것을 그냥 연결해 보는 시험을 한게 다이지 않냐고? 맞다. 그래서 지금 부터는 2ip가 LoRa 관련 기술을 확보하기 위해 어떻게 하는지를 보여주려고(?) 한다.

### 1) LoRaWAN open source 검토

<https://github.com/Lora-net>

<https://github.com/Lora-net/LoRaMac-node>

## 2) Embed OS

이 부분은 설명할 내용이 길어지는 관계로 별도의 문서로 정리하도록 하겠다.

