

LoRaWAN LoRaServer Infrastructure Cheat Sheet

Gateway

Packet forwarder Lora
127.0.0.1:1700
local_conf.json & global_conf.json
./basic_pkt_fwd

abstractspacket-forwarder UDP To JSON over MQTT

LoraGateway Bridge
Server: "tcp://yourdomain.com:1883"
systemctl start lora-gateway-bridge

UPD to MQTT(JSON)

LAN 4G/LTE CAT4 WiFi WiFi AP

subscribed topics related to gateways bridges

MQTT over TLS

System in package LGA 13x11x1.1mm

Ultra low power embedded system

RF Front End		
SX1276	SMT3210	VDD
		ADC
		SPI
		I2C
		JTAG
		UART
		PWM
		USB
		SONY CXD5603GF GPS arms swingin/ swimming Precision Level

128 Bytes with CRC
192 Kbytes Flash
20 Kbytes RAM
+20 dBm RF output
High sensitivity -137 dBm

-40C + 85C Operating temperature
Deep sleep 5uA, Standby 9 mA, Receive mode 17.5 mA

Ports to forwards

MQTT : 1883
Web Portal : 8080

Lora Application web Interface

encryption and decryption of the application payloads
application-key of each device and handles the join-accept in case of OTAA activation
Https://myloraserverdomain:8080

Integration

gRPC || RESTful JSON API
downlink frame-counter ↔ acknowledgement

MQTT OR Google Cloud Platform Pub/Sub
Applications: HTTP / Influx / Grafana



	Lora Server	Application Server	Application Data Base	User integration Application
What ?	LoRaWAN mac-layer, Packet management, deduplication	Node management, Join request, (D)Encryption payloads	LoraServer configuration database, will not store payload data	Integration
How ?	MQTT: publish /receiving PostgreSQL: Store Gateway data Redis: non-persistent data	User friendly Web interface Push data to MQTT, HTTP or InfluxDB	PostgreSQL 9.5+ Gateway data Redis 2.6.0 stores all non-persistent data	Custom Applications developed using data API: RESTful & gRPC To create user GUI dash board, ruled based alerts AI, Statistics, prevention ...
Configuration & Usage	Configuration file /etc/loraserver/ loraserver.toml Start service Sudo systemctl start loraserver Log output journalctl -u loraserver -f -n 50	Configuration file /etc/lora-app-server/lora-app-server.toml Start service Sudo systemctl start lora-app-server Log output journalctl -u lora-app-server -f -n 50	Needed User Loraserver_as Test if database is present: Psql -h localhost -U loraserver_as -W loraserver_as	