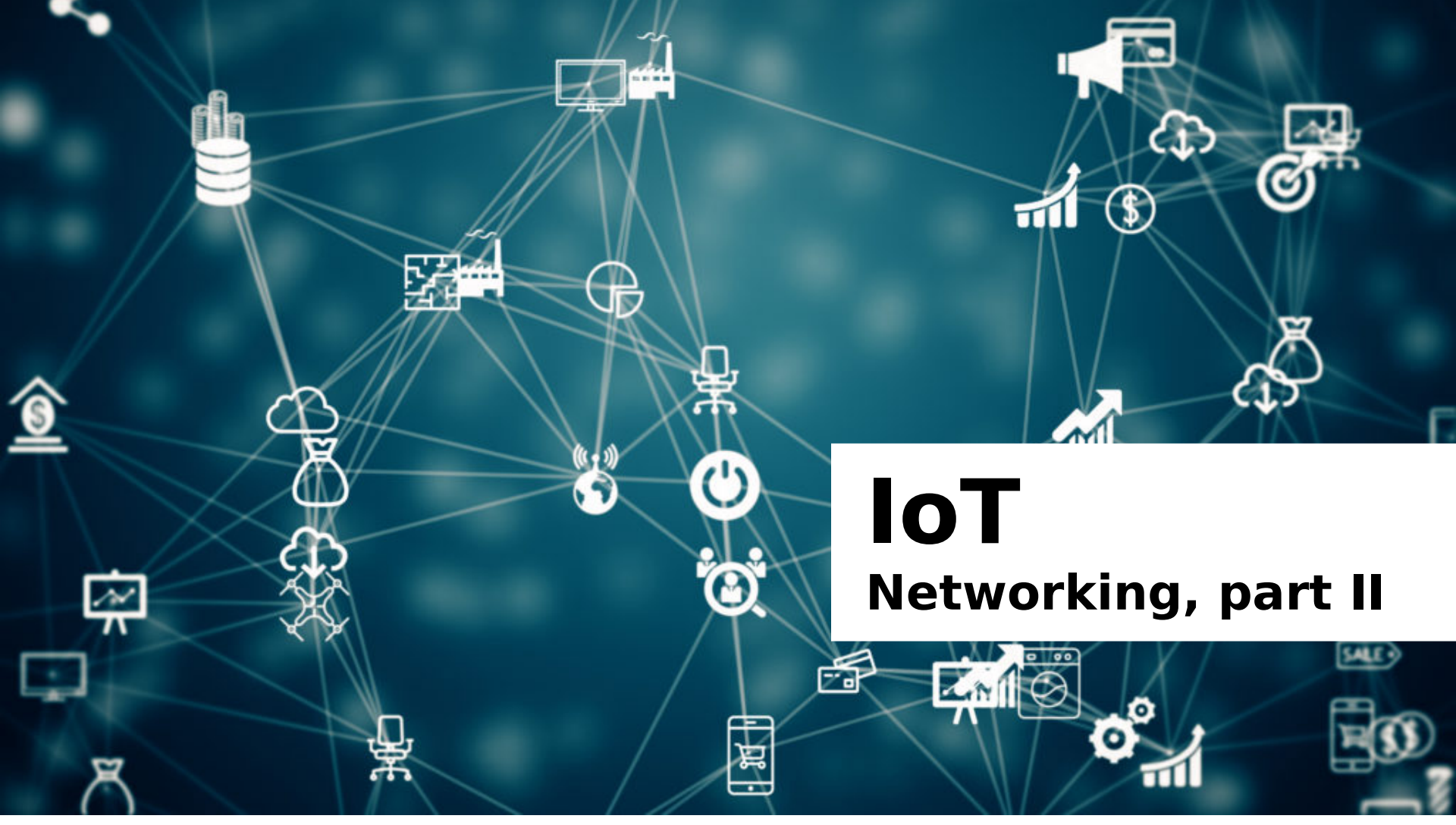


IoT

Networking, part II

Networking / part I / Recap

- Criteria for networking options in IoT:
  Power, reach, bandwidth, cost,
  security, business aspects and more

- Properties of the physical layer: Frequency, bandwidth and their impact

- Basic terms: LPWA(N), LOS/NLOS, Modulation (Spread Spectrum)

- The most relevant options (in 2018) and their main characteristics:
  LoRa, Sigfox, RPMA, Zigbee, Bluetooth, WiFi, Cellular (GSM, LTE-..)

Agenda

- LoRa, LoRaWan, TheThingsNetwork - in depth
- MQTT

- [Networking exercises with

  LoRa, Bluetooth, WiFi

  MQTT, LoRaWan]

LoRa / 1

LoRa is a **proprietary Layer 1 standard** owned by Semtech

Chirp Spread Spectrum (CSS) with forward error coding and interleaving).

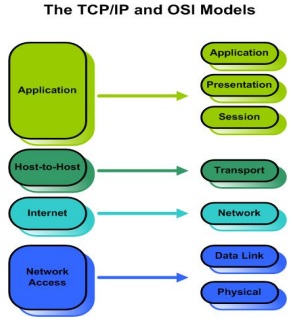
Bandwidth 125/250/500 kHz

Frequency in Europe: **ISM 433/868 Mhz**

Data Rate up to 11 kbps

Focus is on **long range, power efficiency, robustness.**

https://www.semtech.com/lora/what-is-lora



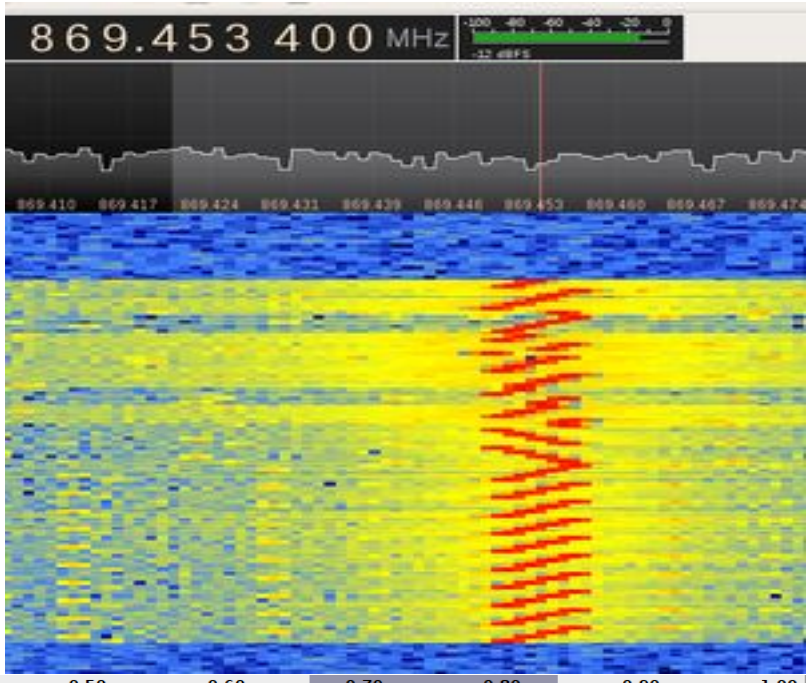
LoRa / 2 / CSS

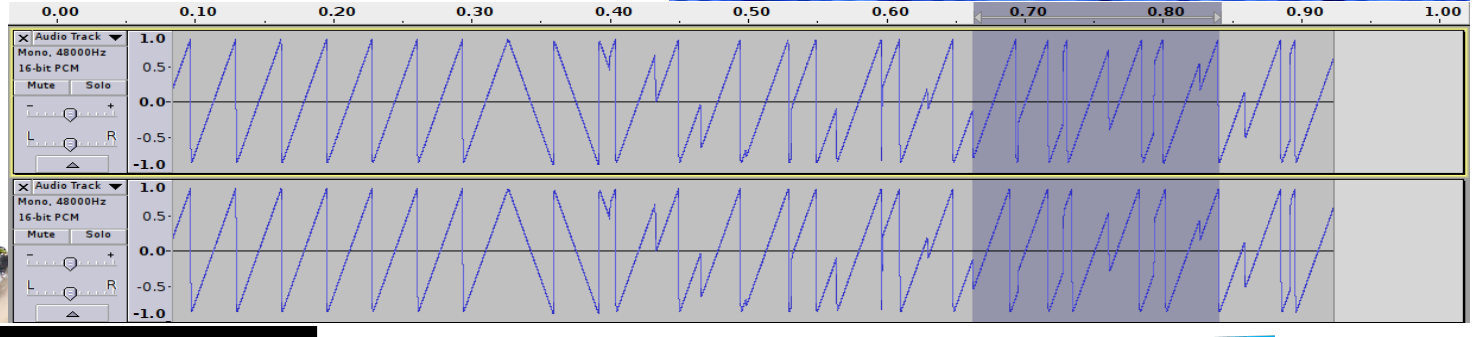
Chirp Spread Spectrum

What is a chirp?

Source: https://revspace.nl/DecodingLora

Preamble (of variable length), here:
10 up, 2 down ->

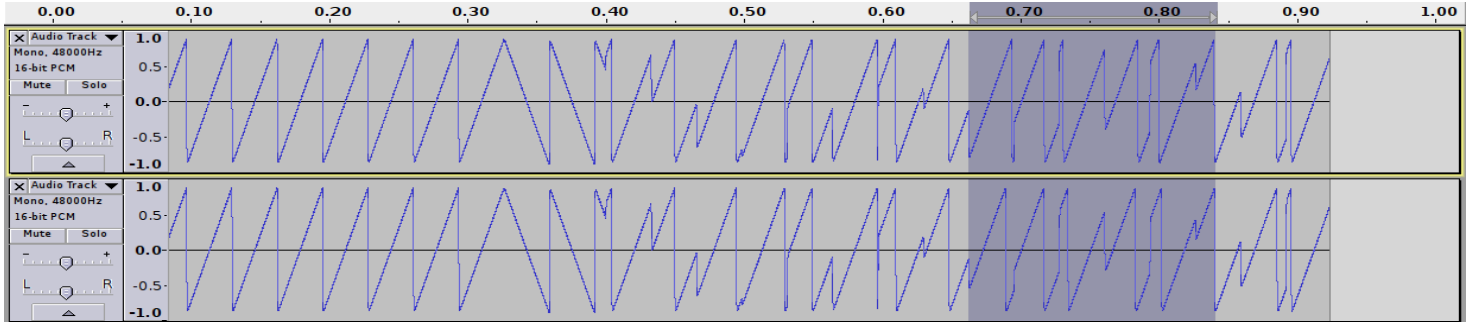




LoRa / 3 / CSS details

Modulation details, reverse engineering:

http://www.semtech.com/images/datasheet/an1200.22.pdf
https://www.lora-alliance.org/portals/0/documents/whitepapers/LoRaWAN101.pdf
https://revspace.nl/DecodingLora#Modulation_basics
https://myriadrf.org/blog/lora-modem-limesdr/
https://static1.squarespace.com/static/54cecce7e4b054df1848b5f9/t/57489e6e07eaa0105215dc6c/1464376943218/Reversing-Lora-Knight.pdf



LoRa / 4 / SF & CR

Spreading Factor SF

$$\mathrm{SF} = \frac{\text{chip rate}}{\text{symbol rate}}$$

(think of it as “one bit is spread out over so and so many pulses”)

Control rate CR, determines depth of forward error coding
(Think of it as saying CCCAAAFFFEEE or CAFECAFECAFE instead of CAFE)

LoRa / 5 / Interleaving

“Mixing up the letters to gain robustness against *burst errors*”

```
Transmitted sentence:              ThisIsAnExampleOfInterleaving...
Error-free transmission:           TIEpfeaghsxlIrv.iAaenli.snmOten.
Received sentence, burst error:    TIEpfe______Irv.iAaenli.snmOten.
after deinterleaving:              T_isI_AnE_amp_eOfInterle_vin_...
```

LoRa / 6 / Data Rate

Data Rate depends on Bandwidth, CR, SF

$$R_b = SF * \frac{\left[\frac{4}{4+CR}\right]}{\left[\frac{2^{SF}}{BW}\right]} * 1000$$

SF = Spreading Factor (6,7,8,9,10,11,12)

CR = Code Rate (1,2,3,4)

BW = Bandwidth in KHz
(10.4,15.6,20.8,31.25,41.7,62.5,125,250,500)

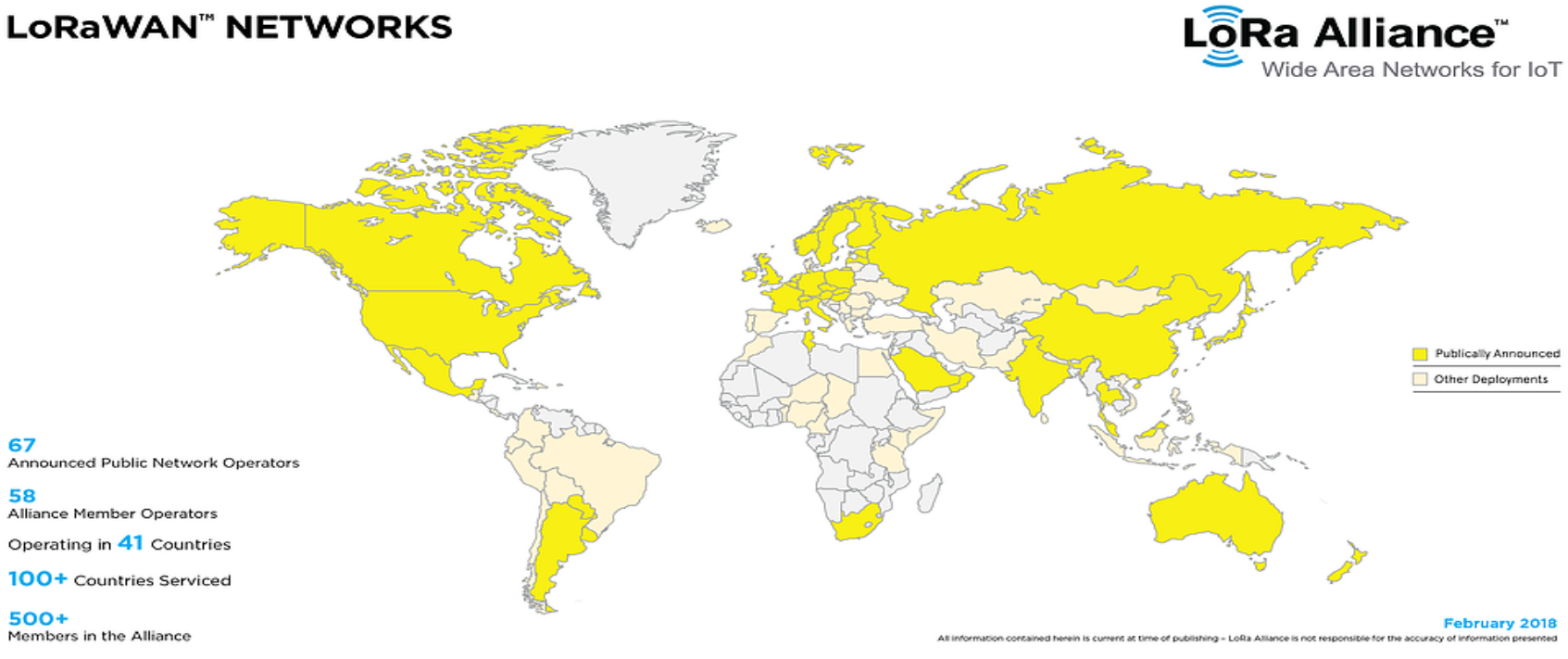
Rb = Data rate or Bit Rate in bps

http://www.rfwireless-world.com/calculators/LoRa-Data-Rate-Calculator.html

LoRaWan / 1

- LoRaWan is an open LPWAN standard building on top of LoRa
- https://www.lora-alliance.org/

LoRaWan / 2


LoRaWAN™ NETWORKS
LoRa Alliance™
Wide Area Networks for IoT
Publically Announced
Other Deployments
67
Announced Public Network Operators
58
Alliance Member Operators
Operating in 41 Countries
100+ Countries Serviced
500+
Members in the Alliance
February 2018
All information contained herein is current at time of publishing – LoRa Alliance is not responsible for the accuracy of information presented

LoRaWan / 3 / concerns

LoRaWAN™ addresses:

architecture
topology
entities
addressing
data rates
mobility
localization
security

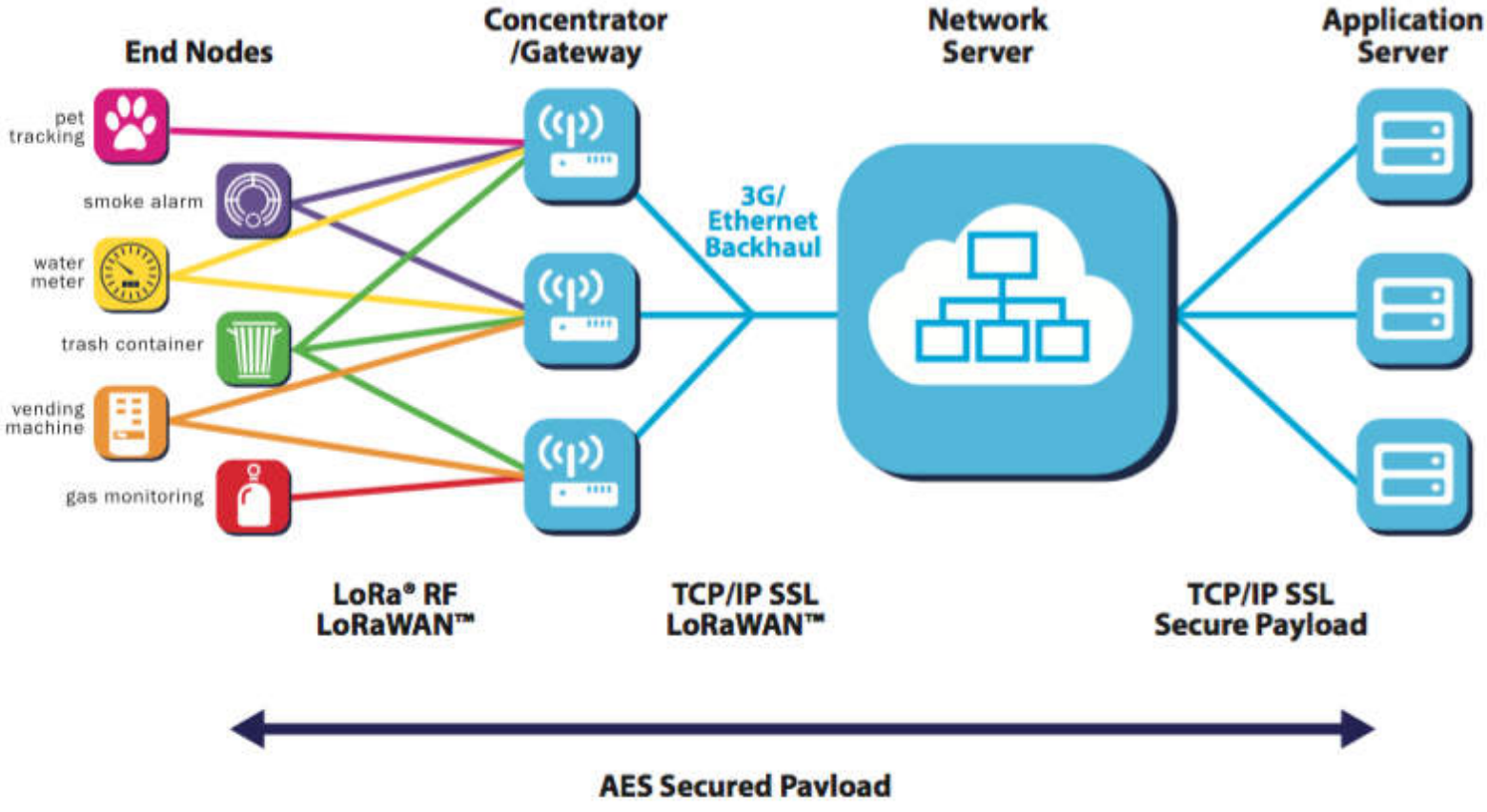
Details:

https://www.lora-alliance.org/What-Is-LoRa/Technology

LoRaWan / 4 / topologies & entities

- **Star-of-stars topology**
- **Gateways** are transparent bridges relaying messages between **end-devices** and a central **network server** in the backend.
- **Gateways are connected** to the network server via **standard IP connections** while end-devices use single-hop wireless communication to one or many gateways.
- All end-point communication generally **bi-directional**, supports **multicast** enabling **software upgrade over the air** or other mass distribution messages

LoRaWan / 5 / architecture



LoRaWan / 6 / device classes

Device classes

A Battery powered, small loads, long breaks, long latency, unicast

B low latency, scheduled receive slots, periodic beacon from gateway, uni/multicast, higher power, 14-30 mA

C no latency, uni/multi, constantly receiving, power hungry

Classes can be dynamically assigned / changed

Source, Details:

https://www.lora-alliance.org/What-Is-LoRa/Technology

LoRaWan / 7 / addressing

Devices and applications

have a 64 bit / 8 byte unique identifier (DevEUI and AppEUI).

When a device joins the network, it receives a dynamic (non-unique) 32-bit / 4 byte address (DevAddr).

Source, Details:

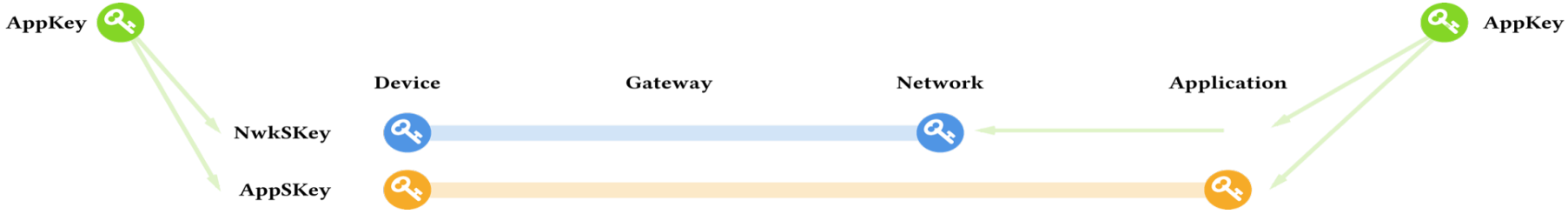
https://www.thethingsnetwork.org/docs/lorawan/

LoRaWan / 8 / Security / keys

Security measures:
three distinct 128-bit AES keys:
The **application key AppKey** is only known by the device and by the application. When a device joins the network (this is called a join or activation), an application session key **AppSKey** and a network session key **NwkSKey** are generated. The NwkSKey is shared with the network, while the AppSKey is kept private.

Source, Details:

https://www.lora-alliance.org/What-Is-LoRa/Technology



LoRaWan / 9 / Security / frame counter

The **frame counter in LoRaWAN** messages is a security measure used to detect **replay attacks**. After validating the MIC, the Broker checks if the Frame counter is valid. As frame counters can only increase, a message with a frame counter that is lower than the last known frame counter should be dropped. Additionally, the Broker has to verify that the gap between the last known frame counter and the counter in the message is not too big. According to the LoRaWAN specification, the maximum gap is 16384.

Source, Details:

https://www.lora-alliance.org/What-Is-LoRa/Technology

LoRaWan / 10 / data rates

LoRaWAN abstracts the PHY data rates of LoRa - for EU / CN:

- EU 863-870 MHz (LoRaWAN Specification (2015), Page 35, Table 14)
- CN 779-787 MHz (LoRaWAN Specification (2015), Page 44, Table 25)
- EU 433 MHz (LoRaWAN Specification (2015), Page 48, Table 31)

DataRate	Modulation	SF	BW	bit/s
0	LoRa	12	125	250
1	LoRa	11	125	440
2	LoRa	10	125	980
3	LoRa	9	125	1'760
4	LoRa	8	125	3'125
5	LoRa	7	125	5'470
6	LoRa	7	250	11'000
7	FSK 50 kbps			50'000

https://blog.dbrgn.ch/2017/6/23/lorawan-data-rates/

LoRaWan / 11 / duty cycles

LoRaWAN implements duty cycle rules made by regulators:

In Europe, duty cycles are regulated by section 7.2.3 of the ETSI EN300.220 standard. This standard defines the following sub-bands and their duty cycles:

- **g** (863.0 – 868.0 MHz): 1%
- **g1** (868.0 – 868.6 MHz): 1%
- **g2** (868.7 – 869.2 MHz): 0.1%
- **g3** (869.4 – 869.65 MHz): 10%
- **g4** (869.7 – 870.0 MHz): 1%

+ duty cycle for join channel: 1%

On top of that, specific networks might have fairplay rules.

The Things Network / 1

The Things Network / 2 / Manifesto

Everything that carries power will be connected to Internet eventually.

Controlling the network that makes this possible means controlling the world. We believe that this power should not be restricted to a few people, companies or nations. Instead this should be distributed over as many people as possible without the possibility to be taken away by anyone. We therefore founded "The Things Network".

The Things Network is an open source, free initiative with the following properties:

It connects sensors and actuators, called "Things", with transceivers called "Things Gateways" to servers called "Things Access".
The first connection is "Over The Air", the second is "Over The Net". The distributed implementation of these concepts is called "The Things Network".
Anyone shall be free to set up "Things" and connect to "Things Gateways" that may or may not be their own.
Anyone shall be free to set up "Things Gateways" and connect to "Things Access" that may or may not be their own. Their "Things Gateways" will give access to all "Things" in a net neutral manner, limited by the maximum available capacity alone.
Anyone shall be free to set up "Things Access" and allow anonymous connections from the Internet. Their "Things Access" will give access to all "Things Gateways" in a net neutral manner, limited by the maximum available capacity alone. Furthermore their "Things Access" will allow connection of other "Things Access" servers for the distribution of data.
The "Over The Air" and "Over The Net" networks shall be protocol agnostic, as long as these protocols are not proprietary, open source and free of rights.
Anyone who perpetrates a "Things Access" or a "Things Gateway" will do so free of charge for all connecting devices and servers.
Anyone making use of the network is allowed to do so for any reason or cause, possibly limited by local law, fully at own risk and realizing that services are provided "as is" and may be terminated for any reason at any moment. The use may be open for anybody, limited to customers, commercial, not-for-profit, or in any other fashion. "The Things Network" providers will not pose restrictions upon its users.

We invite you to sign this Manifesto, and uphold its principles to the best of your abilities.

Source, Details:

The Things Network

THE THINGS
NETWORK

Anyone shall be free to set up "Things" and connect to "Things Gateways" that may or may not be their own.

Anyone shall be free to set up "Things Gateways" and connect to "Things Access" that may or may not be their own. Their "Things Gateways" will give [free] access to all "Things" in a net neutral manner, limited by the maximum available capacity alone.

The Things Network / 3 / Essentials

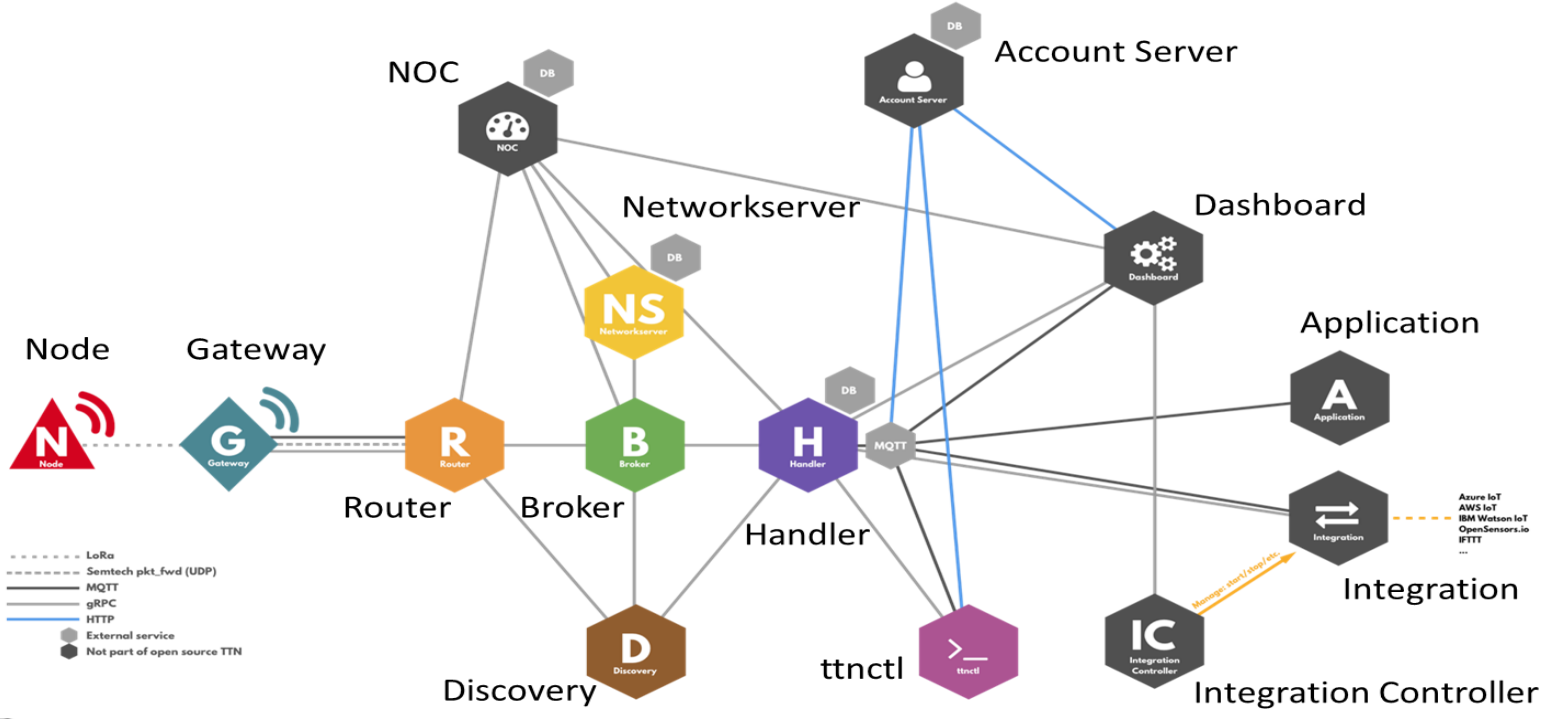
- Open source (except for web GUI)
- Free ... to set up and run their own, in particular:

  Anyone who perpetrates a "Things Access" or a "Things Gateway" will do so **free of charge for all connecting devices and servers** and **without traffic prioritization other than governed by capacity**

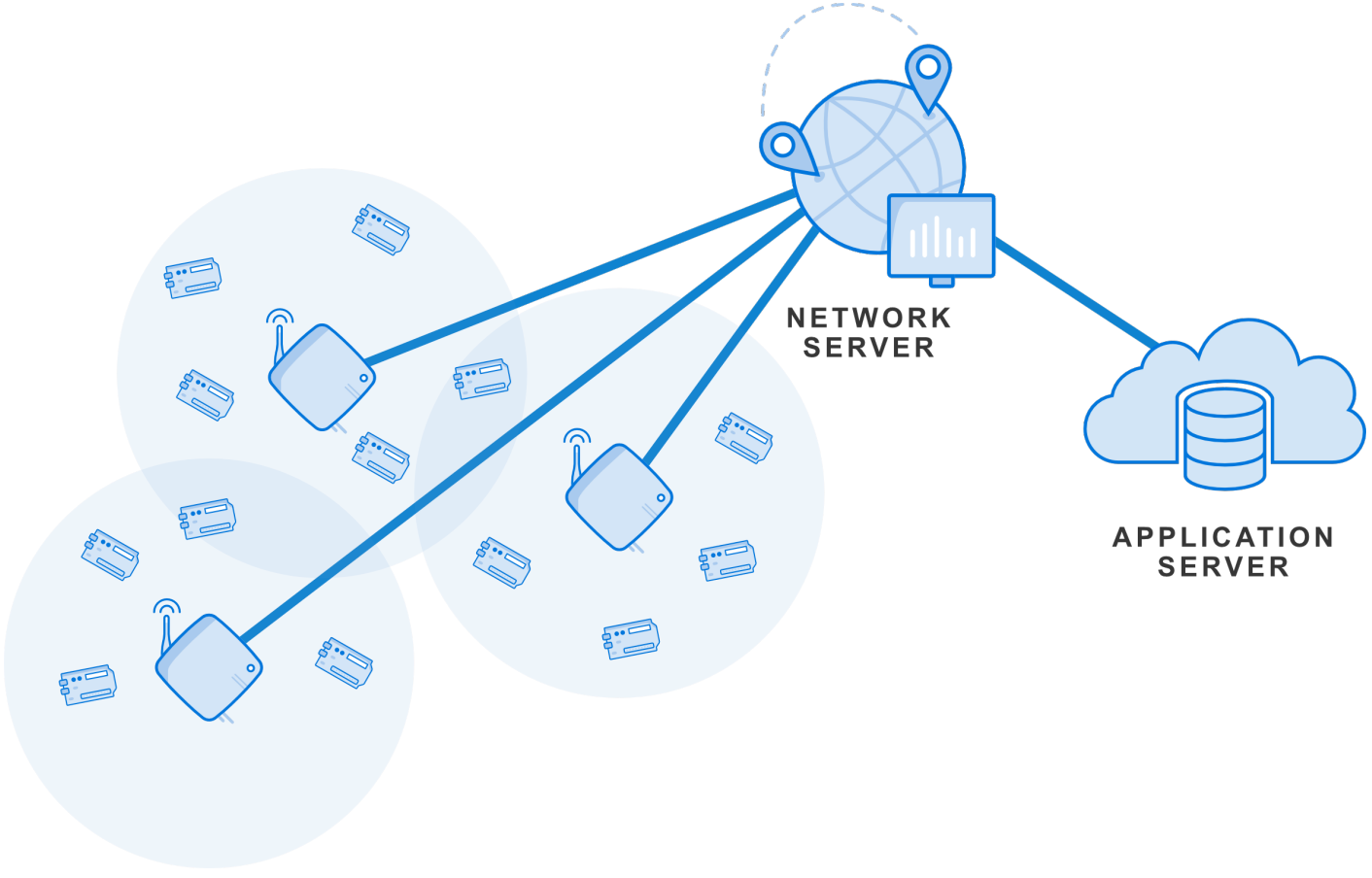
  (-> network neutrality)

  This to some degree explains our current interest in TTN, in an educational context.

The Things Network / 5 / Architecture



The Things Network / 4 / Architecture



The Things Network / 6 / Security

= LoRaWan defined

NwkSKey, AppSKey and AppKey

Challenge of key provision

The Things Network / 7 / Security, cntd

Dynamically activated devices (**OTAA**) use the application key (AppKey) to derive the two session keys during the activation procedure. In The Things Network you can have a default AppKey which will be used to activate all devices, or customize the AppKey per device.

What you will use, in your code:

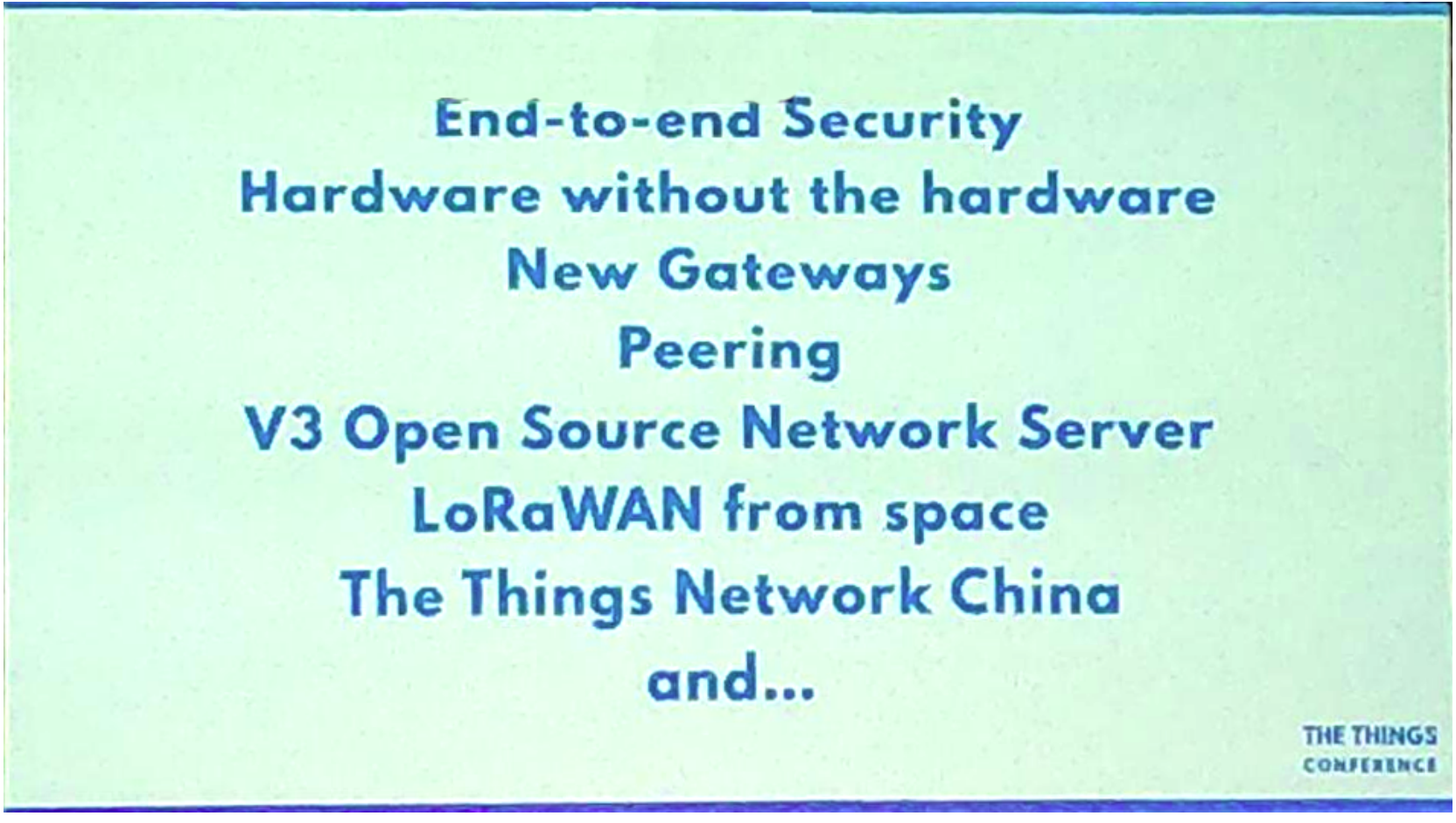
DevEUI, AppEUI, AppKey

Keys will be generated on TTN server, on registration (but can be changed manually)

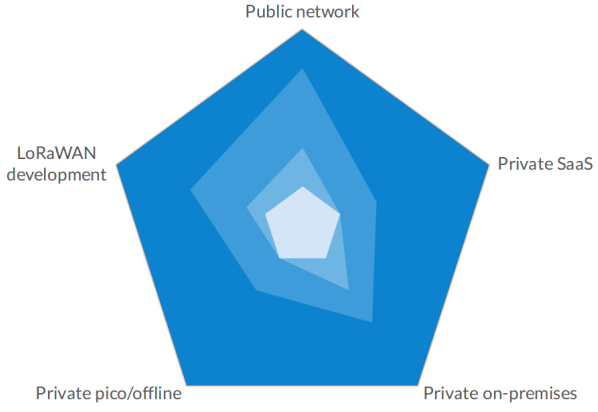
Source, Details:

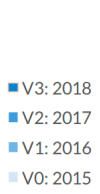
https://www.thethingsnetwork.org/wiki/LoRaWAN/Security

Conference highlights 2019



V3 stack





Source, Johan Stokking, https://www.thethingsnetwork.org

DEPLOYMENT SCENARIOS

Public networks

Public community network and operated public networks

Private networks

Software-as-a-service, on-premises, pico and offline networks

LoRaWAN development

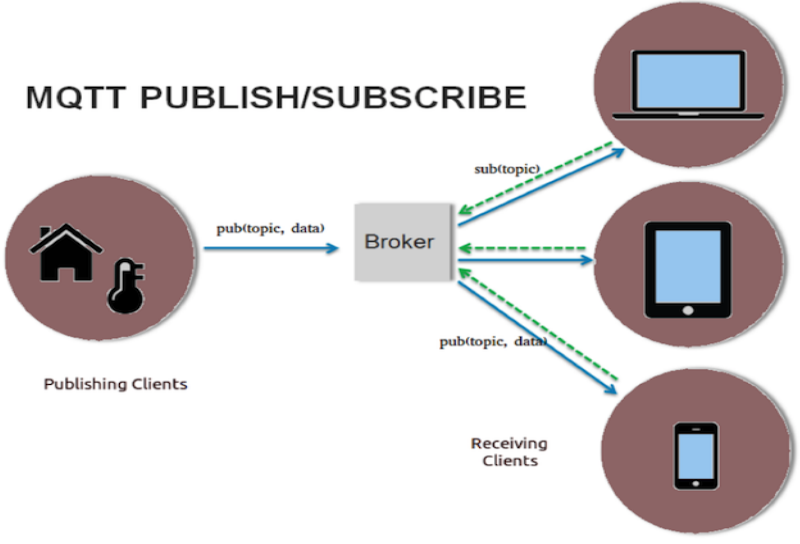
For device makers, application developers and prototype development

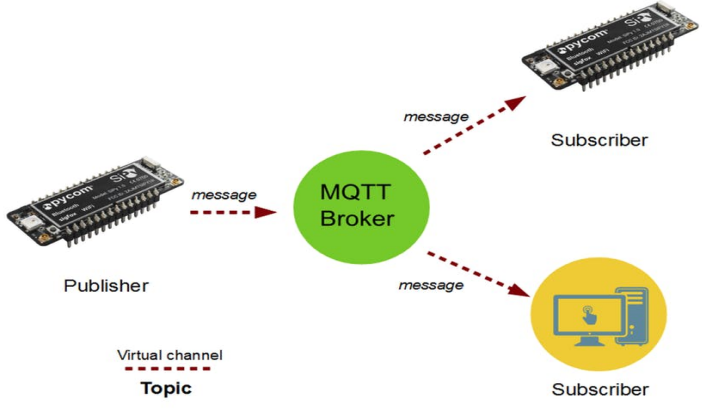
THE THINGS NETWORK

Source, Johan Stokking, https://www.thethingsnetwork.org

MQTT / 1

MQTT (Message Queuing Telemetry Transport) is a **publish-subscribe**-based messaging protocol.
Works on top of **TCP/IP**.
A Message broker or server receives and redistributes





MQTT / 2 / topics

MQTT publishing and subscription is organized by topics:

e.g.
/house/light
Or
/greenhouse/temperature

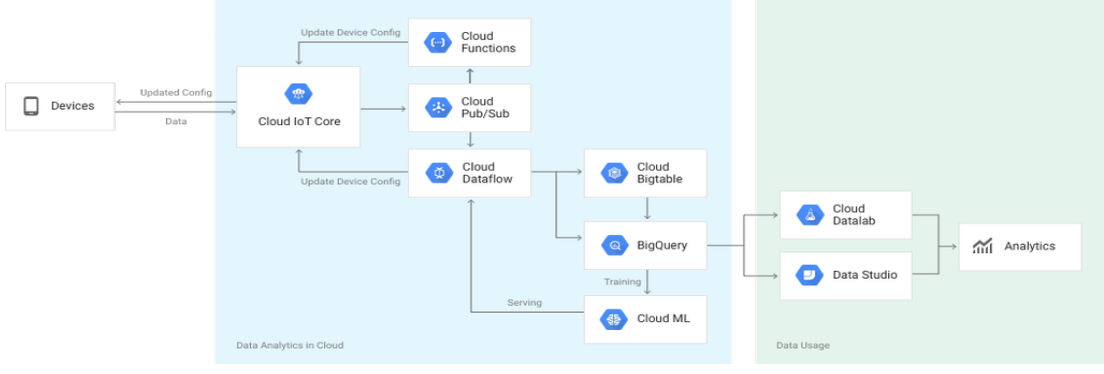
(case sensitive!)

Messages are in **free format,** however, often you will see e.g. json or xml messages.
Some service might restrict message formats.

MQTT / 3 / adoption

MQTT is a widely accepted and deployed standard in services/platforms such as

- Amazon AWS IoT
- Microsoft Azure
- Facebook Messenger (to an unknown degree)
- Google Cloud IoT Core is able to support data “from millions of globally dispersed devices.” Like similar services, Cloud IoT Core supports the standard MQTT and HTTP protocols for talking to devices.



Source: https://techcrunch.com/2018/02/21/googles-cloud-iot-core-is-now-generally-available

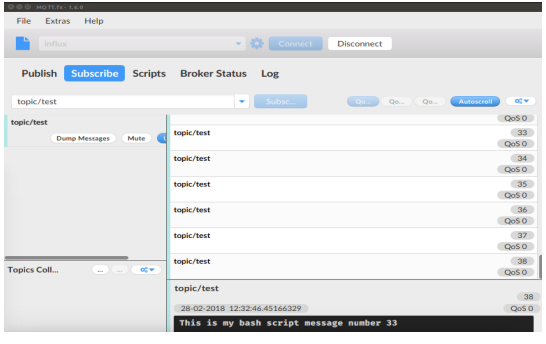
MQTT / 4 / usage

MQTT clients exist for a wide variety of platforms and languages:
https://github.com/mqtt/mqtt.github.io/wiki/software?id=software
https://www.hivemq.com/blog/seven-best-mqtt-client-tools

e.g
MQTT.fx
(available for Win/MacOSX/Linux, http://www.jensd.de/apps/mqttfx/, free)
mqtt-spy
(based on Java 8, http://kamilfb.github.io/mqtt-spy/ , OpenSource)
mosquitto_tools
(commandline, for Win/MacOSX/Linux, https://mosquitto.org/download/ , OpenSource)

And of course for

Arduino (C)
Raspberry (== Debian Linux)
Pycom (micropython)



MQTT / 5 / servers (aka brokers)

A wide choice of servers is available, many of them open source:
https://github.com/mqtt/mqtt.github.io/wiki/servers

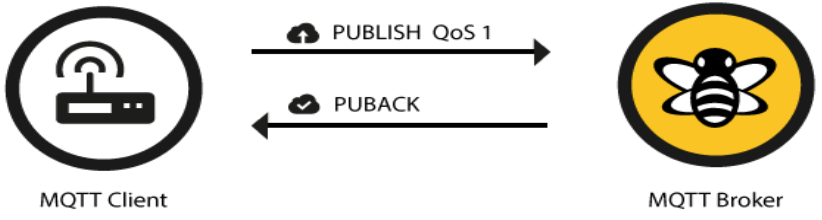
influx.itu.dk runs a public mosquitto server.

MQTT / 6 / QoS

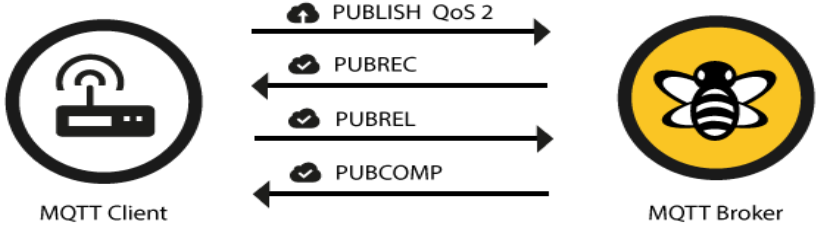
MQTT implements 3 levels of QoS:

At most once (0)
Just send it

At least once (1)
Send and confirm



Exactly once (2)
Send, confirm, reference, stop.



MQTT / 7 / Last Will

MQTT Last Will and Testament

The Last Will and Testament (LWT) feature is used in MQTT to notify other clients about an ungracefully disconnected client. Each client can specify its last will message (a normal MQTT message with topic, retained flag, QoS and payload) when connecting to a broker. The broker will store the message until it detects that the client has disconnected ungracefully, and in tat case, send it out to subscribers.

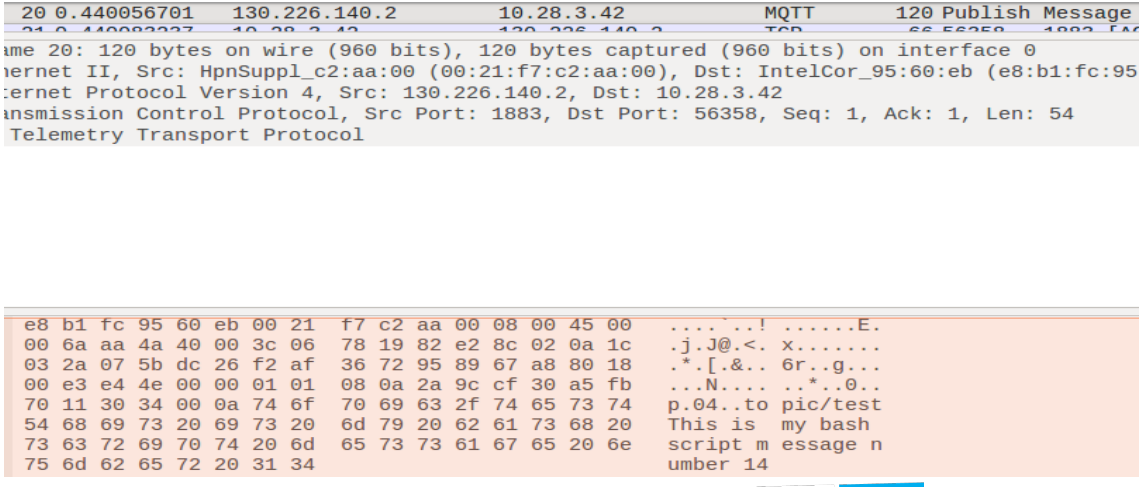
Why is this important?

In a lean communications protocol, dependent devices and services need to know about and have some chance to react to the disappearance of devices.

MQTT / 8 / security

MQTT may be used encrypted or unencrypted - here is mosquittos standard ports:

1883 : MQTT, unencrypted
8883 : MQTT, encrypted
8884 : MQTT, encrypted, client certificate required
8080 : MQTT over WebSockets, unencrypted
8081 : MQTT over WebSockets, encrypted



MQTT / 9 / security

MQTT with TLS/SSL works very much the same as https (which we are familiar with from web usage).
Example with letsencrypt certificates:

```
/etc/mosquitto/conf.d/ssl.conf
listener 8883
certfile /etc/letsencrypt/live/influx.itu.dk/cert.pem
cafile /etc/letsencrypt/live/influx.itu.dk/chain.pem
keyfile /etc/letsencrypt/live/influx.itu.dk/privkey.pem
```

A mosquitto_pub client would publish like this:

```
mosquitto_pub -h influx.itu.dk -p 8883 --capath
/etc/ssl/certs/ -t topic/test -m "encrypted msg"
```

MQTT / 10 / security

MQTT with TLS/SSL works very much the same as https.

In addition to “web style” TLS/SSL, specific client certificates can be demanded.

Username / password protection is also available.

MQTT / 11 / security note

Note:

If devices publish encrypted data,
but the broker allows subscribers to listen unencrypted,
data will be readable on the network!

Take-Aways, Networking 2

- Be able to name protocols and standards on

  PHY, MAC and higher layers -
  Know which protocol belongs where

- Be able to describe the main features of

  LoRa
  LoRaWan

  MQTT

- Be able to explain security measures for these

Exercises

1/ **MQTT**

Find a MQTT client for your laptop OS,
install and configure to send messages to
influx.itu.dk:1883 topic/test
subscribe to topics, create new topics and share

2/ Wireshark monitoring

download and install wireshark.
use wireshark to monitor your communication.
Compare the encrypted and unencrypted communications.

3/ LoPy & MQTT

https://docs.pycom.io/chapter/tutorials/all/mqtt.html
(This guide is for adafruit broker -
how would you send to influx.itu.dk instead?)

4/ LoRaWan and TheThingsNetwork

https://github.com/ITU-PITLab/public/blob/master/Exercises_Networking2.2-LoPy-TTN.txt

For detailed instructions