LoRaWAN - OTA or ABP?

Every newcomer to LoRaWAN eventually needs to know how to get data from a LoRaWAN device. If you're still coming to terms with LoRaWAN itself, check out this gentle introduction to the big picture how and why of LoRaWAN. If you're ready to get your hands dirty, then sooner or later you'll come up against terms like **OTAA**, **ABP**, **DevEUI** and **AppNonce**. You'll probably go searching for answers and be rather underwhelmed.

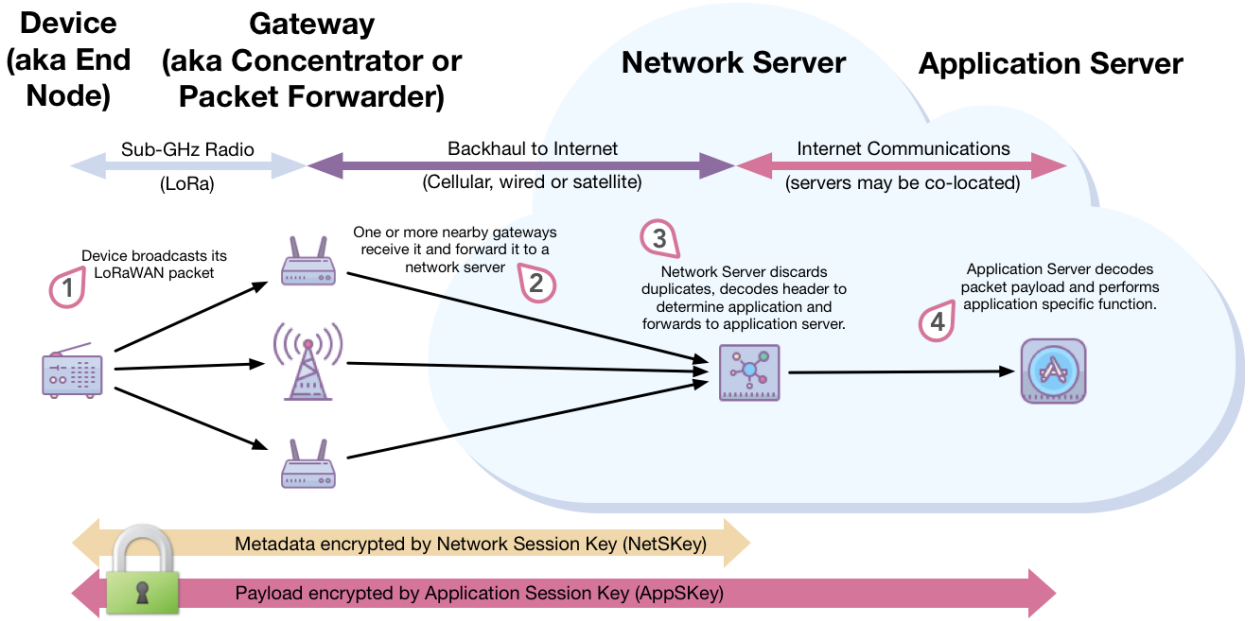
Here, I've attempted to put all the concepts you'll need to wrap your head around this seemingly confusing mess together in one place. I hope this rather succinct one-stop shop saves you months of frustrating work sifting through nuggets of information scattered elsewhere. It's worth nothing that the concepts described here only apply to LoRaWAN 1.0 - not the upcoming LoRaWAN 1.1. And before I get any nitpickers: this article is deliberately incomplete - I wanted to focus on the important concepts you'll need to get going.

LoRaWAN data begins life as a wireless LoRa radio transmission from Device to Gateway. There it is sent via a communications backhaul (such as cellular or Ethernet) to a Network Server in the cloud. The Network Server performs the functions necessary for LoRaWAN to work before forwarding the data to the appropriate Application Server for processing*. The data is encrypted along the whole journey, first by a Network Session Key (**NwkSKey**) and then via an Application Session Key (**AppSKey**).

The Network Server performs the functions necessary for LoRaWAN to work before forwarding the data to the appropriate Application Server for processing. The data is encrypted along the whole journey, first by a Network Session Key (**NwkSKey**) and then via an Application Session Key (**AppSKey**).

Q. But where do NetSKey and AppSKey come from?

A. Depends on the "activation" method: OTAA or ABP



*Note that in reality there is another server called the Join Server which assists the Network and Application Servers during the join process. It provides security functionality important to manufacturers and device makers, but needlessly complicates the picture here.

NEWIE VENTURES

OTAA: Over-The-Air Activation

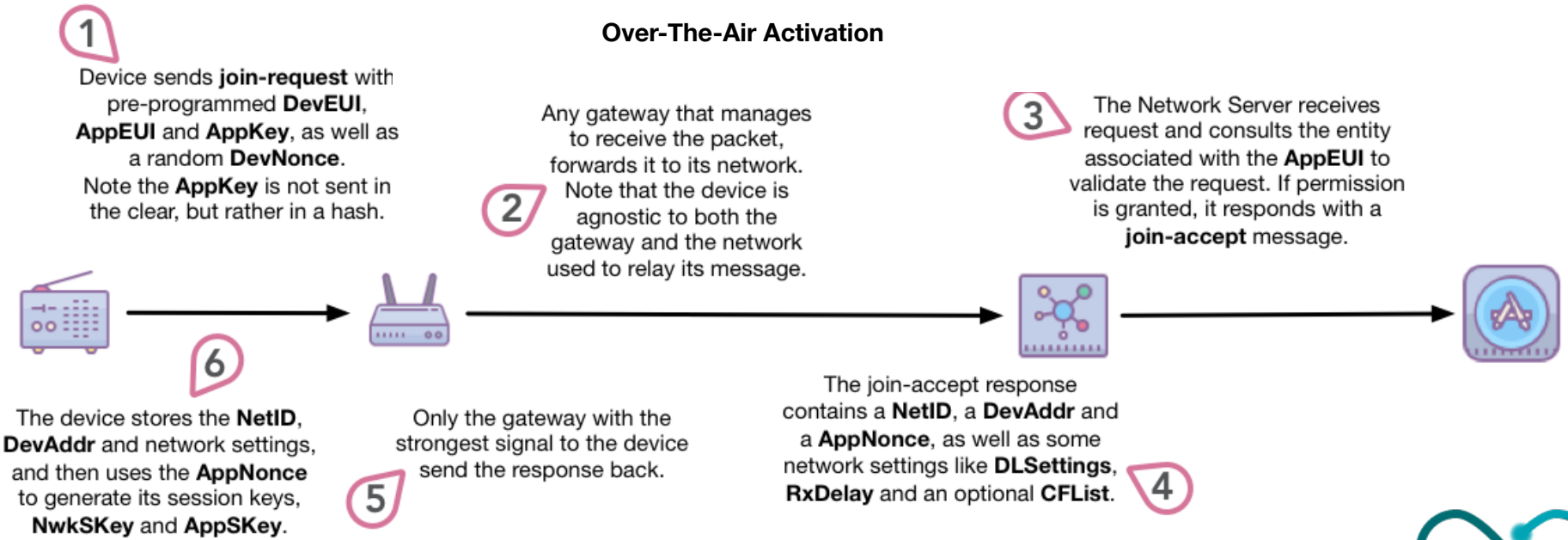
In OTAA, a device is given a **DevEUI**, an **AppEUI** and an **AppKey**. Confusingly enough, the AppKey is used to generate the session keys, **NwkSKey** and **AppSKey**. So watch out for that sneaky 's' or you'll get royally confused. To activate, the device sends a join request and uses the join response to derive the session keys **NwkSKey** and **AppSKey**. The device may store those keys and continue to use them to communicate. If they are lost or the network chooses to expire them, the device must re-join to generate new keys.

Pros:

- Session keys are only generated when required, so cannot be compromised prior to activation.
- If the device changes to a new network, it can re-join to generate the new keys - rather than having to be re-programmed.
- Network settings like **RxDelay** and **CFList** can be specified at join time.

Cons:

- A scheme is required to pre-program each device with a unique **DevEUI** and **AppKey**, and the correct **AppEUI**.
- The device must support the join function and be able to store dynamically generated keys.



NEWIE VENTURES

ABP: Activation By Personalisation

In ABP (Activation By Personalisation), a device does not need a **DevEUI**, an **AppEUI** or an **AppKey**. Instead the session keys **NwkSKey** and **AppSKey** are preprogrammed into the device and the device is pre-registered on the network. When the device wants to communicate, it does so using the session keys without having to use a join procedure first.

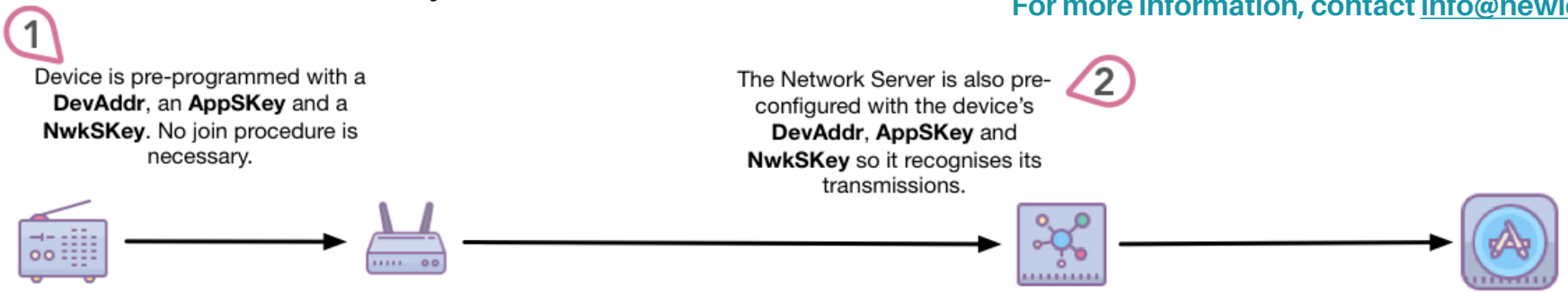
Pros:

- The device does not need the capability or resources to perform a join procedure.
- The device does not need to decide whether a join is necessary at any point, since it is never necessary.
- No scheme is necessary to specify a unique DevEUI or AppKey.

Cons:

- The scheme to generate the **NwkSKey** and **AppSKey** must ensure they are unique, to prevent a widespread breach if a single device is compromised. And the scheme must be secure to prevent the keys being obtained or derived by rogue parties.
- If the device is compromised at any time, even before activation, the keys may be discovered.
- Network settings cannot be specified at join time.
- Events that warrant a change of keys (for example, moving to a new network, the device being compromised, or the keys being expired) require a re-programming of the device.

Activation By Personalisation



Q. So do I use OTAA or ABP?

With care, either method can be just as secure and effective as the other. But unless you have particular requirements,OTAA is the easiest way to achieve some basic security and flexibility when it comes time to deploy. ABP is more likely to be useful in the prototyping stage when you need complete control over the activated devices. Either way, remember that unless you're doing an OTAA join before every message, you have to keep your frame counter up to date for subsequent messages to get through.

Glossary

EUI	Extended Unique Identifier - a globally unique ID
DevEUI	Device EUI - set by manufacturer, unique per device
AppEUI	Application EUI - identifies the end application
App Key	Application Key - used in OTAA to generate session keys
DevAddr	Device Address - identifies a device on a particular network
NwkSKey	Network Session Key - encrypts the packet metadata
AppSKey	Application Session Key - encrypts the packet payload
Nonce	A number that is only used once in cryptographic messages
DevNonce	A random nonce sent from device to network during a join request to prevent rogue device re-playing the join request
AppNonce	A nonce sent from network to device during a join response that allows the device to generate the session keys
NetID	Network Identifier - uniquely identifies the network
DLSettings	Downlink Settings - data rates to be used for receiving
RxDelay	Receive Delay - time between transmit and receive
CFList	Channel Frequency List - frequency setting for each channel

For more information, contact info@newieventures.com.au

NEWIE VENTURES