

# Table of Contents

<b>Фильтрация и буферизация пакетов</b> .....	3
<b>Фильтрация LoRaWAN пакетов</b> .....	3
Фильтрация по полю FPort .....	3
Фильтрация по полю DevAddr .....	4
<b>Буферизация LoRaWAN пакетов</b> .....	5



# Фильтрация и буферизация пакетов

## Фильтрация LoRaWAN пакетов

LoRaWAN сети в силу своих особенностей разворачиваются там, где подключение устройств к другим сетям (WiFi, LTE...) либо **затруднено** либо **дорого**. Поэтому будет логичным предположить, что на практике концентраторы (gateway) LoRaWAN обычно имеют весьма **ограниченное** подключение к сети **Интернет**. Например, концентратор может быть подключен к сети через LTE соединение. В некоторых регионах мира такое соединение может иметь весьма серьезные как административные (высокая стоимость трафика) так и технические ограничения. Например на большом удалении от урбанизированных районов могут быть доступны только сети старых поколений, в которых скорость передачи данных имеет серьезные ограничения.

В связи с вышеизложенным перед администраторами LoRaWAN сетей стоит задача по **экономии** ресурса канала связи шлюза к сети Интернет.

Одним из важных аспектов решения такой задачи является фильтрация **чужого** восходящего (далее "UP") трафика в сторону "Network Server". Как мы знаем в референсных решениях весь "UP" трафик **без разбора** пересылается концентратором на "Network Server". В связи с тем, что в последнее время LoRaWAN сети получают широкое распространение (растет количество проектов и падает цена чипов) в современном эфире бок о бок работает много сетей, конфигурации которых не согласованы друг с другом. Таким образом LoRaWAN концентратор вынужден пересылать большое количество **чужого** трафика на "Network Server". Если не предпринимать дополнительных усилий чужой трафик может **затруднить** эксплуатацию сети либо привести к **дополнительным расходам** на трафик. Одним из способов решения данной проблемы является **фильтрация** LoRaWAN трафика проходящего через концентратор.

В семействе продуктов **JOGL** реализована фильтрация LoRaWAN трафика по двум информационным элементам фреймов:

### Фильтрация по полю FPort

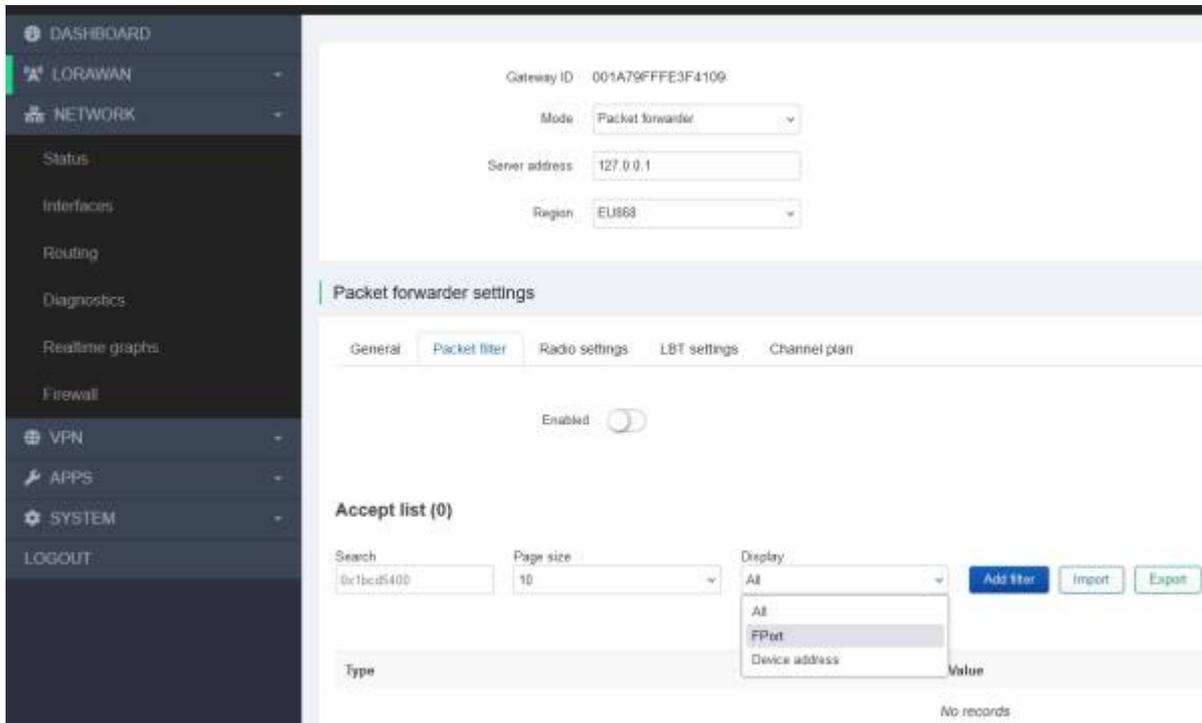
#### 4.3.2 Port field (FPort), [LoRaWAN-Link-Layer-Specification-v1.0.4.pdf](#)

В том случае, если LoRaWAN фрейм **содержит** элемент **FPort** производится проверка его значения на содержание в базе данных "**FPort white list**". Если значение FPort не находится в "**белом списке**", то фрейм **НЕ** пересылается на "Network Server". В противном случае все происходит как обычно.

Для того, чтобы фильтрация по значению FPort имела **смысл** администратор LoRaWAN сети должен предварительно произвести настройку ПО конечных устройств для того, чтобы устройства использовали **определенное** значение FPort. Не все конечные устройства поддерживают данную настройку. Тем не менее даже в том случае, если все "наши" устройства используют фиксированное значение FPort которое мы не можем менять (к примеру 1) все равно **есть смысл** активировать фильтр для того, чтобы **обезопасить** себя в той ситуации, если завтра на наших частотах кем то будет развернута **большая** сеть с

использованием FPort равным (к примеру 100). В таком случае мы не будем тратить ресурсы на пересылку **бесполезного** с точки зрения нашей сети “UP” трафика в сторону “Network Server”.

Модификация базы данных “FPort white list” осуществляется в ручном режиме через соответствующую WEB/UI форму:



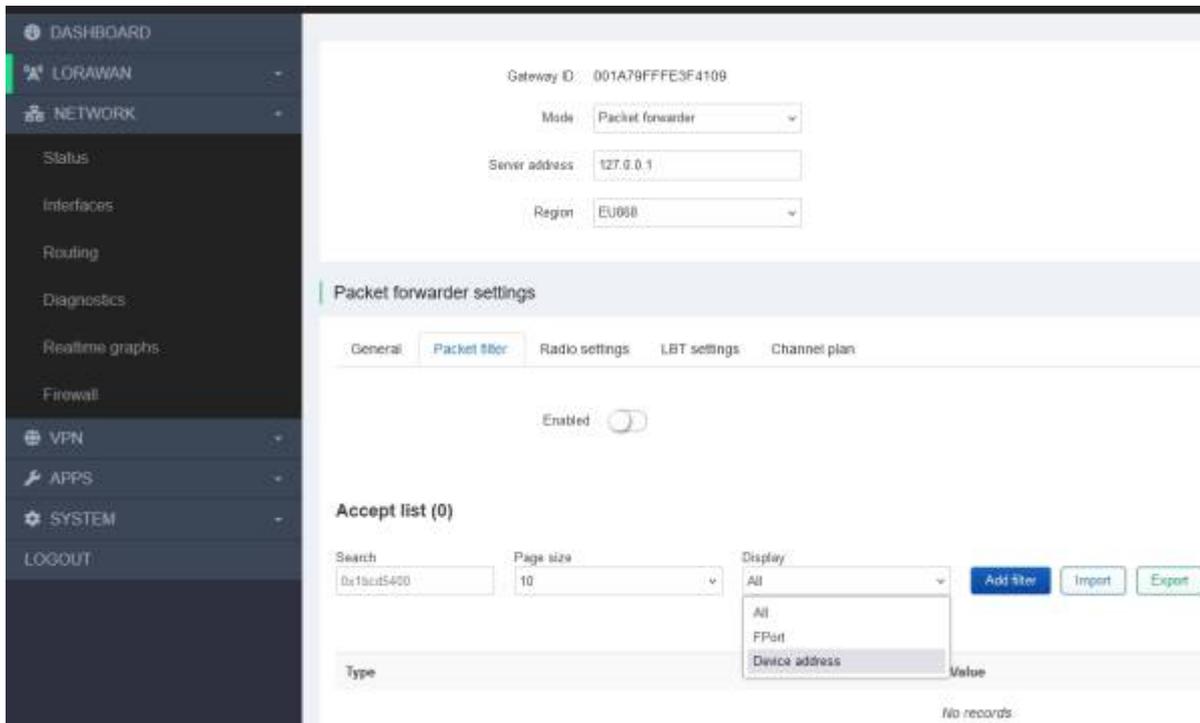
Следует отметить, что фильтрация пакетов осуществляется до буферизации. Таким образом фильтрация экономит не только сетевой трафик но также место в очереди пакетов (смотрите раздел [Буферизация](#)).

## Фильтрация по полю DevAddr

### 4.3.1 Frame header (FHDR) DevAddr, [LoRaWAN-Link-Layer-Specification-v1.0.4.pdf](#)

Значение элемента **DevAddr** проверяется на содержание в базе данных “**DevAddr white list**”. Если значение DevAddr не находится в “**белом списке**”, то фрейм **НЕ пересылается** на “Network Server”. В противном случае все происходит как обычно.

Модификация базы данных “DevAddr white list” осуществляется в ручном режиме через соответствующую WEB/UI форму:



Следует отметить, что в режим фильтрации по значению DevAddr имеет **смысл** только в случае использования активации методом **пред-настройки конечного устройства - режим АВР**. В случае использования режима активации OTAA “Network Server” инициирует генерацию **динамического** значения DevAddr и у нас нет никакой гарантии, что для заданного конечного устройства будет использоваться **определенное значение DevAddr**.

## Буферизация LoRaWAN пакетов

Как было описано выше на практике шлюзы LoRaWAN могут иметь весьма **ограниченное** подключение к сети Интернет. Одной из типичных проблем с которыми сталкиваются администраторы сетей является временное **отсутствие связи** шлюза с “Network Server”. В высоко нагруженных сетях за 1 час отсутствия связи с “Network Server” могут потеряться данные от сотен если не тысяч устройств. Концепция построения LoRaWAN сетей подразумевает **некоторую потерю** сообщений. Но в данной ситуации мы имеем дело со сценарием, когда происходит **массовая безусловная потеря** пакетов по весьма прозаической причине.

Решить данную проблему позволяет **буферизация LoRaWAN фреймов**.

В основе буферизации лежит весьма простой принцип. В случае, если “Network Server” **перестает отвечать на сообщения шлюза** эти сообщения не удаляются и начинают **складываться в специальную очередь** (FIFO). Глубина очереди достаточна для накопления сообщений в течение суток. Попав в такую ситуацию шлюз **периодически пытается передать часть** сообщений на “Network Server”. Когда связь с “Network Server” **восстанавливается** шлюз передает накопившиеся сообщения **соблюдая определенную осторожность** (мы не должны пытаться передать все сообщения **сразу** так как есть вероятность **перегрузить** и без того нестабильный канал связи).

Очередь сообщений хранится в оперативной памяти и **будет потеряна при перезапуске шлюза**.

From:

<https://wiki.jooby.eu/> -

Permanent link:

[https://wiki.jooby.eu/playground/packet\\_filtering\\_buffering](https://wiki.jooby.eu/playground/packet_filtering_buffering)

Last update: **2024/11/14 18:07**