

# **Руководство по интеграции**

**«AirBit Network Server v1.2»**

г. Ярославль, 2018 г.



# Содержание

1. Общие положения.....	4
1.1. Полное наименование системы и ее условные обозначения.....	4
1.2. Заказчик.....	4
1.3. Исполнитель.....	4
1.4. Основания разработки и внедрения.....	4
1.5. Плановые сроки начала и окончания работ.....	4
1.6. Источники и порядок финансирования работ.....	4
1.7. Порядок оформления и предъявления заказчику результатов работ.....	4
1.9. Определения обозначения и сокращения.....	5
2. Описание прикладного API для взаимодействия с AirBit Network Server.....	7
2.1. HTTP REST API.....	7
2.1.1. Получение полезной нагрузки для одного устройства.....	7
2.1.2. Получение полезной нагрузки для нескольких устройств.....	10
2.1.3. Отправка полезной нагрузки на конечное устройство.....	13
2.1.4. Отправка полезной нагрузки в группу многоадресной рассылки (Multicast).....	15
2.1.5. Получение последнего (первого) пакета для списка устройств.....	16
2.2. Webhook.....	19
2.3. MQTT (Mosquitto).....	22
2.3.1. Получение и обработка данных в интерактивном режиме.....	22
2.3.2. Отправка полезной нагрузки на конечное устройство.....	23
2.4. AMQP (RabbitMQ).....	24
2.4.1. Получение и обработка данных в интерактивном режиме.....	24
2.4.2. Отправка полезной нагрузки на конечное устройство.....	25

## **1. ОБЩИЕ ПОЛОЖЕНИЯ**

### **1.1. ПОЛНОЕ НАИМЕНОВАНИЕ СИСТЕМЫ И ЕЕ УСЛОВНЫЕ ОБОЗНАЧЕНИЯ**

Сервер управления сетями стандарта LoRaWAN™ «AirBit Network Server»

Сокращенное наименование: AirBit Network Server

Условное обозначение: AirBit LNS

### **1.2. ЗАКАЗЧИКИ**

ООО «АирБит», Ярославль, ул. Республиканская, 3кб, офис 213

Airbit GmbH Wolfener Str. 32-34 12681 Berlin

### **1.3. ИСПОЛНИТЕЛЬ**

Индивидуальный предприниматель Соколов Сергей Витальевич

### **1.4. ОСНОВАНИЯ РАЗРАБОТКИ И ВНЕДРЕНИЯ**

Договор № 2019070101 от «01» июля 2019 г.

**Система должна разрабатываться в соответствии со следующими нормативными документами:**

1. Федеральный закон РФ от 27.07.2006 г. №149-ФЗ «Об информации, информационных технологиях и о защите информации»;
2. ГОСТ 34.603-92 «Виды испытаний автоматизированных систем»;
3. Санитарно-эпидемиологические нормативы СанПиН 2.2.2./2.4.1340-03 «Гигиенические требования к персональным электронно-вычислительным машинам и организации работы» от 03.06.2003 г.

### **1.5. ПЛАНОВЫЕ СРОКИ НАЧАЛА И ОКОНЧАНИЯ РАБОТ**

Максимальный срок выполнения работ по договору до 25.12.2018.

### **1.6. ИСТОЧНИКИ И ПОРЯДОК ФИНАНСИРОВАНИЯ РАБОТ**

Заказчик реализует проект за счет собственных сил и средств.

### **1.7. ПОРЯДОК ОФОРМЛЕНИЯ И ПРЕДЪЯВЛЕНИЯ ЗАКАЗЧИКУ РЕЗУЛЬТАТОВ РАБОТ**

Разработанная система передается в виде функционирующего программно-аппаратного комплекса на базе средств вычислительной техники Заказчика в сроки, установленные внутренними регламентирующими документами и приказами Заказчика. Приемка системы осуществляется комиссией в составе уполномоченных представителей Заказчика и Исполнителя.

Порядок предъявления системы, ее испытаний и окончательной приемки определяется техническим заданием и договорными условиями на разработку.

## 1.9. ОПРЕДЕЛЕНИЯ ОБОЗНАЧЕНИЯ И СОКРАЩЕНИЯ

<b>Сокращение</b>	<b>Описание</b>
<b>БД</b>	База данных
<b>ГОСТ</b>	Государственный стандарт
<b>API</b>	Application programming interface (интерфейс прикладного программирования — методы и способы взаимодействия с внешними информационными системами)
<b>AMQP</b>	AMQP (Advanced Message Queuing Protocol) — открытый протокол для передачи сообщений между компонентами системы
<b>HTTP</b>	Сокр. от англ. Hyper Text Transfer Protocol – «протокол передачи гипертекста» – протокол передачи данных.
<b>HTTPS</b>	Сокр. от англ. Hyper Text Transfer Protocol Secure – расширение протокола HTTP, поддерживающее шифрование. Данные, передаваемые по протоколу HTTPS, «упаковываются» в криптографический протокол SSL или TLS, тем самым обеспечивается защита этих данных.
<b>JSON</b>	JavaScript Object Notation — структурированный текстовый формат обмена данными, основанный на JavaScript. Как и многие другие текстовые форматы, JSON легко читается людьми.
<b>LORA</b>	Технология модуляции обеспечивающая физическое взаимодействие в сетях M2M
<b>LoRaWAN</b>	Открытый протокол на базе модуляции LORA для высокоёмких (до 1 млн. устройств в одной сети) сетей с большим радиусом действия и низким энергопотреблением
<b>MQTT</b>	Message queue telemetry transport - сетевой протокол, работающий поверх TCP/IP, ориентированный для обмена сообщениями между устройствами по принципу издатель-подписчик
<b>REST, RESTful</b>	Технология сервис-ориентированных приложений упор в которой делается на возможность использования произвольного формата (Текст, XML, JSON и т.д.). Параметры запросов передаются посредством стандартных возможностей протокола HTTP.
<b>Webhook</b>	Отправка HTTP-запроса на внешний URL

	пользователя при наступлении какого-либо события.
--	---

## 2. ОПИСАНИЕ ПРИКЛАДНОГО API ДЛЯ ВЗАИМОДЕЙСТВИЯ С AIRBIT NETWORK SERVER

### 2.1. HTTP REST API

**Механизм авторизации:** HTTP Basic Auth.

Предназначено для решения следующих прикладных задач:

1. Получение полезной нагрузки, которая была направлена конечными устройствами в сеть;
2. Получение полезной нагрузки, которая была направлена приложениями на конечные устройства;
3. Отправка данных на конечные устройства;
4. Отправка данных в группу многоадресной рассылки (multicast).

#### 2.1.1. ПОЛУЧЕНИЕ ПОЛЕЗНОЙ НАГРУЗКИ ДЛЯ ОДНОГО УСТРОЙСТВА

Данные возвращаются отсортированными по времени в порядке убывания (сначала новые). Для получения могут использоваться два URI, которые имеют одинаковый формат запроса. Второй URI возвращает данные в расширенном (extended) формате.

**URI:**

- /api/data/ (получение данных в сокращенном формате);
- /api/data/extended/ (получение данных в расширенном формате).

**Метод:** GET

**Формат запроса:** Query String

**Формат ответа:** JSON

**Описание GET-параметров запроса:**

Наименование	Обязательность	Значение по умолчанию	Тип/формат	Описание
dev_eui	Да	Нет	string	DevEUI устройства, для которого необходимо получить данные
fport	Нет	Нет	integer	Значение FPort для фильтрации возвращаемых данных. Будут возвращены данные только с указанным

				значением.
start_date	Нет	Нет	date YYYY-mm-dd или datetime YYYY-MM-DDThh:mm:ss	Дата начала выборки (UTC)
end_date	Нет	Нет	date YYYY-mm-dd или datetime YYYY-MM-DDThh:mm:ss	Дата окончания выборки (UTC)
utc	Нет	false	boolean	Конвертирует даты start_date и end_date в UTC в соответствии с временной зоной пользователя API.
limit	Нет	1000	integer	Лимит на количество возвращаемых записей
offset	Нет	0	integer	Смещение относительно начала выборки.
page	Нет	Нет	integer	Номер страницы с учетом limit (кол-во записей на страницу). При указании page параметр limit игнорируется. В возвращаемых данных дополнительно заполняется объект pagination, содержащий всю необходимую информацию для постраничного вывода данных.
dir	Нет	up	string	Направление данных uplink или downlink. Допустимые значения: up, down.
mtype	Нет	Нет	integer	Значение поля MType в целочисленном представлении для фильтрации возвращаемых данных. Будут возвращены

				данные только с указанным значением.
--	--	--	--	--------------------------------------

### Пример запроса с использованием утилиты curl:

```
curl -X GET "https://server.air-bit.eu/api/data/?dev_eui=3236323166377615&fport=2&start_date=2018-12-01&end_date=2018-12-31&limit=100&dir=up&mtype=64" -H "accept: application/json"
```

### Пример ответа для URI /api/data/:

```
[
  {
    "data": [
      {
        "created": "2018-12-30T03:05:07.841190",
        "fport": 2,
        "data": "AWMygAooXJwAAAAAAAAAAAAAAAAAAAA",
        "id": 628586
      },
      {
        "created": "2018-12-29T03:05:09.607922",
        "fport": 2,
        "data": "AWMyALkmXJwAAAAAAAAAAAAAAAAAAAA",
        "id": 628178
      }
    ],
    "eui": "3236323166377615"
  }
]
```

### Пример ответа для URI /api/data/extended/:

```
[
  {
    "data": [
      {
        "ack": false,
        "air_time": 82.176,
        "created": "2019-08-26T18:54:01.687296",
        "data": "AWIw2ClkXRgAAAAAAAAAAAAAAAAAAAA",

```

```

    "fcnt": 6,
    "fport": 2,
    "gw_time": null,
    "id": 1557393,
    "mtname": "UNCONF_UP",
    "mtype": 64,
    "srv_time": "2019-08-26T18:54:01.567848"
  },
  {
    "ack": false,
    "air_time": 1974.272,
    "created": "2019-08-26T18:49:03.646905",
    "data": "AWIwrChkXRgAAAAAAAAAAAAAAAAAAAA",
    "fcnt": 5,
    "fport": 2,
    "gw_time": null,
    "id": 1557391,
    "mtname": "UNCONF_UP",
    "mtype": 64,
    "srv_time": "2019-08-26T18:49:03.502326"
  }
],
"eui": "3236323166377615"
}
]

```

### Описание полей ответа:

Наименование	Обязательность	Тип/формат	Описание
eui	Да	string	DevEUI устройства для которого был выполнен запрос
data	Да	array	Массив с данными
created	Да	datetime	Дата и время (UTC) когда запись была сохранена на стороне сетевого сервера в формате ISO

fport	Да	integer	Значение FPort фрейма
data	Да	string	Полезная нагрузка в формате base64
id	Да	integer	ID записи на стороне сетевого сервера
<b>Описание полей для расширенного формата</b>			
mtype	Да	integer	Значение поля MType из фрейма LoRaWAN. Определяет тип пакета и его направление. За подробностями необходимо обратиться к спецификации протокола LoRaWAN.
mtname	Да	string	Символьная мнемоника для целочисленного значения поля MType, может принимать значения: «UNCONF_UP» - unconfirmed uplink; «CONF_UP»- confirmed uplink; «UNCONF_DOWN» - unconfirmed downlink; «CONF_DOWN» - confirmed downlink;
ack	Да	boolean	Состояние флага АСК в полученном кадре
air_time	Да	float	Общая продолжительность передачи пакета в эфире (миллисекунды). Может быть использовано для более точного определения времени устройства.
fcnt	Да	integer	Счетчик кадров FCnt
gw_time	Нет	datetime	Дата и время поступления сообщения на БС (по часам БС) в формате ISO. Может

			быть null в случае, когда БС не имеет источника точного времени.
srv_time	Да	datetime	Время получения пакета сервером (по часам сервера)
<b>pagination</b>	Нет	array	Информация для постраничного вывода данных (в случае указания параметра page)
has_next	Нет	boolean	Имеется ли следующая страница
has_prev	Нет	boolean	Имеется ли предыдущая страница
next_num	Нет	integer	Номер следующей страницы
page	Нет	integer	Номер текущей страницы
pages	Нет	integer	Количество страниц
per_page	Нет	integer	Количество записей на странице
prev_num	Нет	integer	Номер предыдущей страницы
total	Нет	integer	Общее количество записей по фильтру

### 2.1.2. ПОЛУЧЕНИЕ ПОЛЕЗНОЙ НАГРУЗКИ ДЛЯ НЕСКОЛЬКИХ УСТРОЙСТВ

Данные возвращаются отсортированными по времени в порядке убывания (сначала новые).

**Внимание!** При запросе на постраничный возврат данных параметр *page* относится ко всем данным попавшим под фильтр, включая *dev\_eui*, и с учетом сортировки. Например: есть 10 записей за 1 июля 2019 года для устройства №1 и 10 записей за 2 июля 2019 года для устройства №2. При запросе с указанием в фильтре *dev\_eui* обоих устройств, а так же *page=1* и *limit=10* в ответе будут представлены только данные устройства №2.

#### URI:

- /api/data/mult/ (получение данных в сокращенном формате);

- /api/data/mult/extended/ (получение данных в расширенном формате).

**Метод:** POST

**Формат запроса:** application/x-www-form-urlencoded

**Формат ответа:** JSON

**Описание POST-параметров запроса:**

Наименование	Обязательность	Значение по умолчанию	Тип/формат	Описание
dev_eui	Да	Нет	array[string]	Список DevEUI устройств (через запятую), для которых необходимо получить данные
Все остальные параметры полностью соответствуют описанию GET-параметров из п. «2.1.1. Получение полезной нагрузки для одного устройства»				

**Пример запроса с использованием утилиты curl:**

```
curl -X POST "https://server.air-bit.eu/api/data/mult/" -H "accept: application/json" -H "Content-Type: application/x-www-form-urlencoded" -d "dev_eui=3236323166377615,3132343777377F11&fport=2&start_date=2018-12-01&end_date=2018-12-31&limit=1000&dir=up&mtype=64"
```

**Пример ответа:**

```
[
  {
    "data": [
      {
        "created": "2018-12-30T20:41:37.905357",
        "fport": 2,
        "data": "AWMwkPkoXBYAAAAAAAAAAAAAAAAAAAA",
        "id": 628886
      },
      {
        "created": "2018-12-30T19:41:37.999852",
        "fport": 2,
        "data": "AWMwgOsoXBYAAAAAAAAAAAAAAAAAAAA",
        "id": 628869
      }
    ]
  },
  ]
```

```

    "eui": "3132343777377F11"
  },
  {
    "data": [
      {
        "created": "2018-12-30T03:05:07.841190",
        "fport": 2,
        "data": "AWMygAooXJwAAAAAAAAAAAAAAAAAAAA",
        "id": 628586
      },
      {
        "created": "2018-12-29T03:05:09.607922",
        "fport": 2,
        "data": "AWMyALkmXJwAAAAAAAAAAAAAAAAAAAA",
        "id": 628178
      }
    ],
    "eui": "3236323166377615"
  }
]

```

### Описание полей ответа:

Поля ответа полностью соответствуют описанию ответа из п. «2.1.1. Получение полезной нагрузки для одного устройства». Так же как и при получении полезной нагрузки для одного устройства возможен запрос для получения данных в расширенном формате.

#### 2.1.3. ОТПРАВКА ПОЛЕЗНОЙ НАГРУЗКИ НА КОНЕЧНОЕ УСТРОЙСТВО

**URI:** /api/data/

**Метод:** POST

**Формат запроса:** application/x-www-form-urlencoded

**Формат ответа:** JSON

**Описание POST-параметров запроса:**

Наименование	Обязательность	Значение по умолчанию	Тип/формат	Описание

dev_eui	Да	Нет	string	DevEUI устройства, которое отправило данные
up_id	Нет	Нет	integer	Идентификатор входящего uplink пакета, для которого отправляется ответ. При указании up_id и imme=true, будет произведена попытка отправить пакет немедленно в окно RX1 или RX2. При выходе (по времени) за границы RX2 пакет будет поставлен в очередь.
data	Нет	Нет	string	Полезная нагрузка в формате base64, которая будет направлена на устройство
mac	Нет	Нет	string	MAC-команды в формате base64. Необходимо учитывать что при одновременном указании data и mac, объем mac ограничен 15 байтами. Для отправки более длинных MAC-команд, необходимо отправлять их отдельно от data.
fport	Нет	1	integer	Значение FPort, которое будет использовано при отправке фрейма
conf	Нет	false	boolean	Если true, то будет отправлен фрейм требующий подтверждения (confirmed), в противном случае

				будет отправлен фрейм без подтверждения (unconfirmed)
imme	Нет	false	boolean	Если true, то фрейм будет отправлен немедленно для устройств класса «С», а так же для устройств класса «А» при указании <i>up_id</i> , в противном случае отправка будет произведена из очереди в момент получения очередного uplink
fpend	Нет	Нет	boolean	Устанавливает бит FPending сигнализирующий о наличии еще нескольких пакетов для устройства в очереди. Имеет смысл устанавливать для первого пакета, когда первый пакет отправляется немедленно в ответ на uplink ( <i>imme=true</i> и указан <i>up_id</i> ), а несколько последующих пакетов ставятся в очередь ( <i>imme=false</i> ).
prio	Нет	0	integer	Приоритет в очереди (при <i>imme=false</i> ): 0 — наименьший; 2 — наибольший.

### Пример запроса с использованием утилиты curl:

```
curl -X POST "https://server.air-bit.eu/api/data/" -H "accept: application/json" -H "Content-Type: application/x-www-form-urlencoded" -d "dev_eui=343438355A37500E&data=Ag%3D%3D&fport=5&conf=false&imme=false&prio=0"
```

### Пример ответа:

```
{  
  "result": true,  
  "error": null  
}
```

### Описание полей ответа:

Наименование	Обязательность	Тип/формат	Описание
result	Да	boolean	true, если операция успешно выполнена, false в противном случае
error	Нет	string	Текст ошибки в случае когда result=false

#### 2.1.4. ОТПРАВКА ПОЛЕЗНОЙ НАГРУЗКИ В ГРУППУ МНОГОАДРЕСНОЙ РАССЫЛКИ (MULTICAST)

**URI:** /api/data/mcast/

**Метод:** POST

**Формат запроса:** application/x-www-form-urlencoded

**Формат ответа:** JSON

#### Описание POST-параметров запроса:

Наименование	Обязательность	Значение по умолчанию	Тип/формат	Описание
mcaddr	Да	Нет	string	Адрес группы многоадресной рассылки
data	Да	Нет	string	Полезная нагрузка в формате base64, которая будет направлена в группу
fport	Нет	1	integer	Значение FPort, которое будет использовано при отправке фрейма

### Пример запроса с использованием утилиты curl:

```
curl -X POST "https://server.air-bit.eu/api/data/mcast/" -H "accept: application/json" -H "Content-Type: application/x-www-form-urlencoded" -d "mcaddr=005F4CBB&data=Ag%3D%3D&fport=2"
```

### Пример ответа:

```
{  
  "result": true,  
  "error": null  
}
```

### Описание полей ответа:

Наименование	Обязательность	Тип/формат	Описание
result	Да	boolean	true, если операция успешно выполнена, false в противном случае
error	Нет	string	Текст ошибки в случае когда result=false

#### 2.1.5. ПОЛУЧЕНИЕ ПОСЛЕДНЕГО (ПЕРВОГО) ПАКЕТА ДЛЯ СПИСКА УСТРОЙСТВ

Будут возвращены данные по последнему/первому пакету для каждого из устройств с учетом параметров фильтрации.

#### URI:

- /api/data/mult/last/ (получение данных в сокращенном формате);
- /api/data/mult/last/extended/ (получение данных в расширенном формате).

**Метод:** POST

**Формат запроса:** application/x-www-form-urlencoded

**Формат ответа:** JSON

#### Описание POST-параметров запроса:

Наименование	Обязательность	Значение по умолчанию	Тип/формат	Описание
dev_eui	Да	Нет	array[string]	Список DevEUI устройств (через запятую), для которых необходимо

				получить данные
fport	Нет	Нет	integer	Значение FPort для фильтрации возвращаемых данных. Будут возвращены данные только с указанным значением.
start_date	Нет	Нет	date YYYY-mm-dd или datetime YYYY-MM-DDThh:mm:ss	Дата начала выборки (UTC)
end_date	Нет	Нет	date YYYY-mm-dd или datetime YYYY-MM-DDThh:mm:ss	Дата окончания выборки (UTC)
pos	Нет	«last»	string	last — возвращать последний пакет для каждого устройства first — возвращать первый пакет
utc	Нет	false	boolean	Конвертирует даты start_date и end_date в UTC в соответствии с временной зоной пользователя API.
dir	Нет	up	string	Направление данных uplink или downlink. Допустимые значения: up, down.
mtype	Нет	Нет	integer	Значение поля MType в целочисленном представлении для фильтрации возвращаемых данных. Будут возвращены данные только с указанным значением.

### Пример запроса с использованием утилиты curl:

```
curl -X POST "https://server.air-bit.eu/api/data/mult/last/" -H "accept: application/json" -H "Content-Type: application/x-www-form-urlencoded" -d
```

```
"dev_eui=3132343777377F11,3236323166377615&fport=2&start_date=2019-08-01&end_date=2019-08-20&pos=first"
```

### Пример ответа:

```
[
  {
    "data": {
      "created": "2019-08-01T00:12:04.927040",
      "data": "AWMweNlBXRoAAAAAAAAAAAAAAAAAAAA",
      "fport": 2,
      "id": 1531287
    },
    "eui": "3132343777377F11"
  },
  {
    "data": {
      "created": "2019-08-01T00:57:17.978614",
      "data": "AWMybKlBXZwAAAAAAAAAAAAAAAAAAAA",
      "fport": 2,
      "id": 1531293
    },
    "eui": "3236323166377615"
  }
]
```

### Описание полей ответа:

Поля ответа полностью соответствуют описанию ответа из п. «2.1.1. Получение полезной нагрузки для одного устройства». Так же как и при получении полезной нагрузки возможен запрос для получения данных в расширенном формате.

## 2.2. Вебноок

Механизм позволяет реализовать событийную обработку данных поступающих от конечных устройств. При поступлении пакета с полезной нагрузкой от конечного устройства, сетевой сервер сформирует HTTP-запрос, содержащий полезную нагрузку и отправит на URL пользователя методом POST.

URL может быть определен пользователем, как для конкретного устройства, так и для группы устройств. Одновременно может быть указано несколько URL. В качестве авторизации на стороне пользователя может быть использован HTTP Basic Auth.

Сервер формирует POST-запрос с типом данных application/json следующего вида:

```
{
  "ack": false,
  "air_time": 1318.912,
  "data": "/yuYXl0=",
  "dev_cls": 1,
  "dev_eui": "3132343777377F11",
  "fcnt": 1,
  "mac": "Awc=",
  "mtype": 64,
  "mtname": "UNCONF_UP",
  "fport": 4,
  "gw_time": "2019-08-22T13:27:06.427104",
  "in_time": "2019-08-22T13:27:07.746016",
  "up_id": 1554934
}
```

**Описание полей запроса:**

Наименование	Обязательность	Тип/формат	Описание
in_time	Да	datetime	Дата и время поступления сообщения на сервер в формате ISO (по часам сервера)
gw_time	Нет	datetime	Дата и время поступления сообщения на БС (по часам БС) в формате ISO. Может быть null в случае, когда БС не имеет источника точного времени.
air_time	Да	float	Общая продолжительность

			передачи пакета в эфире (миллисекунды). Может быть использовано для более точного определения времени устройства, например в приложении синхронизации времени.
fport	Нет	integer	Значение FPort, может быть использовано для определения приложения, для которого предназначены данные, может быть null
fcnt	Да	integer	Значение поля FCnt (счетчик фреймов) в полученном фрейме
mac	Нет	string	MAC команды в формате base64, может быть null при отсутствии
mtype	Да	integer	Значение поля MType из фрейма LoRaWAN. Определяет тип пакета и его направление. За подробностями необходимо обратиться к спецификации протокола LoRaWAN.
mtname	Да	string	Текстовое описание поле MType, для данного сообщения может принимать значения:  «UNCONF_UP» - unconfirmed uplink;  «CONF_UP» - confirmed uplink.
dev_eui	Да	string	DevEUI устройства, которое отравило данные

dev_cls	Да	integer	Класс устройства: 1 — класс «А» 2 — класс «В» 3 — класс «С»
data	Нет	string	Полезная нагрузка в формате base64, может быть null при отсутствии
up_id	Да	integer	Идентификатор текущего uplink пакета в БД сетевого сервера. Требуется при формировании ответа, если требуется отправить немедленный ответ на входящий uplink. При ответе на входящий uplink, сначала проверяется возможность немедленной отправки в приемные окна RX1/RX2 (при условии, что imme=true), а при отсутствии такой возможности, ответ ставится в очередь.
ack	Да	boolean	Флаг подтверждения получения последнего отправленного confirmed downlink пакета

В веб-интерфейсе управления пользователь может настроить гарантированную отложенную доставку для случаев, когда пользовательский URL недоступен из-за аварии или остановлен на обслуживание. В этом случае сервер будет формировать очередь и попытается доставить все сообщения, вместе с очередным пакетом данных, как только URL вновь станет доступен.

#### Пример ответа, который сервер ожидает от URL пользователя:

```
{"result": true, "message": "Operation executed"}
```

#### Описание полей ответа:

Наименование	Обязательность	Тип/формат	Описание
result	Да	boolean	Результат операции, true, если данные приняты успешно,

			false в противном случае.
message	Нет	string	Произвольное диагностическое сообщение с описанием результата

## 2.3. MQTT (Mosquitto)

### 2.3.1. ПОЛУЧЕНИЕ И ОБРАБОТКА ДАННЫХ В ИНТЕРАКТИВНОМ РЕЖИМЕ

Механизм интеграции по протоколу MQTT позволяет реализовать событийную обработку данных поступающих от конечных устройств, а так же отправлять данные (unicast или multicast) на конечные устройства (downlink). При поступлении пакета с полезной нагрузкой от конечного устройства, сетевой сервер сформирует сообщение, содержащее полезную нагрузку и отправит на MQTT-брокер в указанный топик (подключение и отправка происходят сразу после получения пакета сервером, после публикации сообщения в топик сервер сразу отключается от брокера). В веб-интерфейсе управления сервером, пользователь может указать на какой MQTT-брокер и в какой топик сервер должен отправить данные полученные от устройств.

Брокер и топик могут быть определены пользователем, как для конкретного устройства, так и для группы устройств. Допускается множественное указание брокеров.

В зависимости от настроек брокера может использоваться:

MQTT (TCP-порт 1883, TCP-порт 8883 с TLS/SSL)

MQTT over Websocket (TCP-порт 9001, TCP-порт 9883 с TLS/SSL)

Сервер формирует сообщение с типом данных application/json следующего вида:

```
{
  "ack": false,
  "air_time": 1318.912,
  "data": "/yuYXl0=",
  "dev_cls": 1,
  "dev_eui": "3132343777377F11",
  "fcnt": 1,
  "mac": "Awc=",
  "mtype": 64,
  "mtname": "UNCONF_UP",
```

```
"fport": 4,  
"gw_time": "2019-08-22T13:27:06.427104",  
"in_time": "2019-08-22T13:27:07.746016",  
"up_id": 1554934  
}
```

Все поля соответствуют аналогичным полям описанным в п. «2.2. Webhook».

Так же как и для Webhook, опционально, поддерживается отложенная доставка, в случае когда MQTT-брокер физически недоступен по каким-либо причинам. Все сообщения будут сохранены в очередь и доставлены при первой возможности.

### 2.3.2. ОТПРАВКА ПОЛЕЗНОЙ НАГРУЗКИ НА КОНЕЧНОЕ УСТРОЙСТВО

Network Server подключается к MQTT-брокеру **mqtt.air-bit.eu** в качестве подписчика (subscriber) и ожидает сообщения в топик **send/data/+**, где «+» соответствует *dev\_eui*, например: **send/data/343438355A37500E**. При получении сообщения соответствующего формата (JSON), сервер в зависимости от содержимого:

- отправляет unicast фрейм в сеть;
- ставит пакета в очередь для доставки unicast фрейма в ответ на очередной uplink;
- отправляет фрейм в группу многоадресной рассылки (multicast).

Сервер проверяет совпадение *dev\_eui* из названия топика и содержащегося в сообщении и не производит никаких действий, если они не совпадают.

#### Примеры валидных unicast сообщений:

```
{"dev_eui": "343438355A37500E", "data": "Ag==", "fport": 5, "conf": true,  
"imme": true}
```

```
{"dev_eui": "343438355A37500E", "data": "Ag==", "fport": 5, "conf": true,  
"prio": 2}
```

```
{"dev_eui": "343438355A37500E", "data": "Ag==", "fport": 5, "conf": true}
```

```
{"dev_eui": "343438355A37500E", "data": "Ag==", "fport": 5,  
"up_id": 1412451, "fpend": true}
```

```
{"dev_eui": "343438355A37500E", "mac": "Ag="}
```

Описание всех возможных полей запроса на отправку unicast-сообщений полностью соответствует описанию из п. «2.1.3. Отправка полезной нагрузки на конечное устройство».

### Примеры валидных multicast сообщений:

```
{"mcaddr": "005F4CBB", "data": "Ag==", "fport": 2}
```

Описание всех возможных полей запроса на отправку multicast-сообщений полностью соответствует описанию из п. «2.1.4. Отправка полезной нагрузки в группу многоадресной рассылки (Multicast)».

Сервер не формирует ответа на отправку сообщения. Гарантия доставки определяется только на уровне QoS протокола MQTT. По умолчанию при подключении к брокеру сетевой сервер использует QoS=2.

## 2.4. AMQP (RabbitMQ)

### 2.4.1. ПОЛУЧЕНИЕ И ОБРАБОТКА ДАННЫХ В ИНТЕРАКТИВНОМ РЕЖИМЕ

Механизм интеграции по протоколу AMQP позволяет реализовать событийную обработку данных поступающих от конечных устройств, а так же отправлять данные (unicast или multicast) на конечные устройства (downlink). При поступлении пакета с полезной нагрузкой от конечного устройства, сетевой сервер сформирует сообщение, содержащее полезную нагрузку и отправит на AMQP-брокер в указанную очередь (подключение и отправка происходят сразу после получения пакета сервером, после публикации сообщения в очередь сервер сразу отключается от брокера). В веб-интерфейсе управления сервером, пользователь может указать на какой AMQP-брокер и в какую очередь сервер должен отправить данные полученные от устройств.

Брокер и очередь могут быть определены пользователем, как для конкретного устройства, так и для группы устройств. Допускается множественное указание брокеров.

Сервер формирует сообщение с типом данных application/json следующего вида:

```
{
  "ack": false,
  "air_time": 1318.912,
  "data": "/yuYX10=",
  "dev_cls": 1,
  "dev_eui": "3132343777377F11",
  "fcnt": 1,
```

```
"mac": "Awc=",
"mtype": 64,
"mtname": "UNCONF_UP",
"fport": 4,
"gw_time": "2019-08-22T13:27:06.427104",
"in_time": "2019-08-22T13:27:07.746016",
"up_id": 1554934
}
```

Все поля соответствуют аналогичным полям описанным в п. «2.2. Webhook».

Так же как и для Webhook, опционально, поддерживается отложенная доставка, в случае когда AMQP-брокер физически недоступен по каким-либо причинам. Все сообщения будут сохранены в очередь и доставлены при первой возможности.

#### 2.4.2. ОТПРАВКА ПОЛЕЗНОЙ НАГРУЗКИ НА КОНЕЧНОЕ УСТРОЙСТВО

Network Server подключается к AMQP-брокеру **amqp.air-bit.eu** в качестве потребителя (consumer) и обрабатывает очередь **airbit.lns.downlink** (тип: topic, ключ маршрутизации: **airbit.lns.downlink.\***). При отправке сообщения в очередь следует использовать ключ маршрутизации следующего вида: **airbit.lns.downlink.<dev\_eui>**, например: **airbit.lns.downlink.343438355A37500E**

При поступлении в очередь сообщения соответствующего формата (JSON), сервер в зависимости от содержимого:

- отправляет unicast фрейм в сеть;
- ставит пакета в очередь для доставки unicast фрейма в ответ на очередной uplink;
- отправляет фрейм в группу многоадресной рассылки (multicast).

#### Примеры валидных unicast сообщений:

```
{"dev_eui": "343438355A37500E", "data": "Ag==", "fport": 5, "conf": true, "imme": true}
```

```
{"dev_eui": "343438355A37500E", "data": "Ag==", "fport": 5, "conf": true, "prio": 2}
```

```
{"dev_eui": "343438355A37500E", "data": "Ag==", "fport": 5, "conf": true}
```

```
{"dev_eui": "343438355A37500E", "data": "Ag==", "fport": 5,  
"up_id": 1412451, "fpend": true}
```

```
{"dev_eui": "343438355A37500E", "mac": "Ag="}
```

Описание всех возможных полей запроса на отправку unicast-сообщений полностью соответствует описанию из п. «2.1.3. Отправка полезной нагрузки на конечное устройство».

### **Примеры валидных multicast сообщений:**

```
{"mcaddr": "005F4CBB", "data": "Ag==", "fport": 2}
```

Описание всех возможных полей запроса на отправку multicast-сообщений полностью соответствует описанию из п. «2.1.4. Отправка полезной нагрузки в группу многоадресной рассылки (Multicast)».

Гарантия обработки сообщения определяется протоколом AMQP. Сообщение не будет удалено из очереди до тех пор, пока его получение не будет подтверждено потребителем в соответствии с протоколом AMQP.