



# IOT Vega Server

## Руководство

### **Введение**

В данном документе описаны возможности сервера IOT Vega Server, способы взаимодействия с ним, а также настройки при первом запуске.

## Оглавление

Описание IOT Vega Server .....	3
Возможности.....	4
Установка .....	5
Настройки.....	6
Настройка сервера с помощью ПО IOT Vega AdminTool.....	9

# Описание IOT Vega Server

---

Сетевой сервер IOT Vega Server это инструмент для организации сетей стандарта LoRaWAN любого масштаба.

Предназначен для управления опорной сетью базовых станций, работающих под управлением ПО Packet forwarder от компании Semtech, приема данных с оконечных устройств и передачи их внешним приложениям, а также передачи данных от внешних приложений на LoRaWAN устройства.

Сервер работает по спецификации LoRaWAN 1.02 и поддерживает любые оконечные устройства, работающие согласно данной версии спецификации.

Все принятые от оконечных устройств данные сохраняются во встроенной в IOT Vega Server базе данных и всегда доступны для внешних приложений.

Открытый API, основанный на технологии Web Socket позволяет подключать к IOT Vega Server внешние приложения и использовать возможности LoRaWAN сетей в ваших проектах.

IOT Vega Server выпускается в виде консольного приложения для операционных систем Windows и Linux и является бесплатным для сетей емкостью до 1000 оконечных устройств.

Для управления сервером используется приложение IOT Vega **AdminTool** с простым дружественным интерфейсом. AdminTool открывает перед администратором сервера широкие возможности по управлению сетью LoRaWAN. С AdminTool вы можете добавлять в сеть новые оконечные устройства LoRaWAN, просматривать карту сети, контролировать базовые станции, а также управлять правами пользователей. IOT Vega AdminTool предоставляется бесплатно в виде Web-приложения, а также для операционных систем Windows, Android, IOS.

Для работы с устройствами используется клиентское приложение IOT Vega **Pulse** с широким спектром возможностей извлечения данных, обработки и представления в различных форматах (таблица, график, отчет, диаграмма).

## Возможности

---

- Поддержка любых оконечных устройств LoRaWAN 1.0.1
- Поддержка оконечных устройств класса A и C
- Встроенная база данных
- Удобное приложение администратора
- Построение карты сети
- Управление пользователями сети
- Гибкая настройка подключенных к серверу устройств
- Поддержка произвольных частотных планов
- Онлайн просмотр пакетов с каждого устройства (нисходящие и восходящие пакеты)
- Графики связи для каждого устройства в сети

# Установка

---

Перед обновлением (установкой новой версии) сервера, рекомендуется выполнить резервное сохранение базы данных (БД) и файла настроек по следующим причинам:

- новая версия сервера модифицирует структуру БД, поэтому, в случае необходимости возврата к предыдущей версии, сервер не сможет работать с измененной БД;
- файл с настройками заменяется файлом из установочного пакета, тем самым возвращает старые настройки к значениям по умолчанию. Обновление файла настроек необходимо, поскольку в новой версии сервера добавлены новые настройки.

**Версия для Windows** не требует установки. Необходимо распаковать архив и запустить исполняемый файл.

Для корректной работы сервера необходимо обязательно установить обе версии библиотек, которые находятся в директории **IOT Vega Server/msvc c++ 2013**. После этого можно начинать работать с сервером.

**Версия для Linux** поставляется в виде установочного DEB пакета для 32-х и 64-х разрядных систем.

Процесс установки приложения на Linux:

- скачать файл на ПК;
- установить, выполнив команду: `sudo dpkg -i /путь/к/файлу/iot-vega-server-1.1.1.deb`;
- для конфигурирования необходимо изменить содержимое файла с настройками `/opt/iot-vega-server/settings.conf`;
- для запуска сервера в работу необходимо выполнить скрипт `sudo ./iot-vega-server.sh` в директории `/opt/iot-vega-server` либо написать новый скрипт с указанием пути к библиотекам (директория установки).

# Настройки

---

Сервер работает в автономном режиме, требуется лишь первоначальное конфигурирование перед запуском сервера. Чтобы осуществить первоначальную настройку сервера необходимо зайти в папку с файлами сервера и открыть **settings.conf** с помощью любого текстового редактора (например, «Блокнот»). Содержимое файла выглядит следующим образом:

```
# Host connection settings
[host]
# IP-address for UDP connection (gateway connection)
ip=127.0.0.1
# Port for UDP connection (gateway connection)
udpPort=8001
# Port for TCP (WebSocket) connection
tcpPort=8002
# "path" part of websocket address
websocketPath=/
# Flag of using SSL encryption for WebSocket
useSSL=0
# SSL certificate filename (certificate must be in server's directory)
certFileName=cert.crt
# SSL key filename (key must be in server's directory)
keyFileName=key.key

# LoRaWAN network settings
[lorawan]
# LoRaWAN network identifier (should be random between 1 and 127)
networkID=1
# Flag for using Plug-and-Play gateways function.
# If this value is 1, server would automatically append all gateways which connected to one
usePnPGateway=1

# Super user options
[root]
# Login for super user
root=root
# Password for super user (recommendation: change this password to your own)
password=123

# Console settings (volume of debug information)
[console]
# Maximum level of console messages that will be shown (levels of messages represented below)
maxMsgLevel=20
# Maximum level of console messages that will be saved into LOG file (levels of messages
represented below)
maxLogMsgLevel=0
# Console message levels:
# errors      = 0
# uplink      = 1
# downlink    = 2
# warning     = 3
# info        = 4
# debug       = 20
```

Секция [host] содержит настройки портов и сетевых соединений:

**ip** – IP адрес через который будут работать базовые станции;



**Данный параметр необходимо изменить на корректный IP адрес, через который будут работать базовые станции. В противном случае связи с базовыми станциями не будет**

**udpPort** – порт для базовых станций (БС);

**tcpPort** – порт для внешних приложений (в том числе для AdminTool);

**webSocketPath** – содержимое поля «путь» в адресе WebSocket. Если задать в данном поле /ws, то при подключении из AdminTool нужно будет указывать ws://address:port/ws;

**useSSL** – флаг разрешения использования SSL шифрования для WebSocket;

**certFileName** – имя файла, в котором содержится SSL сертификат (с расширением);

**keyFileName** – имя файла, в котором содержится SSL ключ (с расширением).

Для налаживания корректной работы SSL шифрования необходимо выполнить следующие действия:

- получить SSL сертификат (и ключ), подписанный доверенным центром сертификации (это может быть и бесплатный доверенный центр, например sslforfree или letsencrypt);
- установить OpenSSL версии 1.0.2 (сервер с версией OpenSSL 1.1.0 не работает);
- скопировать файлы сертификата и ключа (для linux достаточно создать ссылки) в директорию с сервером;
- разрешить использовать SSL шифрование, прописать соответствующие имена файлов (с сертификатом и ключом) в конфигурационном файле;
- запустить сервер, проследить за возможными ошибками.

Секция [lora] содержит данные для идентификации в LoRa-сети:

**networkID** – определяет идентификатор LoRaWAN сети. Перед запуском сервера необходимо установить случайное значение в диапазоне от 1 до 127 включительно;



**Если в непосредственной близости будут работать несколько сетей LoRaWAN, которые используют одинаковые NetworkID – каждая из сетей продолжит работу в нормальном режиме, но периодически будут появляться ошибки о некорректности NwkSKey того или иного устройства:**

**"INVALID\_DEVICE\_NETWORK\_SESSION\_KEY"**

**usePnPGateway** – флаг разрешения автоматического добавления БС на сервер. Если эта опция разрешена, сервер любую неизвестную ранее БС будет добавлять в список зарегистрированных БС с параметрами по умолчанию.

Секция [root] определяет идентификационные данные суперпользователя, где **root** – логин, а **password** – пароль.

Секция [console] задает настройки объема выводимой информации в консоль и сохранения ее в LOG файл (параметры данной секции обновляются каждую

минуту, поэтому для изменения уровня выводимых сообщений перезапуск сервера не нужен):

**maxMsgLevel** – максимальный уровень сообщений (включительно), которые будут отображаться в консоли сервера. Расшифровка уровней сообщений приведена ниже;

**maxLogMsgLevel** – максимальный уровень сообщений, которые будут сохраняться в LOG файл.

Уровни сообщений консоли:

- 0 = сообщения с критическими ошибками;
- 1 = исходящие (uplink) сообщения;
- 2 = восходящие (downlink) сообщения;
- 3 = сообщения с предупреждениями;
- 4 = информационные сообщения (зачастую несут информацию отладочного характера);
- 20 = сообщения с отладочной информацией.

Команды для сервера приведены в файле **API IOT Vega Server Rev21.pdf**.



# IOT Vega AdminTool

ПО IOT Vega AdminTool является удобным Web-приложением для администрирования сервера и позволяет добавлять в сеть новые оконечные устройства LoRaWAN, просматривать карту сети, контролировать базовые станции, а также управлять правами пользователей.

Разберем пример подключения новой или редактирования параметров существующей БС на сервере (Рис. 1).

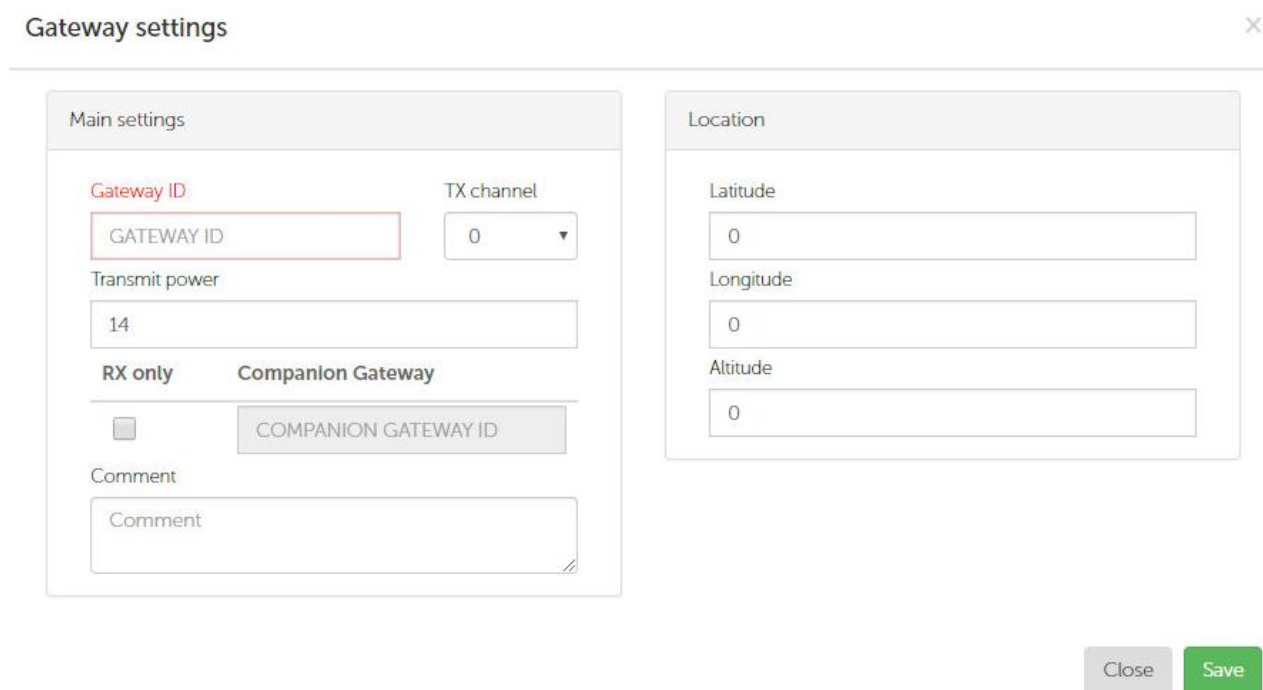


Рис. 1. Окно подключения БС на сервер.

Обязательные настройки:

**Gateway ID** – идентификатор БС (16 шестнадцатеричных символа – 8 Байт);  
**TX channel** – канал БС, используемый для передачи сообщений на оконечные устройства (downlink). Данный параметр указывается в настройках ПО "packet\_forwarder" на БС (обычно в файле "global\_conf.json"). По умолчанию используется 0 канал;

**Transmit power** – мощность вещания БС. Максимальная мощность вещания обычно обусловлена схмотехникой БС и ограничена в ПО "packet\_forwarder"; при превышении данного параметра БС будет возвращать ошибку с соответствующим кодом.

Здесь необходимо оговорить критерии, по которым выбирается необходимое значение мощности вещания.

Обратимся к файлу «global\_conf.json» из настроек ПО «packet\_forwarder» на БС.

В списке допустимых наборов параметров вещания «tx\_lut...» содержится набор мощностей, на которых БС может осуществлять передачу. Для стандартного списка это диапазон от -6 до 27 дБм с разным шагом (всего 16 значений).

Так же имеется параметр «antenna\_gain», который определяет усиление антенны (в дБм).

В итоге, разница величин «Transmit power» и «antenna\_gain» должна соответствовать одному из значений из списка «tx\_lut...». Например, «antenna\_gain» = 3 дБм и планируется передавать данные на мощности 10 дБм («Transmit power»). Разница между «Transmit power» и «antenna\_gain» составляет 7дБм, но такого значения нет в списке «tx\_lut...» (и если использовать такой набор параметров, то при попытке отправки данных, БС будет присылать сообщение с ошибкой о некорректной мощности). Ближайшие разрешенные значения из «tx\_lut...» это 6 и 10 дБм, соответственно, для такой конфигурации разрешенные значения «Transmit power» будут 9 или 13 дБм.

**RX only** – флаг, указывающий, что БС работает только на прием. При установке данного флага, передача данных на оконечные устройства через соответствующую БС запрещена. Данная опция введена для организации «полно-дуплексной БС»: это вариант когда в непосредственной близости установлено две БС и одна из них работает только на прием, а вторая работает в обычном режиме. При этом достигается непрерывность прослушивания радиоэфира;

**Companion Gateway** – идентификатор БС компаньона, работающей в нормальном режиме при организации «полно-дуплексной БС»;

**Comment** – поле для комментария (например, наименования БС);

Область **Location** – содержит поля для ввода координат размещения БС: широта, долгота, высота. Если в БС имеется встроенный GPS приемник, введенные координаты будут обновлены актуальными значениями автоматически.

Разберем пример регистрации нового устройства или изменения параметров существующего (Рис. 2 и Рис. 3).

На Рис. 2 представлен снимок формы для регистрации устройства с указанием основных параметров.

Device settings

Activation by personalisation (ABP)

End-device address (devAddr)

DEVADDR

Application session key (AppSKey)

APPSKEY

Network session key (NwkSKey)

NWKSKEY

Over-the-air activation (OTAA)

Application identifier (AppEUI)

APPEUI

Application key (AppKey)

APPKEY

Main settings

End-device name

device name

End-device identifier (DevEUI)

DEVEUI

End-device class

Location

Latitude

0

Longitude

0

Altitude

0

☐ Expert settings
 

Close Save

Рис. 2. Основные параметры для регистрации устройства на сервере.

Область **Activation by personalization (ABP)** содержит параметры необходимые для регистрации устройства на сервере через ABP:

**devAddr** – адрес устройства в LoRaWAN сети. Это 32-х разрядное число в шестнадцатеричном виде, например 012345AB;

**AppSKey** – сессионный ключ приложения – это строка из 32-х шестнадцатеричных символов;

**NwkSKey** – сетевой сессионный ключ – это строка из 32-х шестнадцатеричных символов.

Область **Over-the-air activation (OTAA)** содержит параметры необходимые для регистрации устройства на сервере через OTAA:

**AppEUI** – EUI идентификатор приложения устройства – строка из 16-ти шестнадцатеричных символов;

**AppKey** – ключ приложения устройства – это строка из 32-х шестнадцатеричных символов.

Для регистрации устройства достаточно указать хотя бы один из типов активации или оба сразу.

Область **Main settings** содержит следующие параметры:

**End-device name** – пользовательское наименование устройства;

**DevEui** – EUI идентификатор устройства (уникальный номер устройства) - строка из 16-ти шестнадцатеричных символов;

**End-device class** – класс устройства. Этот параметр может принимать два значения: CLASS\_A либо CLASS\_C. Поддержка устройств CLASS\_B находится в разработке.

Область **Location** содержит поля для указания координат размещения устройства: широта, долгота, высота.

Class C device settings

Class C device reaction time, (ms)

1000

☐ Use Downlink queue for Class C

Adaptive data rate

☒ Enable server ADR

Preferred data rate

DR5

Preferred transmit power

14 dBm

Device RX settings

RX window

2

RX1 delay

1 s

RX2 data rate

DR0

Join accept delay 1

5 s

Regional settings

Frequency plan

EU868

Nº	Frequency	Enabled
1	FIXED	<input checked="" type="checkbox"/>
2	FIXED	<input checked="" type="checkbox"/>
3	FIXED	<input checked="" type="checkbox"/>
4	<div>867100000</div>	<input checked="" type="checkbox"/>
5	<div>867300000</div>	<input checked="" type="checkbox"/>
6	<div>867500000</div>	<input checked="" type="checkbox"/>
7	<div>867700000</div>	<input checked="" type="checkbox"/>
8	<div>867900000</div>	<input checked="" type="checkbox"/>

RX2 Frequency, Hz

869525000

Рис. 3. Экспертные параметры для регистрации устройства на сервере.

Нажав на галочку **Expert settings**, можно получить доступ к экспертным настройкам устройства (см Рис. 3).

Область **Class C device settings** доступна только устройствам CLASS\_C. Доступны следующие параметры:

**Class C reaction time** – время реакции устройства CLASS\_C. Это время между окончанием приема сообщения сервера устройством и началом отправки возможного ответа от устройства (т.е. по сути время на подготовку возможного ответа). Параметр введен для улучшения качества работы с сообщениями от сервера к устройствам класса C;

**Use downlink queue for Class C** – флаг разрешающий помещать пакеты для отправки в очередь отправки. Зачастую устройства CLASS\_C работают в "online" режиме и если пакет не доставлен до устройства в настоящий момент, то его доставка уже не требуется – в таком случае очередь пакетов не нужна. Даже наоборот, очередь сыграет негативную роль, как накопитель уже устаревшей информации, которая

будет передана в обязательном порядке, занимая эфирное время. По умолчанию опция выключена.

Область **Adaptive data rate** содержит настройки алгоритма ADR, предназначенного для автоматического изменения скорости вещания устройств в зависимости от качества связи (в условиях хорошего приема скорость будет увеличиваться, тем самым снижая время передачи пакетов и увеличивая срок службы устройства от батареи):

**Enable server ADR** – флаг, разрешающий работу алгоритма ADR на сервере для конкретного устройства (если галочка снята, сервер не будет регулировать скорость передачи устройства, даже если алгоритм ADR на самом устройстве активирован);

**Preferred data rate** – значение DR (скорости), к которому будет стремиться алгоритм ADR сервера;

**Preferred transmit power** – мощность передатчика устройства, которую сервер задаст ему при очередном сеансе связи.



**Параметр Preferred transmit power стоит изменять (уменьшать относительно значения по умолчанию) в том случае если качество связи для устройства на максимальной скорости удовлетворительно**

Область **Device RX settings** содержит настройки приемных окон устройства: **RX window** – номер приемного окна (1 или 2), через которое сервер по умолчанию будет передавать данные. Если установить 1-ое окно, то в случае если сервер не может отправить данные в 1-ое приемное окно (например, нет свободной БС), то будет выполнена попытка отправить данные во 2-ое приемное окно;

**RX1 delay** – задержка открытия устройством первого приемного окна (по умолчанию 1 секунда). 2-ое приемное окно открывается всегда (если в первое приемное окно данные не были приняты) через 1 секунду после 1-го;

**RX2 data rate** – скорость передачи данных для 2-го приемного окна;

**Join accept delay 1** – задержка открытия устройством первого приемного окна для получения регистрационной информации при активации в сети способом OTAA (по умолчанию 5 секунд).

Область **Regional settings** содержит настройки частотного плана для соответствующего устройства. Здесь предлагается выбор из двух существующих наборов (официальный европейский частотный план и один из вариантов российского частотного плана), а также имеется возможность настроить уникальный набор частот.

Каждый из наборов частот состоит из:

- трех каналов с фиксированными частотами (данные каналы жестко заданы в устройстве без возможности их изменения через протокол LoRaWAN);
- пяти каналов для приема и передачи сообщений. Если канал не используется, необходимо задать нулевую частоту и отключить соответствующую галочку в настройках **Enabled**;
- одного канала, определяющего частоту приема данных для второго приемного окна.

Маска активных каналов (список флагов **Enabled**) расположена слева от полей ввода частоты и содержит флаги отключения/включения соответствующих каналов

(частот) для передачи данных на устройстве. Данный параметр передается устройству при каждой JOIN процедуре и в каждом сообщении алгоритма ADR (если ADR разрешен).

## Информация о документе

Заголовок	IOT Vega Server
Тип документа	Руководство
Номер документа	B02-server-01
Номер и дата последней ревизии	07 от 27.09.2017

Этот документ применим к следующим продуктам:

Тип продукта	Название продукта
Программное обеспечение	IOT Vega Server
	IOT Vega Admin Tool

## История ревизий

Ревизия	Дата	Имя	Комментарии
01	05.06.2017	КЕВ	Дата создания документа
02	06.06.2017	МАИ	Мелкие правки
03	16.06.2017	КЕВ	Мелкие правки
04	10.07.2017	МАИ	Добавлен раздел «Установка», исправлен раздел «Настройка»
05	14.08.2017	КЕВ	Мелкие правки
06	31.08.2017	МАИ	Добавлен раздел «IoT Vega AdminTool», скорректирован раздел «Настройка»
07	27.09.2017	МАИ	Мелкие правки в разделе «IoT Vega AdminTool»







[vega-absolute.ru](http://vega-absolute.ru)

Руководство пользователя © ООО «Вега-Абсолют» 2017