



# WebdynEasy LoRaWAN

---

User Manual

# Index

Glossary .....	4
Document historical .....	6
1. About this Document .....	7
1.1 Scope .....	7
1.2 Target Audience .....	7
1.3 Product Versions .....	7
1.3.1 Safety Instructions .....	8
1.4 Regulations .....	8
2. General Presentation .....	9
2.1 The LoRaWAN Protocol .....	9
2.2 The Hub .....	10
2.2.1 General description .....	10
2.2.2 Technical Specifications .....	13
3. Installation and Maintenance .....	17
3.1 Unpacking .....	17
3.1.1 Product Contents .....	17
3.1.2 Hub Identification .....	18
3.2 Assembly .....	19
3.2.1 Opening/Closing the Box .....	19
3.2.2 Wall Mounting .....	19
3.2.3 Cellular Network .....	20
3.2.4 LoRa .....	22
3.2.5 Connection .....	22
4. Configuration .....	27
4.1 .....	
Embedded web interface .....	27
4.1.1 Hub Connectivity .....	29
4.1.2 LoRaWAN .....	34
4.1.3 System .....	37
4.1.4 VPN .....	38
4.1.5 Alarms .....	41

4.1.6 Schedules.....	42
4.1.7 Modbus .....	47
4.1.8 Run Actions .....	52
5. ....	
Operation .....	55
5.1 The remote server .....	55
5.1.1 The FTP Server .....	55
5.1.2 Web Service .....	57
5.1.3 MQTT .....	59
5.2 The Configuration .....	60
5.3 The Data .....	61
5.4 Alarms .....	62
5.5 Commands .....	62
5.5.1 “Reboot” Command.....	64
5.5.2 “Factory” Command .....	65
5.5.3 “Update” Command .....	65
5.5.4 “Connect” Command .....	65
5.5.5 “Status” Command .....	65
5.5.6 “Log” Command .....	66
5.5.7 “Settime” Command .....	67
5.5.8 “Modbus” Command.....	67
5.5.9 “Lorawan” Command .....	69
6. Update .....	74
6.1 Local.....	74
6.2 Remote.....	74
7. Appendix: XML configuration file variables.....	76
Offices & Support Contact.....	85

## Glossary

NAME	DESCRIPTION
ABP	Activation By Personalization: ABP activation forces to have the DevAddr and the security key for the peripheral hard-coded in the product. This strategy may seem simpler because the join procedure does not need to be known, but it has security disadvantages.
ADR	Adaptive Data Rate: a data rate and radio transmission power optimisation mechanism. It is used to optimise battery consumption.
APN	Access Point Name: the name of the access point the gateway uses to connect to the Internet via a mobile connection.
AppEUI	The EUI Application is a unique application identifier issued by the IEEE organisation (EUI-64). It is only used in OTAA mode and is used to get the server keys during the JOIN.
AppKEY	The Application Key is product specific. It is only used in OTAA mode and is used to get the server keys during the JOIN.
AppSKey	The Application Session Key is product specific and is used for end to end application data encryption. It is required in ABP mode and is calculated automatically by the server during the JOIN in OTAA mode.
Data Rate	The Data Rate is defined by a digit from 0 to 5 and sets the modulation type, the Spreading Factor and the bandwidth used.
DevEUI	Device EUI: the unique identifier issued by the IEEE organisation (EUI-64).
Device Address	32 bit device identifier that is used to uniquely identify the product on the LoRaWAN server. It is required in ABP mode and is supplied automatically by the server during the JOIN in OTAA mode.
Ftp	File Transfer Protocol: communication protocol used to exchange files over a TCP/IP network.
HTTP	HyperText Transfer Protocol: client-server communication protocol developed for the Web.
IP	Internet Protocol: message protocol in charge of addressing and sending TCP packets over the network.
JSON	JavaScript Object Notation: JSON is an easily interpretable data interchange format.

LoRa	LoRa is radio modulation including the physical connection and the physical layer in the OSI model.
LoRaWAN	LoRaWAN is a transmission protocol that uses LoRa modulation.
MD5	Message Digest 5: cryptographic hash function used to obtain a file's digital imprint.
Modbus	Modbus is a communication protocol routinely used by industry to dialogue with industrial equipment over a network.
NTP	Network Time Protocol: protocol used to synchronise the local hub clock with a time reference via a computer network.
NwkSKey	The Network Session Key is product specific and is used for end to end LoRaWAN network data encryption. It is required in ABP mode and is calculated automatically by the server during the JOIN in OTAA mode.
OTAA	Over The Air Activation: OTAA activation is the preferred and most secure method for connecting to the LoRaWAN network. The product runs a join procedure with the network during which a dynamic DevAddr is assigned and security keys are brokered with the product.
PEM	File format standard for storing certificates and private keys in Base64-encoded text format.
DIN rail	Standard 35 mm metal rail used in Europe in industrial control equipment in racks.
RTU	RTU mode is an RS422/485 hard-wired bus for Modbus.
Spreading Factor (SF)	The spreading factor is the length of the sent frames. The more the signal is spread, the lower the speed. However it increases the product range.
TCP	Transmission Control Protocol: an Internet-based connection-oriented protocol that provides data packet segmenting services that the IP protocol sends over the network. This protocol provides a reliable data transfer service. See also IP.
TCP/IP	Transmission Control Protocol/Internet Protocol: a set of network protocols that provide interconnection services between computers of different hardware architectures and operating systems. TCP/IP includes standards for communication between computers and conventions for network interconnection and routing.

UDP	User Datagram Protocol: non connection-oriented protocol of the TCP/IP model transport layer. This protocol is very simple because it does not provide error checks (it is not connection-oriented...).
VPN	Virtual Private Network: secure and encrypted connection between the concentrator and a private network, thus allowing isolation from public telecommunications networks.
XML	Extensible Markup Language: generic tagging computer metalanguage. The purpose of XML is to facilitate the automated exchange of complex content between heterogeneous information systems.
XSD	XML Schema Definition: file used to validate XML tags and data in an XML file.

## Document historical

VERSION	DESCRIPTION
V0.11	Creation
V1.0	Added VPN
V3.12	Add MQTT data support Add LoRaWAN mode C

# 1. About this Document

This guide describes the hub assembly, installation and configuration as well as its remote operation.

## 1.1 Scope

This technical description is valid for WebdynEasy LoRaWAN hubs from hardware version V1 and software version V1.0 onwards.

## 1.2 Target Audience

This guide is intended for WebdynEasy LoRaWAN hub installers and users, but also for people using our Sens'RF LoRaWAN sensors.

## 1.3 Product Versions

LoRaWAN hub:

REFERENCES	VERSIONS
WG0610-A01	WebdynEasy LoRaWAN

LoRaWAN Webdyn compatible sensor:

REFERENCES	DESCRIPTION
WG0307-D01-EU	Sens'RF-LoRaWAN-Pulse (no external power supply)
WG0307-D02-EU	Sens'RF-LoRaWAN-Pressure Humidity & Temp. (no external power supply)
WG0307-D03-EU	Sens'RF-LoRaWAN-TIC (no external power supply)
WG0307-D08-EU	Sens'RF-LoRaWAN-Analog (0-10V/4-20mA) (no external power supply)
WG0307-D11-EU	Sens'RF-LoRaWAN-Pulse (with external power supply)
WG0307-D12-EU	Sens'RF-LoRaWAN-Pressure Humidity & Temp. (with external power supply)
WG0307-D13-EU	Sens'RF-LoRaWAN-TIC (with external power supply)
WG0307-D18-EU	Sens'RF-LoRaWAN-Analog (0-10V/4-20mA) (with external power supply)

### 1.3.1 Safety Instructions

It is imperative to follow all the safety instructions in this guide.

Failure to follow these instructions can damage equipment and endanger people.



#### Electric connections

- All wiring must be carried out only by a specialized qualified electrician.
- Please follow all the safety instructions featured in the equipment documentation.



The WebdynEasy product can be damaged by electrostatic discharges (ESD). When the equipment is open, do not carry out any operations other than those described in this manual. Avoid any contact with the components.



Class 3 equipment: the device operates on safety extra-low voltage (SELV) (50V maximum). The voltage reduction must be obtained using a safety transformer providing safe galvanic isolation between primary and secondary.



Do not install the equipment near a heat source or at a height greater than 2m.



To clean the product, only use a slightly damp cloth to gently clean and wipe the surfaces. Never use aggressive chemical agents or solvents that could alter the plastic material or corrode the metal parts.



To optimise radio and cellular modem reception sensitivity, it is imperative to leave 20 cm free space around the antennas.

### 1.4 Regulations

The product complies with the European directives according to the EU Declaration of Conformity available from Webdyn or on website: [www.webdyn.com](http://www.webdyn.com).



Recycling:

The European directives enacted into national law covering battery waste and electric and electronic equipment provide the framework for the actions needed to limit the negative impact of the product's end of life. These products are collected separately. Use an authorised battery collection and processing centre or contact Webdyn.

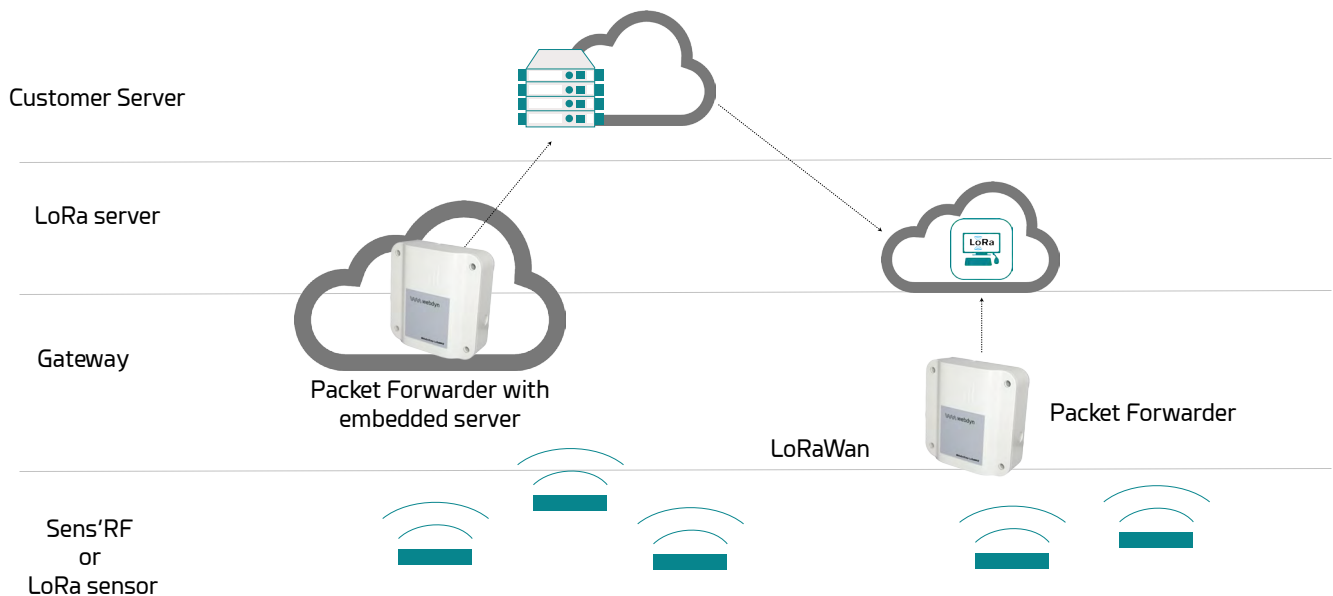


## 2. General Presentation

The WebdynEasy LoRaWAN hub is part of a line of Webdyn hubs specific to wireless networks. The hub's main function is to be a LoRaWAN gateway to create a LoRaWAN network and collect the data from the different LoRa sensors deployed nearby. The LoRaWAN gateway has 2 running modes:

- Packet Forwarder
- Packet Forwarder with embedded LoRaWAN server

The hub is also used to communicate with Modbus devices in IP or RTU mode.



Principle diagram for a complete LoRaWAN solution

### 2.1 The LoRaWAN Protocol

LoRaWAN is a communication protocol that uses LoRa modulation. This communication protocol uses several radio bands (ISM) that are available in the 868 MHz range in Europe without a licence.

In a LoRaWAN network, the radio modules are not paired to a single base station. The data they send is relayed by multiple base stations. Each one sends the information received from a radio module to the management server via a gateway. The intelligence and complexity are located on the server which manages data redundancy, integrity checks, receipt confirmation and the adaptation of the sensor data rate and emission power.

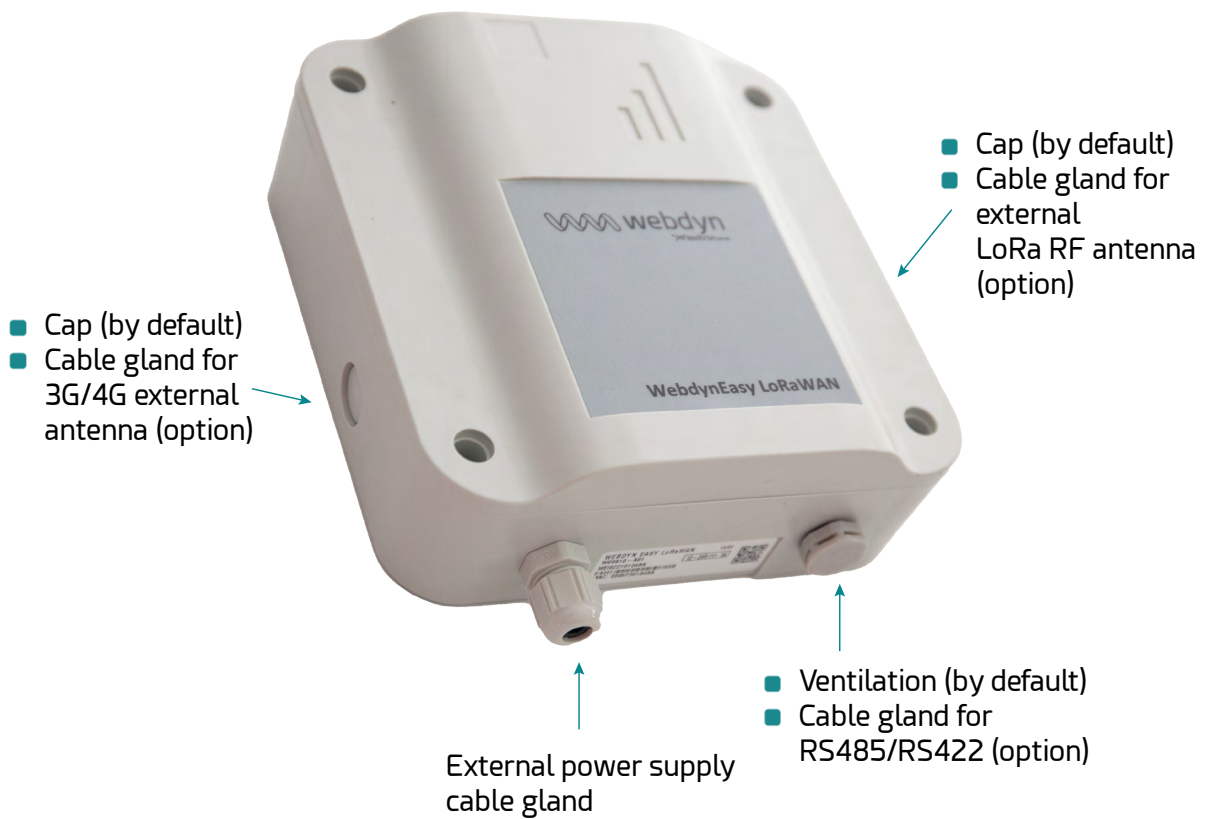
## 2.2 The Hub

The purpose of the hub is to collect LoRaWAN and/or Modbus data and regularly send it to a remote server (IS) using Ethernet or 3G/4G.

### 2.2.1 General description

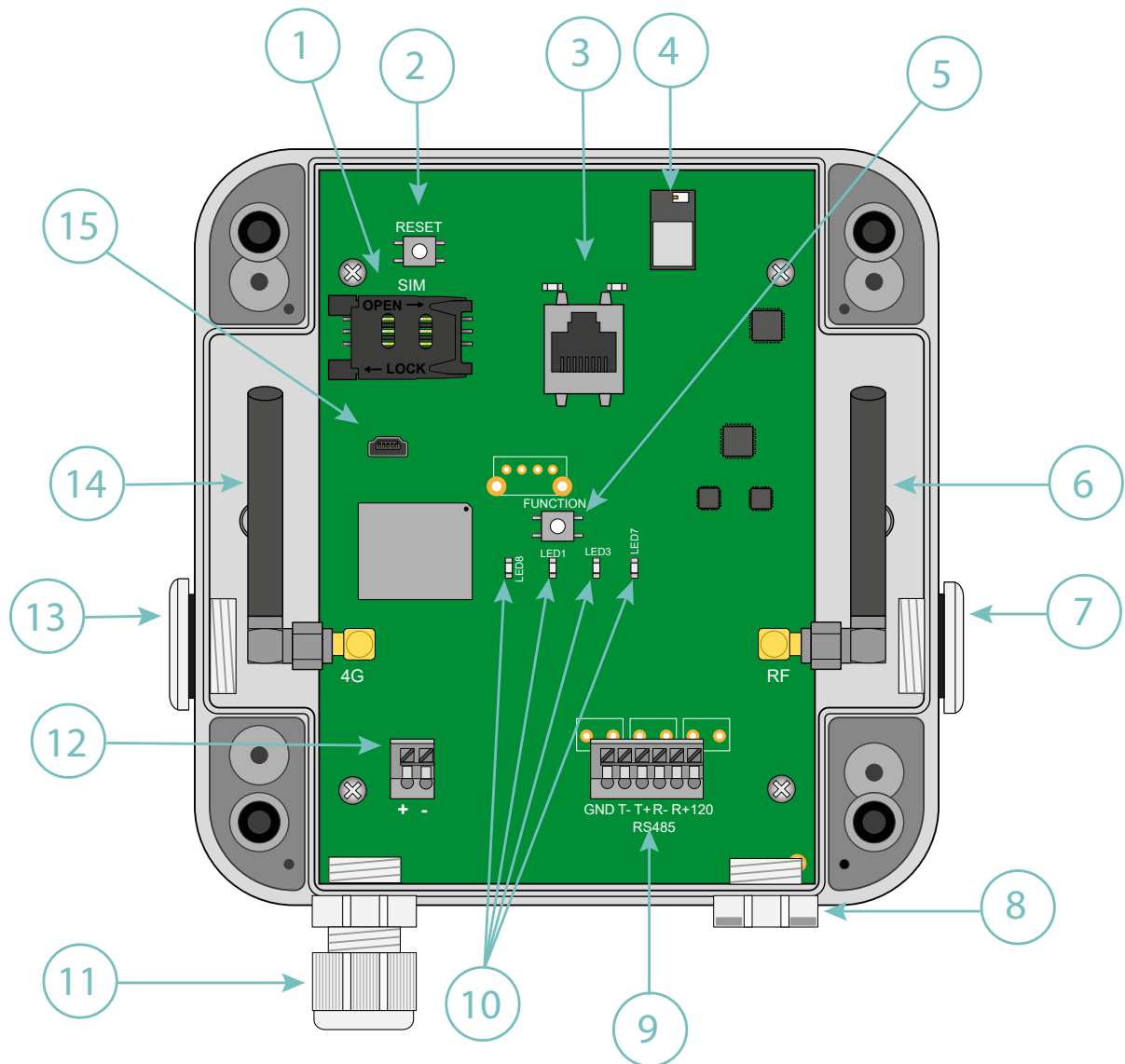
#### Exterior

Box front face:



## Interior

Interior of the box:



1. SIM card holder
2. Reset button
3. RJ45 connector and LEDs
4. BLE Bluetooth (future use)
5. Request button (identified FUNCTION on the board)
6. LoRa RF radio SMA antenna
7. Box output for the LoRa RF radio external antenna (option)
8. Box RS485/422 output

9. 1x RS485/422 port

10. Indicators:

- LED 8: Power
- LED 3: Modem
- LED 1: CPU
- LED 7: LoRa

11. Box output for external power supply

12. Terminal block for external 12/24V power supply

13.Box output for the 3G/4G modem external antenna (option)

14. 3G/4G Modem SMA Antenna

15. Mini-USB connector (reserved)

Indicators:

LED	DESCRIPTION
Power	Lights when the product is powered
CPU	Lights depending on the CPU activity
LoRa	Off by default and flashes to indicate LoRaWAN radio traffic
Modem	Lights when the Modem sets up an IP connection Lights for 1 second on receipt of a text message Following a long press on the Request button, it indicates the received signal level (RSSI) using a number of flashes (0 to 5 times) 0 – signal power $\leq$ -112 dBm 1 – signal power between -111 dBm and -96 dBm 2 – signal power between -96 dBm and -81 dBm 3 – signal power between -81 dBm and -66 dBm 4 – signal power between -66 dBm and -51 dBm 5 - signal power $>$ -51 dBm

Buttons:

BUTTON	DESCRIPTION
Request	Short press (less than 2 seconds) => Connection request  Long press (more than 2 seconds) => Displays the Modem signal reception level (see Modem LED)  3 successive long presses in less than 15 seconds => Return to factory settings
Reset	Hub reboot (Hard Reset)



Never press the RESET button 7 times in less than 30 seconds. This would switch the hub to a special mode that prevents it from starting. To exit the mode, a new hub RESET is required.






End users must make sure their installation using remote antennas meets applicable EMC standards.

## 2.2.2 Technical Specifications

### General Specifications

PARAMETERS	VALUES
External power supply	+12/24V DC from an external power supply
Consumption	10 Watts maximum
Flash memory	50 Mb (shared between compressed and uncompressed files)
Dimensions	160 x 150 x 55 mm
Box	ASA IP67 box
Weight	0.450 kg
Operating temperature	-20°C/+55°C
Storage temperature	-20°C/+70°C
Humidity	25 - 75 %

Pollution rating	2
Certification	RED ROHS REACH
Regulation	 CE marking created in the framework of European technical harmonisation legislation. It is mandatory for all products covered by one or more European regulatory texts (directives or regulations).
	 Symbol indicating that the waste must be collected via a specific channel and must not be disposed of as household waste.
	 Symbol indicating that the product must be recycled.

## Technical Specifications

PARAMETERS	VALUES
LoRa radio interface	863MHz -870MHz
Modem interface	3G: HSPA+, UMTS (B1, B8) 4G: Cat-1, Bands B1, B3, B7, B8, B20, B28
-Serial interface	1 RS422/RS485 Modbus RTU port
Ethernet network interface	10/100 Mbit/s

RF BAND	EMISSION FREQUENCIES	MAX. POWER
3G 2100MHz (B1)	1920–1980 MHz	23 dBm class 3b
3G 900 MHz (B8)	880–915 MHz	23 dBm class 3b

4G 2100 MHz (B1)	1920–1980 MHz	23 dBm class 3
4G 1800 MHz (B3)	1710–1785 MHz	23 dBm class 3
4G 2600 MHz (B7)	2500–2570 MHz	23 dBm class 3
4G 900MHz (B8)	880–915 MHz	23 dBm class 3
4G 800MHz (B20)	832–862 MHz	23 dBm class 3
4G 700MHz (B28)	703–748 MHz	23 dBm class 3

## LoRa Specifications

PARAMETERS	VALUES
Channels	8 simultaneous channels: <ul style="list-style-type: none"> <li>• 863-870 MHz (Europe)</li> <li>• 865-867 MHz (India)</li> </ul>
Max sensitivity	-141dBm (125kHz in SF12)
Supported DataRate	DR0-DR5
Supported Bandwidth	125/250 kHz
Max TX power	+14dBm
Activation mode	ABP or OTAA
Default frequencies	Europe: 867.1 MHz, 867.3 MHz, 867.5 MHz, 867.7 MHz, 867.9 MHz, 868.1 MHz, 868.3 MHz, 868.5 MHz India: 865.0625 MHz, 865.4025 MHz, 865.985MHz

Software Specifications

PARAMETERS	VALUES
LoRaWAN server	<ul style="list-style-type: none"><li>LoRaWAN V1.0.2 class A protocol</li><li>1000 LoRaWAN sensors supported</li><li>10 gateways supported</li></ul>
Modbus	Monitoring in RTU and TCP mode
OpenVPN	V2.5.4

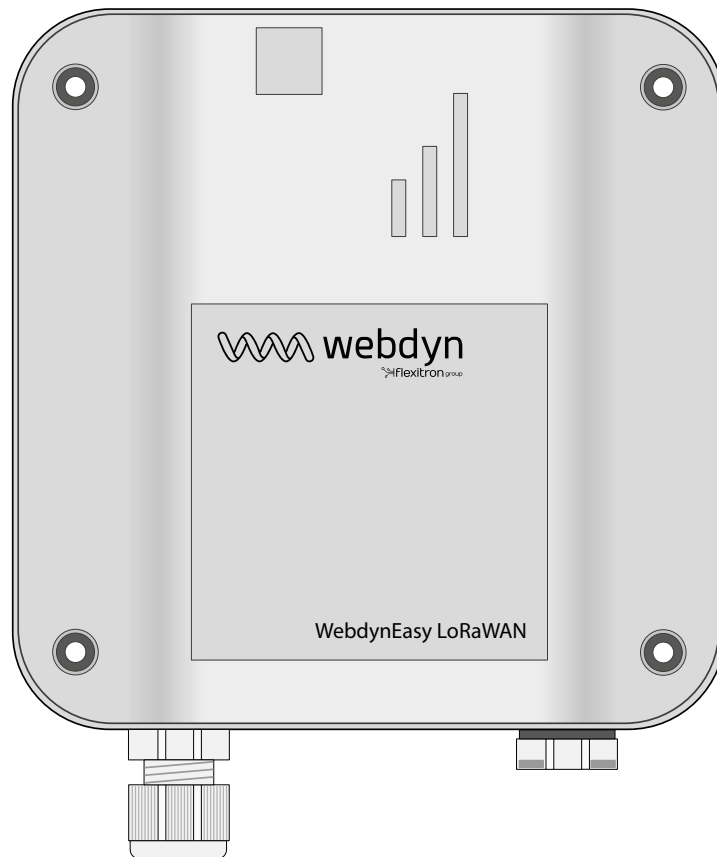


## 3. Installation and Maintenance

### 3.1 Unpacking

#### 3.1.1 Product Contents

Start by checking the contents before starting any installation work. If there are missing or damaged items, contact Webdyn support. (see section 7: “Support”)



WebdynEasy LoRaWAN hub  
(Ref.: WG0610-A01)

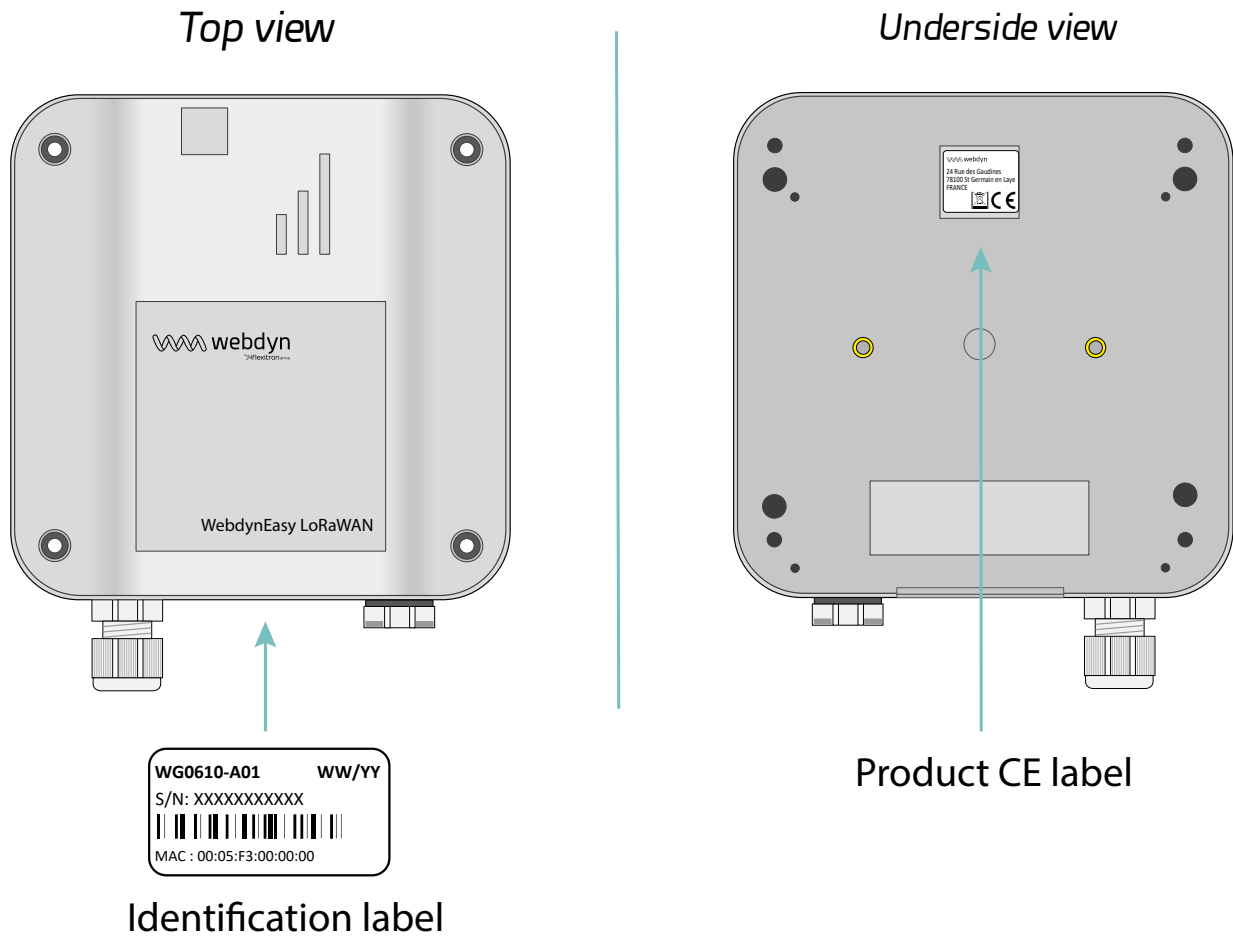
The following are shipped with the hub:

- a curved SMA antenna for the modem (internal)
- a curved SMA antenna for the radio (internal)

### 3.1.2 Hub Identification

#### Identification Label

The WebdynEasy LoRaWAN hub can be identified from its identification label located on the box.



This label features:

- Product name (WG0610-A01)
- The date of manufacture (in WW/YY format at the top right)
- The serial number in character and 128 barcode format
- The MAC (Ethernet) address in character format

#### Software Version

The software version can be found on the hub web interface. The software version is given on the “Overview” tab (See section 4.1.1: “Hub connectivity”).

## 3.2 Assembly

It is important to comply with the environment conditions described in section 2.2.2.1: “General specifications” before installing, as well as the following conditions:

- Protect the product from dust, moisture, aggressive substances and corrosion.
- The distance between the hub and Modbus equipment must not exceed the maximum authorised distance for the corresponding interface type (RS485 or RS422) (See section 3.2.5.2: “RS485/RS422 Bus”).
- If the Modem connection is used, make sure there is optimum reception when installing. Check the RSSI which is available on the embedded web page (See section 4.1.1.1: “Modem”).



To optimise Modem and LoRa radio reception sensitivity, it is essential to leave 20 cm free space around the antennas.

### 3.2.1 Opening/Closing the Box

#### **Follow these steps to open the hub box:**

If the box is wall-mounted:

- Open the 2 doors on the front panel.
- Unscrew the 4 wall mounting screws in the recesses under the doors.

Then follow these steps:

- Unscrew the 4 screws behind the box.
- Remove the cover.

#### **Follow these steps to close the hub box:**

- Place the cover on the box base, make sure the seal is properly fitted.
- Screw in the 4 screws on the back of the box.

### 3.2.2 Wall Mounting

The WebdynEasy can be wall-mounted. Before wall-mounting, first close the box (see section 3.2.1: “Opening/closing the box”)



Screws and anchors are not included in the kit. You must choose the correct type of screw for the type of wall you are fixing the hub to (4 mm diameter screw, minimum length 25 mm).

#### **Follow the steps below to fix the hub to a wall:**

- Open the 2 doors on the front panel.

- Screw the 4 wall mounting screws into the recesses under the doors.
- Close both doors on the front.

### 3.2.3 Cellular Network

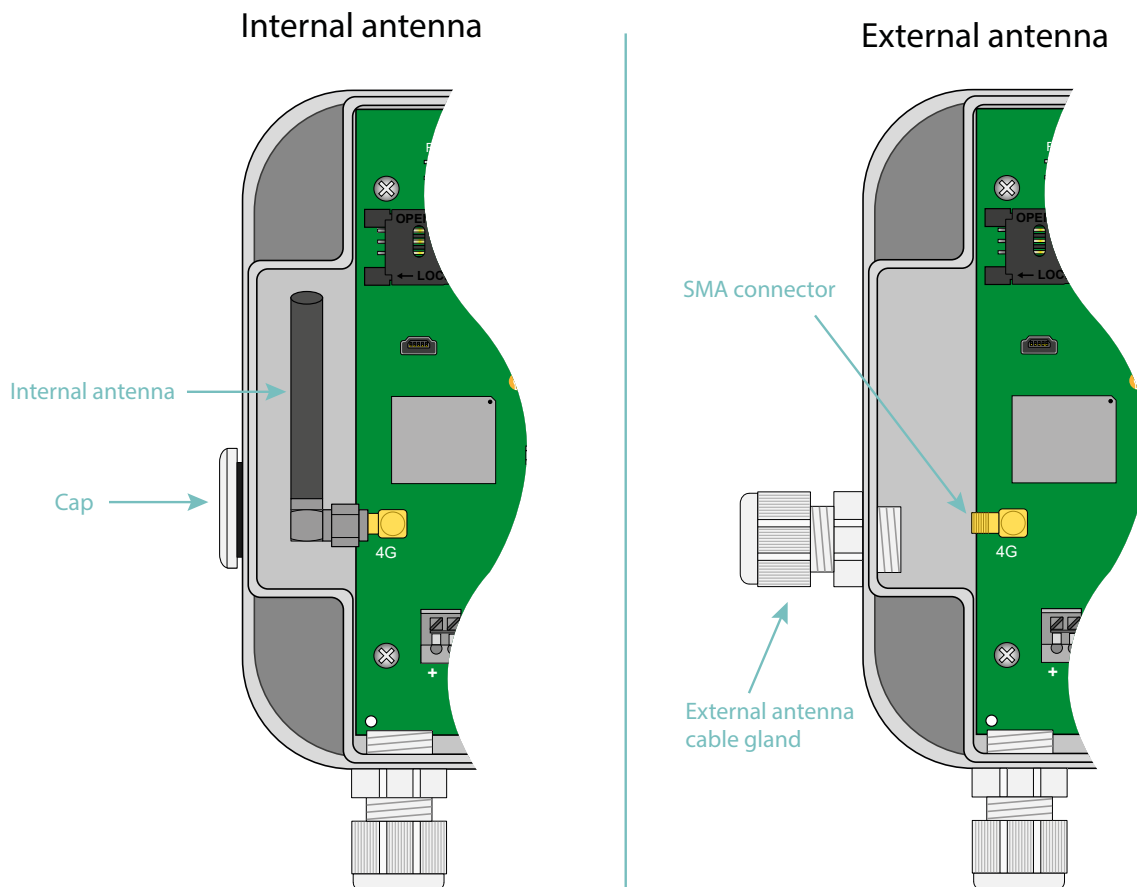
The WebdynEasy hub includes a 3G and 4G network compatible modem.

#### 3.2.3.1 Antenna

The hub has a female SMA connector labelled “4G” on the board to connect a modem antenna. The product is delivered with an internal antenna. An external antenna can be connected to the product. To do this, unscrew the cap on the box and fit a M16\*1.5 cable gland (not included).



If the WebdynEasy hub were to be installed in a metal box or in a location that does not have proper signal reception, the use of a remote antenna is strongly recommended. Be careful to use an antenna compatible with the connector and frequencies used.



End users must make sure their installation using remote antennas meets applicable EMC standards.

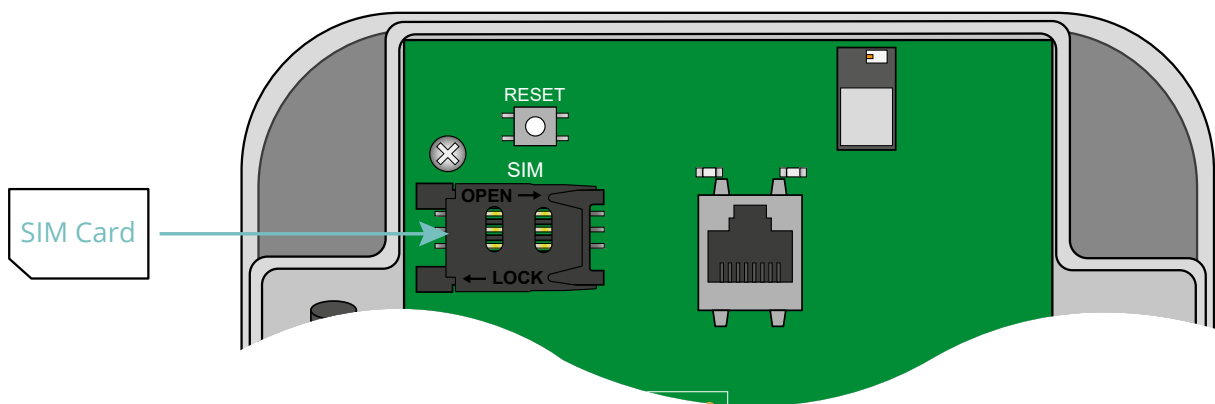
### 3.2.3.2 SIM Card

To use the 3G or 4G modem connection to allow the hub to communicate with the remote server, the box must be opened (see section 3.2.1: “Opening/closing the box”) and a mini SIM card inserted into the SIM card housing inside the hub.

The hub is compatible with all market operators as well as with all mini SIM 2FF 25 x 15mm format SIM cards.

To check that the WebdynEasy is operating properly, insert a SIM card with the following specifications:

- Possibility of receiving and sending text messages.
- 3G and 4G communication included.



To insert the SIM card into the product, slide the holder flap to the right (in the OPEN direction). Slide the SIM card into the flap. Then close the flap by sliding it to the left (in the LOCK direction)



Webdyn does not supply any SIM cards. Please contact an M2M operator that supports the 3G and LTE-M network.



Please contact your SIM card provider to find out what information to enter to configuration the modem.

By default, the hub configuration does not request a PIN code (PIN Mode: Off). If you want to enable the hub PIN code, it is preferable to configure it before the SIM card is installed. (see section 4.1.1: Hub connectivity)

There are three possibilities:

- The PIN code is disabled: modem communication is active.
- The PIN code is enabled and the entered PIN code is correct: modem communication is active.
- The PIN code is enabled and the entered PIN code is incorrect: modem communication is in error.

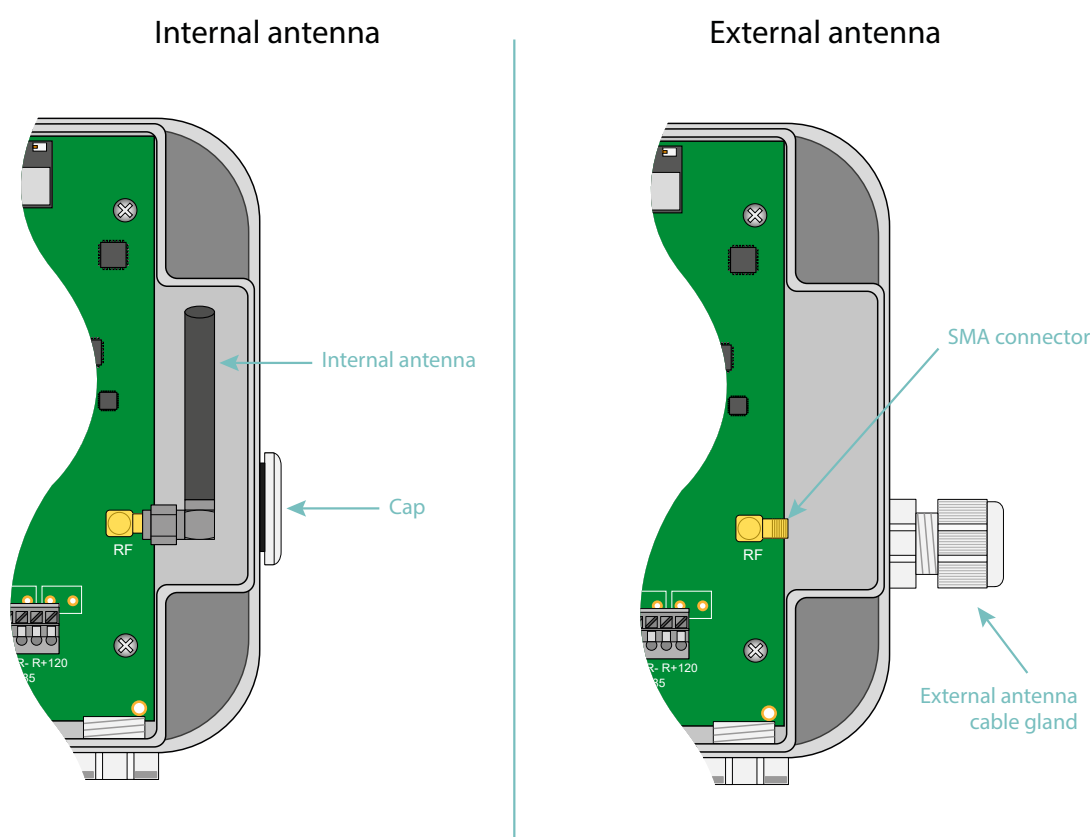


If the SIM card has an enabled PIN code and it is incorrect the first time the hub is started, it will be blocked after 3 attempts. It can be unlocked using a mobile phone using the PUK code provided by the operator.

### 3.2.4 LoRa

The hub has a female SMA connector labelled “RF” on the board to connect a radio antenna. The product is delivered with an internal antenna. An external antenna can be connected to the product. To do this, unscrew the cap on the box and fit a M16\*1.5 cable gland (not included).

To optimise the radio range, it is important to install the radio antenna as high as possible and to place it carefully, avoiding obstacles as far as possible. As a priority, move it away from any metal (cupboard, beams...) or concrete (reinforced concrete, walls...) obstacles as they greatly attenuate radio waves.



End users must make sure their installation using remote antennas meets applicable EMC standards.

### 3.2.5 Connection

#### 3.2.5.1 Power Supply

The WebdynEasy hub must have a 12V or 24V DC power supply. Power is supplied from terminal block J11 on the bottom left side of the board.



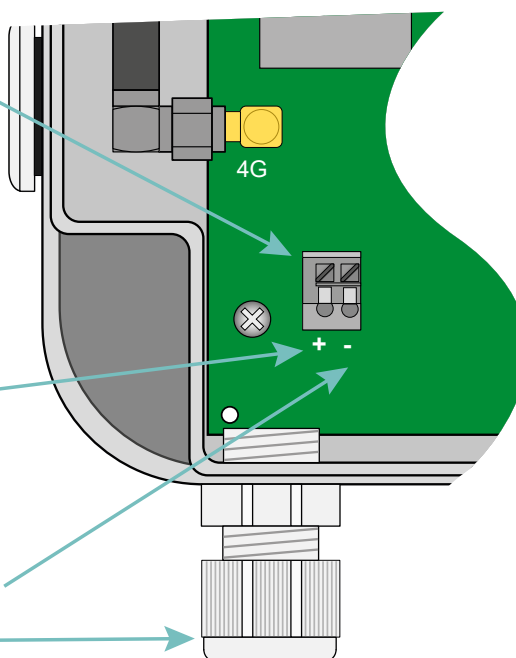
End users must use a CE certified power supply of less than 15 watts. The distance between the power supply and the product must not exceed 3 metres. End users must make sure their installation meets applicable EMC standards.

Terminal block for external  
12V/24V power supply ==

12V/24V terminal ==

Earth

External power  
supply cable gland



Make sure the power supply wires are connected to the proper terminals.

Product power consumption varies depending on its configuration. Make sure the power supply used can provide a minimum power of 10Watts.

### 3.2.5.2 RS485/RS422 Bus

The RS485/RS422 communication bus is only used for RTU mode modbus, RS485 is screen printed at the bottom right of the board. This interface is Half Duplex (2 wires) and Full Duplex (4 wires) compatible.

If several modbus RTU devices are connected, the wiring must be “serial”. The cable arrives at a modbus module and exits towards the next one.

To guarantee proper data bus operation, an RS485 bus must feature a 120 Ohm terminator at each end. The WebdynEasy hub can be located at the end of the RS485 communication bus or in the middle. As the hub has a 120 Ohms resistor, it made need to be enabled depending on the hub position on the bus (see wiring).

There are 3 separate considerations for the choice of cable type:

- On installations requiring short lengths with no electric interference, plan on using a 2 pair 6/10 rigid screened cable.
- On larger installations of which the cable length is less than 500 m, plan on a 2 pair 8/10 rigid screened cable.
- When the cable distance is more than 500 m, and even more so if there is electric interference, plan for a shielded 2 pair 0.34 mm<sup>2</sup> cable.



The maximum RS485 bus length is 1000 metres.



Recommendations for RS485/RS422 BUS wiring:

- The modules must be connected one after the other.
- Star connections are prohibited.
- The cables must either be screened or shielded, twisted pair per pair (see above: “cable type for RS485 bus connection”).
- The cable screen or shielding must be connected to the hub box earth and not to the 0 V (only connect one end of the screen).
- Avoid any return trips in the same cable.

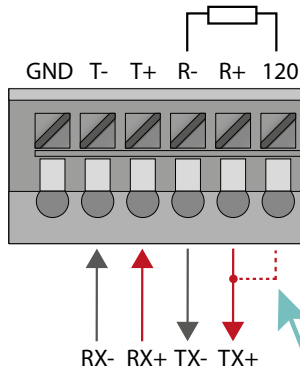
RS485 wiring on the hub side:

- Strip the RS485 communication cable sheath over about 4cm.
- Shorten the shielding down to the cable sheath.
- Strip the wires over about 6 mm.
- Connect the conductors to the terminal block marked RS485, following the assignments in your RS485 communication bus.

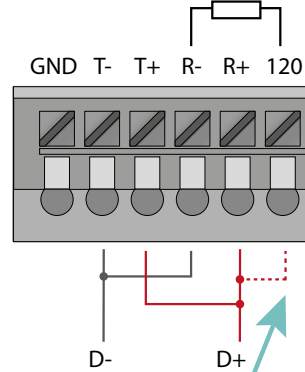


RS485/RS422 assembly:

4 wires



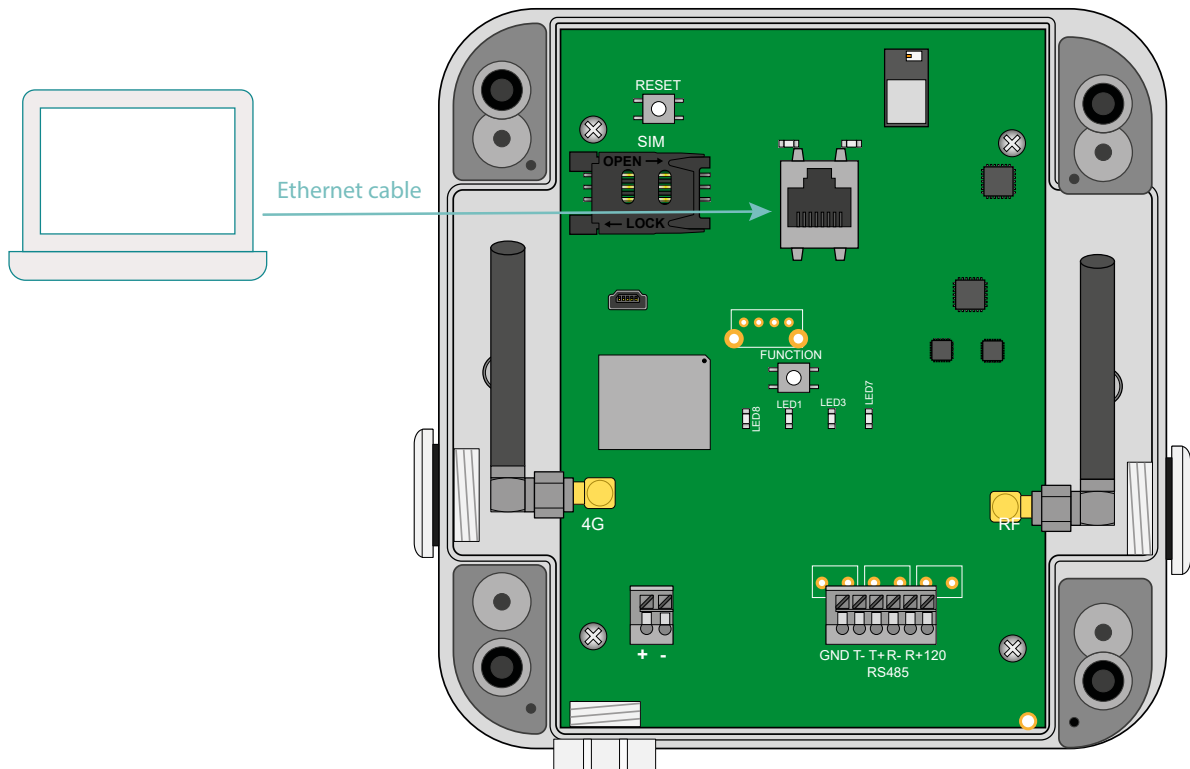
2 wires



Only to be wired if the hub is at the end of the line

### 3.2.5.3 Ethernet

To configure the hub, first open the box to access the RJ45 connector (see section 3.2.1: “Opening/closing the box”). Once open, connect the hub to the computer using an Ethernet cable.



A static IP address needs to be configured on the computer in the same IP address range and the same subnet as the WebdynEasy LoRaWAN hub.



The WebdynEasy LoRaWAN hub default configuration parameters are the following:

IP address: 192.168.1.12

Subnet mask: 255. 255. 255.0

DHCP: Deactivated

The next step is used to configure a PC network address to access the WebdynEasy LoRaWAN hub:

**Configuring a second IP address on the PC:**

- Under Windows 10, click Start/Settings/Network & Internet. The “Network Status” window is displayed.
- Click on “Ethernet” on the left of the window, then “Network and sharing centre” on the right.
- The “Network and sharing centre” window is displayed.
- Click “Ethernet” connections. The “Ethernet Status” window is displayed.
- Click “Properties”.
- Select “Internet Protocol (TCP/IPv4)” then click the “Properties” button.
- Then click “Advanced”.
- In the “IP Address” zone, click “Add”.
- Enter IP address 192.168.1.xxx (xxx between 1 and 254 and not equal to 12) and the subnet mask 255. 255. 255.0.
- Click “Add”.
- To validate the settings, click OK in each one of the three windows.
- Close the Network connection and remote access window.

It is now possible to easily modify the hub configuration using its embedded web interface using the computer’s web browser. (See section 4.1.1 : “Hub connectivity”)

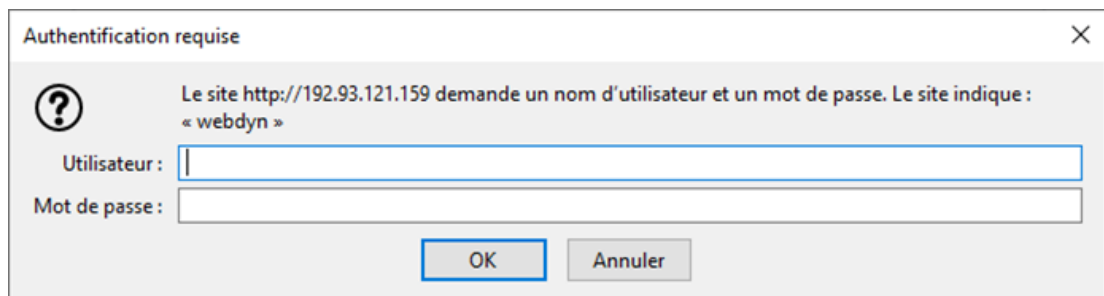
## 4. Configuration

The first time the WebdynEasy LoRaWAN hub is configured, the Web interface embedded in the product is used.

### 4.1 Embedded web interface

To access hub's embedded web interface, proceed as follows:

- Launch the web browser. The web interface is compatible with the latest versions of the following browsers: Firefox, Chrome and Edge. Older versions may work but they are not supported (IE 7 for example).
- Enter the hub IP address in your browser (the default address is: `http://192.168.1.12`) to access the WebdynEasy LoRaWAN home page.
- An identification window should be displayed:



Enter your login and the password:

LOGIN	PASSWORD	RESTRICTIONS
admin	high	None
install	medium	System, LoRaWAN, Modbus, Actions Read-only schedules
data	low	Actions only



Password: to secure access to the hub, we recommend changing the default passwords following the first configuration. Passwords are changed in the XML configuration file (see: "Appendix A: XML configuration file variables")

- The home page is displayed:



The “Overview” tab provides an overall view of WebdynEasy LoRaWAN operation



If web pages are accessed during the hub initialisation phase, the



logo is

displayed. Wait for the hub to be fully initialised to access the web pages

## 4.1.1 Hub Connectivity

The “Connectivity” tab is used to configure the hub to communicate with the remote server.

### 4.1.1.1 Modem

**Modem**

PIN Mode:

Off

PIN Code:

1234

APN:

tm

Login:

Password:

Mode:

AlwaysOn

Disconnect delay (s):

60

PARAMETERS	DESCRIPTION
Pin Mode	Off: The SIM card PIN code must be disabled  Manual: The SIM card PIN code must be entered in the PIN Code box
PIN Code	The SIM card PIN code must be entered if Manual is selected in PIN Mode
APN	Your mobile operator’s APN name (required for an IP connection)
Login	Your mobile operator’s user name (optional depending on the operator)
Password	Your mobile operator’s password (optional depending on the operator)
Mode	OnDemand: The hub only connects when it needs to communicate with the remote server. It cuts the connection when the data transfer is complete after a duration that can be configured in Disconnect delay.  AlwaysOn: The modem is always connected. The hub uses the modem continuously for all IP communications.  AlwaysOff: This mode should be used if there is an Ethernet connection to the remote server, but with a SIM card inserted in the hub. The connection never uses the modem, but the hub can receive and send text messages.

Disconnect delay (s)	Waiting time value in seconds for OnDemand mode between the end of data exchanges and disconnection
----------------------	---



Refer to your mobile operator to obtain the information for your SIM card (APN, login, password).

#### 4.1.1.2 Ethernet

Ethernet

IP:  •  •  •

Netmask:  •  •  •

Gateway:  •  •  •

☐ Use DHCP

---

DNS

DNS servers:  •  •  •

PARAMETERS	DESCRIPTION
IP	The IP address used to access the WebdynEasy LoRaWAN hub via the Ethernet network.
Netmask	Your Ethernet network subnet mask. This mask limits the Ethernet network to defined IP addresses and separates the network ranges from each other.
Gateway	Your Ethernet network gateway address. The gateway address is the IP address for the device that connects to the internet. The address entered here is usually your ADSL/fibre router address.
Use DHCP	Ethernet parameters can be obtained automatically if the network infrastructure allows. In that case, select dynamic mode and refer to your DHCP server configuration to find your hub IP address.
DNS servers	List of DNS servers. DNS (Domain Name System) servers translate explicit internet addresses (for example, <a href="http://www.webdyn.com">www.webdyn.com</a> ) into their corresponding IP addresses. Enter the DNS server addresses you received from your internet service provider (ISP) here. You can also enter your router IP address. You can also use the google DNS: "8.8.8.8".



The hub can only use the Ethernet connection to access the server if the modem connection is disabled ("off" or "alwaysoff"). Otherwise the hub will attempt to connect using the modem.

4.1.1.3 FTP

FTP

Address:

ftp3.webdyn.com

Login:

login

Password:

password

Root:

/LoRaWAN

PARAMETERS	DESCRIPTION
Address	IP address or remote FTP server name (default port: 21). The FTP port can be changed by adding ": " and then a port number.
Login	The login used by the hub to connect to the remote FTP server.
Password	The password used by the hub to connect to the remote FTP server.
Root	Remote FTP server root directory.



The directory tree structure is to be created on the remote FTP server before any FTP connections. (see section 5.1.1.1: “The FTP server: Configuration”)

4.1.1.4 Web Services

Web services

URL:

http://192.93.121.120:500

Login:

login

Password:

password

Proxy:

Trust model

Verify peer

Upload POST path:

/upload

PARAMETERS	DESCRIPTION
URL	IP address or name of the remote web server (Default port: 80). Possibility to modify the port of the web server by adding “:” then the port number.
Login	Username used by the hub to connect to the remote web server.
Password	Password used by the concentrator to connect to the remote web server.

Proxy	Proxy IP address or hostname (Default port: 1080). Ability to modify the proxy port by adding “:” then the port number. The proxy is optional if it is not used, the empty field must be empty.
Trust model	Verification of authentication certificates (only for secure HTTPS connections): <ul style="list-style-type: none"> <li>• Verify peer: Verification of authentication certificates.</li> <li>• Trust peer: Accepts all authentication certificates (not recommended).</li> </ul>
Upload POST path	Path on the remote web server.

#### 4.1.1.5 MQTT

PARAMETERS	DESCRIPTION
<i>Adress</i>	IP address or name of the remote web server (Default port: 1883)  Possibility to modify the port of the mqtt server by adding “:” then the port number
<i>Client ID</i>	Client’s MQTT identifier
<i>Login</i>	Username used by the hub to connect to the remote MQTT server
<i>Password</i>	Password used by the hub to connect to the remote MQTT server
<i>Keepalive interval (s)</i>	Time in seconds for sending keep-alive frame
<i>Topic</i>	Topic of MQTT messages used
<i>Trust model</i>	Verification of authentication certificates (only for MQTTS secure connections): <ul style="list-style-type: none"> <li>• Verify peer: Verification of authentication certificates.</li> <li>• Trust peer: Accepts all authentication certificates (not recommended)</li> </ul>



4.1.1.6 NTP

Time

Alarm threshold (s):

0

NTP

NTP servers:

pool.ntp.org

PARAMETERS	DESCRIPTION
Alarm threshold (s)	Difference in seconds between the hub time and the NTP synchronisation time beyond which an alarm is issued.
NTP servers	Addresses of the NTP servers used for the hub clock synchronisation.

i

At the first connection, NTP synchronisation is carried out and the next NTP synchronisation will be carried out during another connection after a minimum period of time. The minimum time between NTP synchronisations can be configured using the “min\_syn\_interval “ variable in seconds.

4.1.1.7 Upload

Upload

Configuration

Method:

FTP

Supervision data

Method:

FTP

Alarms

Method:

FTP

Data

Method:

FTP

Format:

XML

Schedule:

NOT SET

The concentrator can deposit the following data on the remote server:

NAMES	DESCRIPTION	FORMAT
Configuration	Hub configuration data	• XML
Supervision data	Hub monitoring data	• XML
Alarms	Alarms	• XML
Data	LoRaWAN and/or modbus data	• XML • JSON

For each type of data, the concentrator can deposit the data by:

- FTP
- Web Service



If commands are sent by Web Service, the concentrator responds to the commands through alarms. In this case, it is necessary to configure the alarms in Web Service.

The data repository must be associated with a Schedule by entering its configured unique identifier (see chapter 4.1.6: “Schedules”).



Consult chapter 5.2: “Configuration” to find out about the format and content of the configuration, supervision, alarm and data files.



The directory tree on the remote FTP server must be created before any file upload. (see chapter 5.1.1.1: “The FTP server: SettingsSettings”).

## 4.1.2 LoRaWAN

The “LoRaWAN” tab is used to configure the Packet Forwarder and the LoRaWAN server. These 2 parts are completely separate. The Packet Forwarder can be used with a remote server and the embedded LoRaWAN server can use an external Packet Forwarder.

### 4.1.2.1 Packet Forwarder

In Packet Forwarder mode, the WebdynEasy LoRaWAN has a gateway role. The gateway continuously polls the LoRa radio interface and sends all received frames to the LoRaWAN server (remote or embedded) using an IP connection.

For the Packet Forwarder to operate, it must set up a permanent IP connection to the server using its Ethernet interface or Modem in AlwaysOn mode.

Packet Forwarder

Server address:

127.0.0.1

Upstream server port:

1700

Downstream server port:

1700

Keepalive interval [s]:

10

Push timeout [ms]:

10

PARAMETERS	DESCRIPTION
Server address	LoRaWAN server IP address or name. To use the embedded LoRaWAN server on the hub, use the following address: “127.0.0.1”
Upstream server port	Packet Forwarder outgoing UDP port number.
Downstream server port	Packet Forwarder incoming UDP port number.
Keepalive interval [s]	Time in seconds to send a keep alive frame.
Push timeout [ms]	Maximum waiting time in milliseconds to acknowledge the frame sent to the LoRaWAN server.

i

The supported Packet Forwarder is the Semtech forwarder.

4.1.2.2 LoRaWAN server

The LoRaWAN server manages the LoRaWAN sensors as a private network. It includes all the LoRaWAN network functions (gateway, LoRaWAN server and application server). All received data is stored in files and all available data is uploaded at every FTP connection.

i

To use the hub Packet Forwarder enter the following IP address in “Server address”: “127.0.0.1”. Also check that the server ports (“Upstream server port” and “Downstream server port”) are set to 1700.

Server Configuration

Net ID:

1532673

ADR

Enable:

☒

Margin [db]:

5

Uplink count:

20

PARAMETERS	DESCRIPTION
Net ID	24 bit hexadecimal value used to identify LoRaWAN networks. If you enter 0, when the hub reboots, it will use its factory NetID.
Enable	Check to enable ADR (Adaptive Data Rate).
Margin [dB]	Margin in dB to calculate ADR between 1 and 30.
Uplink count	The number of uplinks needed for ADR between 1 and 65535 included.



In order to optimize the sensor stacks and the LoRaWAN bandwidth, it is strongly recommended to leave ADR enabled and the default configuration of Margin and uplink count.



For the calculation of the ADR, the concentrator needs at least 20 uplinks, i.e. the Uplink Count variable is used after the first 20 uplinks received by the concentrator before sending ADR commands to the sensor LoRaWAN.

LoRaWAN sensor:

The server supports 2 activation modes:

- ABP (Activation By Personalization): the DevAddr, NwkSKey and AppSKey parameters must be entered.
- OTAA (Over The Air Activation): the DevEUI and AppKey parameters must be entered, the AppSKey and NwkSKey keys are generated and saved at JOIN time.

The image shows a web form titled 'Endpoint'. It contains five input fields labeled 'DevEUI:', 'AppKey:', 'DevAddr:', 'AppSKey:', and 'NwkSKey:'. At the bottom of the form are two buttons: 'Cancel' and 'Apply'.

PARAMETERS	DESCRIPTION	ABP	ABP
DevEUI	Unique sensor identifier (EUI64) in 8 byte hexadecimal format	empty	•
AppKEY	Hexadecimal 16 byte encryption key used by the network to derive the session keys.	empty	•

DevAddr	Sensor address in 4 byte hexadecimal format	•	auto
AppSKey	Encryption key between the sensor and the application server in 16 byte hexadecimal format	•	auto
NwkSKey	Encryption key between the sensor and the LoRaWAN server in 16 byte hexadecimal format	•	auto



AppEUI is not used by the embedded hub server.

The LoRaWAN sensor data is uploaded in XML format (see section 4.1.1.5: “Upload”) to the DATA directory on the remote FTP server (see section 5.3 “The data”).

### 4.1.3 System

When the Modbus protocol is enabled on the RS485 port, the serial port parameters must be defined.

PARAMETERS	DESCRIPTION
Mode	Off: RS485 disabled Modbus: RS485 enabled in Modbus mode
Baudrate	4800 9600 <b>19200</b> (default value) 38400 57600 115200

Data bits	5
	6
	7
	<b>8</b> (default value)
	9
Parity	None
	Odd
	<b>Even</b> (default value)
Stop bits	<b>1</b> (default value)
	2

#### 4.1.4 VPN

The hub supports VPN from OpenVPN V2.5.4 (<https://openvpn.net/>).

OpenVPN

Enable: ☒

Protocol:

Server

Address:

Port:

Cipher:

Auth:

CA: 

-----BEGIN CERTIFICATE-----

MIIFMTCCAxmgAwIBAgIJALZAhXiaEan1MA0GCSqGSIb3DQEBCwUAMBQxEjAQBglNVBAMMCVdYmR5biBDQTAqFw0xOTA2MTcxMTQzMjJhGA8yMTE5MDUyNDExNDMxMlowFDESMBAGA1UEAwVJV2VhZHUuIENBMIIjXANBgkqhkiG9w0BAQEFAAOCAg8AMIICCgKCAgEApTNkOoFT+HpFS4vAlmUd2upTYgobJfQ0NoS1LxNLC79WIFRydnwcf4sdGHVejXIZ3zGCFvDuzKabwJpS38XFBnR9qWJPKDcB57+MFsay4QFBRzQ8O+Ibya4boLKRRi6Ivw0oLRi5vWSpfBsDTd36cvEq5Vp857Vzf44EJhbDGHhIaGmSJwZSk5IG9+IqbRBUD+/m3eZZUDz2A8abvc1Px3mnNpF0vUslP4kG2+nz4V9R2oXuoU6HSqipPC4YoaF6YEPxXFLyGbZKGrEa0zvTc7QFTsQooYIL7aRMOuwgIaWF4IzWaEU+o6rZXvHLP  
TmaYk/ciaolGNJwdTfoca8Wfyu3ZolikhRo0KEfL9II

-----END CERTIFICATE-----

Cert: 

-----BEGIN CERTIFICATE-----

MIIFUzCCAzugAwIBAgIQObrCQvrfPFZktu0g33L8TAN8gkqhkiG9w0BAQsFADAUMRIWEAYDVQQDDAAXZWJkeW4gQ0EwHhcNMjAwMzEyMTA1MDQ1WWhcNMzAwMzEyMTA1MDQ1WjAqFMR0wGwYDVQQDDBRqZWVhZHUuIENBMIIjXANBgkqhkiG9w0BAQEFAAOCAg8AMIICCgKCAgEApTNkOoFT+HpFS4vAlmUd2upTYgobJfQ0NoS1LxNLC79WIFRydnwcf4sdGHVejXIZ3zGCFvDuzKabwJpS38XFBnR9qWJPKDcB57+MFsay4QFBRzQ8O+Ibya4boLKRRi6Ivw0oLRi5vWSpfBsDTd36cvEq5Vp857Vzf44EJhbDGHhIaGmSJwZSk5IG9+IqbRBUD+/m3eZZUDz2A8abvc1Px3mnNpF0vUslP4kG2+nz4V9R2oXuoU6HSqipPC4YoaF6YEPxXFLyGbZKGrEa0zvTc7QFTsQooYIL7aRMOuwgIaWF4IzWaEU+o6rZXvHLP  
TmaYk/ciaolGNJwdTfoca8Wfyu3ZolikhRo0KEfL9II

-----END CERTIFICATE-----

Key: 

-----BEGIN RSA PRIVATE KEY-----

MIUKgIBAAKCAgEA0Lwd/GVOTYvzrfY/Z7hImW25xNjLSaE5yg6vwzoqA1NBrJVNT2NyhulAMwTqMtCjaep55+n5uwCkajDfCLNNGIBK0lyxeqyvNevYKnr8N5/wEFW7QEJU8NY+aoOsRWSu+Bmp+Rnm8H4wS8DA56LAB/01owgv+N6FaIACX0R45SWBKqIvFazgTT+CtV0EaaUTVE5QRfsZApHoNFRozI1NC1dswcwKHQJPN5OjUfFBZQ0IDY6aHkKcB6a84KHq97Gsu/PUJqTr20DE/pyMVHO46M6mJ1qHGJnCIQ/dcWHeiHmaHEAzhKaZfyD8G1ze0eILPmPSrKCb0B2hOScc+9xtkJ7V9K63XoGxTmSZUIRj5sn4/B3Srm/AjfbhhdCFTmCN0ATWdnUAsWjryno0tgUXNZrrqGWILLQazKsdJm1ghVATtoRzrCKKtVfASioplU9v0QUaoO9vLPwCelDztfCh4J0cJH6lhoEvosrv5ORD

-----END RSA PRIVATE KEY-----

Channel Security

Method:

Key: 

-----BEGIN OpenVPN Static key V1-----

12e269fae5bb7dbc4a904faeac6ca8af66f3a5d66718cf89b83ce03a3ec9fd08f6b153029bc998da300e5ceb010fe30f28789b437eca1fe42aa445cba25f948c59d955de355db0fb427718431f42635b32be8ccd4626b3f4a9e8a2f9cc71d4f51b4b1171db261285ce1566cac825f4c95c9f1592543d53652be954adabc0a406318d6c655fab46c0ecc67289f98b66031fdbb5a8313a6a68f5aecb4c1f5d8eb1a67700cbadaaeaf7f346ff90c6edd9687f2e791f7efc4871faf6539add97ae3486d360a74c360ec4593645e8871fe16a28c8391969dd075cdb424e4214350a3bd89ea75651087552eb889021e536b3c99f8034478155dba939ef87f0499a9405-----END OpenVPN Static key V1-----

-----END OpenVPN Static key V1-----

webdyn | 39

contact@webdyn.com | webdyn.com  
V3.12 subject to changes | Webdyn © by Flexitron Group

PARAMETERS	DESCRIPTION
Enable	Check to enable VPN
Protocol	Communication protocol used: <ul style="list-style-type: none"> <li>• tcp</li> <li>• udp</li> </ul>
Address	IP address or VPN server name
Port	VPN server port (usually 1194)
Cipher	Data packet encryption algorithm (optional). List available on OpenVPN ("openvpn --show-ciphers" command).
Auth	HMAC hash algorithm to authenticate data packets. If "TLS Auth" is entered then the hashing algorithm also applies to control packets. If the field is empty, the value used by default is "SHA1" (optional). List available on OpenVPN ("openvpn --show-digests" command).
CA	CA root certificate. In PEM file format
Cert	Local client signed certificate. In PEM file format
Key	Local client's private key. In PEM file format

#### Channel Security:

PARAMETERS	DESCRIPTION
Method	List of methods for control channel security: <ul style="list-style-type: none"> <li>• none: none</li> <li>• tls-auth: static key used for the HMAC hashing algorithm on control packets.</li> <li>• tls-crypt: Same as tls-auth, but also encrypts the TLS control channel.</li> <li>• tls-crypt-v2: Same as above, but uses one key per client instead of a shared group key.</li> </ul>
Key	Key for control channel security. In PEM file format.



To find out what information to enter for the VPN configuration, please contact the network administrator of the VPN server.





The configuration of an NTP server is mandatory for the use of a VPN in order to verify the validity of the certificates. (see chapter 4.1.1.5: “NTP”).

## 4.1.5 Alarms

The hub can generate system alarms.

**System alarms**

Modem IP: Off ▼

MSISDN: Off ▼

SW Version: On ▼

---

Defaults

Ignored:

Delayed:

There are 3 types of system alarm:

- Modem IP: alarm generated if the IP address obtained during a modem connection changes.
- MSISDN: alarm generated if the SIM card inserted in the hub is replaced.
- SW Version: alarm generated if the firmware or core versions change (following an update).

Each alarm source can be enabled separately and immediately transferred to the remote server (On) or transferred at the next connection (Delayed).

The hub malfunction alarms (“Default”) are sent to the remote server immediately by default. They can however be disabled (“Ignored”) or delayed (“Delayed”) and sent at the next connection. To do that, the codes must be entered in the fields corresponding to the required behaviour.

Below are the available codes and faults:

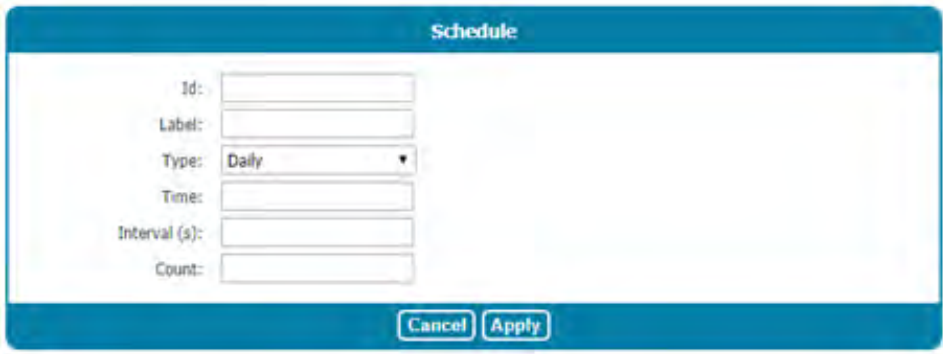
CODE	DESCRIPTION
D_MODEM	Modem fault
D_MODEM_SIM_MISS	SIM card missing
D_MODEM_SIM_CODE_FAIL	SIM code error
D_MODEM_PUK	SIM card locked
D_MODEM_REG_DENIED	Network registration declined

The fault codes ignored by the hub can be listed in the Ignored box. If several fault codes are entered, they must be separated by a comma “,”.

The fault codes to be transferred by the hub at the next connection can be listed in the Delayed box. If several fault codes are entered, they must be separated by a comma “,”.

### 4.16 Schedules

The scheduler is in charge of all regular tasks. The scheduler configuration is a list of schedules. Each schedule has a unique identifier which is used to link one or more tasks to a schedule. They can be used separately to trigger data collection and data uploads.



Each schedule is configured as follows:

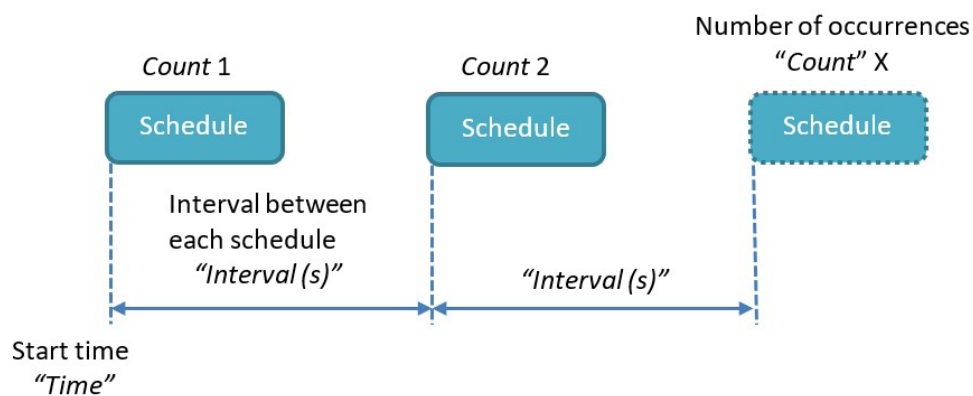
PARAMETERS	DESCRIPTION
Id	Unique schedule identifier. The identifier must be an integer. (between 1 and 2,147,483,647 included).
Label	Purely descriptive schedule name.
Type	Daily, Weekly, Monthly, Yearly or Follower: see description below.
Time	Time of the first occurrence in “HH:MM:SS” format (not used for “Yearly” type schedules).
Day of Week	Number of the day in the week of the first occurrence (1=Monday, 7=Sunday) (only used for “Weekly” type schedules).
Day of Month	Number of the day in the month of the first occurrence (only used for “Monthly” type schedules).
Date & Time	Date and time of the first occurrence in a given period (only used for “Yearly” type schedules).

Interval (s)	Interval between occurrences (in seconds).
Count	Number of occurrences (at least 1).
Parent	Reference to the parent schedule for a “Follower” type schedule.

Configuring the different schedule types:

- “Daily” type schedule:

Every day, the first occurrence is given by the time entered in the “Time” field. The number of events in the day is given by the “Count” field and the interval between each event by the “Interval” field



The Time format is the following: HH:MM:SS (for example 09:30:00)

The “Count” value is between 1 and 2,147,483,647 included

The “Interval” value is between 0 and 2,147,483,647 included



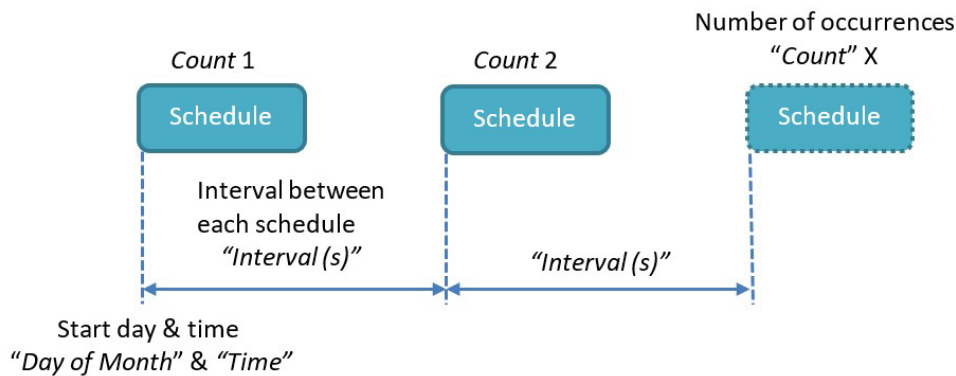
“Count”: if the schedule is to be triggered throughout the day at regular intervals, you can enter the maximum value (namely 2,147,483,647) in “Count”.

Example:

NEED	TYPE	TIME	DAY/ WEEK	DAY/ MONTH	DATE/ TIME	INTERVAL	COUNT
Every day at 14:00:00	Daily	14:00:00				0	1

- Weekly type schedule:

Every week, the first occurrence is given by the day of the week entered in the “Day of week” field and the time entered in the “Time” field.



The “Day of week” is between Monday and Sunday

The Time format is the following: HH:MM:SS (for example 09:30:00)

The “Count” value is between 1 and 2,147,483,647 included

The “Interval” value is between 0 and 2,147,483,647 included



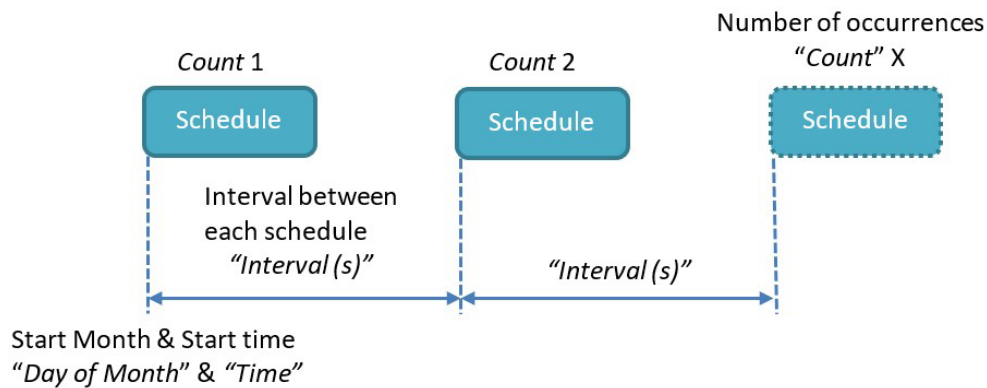
“Count”: if the schedule is to be triggered throughout the week at regular intervals, you can enter the maximum value (namely 2,147,483,647) in “Count”.

Example:

NEED	TYPE	TIME	DAY/ WEEK	DAY/ MONTH	DATE/ TIME	INTERVAL	COUNT
Every Tuesday at 15:00:00	Weekly	15:00:00	Tuesday			0	1
Every hour between 8:00 and 18:00 every Tuesday	Weekly	08:00:00	Tuesday			3600	11

- Monthly type schedule:

Every month, the first occurrence is given by the day of the month entered in the “Day of month” field and the time entered in the “Time” field.



The “Day of month” format is between 1 and 31 included

The Time format is the following: HH:MM:SS (for example 09:30:00)

The “Count” value is between 1 and 2,147,483,647 included

The “Interval” value is between 0 and 2,147,483,647 included



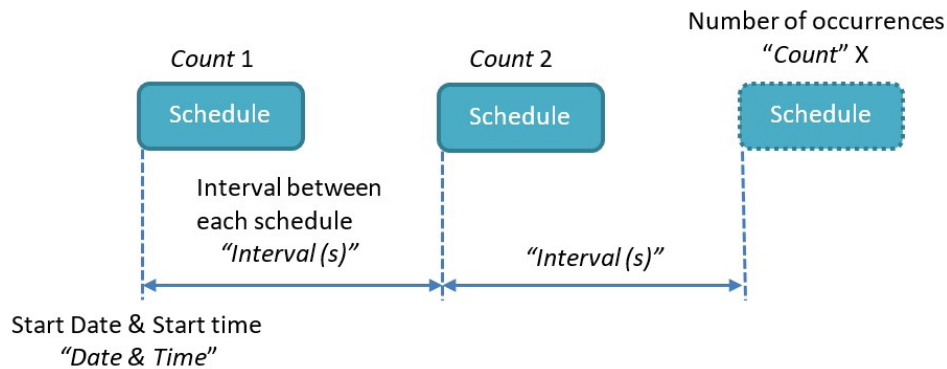
“Count”: if the schedule is to be triggered throughout the month at regular intervals, you can enter the maximum value (namely 2,147,483,647) in “Count”.

Example:

NEED	TYPE	TIME	DAY/ WEEK	DAY/ MONTH	DATE/ TIME	INTERVAL	COUNT
Every 2nd day of the month at 00:00:00	Monthly	00:00:00		2		0	1

- Yearly type schedule:

Every year, the first occurrence is given by the date entered in the “Date & Time” field.



The “Date & Time” format is the following: YEAR-MM-DDTHH:MM:SS (for example, for a first occurrence on 11 February at 13:00: Time = 2019-02-11T13:00:00).

The “Count” value is between 1 and 2,147,483,647 included

The “Interval” value is between 0 and 2,147,483,647 included



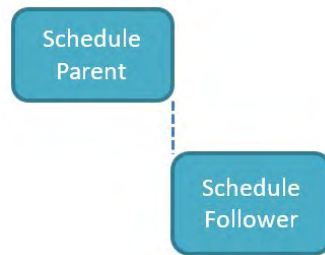
“Count”: if the schedule is to be triggered throughout the year at regular intervals, you can enter the maximum value (namely 2,147,483,647) in “Count”.

Example:

NEED	TYPE	TIME	DAY/ WEEK	DAY/ MONTH	DATE/ TIME	INTERVAL	COUNT
Every 2 hours between 8:00 and 20:00 on 31 December	Yearly				2019-12-31 T08:00:00	7200	7

- Follower type schedule:

A “Follower” type schedule will be triggered after the end of every reference schedule occurrence. The “Parent” schedule cannot be of the “Follower” type.



This type is used to trigger a data upload to the remote server after the scheduled data collection is complete, for example.

Example:

You want to collect all the Modbus module data every day at midnight and upload the data immediately afterwards. You can configure a “Daily” type schedule for the data collection and another “Follower” type schedule linked to the first schedule to upload the data.

## 4.1.7 Modbus

The WebdynEasy LoRaWAN hub is exclusively a Modbus RTU and TCP Master.



If Modbus RTU slaves are used, the Modbus protocol must be enabled on the RS485/RS422 port (see section 4.1.3: “System”)

On the local web interface “Modbus” tab, you can configure the maximum response time for the Modbus RTU and TCP protocols.

Settings	
RTU	
Timeout (ms):	2000
Turnaround (ms):	100
TCP	
Timeout (ms):	2000

PARAMETERS	DESCRIPTION
<b>RTU</b>	
Timeout (ms)	Modbus RTU response waiting time in ms
Turnaround (ms)	RTU turnaround time in ms
<b>TCP</b>	
Timeout (ms)	Modbus TCP response waiting time in ms

A Modbus slave is defined by a label, a dataset, a Modbus address and a schedule. For a Modbus TCP Modbus slave, an IP address is needed.

PARAMETERS	DESCRIPTION
Label	Purely descriptive name.
Dataset	Associated dataset identifier (Declared dataset list).
Address	Modbus address (from 1 to 247).
IP	IP Address (empty for RTU devices).
Schedule	Schedule identifier (Declared schedule list) .



A dataset defines the variables available on a Modbus slave and how to retrieve them. Dataset configuration:

Modbus Dataset

Id:

Label:

Polling:

☐

Variables

Name

Type

Address

Size

Format

Flags

Threshold low

Threshold high

Threshold hysteresis

INVALID ()

INVALID ()

Cancel

Apply

PARAMETERS	DESCRIPTION
Id	Unique Modbus data set identifier (integer).
Label	Data set name (for information).
Polling	Continuous Modbus slave polling.

Variable configuration, each variable being defined by the following parameters:

PARAMETERS	DESCRIPTION
Name	Variable name (for information only).
Type	Variable type: <ul style="list-style-type: none"><li>Coil (0x1/0x5,0xF)</li><li>Discrete input(0x2)</li><li>Holding register (0x3/0x6,0x10)</li><li>Input register (0x4)</li></ul>
Address	Extended 16-bit register address.
Size	Size in bits for Discrete inputs and Coils, and in bytes for Input and Holding registers.

Format	Variable format: <ul style="list-style-type: none"> <li>• Raw (raw data)</li> <li>• Boolean: 0 or 1)</li> <li>• Integer (whole number)</li> <li>• Float (number with a decimal point)</li> <li>• ASCII (text)</li> </ul>
Flags	List of options to apply: (optional) <ul style="list-style-type: none"> <li>• cmd_only</li> <li>• little_endian</li> <li>• no_opt</li> <li>• signed</li> <li>• is_status</li> <li>• is_alarm</li> </ul>
Threshold low	Low threshold value (optional).
Threshold high	High threshold value (optional).
Threshold hysteresis	Hysteresis applied to both thresholds (optional).

The “Polling” variable is used to enable continuous Modbus slave polling. When disabled, the Modbus slave is only polled when the associated schedule is triggered.

The variable type defines the function code to read or write the variable. See table below:

TYPE	DESCRIPTION	READ (multiple)	WRITE (single)	WRITE (multiple)
S0	Coil	0x01	0x05	0x0F
S1	Discrete input	0x02	-	-
S3	Input register	0x04	-	-
S4	Holding register	0x05	0x06	0x10

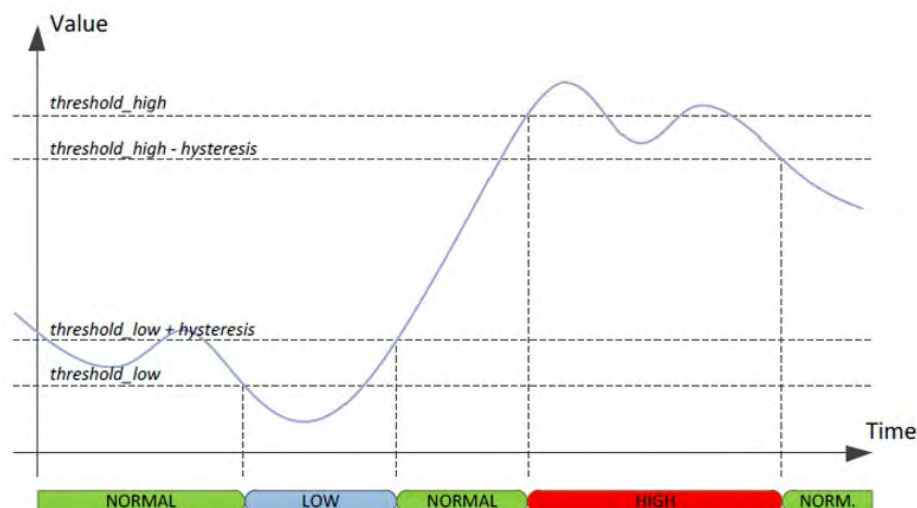
The available formats are the following:

FORMAT	DESCRIPTION	COIL	REGISTER
raw	Data represented as: <ul style="list-style-type: none"> <li>binary string for “Discrete input” and “Coils”</li> <li>hexadecimal string for “registers”</li> </ul>	X	X
boolean	True or False	X	
integer	8, 16 or 32 bit whole number		X
float	16 or 32 bit floating point number (IEEE 754)		X
ascii	ASCII character string		X

The “Flag “ field can be supplemented by one or more options. For multiple options, the options must be separated by a comma “,”. Below is a list of the available options:

PARAMETERS	DESCRIPTION
cmd_only	The variable will not be read by the Modbus device, but can be written.
little_endian	Interprets registers in little-endian.
no_opt	A Modbus request will be used to read this variable.
signed	The variable contains a signed value.
is_status	Indicates that the variable contains an information status.
is_alarm	All changes to the status variable will trigger an alarm.

When the “is\_status” option is defined, or at least one threshold is defined, the variable is considered to be a status variable. This means that if the status changes, the variable value is saved in the data file. Below is a diagram describing the status changes depending on the thresholds and hysteresis.



If the variable is a status variable and the “is\_alarm” option is present, an alarm file is generated each time the status changes. The “is\_alarm” option has no effect if the variable is not a status variable (“is\_status” option).



When the monitoring parameters Threshold low, Threshold high or Threshold hysteresis are used, the Polling mode must be enabled to monitor the variable continuously.

## 4.1.8 Run Actions

The web interface “Actions” tab is used to run certain actions locally.

### 4.1.8.1 Remote server connection request. Request

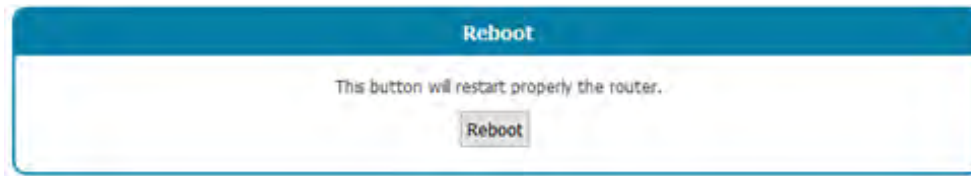


The “Request” button has the same effect as the physical button on the front of the product.

When this button is pressed, a pop-up window appears displaying all the connection steps, in particular NTP synchronisation, the INBOX directory check, and indicates all the uploaded files.

#### 4.1.8.2 Reboot request: Reboot

This button is used to reboot the hub.

A rectangular button with a blue header bar containing the text "Reboot". Below the header, the text "This button will restart properly the router." is displayed. At the bottom center, there is a smaller button labeled "Reboot".

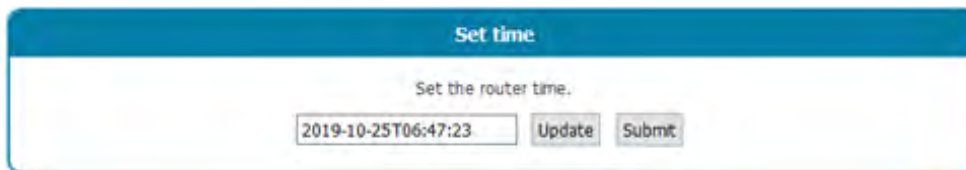
#### 4.1.8.3 Log download: Download logs

This button is used to download the logs for the last actions run on the hub.

A rectangular button with a blue header bar containing the text "Download logs". Below the header, the text "Download Gateway logs: [trace.log](#)" is displayed.

#### 4.1.8.4 Manual time set: Set time

This form is used to update the hub time if an internet connection is unavailable or if an NTP server has not been entered.

A rectangular form with a blue header bar containing the text "Set time". Below the header, the text "Set the router time." is displayed. Underneath, there is a text input field containing "2019-10-25T06:47:23", followed by two buttons labeled "Update" and "Submit".

By clicking the “Update” button, the computer date and time are copied into the form in the correct format.

If you want to enter the date and time manually, the format must be the following: YEAR-MM-DDThh:mm:ss

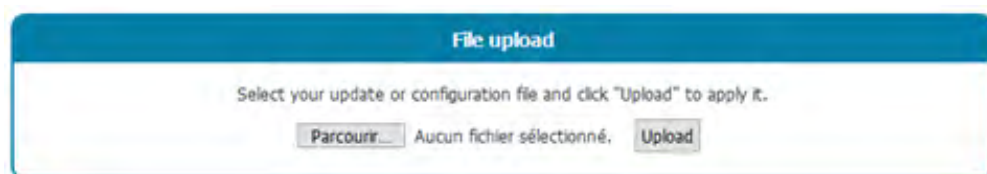
Where:

- YEAR: 4 digit year.
- MM: 2 digit month in the year.
- DD: 2 digit day in the month.
- hh: 2 digit hour.
- mm: 2 digit minutes.
- ss: 2 digit seconds.

The new date is only applied after the form has been validated by pressing the “Submit” button.

#### 4.1.8.5 System file upload: File upload

This form is used to locally upload a file on the hub.



The screenshot shows a web form titled "File upload" with a blue header. Below the header, there is a text instruction: "Select your update or configuration file and click 'Upload' to apply it." Underneath this text is a file selection area containing a "Parcourir..." button, the text "Aucun fichier sélectionné," and an "Upload" button.

Only configuration files and updates are accepted using this form.

## 5. Operation

### 5.1 The remote server

The hub communicates with a remote server using the FTP protocol. This server is used to manage the hub remotely.

The remote server has several roles:

- Report data and alarms collected locally by the hub: each time a connection is made to the server, whether by manual request, the triggering of an alarm or the triggering of the Connection Schedule, the hub takes advantage of the connection to the server to upload its stored data.
- Save a copy of the configuration: a backup of the hub configuration is available in the server “CONFIG/” directory. Each time the hub configuration is changed (locally or remotely), the hub sends a copy of its configuration to this directory.
- Reconfigure the hub or trigger actions on it: the configuration or command files must be uploaded to the server in an INBOX directory associated with the hub.
- Monitor the hub and assist in diagnosis: the hub can upload hub status files and logs for diagnostic purposes.

#### 5.1.1 The FTP Server

##### 5.1.1.1 Configuration

The FTP server is defined by the following parameters:

- An address: This can be an IP address or a domain name. If a domain name is used with an Ethernet connection, a DNS server must be configured in the hub so that the domain name can be translated to an IP address.
- The FTP connection port (default 21) can be changed by adding the port to be used after the ‘:’ character to the end of the address. The format to be used is as follows: “address:port” (e.g. “192.168.1.2:8021”).
- A login and a password: The parameters are used to define the FTP account to be used.
- A root directory: The root directory can be the FTP server root “/” or a series of subdirectories (for example “WebdynEasy\_LoRaWAN/00C8B5/”).

Below the root directory, the FTP server must have the following directories:

NAME	RIGHTS	DESCRIPTION
CONFIG/	Write	Contains the configuration image. The configuration is saved in a file called: "<uid>.xml"
DATA/	Write	Contains collected data. The name of the data file respects the following format: "<uid>-<timestamp>.xml.gz" or "<uid>-<timestamp>.json.gz"
ALARM/	Write	Contains the alarms. The alarm file name is in the following format: "<uid>-<timestamp>.xml.gz"
SUPERVISION/	Write	Contains the status files and the logs. The file names are in the following format: "<uid>-<timestamp>.json.gz"
INBOX/<uid>/	Read/Write	Mailbox to send a configuration or a command to the hub.
BIN/	Read	Contains the update files.

Where:

- <uid>: Hub identifier
- <timestamp>: The timestamp format is "YEARMMDD-HHMMSS" so that an alphabetical sort of the directory gives the chronological order

The data, alarm and supervision files are compressed in Gzip ".gz" format.

The minimum access rights to the different directories must be defined as specified in the table above.



The hub will not create the directories if they are missing. If the directories are missing, or if the rights are insufficient, please contact the server administrator.

#### 5.1.1.20operation

The hub always uploads files to the FTP server using a 2-step process:

- At the start of the transfer the file has an additional ".tmp" extension.
- When the file transfer is complete, it is renamed by removing the ".tmp" extension.

This process allows the remote server to easily differentiate between files being uploaded and files that are completely uploaded.





The files exchanged with the remote server comply with the formats described in the schema files (XSD files). Each firmware version is delivered with its associated schema files and is available on our web site (see section 7: “Support”)



The XML schemas specifying the different XML file formats used by the hub may change in future versions when new functions are added. These changes will be made in such a way that the previous XML files remain compatible with the new XML schemas. Similarly, as the XML files generated by the hub may contain additional elements, their processing must be implemented so that the new elements are ignored.

### 5.1.1.3 File Format

The data, alarm, command and configuration files exchanged with the server are in XML format.

## 5.1.2 Web Service

### 5.1.2.1 Settings

The Web Service is defined by the following parameters:

- A URL: the URL can be an IP address or the name of the remote web server. When using a domain name with an Ethernet connection, a DNS server must be configured in the concentrator to allow resolution of the domain name into an IP address.
- It is possible to modify the port of the web server (by default 80) by adding at the end of the URL, the port to use after the ‘:’ character. The format to use is: “url:port” (for example: “192.168.1.2:5000”).
- An identifier and a password: these parameters are used to define the Web Service account to be used.
- A path: the path on the web server.

The web server must contain the following subpaths:

NAME	RIGHTS	DESCRIPTION
CONFIG/	Write	Contains the configuration image. The configuration is saved in a file called: “<uid>.xml”
DATA/	Write	Contains collected data. The name of the data file respects the following format: “<uid>-<timestamp>.xml.gz” or “<uid>-<timestamp>.json.gz”
ALARM/	Write	Contains the alarms. The alarm file name is in the following format: “<uid>-<timestamp>.xml.gz”

SUPERVISION/	Write	Contains the status files and the logs. The file names are in the following format: “<uid>-<timestamp>.json.gz”
INBOX/<uid>/	Read/Write	Mailbox to send a configuration or a command to the hub.
BIN/	Read	Contains the update files.

With:

- <uid>: Concentrator ID
- <timestamp>: The timestamp format is “YYYYMMDD-HHMMSS” so an alphabetical sorting of the directory gives the chronological order

Data, alarm and supervision files are compressed in Gzip “.gz” format.

The minimum access rights to the different access paths must be defined as specified in the table above.

### 5.1.2.2 Functioning

The hub uploads the files to the web server using an HTTP POST request in the following format:

- CONFIG : http://<ws\_address>/<ws\_upolad\_path>/config
- DATA : http://<ws\_address>/<ws\_upolad\_path>/data
- ALARM : http://<ws\_address>/<ws\_upolad\_path>/alarm
- SUPERVISION : http://<ws\_address>/<ws\_upolad\_path>/supervision
- INBOX : http://<ws\_address>/<ws\_upolad\_path>/inbox ?uid=<uid>

The hub retrieves files from the web server using an HTTP GET request in the following format:

- INBOX : http://<ws\_address>/<ws\_upload\_path>/inbox/<update\_file>?uid=<uid>
- BIN : http://<ws\_address>/<ws\_upload\_path>/bin/<update\_file>?uid=<uid>



The files exchanged with the remote server comply with the formats described by the schema files (XSD files). Each firmware version is delivered with its associated schematic files and available on our website (see chapter 7: “Support”).



The XML schemas specifying the format of the various XML files used by the hub may change in future releases when new features are added. These changes will be made so that the old XML files remain compatible with the new XML schemas. Also, since the XML files generated by the hub may contain additional elements, their processing must be implemented so that the new elements are ignored.

### 5.1.2.3 File format

The alarm, command and configuration files exchanged with the server are in XML format. The data files are either in XML format or in JSON format.

## 5.1.3 MQTT

### 5.1.3.1 Settings

- The MQTT Server is defined by the following parameters:
- An address: This address can be an IP address or the name of the remote web server. When using a domain name with an Ethernet connection, a DNS server must be configured in the concentrator to allow resolution of the domain name into an IP address.
- It is possible to modify the port of the MQTT server (by default 1883) by adding at the end of the address, the port to use after the ':' character. The format to use is: url:port (for example: "192.168.1.2:5000").
- An identifier and a password: These parameters are used to define the MQTT server account to use.
- The topic: The topic of the MQTT messages to use

With :

<uid>: Concentrator ID

### 5.1.3.2 Operation

The hub sends the data to the MQTT server in the format specified in the upload section. There is no configuration management in MQTT. There is no command in MQTT.

### 5.1.3.3 Data Format

Alarms, data and supervision are either in XML format or in JSON format.

Regarding the XML format:



The data exchanged with the remote server respects the formats described by the schema files (XSD files). Each firmware version is delivered with its associated schematic files and available on our website (see chapter 7: "Support")



The XML schemas specifying the format of the various XML files used by the hub may change in future versions when new features are added. The XML files generated by the concentrator may contain additional elements, their processing must be implemented so that the new elements are ignored.

## 5.2 The Configuration

The hub allows remote configurations using a configuration file or text messages

Configuration file:

The WebdynEasy LoRaWAN hub configuration file is in XML format. Please refer to the configuration XSD file for your firmware version to get the details of the configuration file formats.

The appendix to this manual (Appendix A - Variable list) contains the list of variables and their meaning.

A backup of the current configuration is available on the remote server in the “/CONFIG” directory. Whether after a local or remote modification of the configuration, the hub sends its new configuration to the remote server.

A configuration file can be sent locally via the web interface, or remotely via the FTP “INBOX” directory.

- Locally: on the Actions” tab, select the required configuration file using the “File Upload” form, then validate the selection by clicking the “Upload” button. The file will be sent to the hub and applied.

- Remotely: upload the configuration file to the FTP “INBOX” directory on your hub (“INBOX/<uid>/”, where <uid> is your hub identifier). On the next connection to the FTP server, the hub will carry out 3 steps:
  - Download the configuration file available on the server.
  - Delete the server configuration file.
  - Apply the new configuration.



A pre-defined name for the configuration file is not needed.

If there is an error in the configuration file (corrupt file, incorrect value, ...), the file will not be applied and an alarm will be generated on the server. Check the coherence of your configuration file with the XSD file for your firmware version before sending it to your hub.

There is no need to send the entire configuration back to your hub. A configuration file can be complete or partial. A configuration file containing only one variable can therefore be sent.

By default, the configuration sent to the hub overwrites the current configuration. Only the variables in the configuration file will be overwritten. However, the default values can be applied to all the variables before the new values are applied. To do that, in the main “config” tag, add the “factory=true” attribute:

```
<config
  xmlns="http://www.webdyn.com/GWL_config_20190719"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.webdyn.com/GWL_config_20190719 config.
  xsd"
  factory="true">
    <uid>00C8B4</uid>
    <name>WG_00C8B4</name>
    <enable_local_config>true</enable_local_config>
    <com>
      <modem>
        <pin>
          ...
        </pin>
      </com>
    </uid>
  </config>
```



Refer to “Appendix A – Variable list” to see the list of variables and their possible values.

## 5.3 The Data

The data is uploaded to the “DATA/” directory of the remote server, in the form of files either in XML format or in JSON format, and compressed in Gzip “.gz” format.

Below is the format of the data file names: <uid>-<timestamp>.xml.gz or <uid>-<timestamp>.json.gz.

With :

- <uid>: Concentrator ID
- <timestamp>: The format of the timestamp is “YYYYMMDD-HHMMSS” so that an alphabetical sorting of the directory gives the chronological order

Example :

00C8B4-20191029-112704.xml.gz or 00C8B4-20191029-112704.json.gz

The format of the data files is described by the data XSD file. XSD files can evolve according to firmware

versions. They are delivered with each update.

The frequency of sending files to the remote server can be defined by a Schedule. (see chapter 4.1.6: “Schedules” and chapter 4.1.1.6: “Upload”).

However, during a connection to the server, following a manual request or the triggering of an alarm, the concentrator takes advantage of the connection to deposit the data in memory.

## 5.4 Alarms

The alarms are uploaded in XML format files compressed to Gzip “.gz” format. They are uploaded to the “ALARM/” directory of the remote server.

The alarm file name format is identical to the data file format. Below is the Alarm file name format: <uid>-<timestamp>.xml.gz

Where:

- <uid>: Hub identifier
- <timestamp>: The timestamp format is “YEARMMDD-HHMMSS” so that an alphabetical sort of the directory gives the chronological order

Example: 00C8B4-20191029-090507.xml.gz

The alarm file format is described by the alarm XSD file. XSD files may change depending on firmware versions. They are shipped with every update.

Alarms can be configured to be uploaded immediately they are triggered (On), at the next connection (Delayed) or disabled (Off). (see section 4.1.5: “Alarms”)

## 5.5 Commands

Actions can be run on the hub remotely. To do this, the hub must be send a command. This command can be sent using an XML format command file, or by text message.

- XML command file: the command file must be uploaded to the remote server “INBOX” directory for the hub (“INBOX/<uid>/”, where <uid> is the hub identifier). In the same way as the configuration files. All the files in this directory will be downloaded before being deleted and run.

The command file format is described by the command XSD file. XSD files may change depending on firmware versions. They are shipped with every update.

- Test message: the text message format must be the following:

```
cmd=command  
param1=value1  
param2=value2  
...  
parami=valuei
```

or

```
cmd=command;param1=value1;param2=value;...;parami=valuei
```

Where:

- command: the command to be sent
- param1, param2, ..., parami: command parameters
- value1, value2, ..., valuei: parameter values



- command: command to send
- param1, param2, ..., parami: command parameters
- value1, value2, ..., valuei: parameter values

All commands accept two optional parameters “uid” and “cid”:

- uid: unique hub identifier
- cid: command identifier

Commands will be rejected if the included uid parameter does not match the hub uid.

The cid can be freely chosen by the command issuer. It will be included with any associated download.

Below is a list of the commands available on the hub:

COMMAND	SUBCOMMAND	DESCRIPTION	RETURN
reboot		Restarting the product	Any
factory		Return to factory settings	Any
update		Hub software update	Alarm

connect		Immediate connection to the remote server	Login
status		Hub Status Recovery	Monitoring+SMS
log		Logbook recovery	supervision
settime		Concentrator time setting	Alarm
modbus	write	Writing to a modbus slave	Alarm
lorawan	send	Downlink LoRaWAN frame sending	Alarm
lorawan	add	Adding a sensor	Alarm
lorawan	delete	Deleting a sensor	Alarm



If several commands are sent at the same time, the “reboot”, “factory” and “update” commands can result in the commands following them being lost.

### 5.5.1 “Reboot” Command

The “reboot” command is used to trigger an immediate product reboot. There is no return/ acknowledgement after this command is sent.

No subcommands or parameters are required for this command.

Example:

- By XML file:

```
<commands
xmlns="http://www.webdyn.com/GWL_command_20190719"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.webdyn.com/GWL_
command_20190719 command.xsd">
  <cmd uid="00C8B4">
    <reboot />
  </cmd>
</commands>
```

- By text message:

```
cmd=reboot
uid=00C8B4
```



## 5.5.2 "Factory" Command

The "factory" command is used to restore the hub factory settings. There is no return/acknowledgement after this command is sent.

No subcommands or parameters are required for this command.

Example:

- By XML file:

```
<commands
xmlns="http://www.webdyn.com/GWL_command_20190719"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.webdyn.com/GWL_
command_20190719 command.xsd">
  <cmd>
    <factory />
  </cmd>
</commands>
```

- By text message:

```
cmd=factory
```

## 5.5.3 "Update" Command

(see section 6.2: "Update Remote")

## 5.5.4 "Connect" Command

The "connect" command is used to trigger an immediate product connection to the remote server. There is no return/acknowledgement after this command is sent.

No subcommands or parameters are required for this command.

Example:

- By text message:

```
cmd=connect
```

## 5.5.5 "Status" Command

The "status" command is used to retrieve product status information. When the request is made using a file, a status file is uploaded to the remote server "SUPERVISION/" directory. When the request is made by text message, the answer is sent to the command issuer by text message.

No subcommands or parameters are required for this command.

Example:

- By XML file:

```
<commands
xmlns="http://www.webdyn.com/GWL_command_20190719"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.webdyn.com/GWL_
command_20190719 command.xsd">
  <cmd cid="status cmd 1">
    <status />
  </cmd>
</commands>
```

- By text message:

```
cmd=status
cid=status cmd 1
```

### 5.5.6 "Log" Command

The "log" command is used to retrieve the hub log. The log is uploaded to the remote server "SUPERVISION/" directory.

No subcommands or parameters are required for this command.

Example:

- By XML file:

```
<commands
xmlns="http://www.webdyn.com/GWL_command_20190719"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.webdyn.com/GWL_
command_20190719 command.xsd">
  <cmd>
    <log />
  </cmd>
</commands>
```

- By text message:

```
cmd=log
```

### 5.5.7 "Settime" Command

The "settime" command allows you to update the date and time of the concentrator with the desired time.

To do this, in the "time" attribute, you must indicate the desired date and time in the following format: YYYY-MM-DDThh:mm:ss

With :

- YYYY: 4-digit year
- MM: Month in the year on 2 digits
- DD: Day in the month on 2 digits
- hh: Hour on 2 digits
- mm: Minutes in 2 digits
- ss: 2-digit seconds



If an NTP server is configured, the concentrator date and time will automatically update when connecting to the remote server.

Example:

- By XML file:

```
<commands>
  <cmd>
    <settime>
      <time>2021-05-23T16:03:23</time>
    </settime>
  </cmd>
</commands>
```

- By text message:

```
cmd=settime
time=2021-05-23T16:03:23
```

### 5.5.8 "Modbus" Command

The "modbus" command is used to write values to the Modbus slave registers configured on the hub.

To do that, the "write" sub-command, the data to be written in the "data" attribute, the list of slaves and registers in which the value must be written must be indicated.

The command is saved in the supervision actions and sent to the remote server "SUPERVISION/" directory.

The slave addresses must be in the following format:

- Modbus RTU:

```
<modbus_address>/<register_type>@<register_address>
```

Example: 45/S3@0x0056

- Modbus TCP:

```
<device_ip>:<modbus_address>/<register_type>@<register_address>
```

Example: 192.168.0.17:223/S3@0x0F52

Example:

- By XML file:

```
<commands
xmlns="http://www.webdyn.com/GWL_command_20190719"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.webdyn.com/GWL_command_20190719 command.xsd">
  <cmd>
    <modbus subcmd="write" data="0xFF">
      <address>45/S4@0x0056</address>
      <address>192.168.0.17:223/S3@0x0F52</address>
    </modbus>
  </cmd>
</commands>
```

- By text message:

```
cmd=modbus
subcmd=write
data=0xFF
address=45/S4@0x0056
address=192.168.0.17:223/S3@0x0F52
```

## 5.5.9 “Lorawan” Command

The “lorawan” command is used to send commands to the concentrator. There are several sub-commands which are:

- “send”: sends “downlink” frames to the sensor.
- “add”: allows you to add a sensor to the concentrator.
- “delete”: allows you to delete a sensor in the concentrator.

Following the command, an alarm is generated and placed in the “ALARM/” directory specifying the result of the command.

### 5.5.9.1 “Send” subcommand

The “send” sub-command sends “downlink” frames to the sensor.

For this, it is necessary to fill in the following attributes:

- “devaddr”: the DEVADDR of the sensor which identifies the sensor (in hexadecimal format).
- “Deveui”: the DEVEUI of the sensor which identifies the sensor (in hexadecimal format).
- “fport”: the port number of the sensor to use sensor (in decimal format).
- “data”: the data to be sent in hexadecimal format.

In class A, the concentrator sends “downlink” frames just after an “uplink” frame from the sensor. The concentrator prepares the message and stores it for a maximum of 48 hours. Pass this time, an alarm will be sent to notify that the time is exceeded. An alarm is also sent to signal the sending of the frame to the sensor.

Example :

- By XML file:

```
<commands>
  <cmd cid="cmd1">
    <lorawan subcmd="send">
      <devaddr>01020304</devaddr>
      <fport>1</fport>
      <data>0AF0C4</data>
    </lorawan>
  </cmd>
</commands>
```

```

<commands>
  <cmd cid="change_send_period_to_10min"
    <lorawan subcmd="send">
      <deveui>E498ED0000000000</deveui>
      <fport>1</fport>
      <data>600401</data>
    </lorawan>
  </cmd>
</commands>

```

- By text message:

```

cmd=lorwan
subcmd=send
devaddr=01020304
fport=1
data=0AF0C4

```

Example of alarms in case of success:

```

<alarms>
  <command>
    <date>2021-01-25T15:00:00</date>
    <cid>change_send_period_to_10min</cid>
    <source>ws</source>
    <error>none</error>
    <description>commande queued</description>
  </command> <command>
    <date>2021-01-25T15:05:00</date>
    <cid>change_send_period_to_10min</cid>
    <source>ws</source>
    <error>none</error>
    <description>commande sent</description>
  </command>
</alarms>

```

Example of an alarm in the event of times exceeded:

```
<alarms>
  <command>
    <date>2021-01-27T15:00:00</date>
    <cid>change_send_period_to_10min</cid>
    <source>ws</source>
    <error>other</error>
    <description>message timeout</description>
  </command>
</alarms>
```

#### 5.5.9.2 “Add” subcommand

The “add” subcommand adds a sensor to the concentrator.

For this, it is necessary to fill in the following attributes:

- “Deveui”: the DEVEUI of the sensor which identifies the sensor (in hexadecimal format).
- “appskey”: the APPSKEY of the sensor if the sensor is in ABP mode (in hexadecimal format).
- “nwkskey”: the NWKSKEY of the sensor if the sensor is in ABP mode (in hexadecimal format).
- “appkey”: the APPKEY of the sensor if the sensor is in OTAA mode (in hexadecimal format).

Example of adding a sensor in OTAA mode:

- By XML file:

```
<commands>
  <cmd cid="add_endpoint_key">
    <lorawan subcmd="add">
      <deveui>E498ED0000000000</deveui>
      <appkey>000102030405060708090A0B0C0D0E0F</
appkey>
    </lorawan>
  </cmd>
</commands>
```

- By text message:

```
cmd=lorawan
subcmd=add
deveui=E498ED0000000000
appkey=000102030405060708090A0B0C0D0E0F
```

Example of adding a sensor in ABP mode:

- By XML file:

```
<commands>
  <cmd cid="change_send_period_to_10min">
    <lorawan subcmd="send">
      <devaddr>00000F6A</devaddr>
      <appskey>000102030405060708090A0B0C0D0E0F<
appskey>
      <nwkskey>000102030405060708090A0B0C0D0E0F</
nwkskey>
    </lorawan>
  </cmd>
</commands>
```

- By text message:

```
cmd=lorawan
subcmd=add
devaddr=00000F6A
appskey=000102030405060708090A0B0C0D0E0F
nwkskey=000102030405060708090A0B0C0D0E0F
```

### 5.5.9.3 “Delete” subcommand

The “delete” subcommand allows you to delete a sensor from the concentrator.

For this, it is necessary to fill in the following attributes:

- “devaddr”: the DEVADDR of the sensor which identifies the sensor (in hexadecimal format).
- “Deveui”: the DEVEUI of the sensor which identifies the sensor (in hexadecimal format).

Example:

- By XML file:

```
<commands>
  <cmd cid="add_endpoint_key">
    <lorawan subcmd="delete">
      <deveui>E498ED0000000000</deveui>
    </lorawan>
  </cmd>
</commands>
```



```
<commands>
  <cmd cid="add_endpoint_key">
    <lorawan subcmd="delete">
      <devaddr>00000F6A</devaddr>
    </lorawan>
  </cmd>
</commands>
```

- By text message:

```
cmd=lorawan
subcmd=delete
deveui=E498ED0000000000
```

## 6. Update

The WebdynEasy LoRaWAN hub can be updated locally or remotely. The latest firmware version ("GatewayLoRaWAN\_x.x.x.cwe") is available for download from our web site at the following address: <https://www.webdyn.com/support/lorawan/>

### 6.1 Local

To update the hub locally, use its web interface and go to the "Actions" tab and then follow the "File Upload" system file upload procedure (see section 4.1.8.5: "System file upload: File upload").

### 6.2 Remote

For a remote update, the file containing the update must be uploaded to the "BIN" directory on the remote server, and an "update" command must be sent to the hub.

The update command can be sent either in a command file or by text message. The command must include the name of the file containing the update ("firmware" field) and its associated MD5 code ("checksum" field).



It is strongly recommended to use a command file (XML).

Example:

- By XML file:

```
<commands
xmlns="http://www.webdyn.com/GWL_command_20190719"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.webdyn.com/GWL_
command_20190719 command.xsd">
<cmd>
<update>
<firmware>GatewayLoRaWAN_1.3.0.cwe</firmware>
<checksum>c1fb7d81f3d53a8b7bf94098115249d3</checksum>
</update>
</cmd>
</commands>
```

- By text message:

```
cmd=update  
firmware=GatewayLoRaWAN_1.3.0.cwe  
checksum=c1fb7d81f3d53a8b7bf94098115249d3
```

## 7. Appendix: XML configuration file variables



All the “names +tree structures” highlighted in blue are lists and can be created multiple times.

NAME+TREE STRUCTURE	DESCRIPTION	TYPE	DEFAULT VALUE	NOT USED (future use)
/uid	Hub identifier	Hexadecimal 3 bytes	3 last bytes of the MAC address	
/name	Optional product name	Text	“WG_”+ 3 last bytes of the MAC address	
/enable_local_config	Enables/Disables access to the local configuration	Boolean (true, false)	false	
/com/modem/pin/mode	Enable/Disable the PIN	List: •Off •manual	off	
/com/modem/pin/code	PIN code	Whole number from 4 to 6 digits		
/com/modem/apn	APN	Text		
/com/modem/login	APN login	Text		
/com/modem/password	APN password	Text		
/com/modem/mode	Modem connection mode	List: •ondemand •alwaysOn •alwaysOff	alwaysOff	
/com/modem/delay	Time before disconnection in seconds	Integer (min 0 max 65535)	60	
/com/ethernet/use_dhcp	Enable/Disable the DHCP client	Boolean (false, true)	false	
/com/ethernet/ip	IP address	IP format: "xxx.xxx.xxx.xxx"	192.168.1.12	
/com/ethernet/netmask	Subnet mask	IP format: "xxx.xxx.xxx.xxx"	255.255.255.0	
/com/ethernet/gateway	Local network gateway	IP format: "xxx.xxx.xxx.xxx"		
/com/ethernet/dns/server	List of DNS servers	IP format: "xxx.xxx.xxx.xxx"		
/com/ftp/address	FTP server address + port (optional). If the port is not entered, by default FTP will use 21	IP format: "xxx.xxx.xxx.xxx" or domain name: "xxxxxxxxx.xxx" + port (optional): ":xxxx"		
/com/ftp/login	FTP account login	Text		
/com/ftp/password	FTP account password	Text		
/com/ftp/mode	FTP passive or active connection mode	List: •passive •active	passive	

/com/ftp/secured	Enable/Disable secure mode (FTPS)	Boolean (true, false)	false
/com/ftp/trust_model	Secure mode operating mode	List:	verify_peer
/com/ftp/root_path	FTP server root directory	Text	
/com/ftp/ws_notification	Enable/Disable sending notifications	List: • none • put • get • both	none
/com/ws/address	Web Services server address		
/com/ws/login	Web Services server login		
/com/ws/password	Web Services server password		
/com/ws/webservice_proxy	Proxy server address (optional)		
/com/ws/trust_model	Secure mode operating mode: • Trust peer Verify peer		
/com/ws/upload_path	Web Services root directory		
/com/mqtt/address	MQTT server address + port (optional)  If the port is empty, by default the MQTT will use the 1883	IP format: "xxx.xxx.xxx.xxx"  Or  Domain name: "xxxxxxxxx.xxx"  + port (optional): ":xxxx"	
/com/mqtt/client_id	Client identifier in the MQTT Text protocol	Text	
/com/mqtt/login	MQTT Account ID	Text	
/com/mqtt/password	MQTT account password	Text	
/com/mqtt/keepalive	Time in seconds to send keepalive frame	Integer (min 0, max 65535)	60
/com/mqtt/topic	MQTT message topic starts with this chain		
/com/mqtt/trust_model	Safe Mode Operation Mode	List: • verify peer • trust peer	verify peer
/com/mqtt/ca	CA Root Certificate	PEM File Format	
/com/mqtt/cert	Local Client Signed Certificate	PEM File Format	
/com/mqtt/key	Local client private key	PEM File Format	

/com/keepalive/file	File type to send	List: • "log" • "supervision" • empty: keepalive au format "[UID]- [TIME %Y%m%d- %H%M%S]- keepalive"	•
/com/keepalive/ Schedule	Schedule login to send regular keepalive		•
/com/request/upload	Enable/Disable server connection after pressing the "REQUEST" button		•
/com/request/include_status	Send a supervision file to the server after pressing the "REQUEST" button		•
/com/request/sms_status_recipient	Phone number for the status text message recipient after the "REQUEST" button is pressed. International format phone number		•
/com/time/ntp/server	List of NTP server addresses	IP format: "xxx. xxx.xxx.xxx" or domain name: "xxxxxxxxxx.xxx"	
/com/time/timezone	Timezone in tz format	List: (see <a href="http://en.wikipedia.org/wiki/Zone.tab">http:// en.wikipedia.org/ wiki/Zone.tab</a> )	•
/com/time/alarm_threshold	Alarm trigger threshold in seconds	Integer (min 0 max 65535)	0
/com/time/min_sync_interval	Minimum time between 2 NTP synchronisations (in seconds)	Whole number (min 0 max 4,294,967,295)	86400
/com/vpn/openvpn/enable	Enables the client OpenVPN	List: • true • false	false
/com/vpn/openvpn/protocol	VPN IP protocol	List: • tcp • udp	
/com/vpn/openvpn/server/address	IP address or name of the VPN server	IP format: "xxx. xxx.xxx.xxx" or domain name: "xxxxxxxxxx.xxx"	
/com/vpn/openvpn/server/port	VPN sever port (normally 1194)	Integer (min 1 max 65535)	
/com/vpn/openvpn/server/cipher	Data packet encryption algorithm (optional)	List: (see list in OpenVPN "openvpn --show- ciphers")	
/com/vpn/openvpn/server/auth	VPN authentication	List: (see the list in OpenVPN "openvpn --show- digests")	SHA1
/com/vpn/openvpn/server/ca	CA certificate	PEM file format	
/com/vpn/openvpn/server/cert	Client certificate	PEM file format	

/com/vpn/openvpn/server/key	Client private key	PEM file format		
/com/vpn/openvpn/server/tls_auth (obsolete)	Static key used for the HMAC hashing algorithm on control packets (Obsolete, see Key variable below)	PEM file format		
/com/vpn/openvpn/server/control_channel_security/method	List of methods for control channel security	List: •none •tls-auth •tls-crypt •tls-crypt-v2	none	
/com/vpn/openvpn/server/control_channel_security/key	Key for control channel security	PEM file format		
/com/firwall	Firewall			•
/com/mdns/enable	Activation of the mDNS protocol intended to resolve the name of the hub “UID” into an IP address.	List: •true •false	true	
/upload/config/method	Configuration file management communication protocol	List: •none •ftp •ws	ftp	ws
/upload/config/omit_password	Hide the “password” tags in the XML file	Boolean (true, false)	false	
/upload/supervision/method	Communication protocol to send supervision files	List: •MQTT •none •ftp •ws	ftp	ws
/upload/alarm/method	Communication protocol to send alarm files	List: •MQTT •none •ftp •ws	ftp	
/upload/data/method	Data file management communication protocol	List: •MQTT •none •ftp •ws	ftp	
/upload/data/format	Data file format for data	List: •xml •json	xml	
/upload/data/schedule	Schedule identifier for sending data	Integer (min 1 max 65535)		
/upload/commun/size_limit	Maximum non compressed data file size in Mb in XML format. For JSON format the unit is kilobytes	Integer (min 0 max 30) for XML and 30000 for JSON	10	
/alarm/sources/modem_ip	Modem IP address change alarm configuration	List: •on •off •delayed	off	

/alarm/sources/msisdn	SIM card change alarm configuration • on: enabled and immediate send • off: disabled delayed: enabled and send at next connection	List: • on • off • delayed	off
/alarm/sources/sw_version	Software update alarm configuration (firmware or core) • on: enabled and immediate send • off: disabled delayed: enabled and send at next connection	List: • on • off • delayed	on
/alarm/sources/defaults/ignored	List of fault to ignore: • D_MODEM • D_MODEM_SIM_MISS • D_MODEM_SIM_CODE_FAIL • D_MODEM_PUK • D_MODEM_REG_DENIED	Text (fault list separated by a comma “,”)	
/alarm/sources/defaults/delayed	List of faults that cannot be sent immediately but only at the next connection: • D_MODEM • D_MODEM_SIM_MISS • D_MODEM_SIM_CODE_FAIL • D_MODEM_PUK • D_MODEM_REG_DENIED	Text (fault list separated by a comma “,”)	
/scheduler/schedules/schedule/	Schedule list		
/scheduler/schedules/schedule/id	Schedule identifier	Whole number (min 1 max 2,147,483,647)	
/scheduler/schedules/schedule/label	Schedule name	Text	
/scheduler/schedules/schedule/type	Schedule type	List: • Daily • Weekly • Monthly • Yearly • Follower	Daily
/scheduler/schedules/schedule/parent	Parent schedule identifier for a “follower” type schedule	Integer (min 1 max 65535)	
/scheduler/schedules/schedule/start/time	Trigger time for the first schedule iteration for a “daily”, “weekly” or “monthly” type schedule	Time format: “hh:mm:ss”	
/scheduler/schedules/schedule/start/datetime	Trigger date and time for the first schedule iteration for a “yearly” type schedule	Date and time format: “year-mm-ddThh:mm:ss”	
/scheduler/schedules/schedule/start/dayofweek	Trigger day in the week for the first schedule iteration for a “weekly” type schedule	List: • Monday • Tuesday • Wednesday • Thursday • Friday • Saturday • Sunday	
/scheduler/schedules/schedule/start/dayofmonth	Trigger day in the month for the first schedule iteration for a “monthly” type schedule	Integer (min 1 max 31)	



/scheduler/schedules/schedule/interval	Interval between occurrences (in seconds)	Integer (min 0 max 4,294,967,295)	
/scheduler/schedules/schedule/count	Number of occurrences	Integer (min 1 max 65535)	
/modbus/tcp/timeout	Max. time without response from Modbus/TCP slaves (in ms)	Integer (min 0 max 65535)	2000
/modbus/rtu/timeout	Max. time without response from Modbus RTU slaves (in ms)	Integer (min 0 max 65535)	2000
/modbus/rtu/turnaround	Modbus RTU turnaround time (in ms)	Integer (min 0 max 65535)	100
/modbus/datasets/dataset/	List of Modbus datasets	Integer (min 1 max 65535)	
/modbus/datasets/dataset/id	Dataset identifier	Integer (min 1 max 65535)	
/modbus/datasets/dataset/label	Dataset name	Text	
/modbus/datasets/dataset/vars/var/	List of Dataset variables		
/modbus/datasets/dataset/vars/var/name	Variable name	Text	
/modbus/datasets/dataset/vars/var/type	Variable type	List: <ul style="list-style-type: none"> <li>•S0: Coil (0x1/0x5,0xF)</li> <li>•S1: Discrete input (0x)</li> <li>•S3: Input register (0x3/0x6,0x10)</li> <li>•S4: Holding register (0x4)</li> </ul>	
/modbus/datasets/dataset/vars/var/address	1st variable register address	Hexadecimal 2 bytes	
/modbus/datasets/dataset/vars/var/size	Variable size	Unsigned integer	
/modbus/datasets/dataset/vars/var/format	Variable format	List: <ul style="list-style-type: none"> <li>•raw</li> <li>•boolean</li> <li>•integer</li> <li>•float</li> <li>•ascii</li> </ul>	
/modbus/datasets/dataset/vars/var/flags	Variable options (optional)	List: <ul style="list-style-type: none"> <li>•cmd_only</li> <li>•little_endian</li> <li>•no_opt</li> <li>•signed</li> <li>•is_status</li> <li>•is_alarm</li> </ul>	
/modbus/datasets/dataset/vars/var/threshold/low	Low threshold (optional)	Number (double)	
/modbus/datasets/dataset/vars/var/threshold/high	High threshold (optional)	Number (double)	
/modbus/datasets/dataset/vars/var/threshold/hysteresis	Hysteresis (optional)	Number (double)	

/modbus/datasets/dataset/boundaries			•
/modbus/datasets/dataset/polling	Enable continuous polling	Boolean (true, false)	false
/modbus/modules/module/	List of Modbus modules		
/modbus/modules/module/label	Modbus slave name	Text	
/modbus/modules/module/dataset	Identifier of the dataset to use	Integer (min 1 max 65535)	
/modbus/modules/module/address	Modbus slave Modbus address	Integer (min 1 max 247)	
/modbus/modules/module/ip	Modbus/TCP slave IP address	IP format: "xxx.xxx.xxx.xxx" or domain name: "xxxxxxxxxxx.xxx"	
/modbus/modules/module/schedule	Modbus slave collection schedule identifier	Integer (min 1 max 65535)	
/system/log/level	Log trace level. Only for debug (contact support)	Level 1 (high) to 5 (low)	5
/system/password/admin	Administrator password	Text	high
/system/password/install	Installer password	Text	medium
/system/password/data	User password	Text	low
/system/ports/rs485/mode	RS485 port configuration	List: •Off •Modbus	Off
/system/ports/rs485/baudrate	RS485 port speed (in bauds)	List: •4800 •9600 •19200 •38400 •57600 •115200	19200
/system/ports/rs485/data	RS485 port data bit number	List: •5 •6 •7 •8 •9	8
/system/ports/rs485/parity	RS485 port parity	List: •None •Odd •Even	Even
/system/ports/rs485/stop_bit	RS485 port stop bit number	List: •1 •2	1
/system/upload/direct_mode	Forces the deposit of data on the remote server after receiving data from a sensor.	List: •0: disabled •1: enabled	0
/lorawan/region	Region name for LoRaWAN parameters	List: •EU868 •IN865	EU868

/lorawan/channels/channel	Channel 4 frequency (in Hz)	Whole number (min 863,000,000 max 870,000,0000)	867100000
/lorawan/channels/channel	Channel 5 frequency (in Hz)	Whole number (min 863,000,000 max 870,000,0000)	867300000
/lorawan/channels/channel	Channel 6 frequency (in Hz)	Whole number (min 863,000,000 max 870,000,0000)	867500000
/lorawan/channels/channel	Channel 7 frequency (in Hz)	Whole number (min 863,000,000 max 870,000,0000)	867700000
/lorawan/channels/channel	Channel 8 frequency (in Hz)	Whole number (min 863,000,000 max 870,000,0000)	867900000
/lorawan/packet_forwarder/server/address	LoRaWAN server address (embedded server: 127.0.0.1)	IP format: "xxx.xxx.xxx.xxx" or domain name: "xxxxxxxxx.xxx"	127.0.0.1
/lorawan/packet_forwarder/server/port_up	Packet Forwarder outgoing UDP port number	Integer (min 1 max 65535)	1700
/lorawan/packet_forwarder/server/port_down	Packet Forwarder incoming UDP port number	Integer (min 1 max 65535)	1700
/lorawan/packet_forwarder/keepalive_interval_s	Time in seconds to send a keepalive frame	Integer (min 0 max 65535)	10
/lorawan/packet_forwarder/push_timeout_ms	Maximum waiting time in milliseconds to acknowledge the frame sent to the LoRaWAN server.	Integer (min 0 max 65535)	10
/lorawan/packet_forwarder/forwarder_crc_valid	LoRaWAN packet processing with a valid CRC (do not modify, only used for testing)	Boolean (true, false)	true
/lorawan/packet_forwarder/forwarder_crc_error	LoRaWAN packet processing with error CRC (do not modify, only used for testing)	Boolean (true, false)	false
/lorawan/packet_forwarder/forwarder_crc_none	LoRaWAN packet processing without CRC (do not modify, only used for testing)	Boolean (true, false)	false
/lorawan/packet_forwarder/public	Type of LoRaWAN public network preamble (public: 0x34, private: 0x12) (do not modify, only used for testing)	Boolean (true, false)	true
/lorawan/server/netid	LoRaWAN network identifier (see LoRaWAN specification) If you enter 0, when the hub reboots, it will use its factory NetID.	Hexadecimal 3 bytes	Calculated automatically from its MAC address
/lorawan/server/adr/enable	Enable ADR	Boolean (true, false)	true

/lorawan/server/adr/margin_db	Margin in dB to calculate ADR	Integer (min 1 max 30)	5
/lorawan/server/adr/uplink_count	The number of uplinks needed for ADR	Integer (min 1 max 65535)	20
/lorawan/server/udp_port	LoRaWAN server UDP port	Integer (min 1 max 65535)	1700
/lorawan/server/backup_interval	Automatic configuration backup interval with the Fcntup and Fcntdown counters. (in seconds)	Whole number (min 1 max 4,294,967,295)	86400
/lorawan/server/modules/module/	List of LoRaWAN modules		
/lorawan/server/modules/module/deveui	Unique sensor identifier (EUI64)	Hexadecimal 8 bytes	
/lorawan/server/modules/module/appkey	Encryption key used by the network to derive the session keys.	Hexadecimal 16 bytes	
/lorawan/server/modules/module/devaddr	Sensor address	Hexadecimal 4 bytes	
/lorawan/server/modules/module/appskey	Encryption key between the sensor and the application server	Hexadecimal 16 bytes	
/lorawan/server/modules/module/nwkskey	Encryption key between the sensor and the LoRaWAN server	Hexadecimal 16 bytes	
/lorawan/server/modules/module/fcntup	Uplink frame counter (to the server)	Whole number (min 0 max 4,294,967,295)	
/lorawan/server/modules/module/fcntdown	Downlink frame counter (to the sensor)	Whole number (min 0 max 4,294,967,295)	
/lorawan/server/modules/module/lclass	LoRaWAN classe du capteur	a or c	a

# Offices & Support Contact

## SPAIN

C/ Alejandro Sánchez 109  
28019 Madrid  
Phone: +34.915602737  
Email: [contact@webdyn.com](mailto:contact@webdyn.com)

## FRANCE

26 Rue des Gaudines  
78100 Saint-Germain-en-Laye  
Phone: +33.139042940  
Email: [contact@webdyn.com](mailto:contact@webdyn.com)

## INDIA

803-804 8th floor, Vishwadeep Building  
District Centre, Janakpurt, 110058 Delhi  
Phone: +91.1141519011  
Email: [contact@webdyn.com](mailto:contact@webdyn.com)

## PORTUGAL

Av. Coronel Eduardo Galhardo 7-1°C  
1170-105 Lisbon  
Phone: +351.218162625  
Email: [comercial@lusomatrix.pt](mailto:comercial@lusomatrix.pt)

## TAIWAN

5F, No. 4, Sec. 3 Yanping N. Rd.  
Datong Dist. Taipei City, 103027  
Phone: +886.965333367  
Email: [contact@webdyn.com](mailto:contact@webdyn.com)

## SUPPORT

### Madrid Offices

Phone: +34.915602737  
Email: [iotsupport@mtxm2m.com](mailto:iotsupport@mtxm2m.com)

### Saint-Germain-en-Laye Offices

Phone: +33.139042940  
Email: [support@webdyn.com](mailto:support@webdyn.com)

### Delhi Offices

Phone: +91.1141519011  
Email: [support-india@webdyn.com](mailto:support-india@webdyn.com)

### Taipei City Offices

Phone: +886.905655535  
Email: [iotsupport@mtxm2m.com](mailto:iotsupport@mtxm2m.com)