

# PowerPDU 4C

&

# NETIO 4

# NETIO 4All

# MANUAL

FIRMWARE 3.4.0

and later

6.5.2020

# Table of Contents

Introduction.....	4
1 Safety notices .....	4
2 General characteristics .....	5
3 Specification.....	6
3.1 Features .....	6
3.2 NETIO PowerPDU 4C specifications .....	7
3.3 Drawings .....	8
3.4 NETIO 4 / 4All specifications.....	9
4 Configuration and control .....	10
4.1 Detecting and configuring the IP address.....	10
4.1.1 Initial configuration over a LAN.....	10
4.1.2 Initial configuration over Wi-Fi - only NETIO 4 and NETIO 4All .....	12
4.2 Login to device web .....	13
4.3 Restoring factory defaults.....	13
5 Web interface .....	14
5.1 Outputs .....	14
5.1.1 General.....	15
5.1.2 Scheduler/Timer.....	16
5.1.3 Watchdog function .....	17
5.1.4 Consumption metering - only NETIO PowerPDU 4C and NETIO 4All .....	20
5.2 M2M API Protocols.....	22
5.2.1 SNMP.....	23
5.2.2 TELNET/KSHELL.....	27
5.2.3 MQTT .....	29
5.2.4 JSON API .....	33
5.2.5 XML API.....	36
5.2.6 URL API.....	38
5.2.7 Modbus/TCP .....	39
5.3 Cloud .....	41
5.4 Users .....	44
5.5 Schedules.....	46
5.6 Actions (Lua scripts).....	48
5.7 Settings .....	52
5.7.1 Network mode .....	52
5.7.2 Wi-Fi settings.....	58
5.7.3 Network configuration.....	60
5.7.4 Security settings.....	62
5.7.5 Date / Time .....	64
5.7.6 E-mail.....	65
5.7.7 Firmware update .....	66
5.7.8 System.....	68
5.8 Log.....	70
6 Controlling the outputs manually.....	71
7 LED status indicators .....	71
8 Acoustic indication .....	72
9 NETIO Mobile2 for Android .....	72
10 Troubleshooting.....	73
10.1 Forgotten password, restoring factory defaults .....	73
10.2 Firmware upgrade problems .....	73
10.3 Fuse reset - only NETIO 4 and NETIO 4All .....	73
11 Application Notes (ANxx) .....	74
Conclusion.....	75
Declaration of conformity.....	76
NETIO products overview.....	78



## Introduction

Thank you for purchasing this product of NETIO products a.s. Before using your product, please read this User Manual (MAN) and the included Quick Installation Guide (QIG) to avoid problems with incorrect installation or use.

### Caution:

*The product works with mains voltage. Mishandling may damage it or result in injury or death.*

## 1 Safety notices

1. The manufacturer is not liable for any damage caused by incorrect use of the device or by operating it in an unsuitable environment.
2. The device is not rated for outdoor use.
3. Do not expose the device to strong vibrations.
4. Unauthorized modifications may damage the device or cause a fire.
5. Protect the device from liquids and excessive temperatures.
6. Make sure the device does not fall.
7. Only electrical appliances approved for use in the electrical network may be connected to the device.
8. Do not connect multiple devices in series.
9. The cable plug must be easily accessible.
10. The device is completely switched off only when unplugged from the wall socket.
11. If the device malfunctions, disconnect it from the electrical power supply and contact your vendor.

## 2 General characteristics

NETIO PowerPDU 4C, NETIO 4 and NETIO 4All (NETIO 4x) are the multifunctional Power Distribution Units (PDU). With LAN/Wi-Fi network interfaces, the output power ports for connected appliances can be controlled over the Web interface, with our mobile app, using Lua scripts or over one of the available open M2M API protocols.

To control the device over the Web, all you need is a web browser; NETIO 4x includes a built-in Web server.

NETIO 4x is a designation covering several product versions with different electrical socket types. In this manual, screenshots of the web user interface show only one socket type; the sockets of your product may be different from the pictures herein.

This manual covers the following models:

**NETIO PowerPDU 4C** IEC 320 C13/C14 el. connection and metering for each socket. No Wi-Fi.

**NETIO 4** Basic model with integrated Wi-Fi and electrical sockets.

**NETIO 4All** NETIO 4 with electricity consumption metering for each socket.



NETIO PowerPDU 4C



NETIO 4



NETIO 4All

## 3 Specification

### 3.1 Features

- Four controlled 110/230V power outputs
  - Each power output has a status LED and a control button
  - Each output has an integrated surge protector
  - After restarting or powering up the device, each output is set to its pre-configured state (ON / OFF / LAST - the last state before the device was powered off)
  - Wi-Fi interface: IEEE 802.11abgn @ 2.4GHz (secured / unsecured) - only NETIO 4 / 4All
  - LAN interface: 100 Mbps (NETIO PowerPDU 4C includes an integrated 2-port switch)
  - Robust design, metal housing
  - 1.2m lead cable + power switch at the device (external cable for NETIO PowerPDU 4C)
  - Built-in web server for controlling the device, HTTPs support
  - Encrypted login into device configuration
  - User accounts with configurable access rights
  - User interface localized into EN, DE, ES, IT, CZ
- 
- Mobile app for controlling outputs is available for iOS and Android devices
  - Watchdog function for monitoring the state of connected network equipment and restarting it if needed
  - Scheduler function controls the outputs according to a user-defined time schedule
  - Lua scripting enables system integration and customization by system integrators
- 
- M2M API - supported protocols: HTTP, HTTPS, SNMP v1/v3, XML, JSON, MQTT, Modbus/TCP, SIP (VoIP), URL API (HTTP get)
  - Other supported protocols: SMTP, DNS, NTP, UPnP, DHCP
  - Supported security protocols: SSL, WEP, WPA, WPA2, TKIP, AES, MD5
  - Supported encryption protocols: DES, AES
- 
- NETIO PowerPDU 4C and NETIO 4All: Independent power measurements for each output [V, Hz, A, W, Wh, PF ( $\cos \varphi$ )]
  - NETIO PowerPDU 4C: RS-232 serial port

### 3.2 NETIO PowerPDU 4C specifications

Power	100-240 V; 50/60 Hz; 10 A
Switched outputs	NETIO PowerPDU 4C: 10 [8] A total per device / 8 A per output
Internal consumption	Max 4.1 W
Output relay	Micro-disconnection ( $\mu$ ) (resistive load) 1E5 switching cycles, max. 1.5 kV pulse voltage Switch heat and fire resistance class 1
Electrical load	ZCS (Zero Current Switching) on each output  - Resistance load compatible - Capacitive load compatible - Inductive load compatible
Interfaces	2x Ethernet RJ-45 10/100 Mbit/s 1x RS-232 (green 3-pin terminal block - Rx, Tx, GND)
Environment	IP30, protection rating = class 1 Operating temperature 0-50 °C Device rated for pollution degree 2. Designed for continuous operation in altitudes up to 2000 m. Does not require additional cooling

### 2x network interface

NETIO PowerPDU 4C features two RJ45 Ethernet jacks.

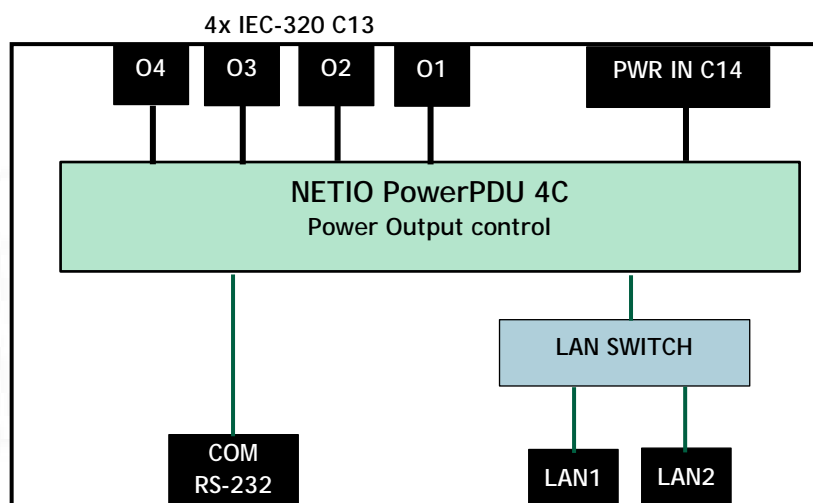


Figure 1. NETIO PowerPDU 4C block diagram

### 3.3 Drawings

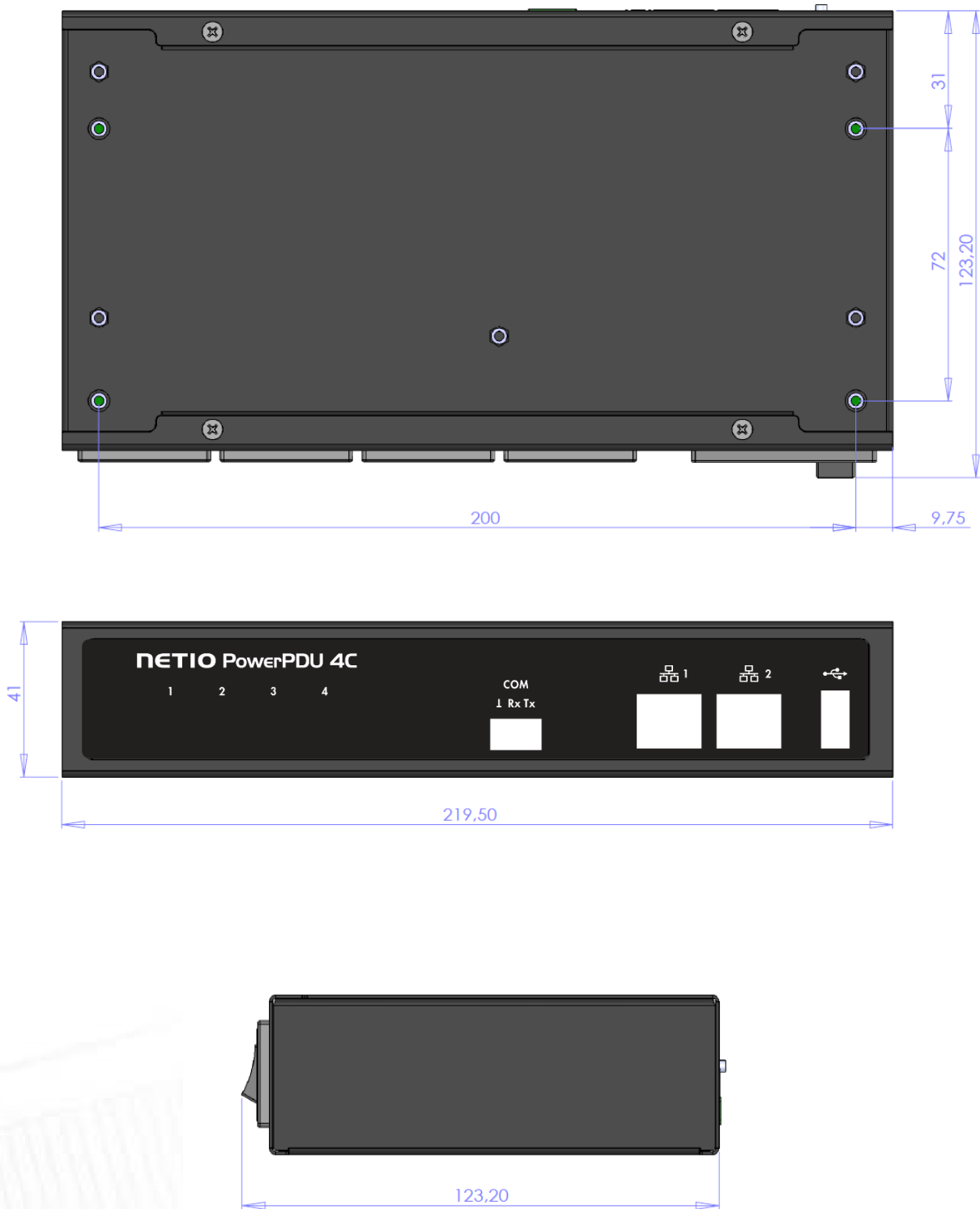


Figure 2. NETIO PowerPDU 4C dimensions



### 3.4 NETIO 4 / 4All specifications

Power	230 V; 50 Hz; max 15 A
Switched outputs	NETIO 4/4All: 15 [8] A total per device / 8 A per output
Electrical load	Resistance load only
Fuse	Built-in 15 A protection fuse
Internal consumption	Max 4.1 W
Output relay	Micro-disconnection ( $\mu$ ) (resistive load) 1E5 switching cycles, max. 1.5 kV pulse voltage Switch heat and fire resistance class 1
NETIO 4 interfaces	1x Ethernet RJ-45 10/100 Mbit/s 1x Wi-Fi 802.11b/g/n 2.4 GHz (external 2 dBi antenna)
NETIO 4All interfaces	1x Ethernet RJ-45 10/100 Mbit/s 1x Wi-Fi 802.11b/g/n 2.4 GHz (3 dBi antenna, RSMA connector) 1x Bluetooth 4.0 Low Energy 2.4 GHz (BLE) (3 dBi, RSMA connector)
Environment	IP30, protection rating = class 1 Operating temperature 0 - 50 °C Device rated for pollution degree 2. Designed for continuous operation in altitudes up to 2000 m. Does not require additional cooling

## 4 Configuration and control

### Before the first use

1. Connect your NETIO 4x to a LAN with a network cable.
2. Connect the power cable of your NETIO 4x to a mains electrical outlet.
3. Switch your NETIO 4x on with the rocker switch at the back or on the side.
4. Allow about a minute for your NETIO 4x to start up.

### System requirements (controlling system)

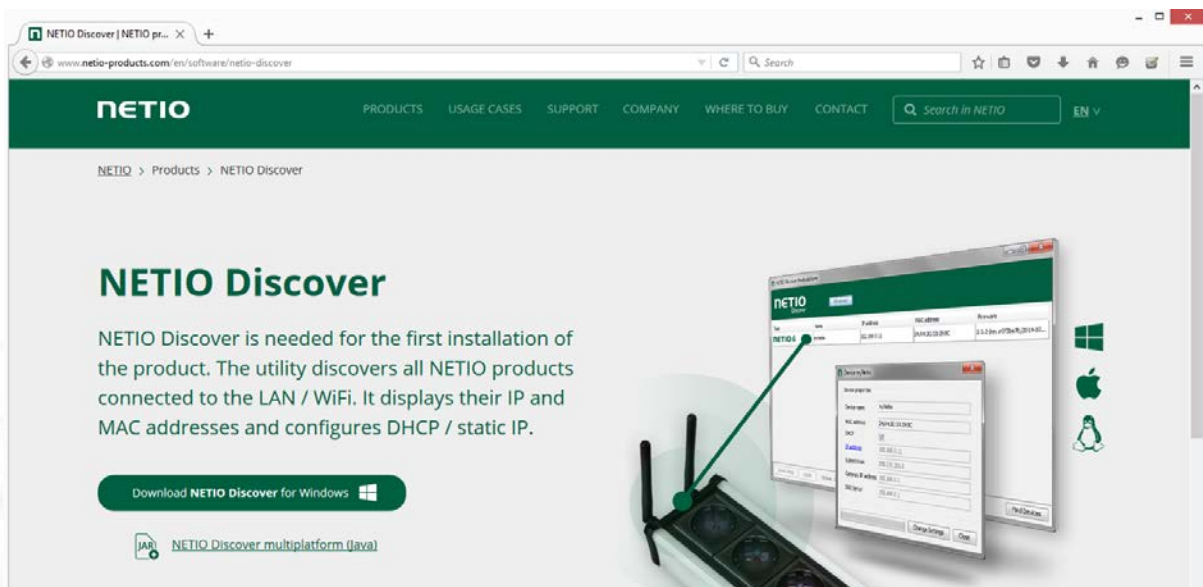
To be able to fully control the Netio 4x electrical sockets, the controlling system needs a web browser (Microsoft Internet Explorer, Opera, Mozilla Firefox, Chrome or other) with enabled JavaScript and cookies. Socket states can be changed in other ways, too; for details, see the [Mobile app](#) and [M2M API protocols](#) sections.

## 4.1 Detecting and configuring the IP address

### 4.1.1 Initial configuration over a LAN

In order to work correctly in an IP network, the device must have a correctly configured IP address.

To find the current IP address, use the NETIO Discover utility, available for download at: <http://www.netio-products.com/en/software/netio-discover> Depending on your operating system, choose the .exe file (Windows) or the .jar file (Linux or Mac). To run the .jar file, JAVA RE is needed. It is available for download at: [www.java.com](http://www.java.com)



To successfully discover the IP address, the controlling system must be in the same LAN as the NETIO 4x device.

If your network uses a DHCP server, simply click the IP address in NETIO Discover to open the web interface.

Without a DHCP server in the network, NETIO 4x defaults to the following IP address at its wired Ethernet interface: **192.168.1.78** For the NETIO 4x device to be accessible, the controlling system must use an IP address in the same subnet, such as 192.168.1.77.

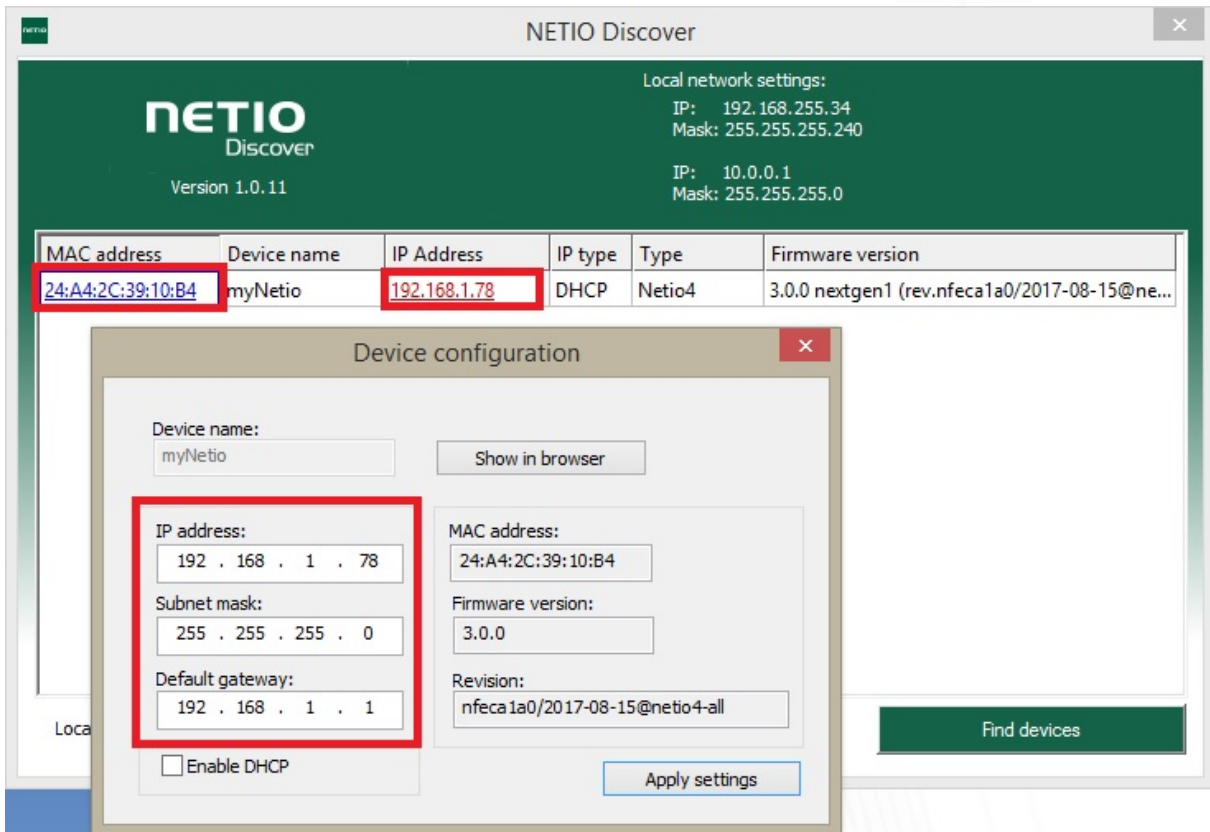


Figure 3. Configuring the IP address in NETIO Discover

To change the IP address, click the value in the MAC address column, uncheck Enable DHCP and specify the IP address, Subnet mask and Default gateway as required. After applying the settings, NETIO 4x will be accessible at the specified address.

### 4.1.2 Initial configuration over Wi-Fi - only NETIO 4 and NETIO 4All

By default, NETIO 4 and NETIO 4All act as a Wi-Fi access point. They can be connected to from a Wi-Fi client device without a wired network.

The procedure depends on the operating system. For example, in Windows 10, go to

*Settings > Network & Internet > Wi-Fi > Show available networks*

and select the Netio4 network. After selecting the network, enter 12345678-Netio4 as the network key/default password. The DHCP server in Netio 4x automatically assigns an IP address to your device. The default IP address of the Netio 4/4All Wi-Fi interface is 192.168.2.78 (NOTE: It is different from the default IP of the Ethernet interface - 192.168.1.78 (eth) vs. 192.168.2.78 (wifi)).

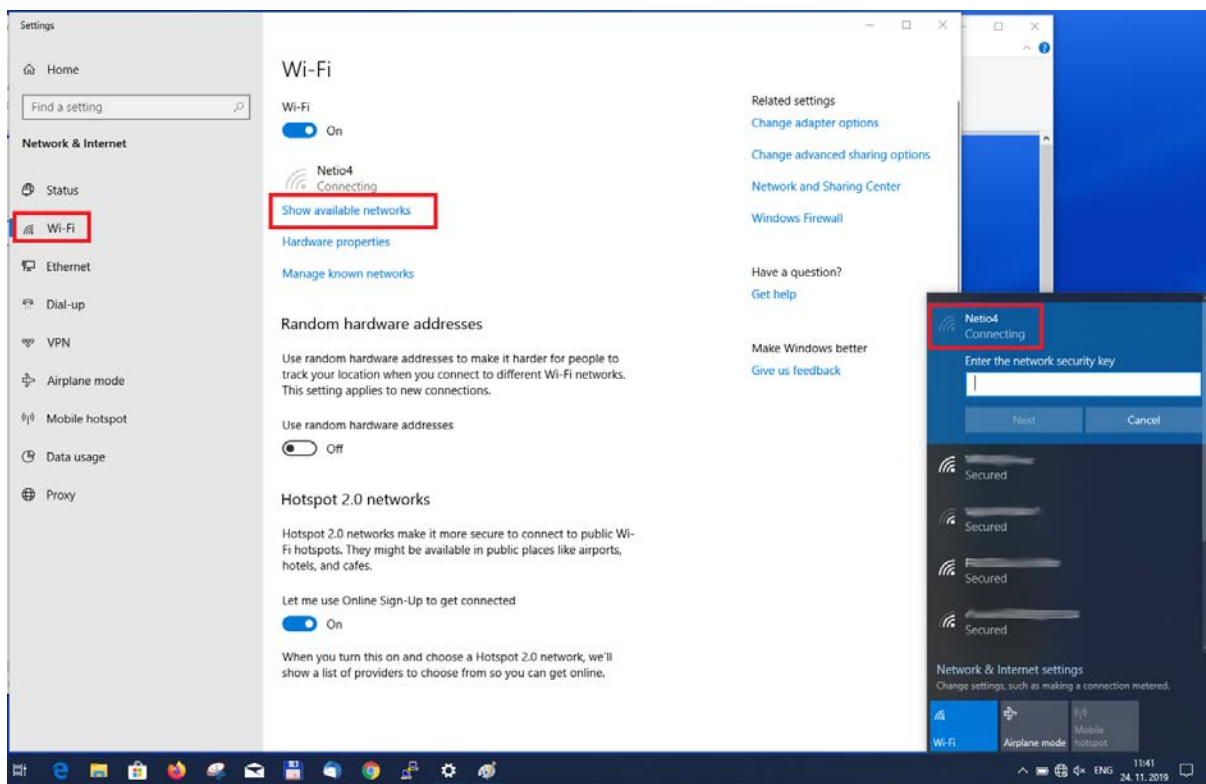


Figure 4. Connecting to the Wi-Fi network

## 4.2 Login to device web

To log in, use **admin** / **admin**.  
(default login username / password)

## 4.3 Restoring factory defaults

This operation deletes all user settings and restores default values. It is useful when the device is in an unknown state or does not behave as described in this manual. Reset to factory defaults can be performed over the web interface (see the [System settings](#) section) or by **holding down buttons 1 and 2 while powering up** the device. Hold the buttons pressed until the device beeps twice. After restoring the defaults, NETIO 4x again beeps twice.





### Forgotten password

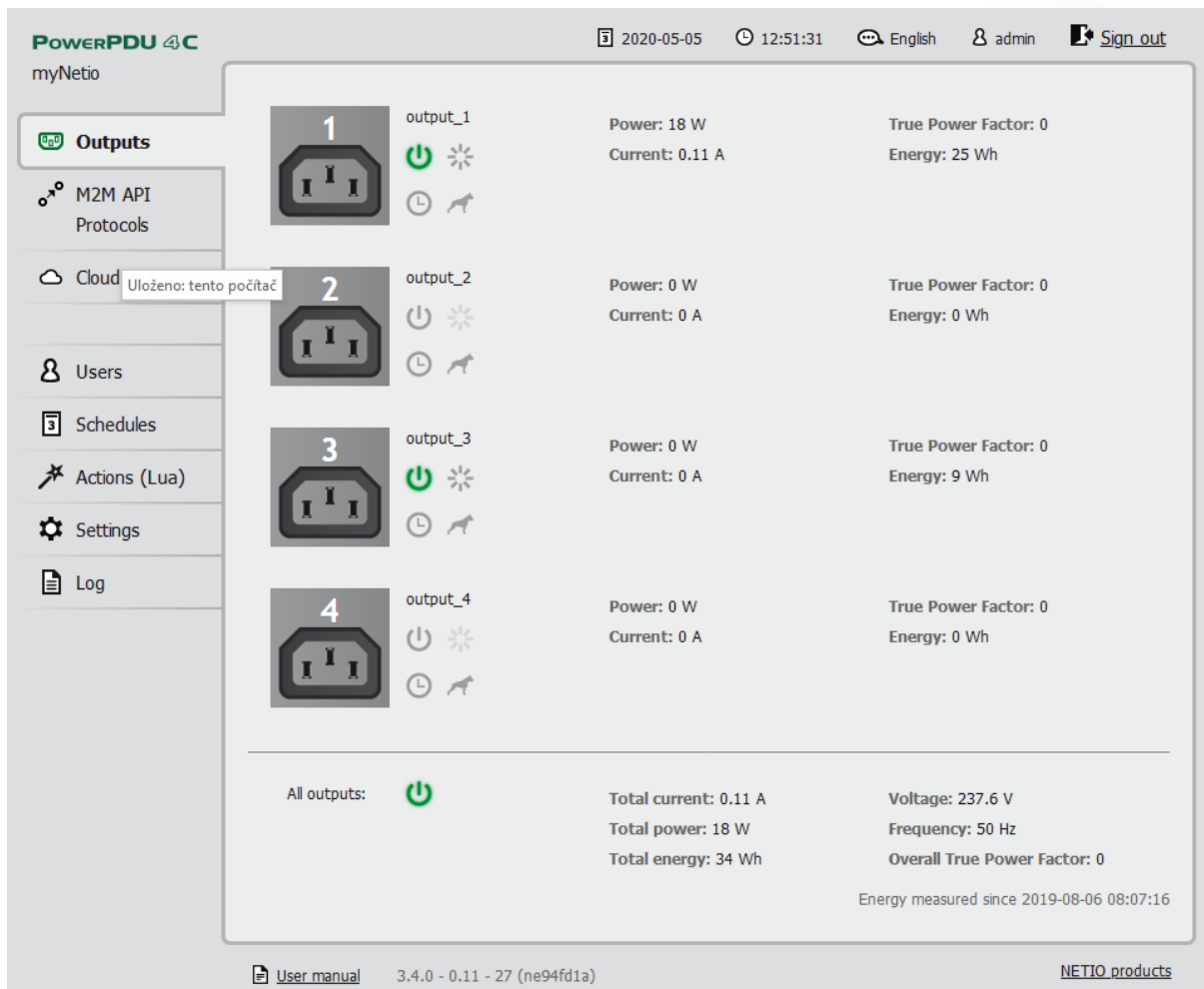
The reset to factory defaults is most often used when the password has been forgotten. After restoring the factory defaults, the username and password to access NETIO 4x will be "**admin**" / "**admin**".

## 5 Web interface

### 5.1 Outputs

#### Controlling the outputs

In the left menu, choose  **Outputs**. An overview of outputs appears. The four buttons next to each output control the respective output and its automated functions. The power button  below the line at the bottom switches all outputs on/off at the same time.



The screenshot shows the 'Outputs' page in the NETIO web interface. The page is titled 'POWERPDU myNetio' and includes a navigation menu on the left with options: Outputs, M2M API Protocols, Cloud (Uloženo: tento počítač), Users, Schedules, Actions (Lua), Settings, and Log. The main content area displays four individual output controls, each with a power button, a restart button, and a timer button. The status of each output is shown as follows:


Output	Power	Current	True Power Factor	Energy
output_1	18 W	0.11 A	0	25 Wh
output_2	0 W	0 A	0	0 Wh
output_3	0 W	0 A	0	9 Wh
output_4	0 W	0 A	0	0 Wh


Below the individual outputs, there is a summary for 'All outputs' with a power button and the following statistics:


Parameter	Value
Total current	0.11 A
Total power	18 W
Total energy	34 Wh
Voltage	237.6 V
Frequency	50 Hz
Overall True Power Factor	0

Additional information at the bottom of the summary includes 'Energy measured since 2019-08-06 08:07:16'. The footer of the page contains 'User manual 3.4.0 - 0.11 - 27 (ne94fd1a)' and 'NETIO products'.

Figure 5. Controlling the outputs

The  **Power** button directly controls the respective output. When the output is on, the button is green; when the output is off, the button is grey.

The  **Restart** button power-cycles the respective output, with the defined switch-off delay. This “Short ON/OFF delay” can be configured for each output. Only an output that is switched on can be restarted. The default delay is 5000 ms (5 seconds).

The  **All outputs** button under the line at the bottom controls all outputs together. Depending on the current state, all outputs can be switched on or off together.


The  **Timer** button indicates an active Timer function when lit (see below for details).

The  Watchdog button indicates an active Watchdog function when lit (see below for details).

When the button is green, the Timer / Watchdog function is active. When the button is grey, the function is inactive. Both functions can be configured for each output individually at the "Timer/Scheduler" and "Watchdog" tabs.

### 5.1.1 General

Click the icon with the output number to open detailed configuration for that output.

The  General tab configures basic parameters for controlling the output.

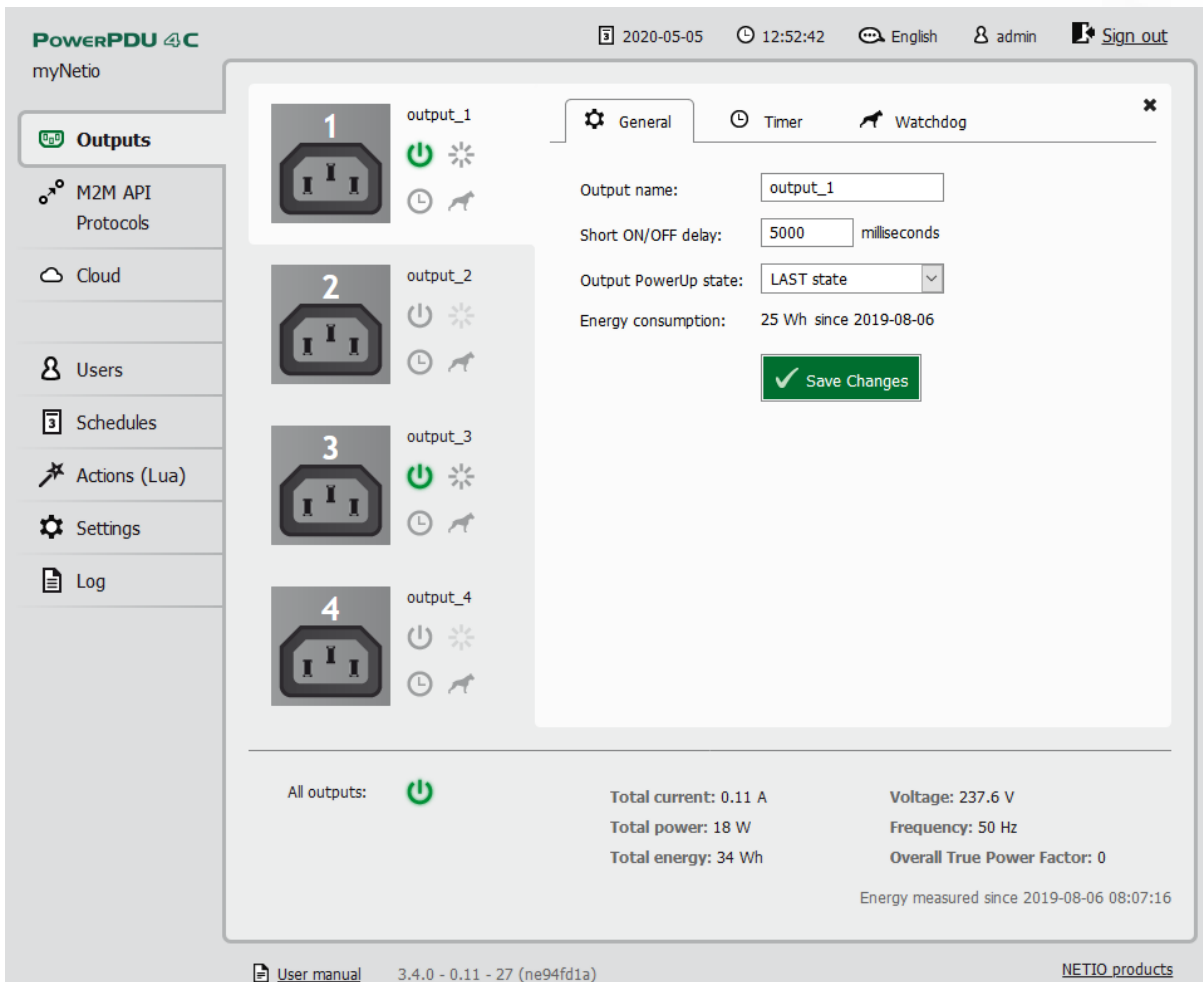



Figure 6. General output configuration

Output name is displayed above the four control buttons and improves clarity.


**Short ON/OFF delay** is an integer specifying, in milliseconds, the pulse duration for the Short OFF (power cycling) and Short ON actions. During this interval, any attempt to switch the output state via Lua scripts or M2M API protocols is ignored.

**Output PowerUp state** defines, for each output, the state of the output when the device is powered up or restarted. The output can be switched on (ON), switched off (OFF), or set to the last state before the restart (LAST).


Click **Save Changes** to save the settings.

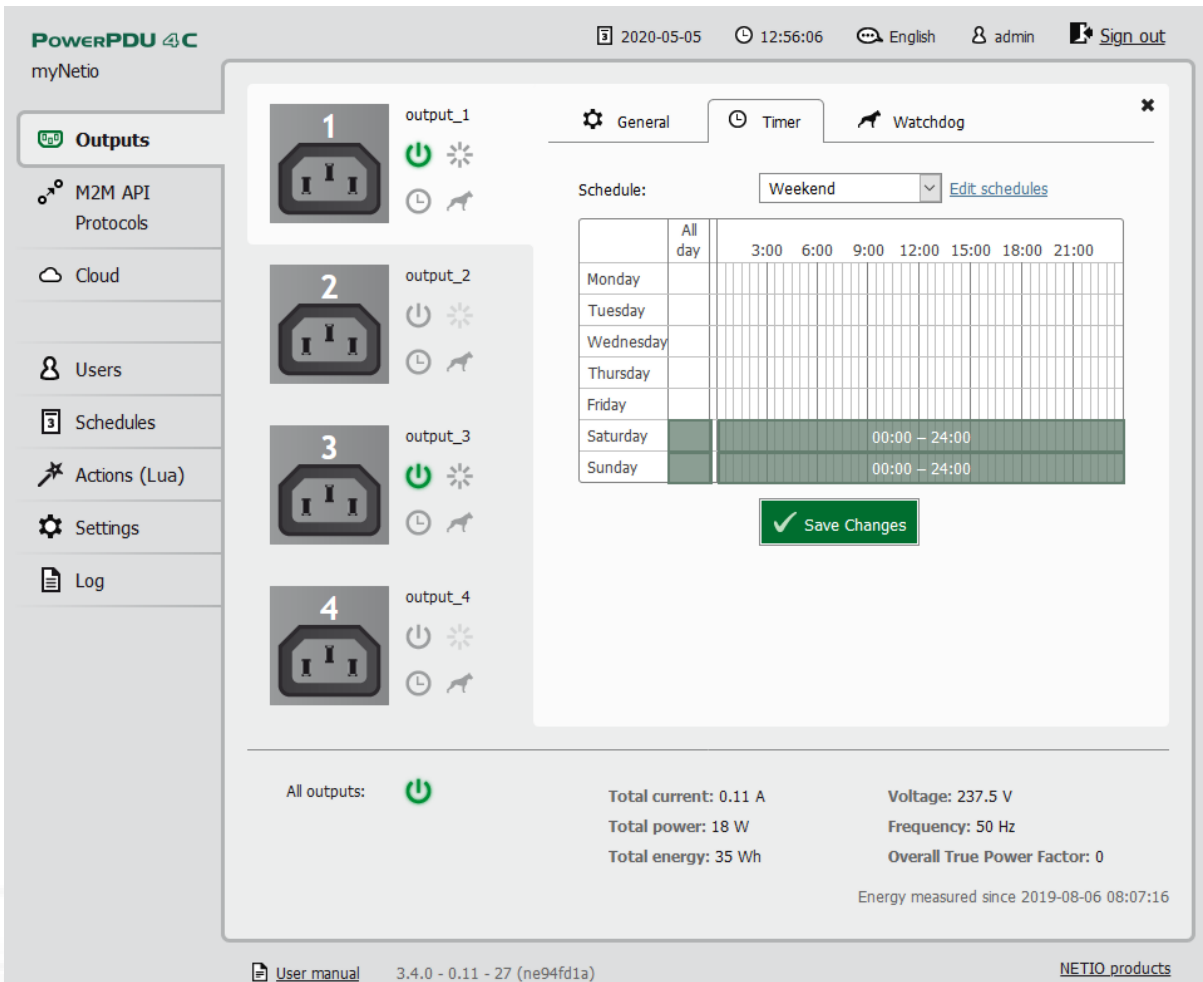
To close the configuration dialog, click the  symbol in the top right corner.

## 5.1.2 Scheduler/Timer

The  Scheduler/Timer function can be used to switch the output on and off at specified times. Select a schedule to specify when the output should be turned on. After selecting a schedule, the on-times for the output according to that schedule are shown graphically. Click **Save Changes** to confirm the schedule change. After the changes are saved, the Timer/Scheduler is immediately active and sets the output state accordingly. If an output that is currently switched on would be switched off by activating the timer/scheduler, by default such a change needs to be confirmed.

In the standard WEB GUI, the displayed schedule cannot be changed. To change a schedule, click **Edit schedules**.

To manually enable or disable the Scheduler/Timer for a given output, click the  Scheduler/Timer button next to that output. When the Scheduler/Timer function is disabled, the outputs stay in their current states. When it is enabled, the outputs are set according to the schedule.



**POWERPDU 4C**  
myNetio

2020-05-05 12:56:06 English admin Sign out

**Outputs**

M2M API Protocols

Cloud



Users



Schedules



Actions (Lua)



Settings

Log

**1** output\_1  

**2** output\_2  


**3** output\_3  


**4** output\_4  

General Timer Watchdog

Schedule: Weekend [Edit schedules](#)

	All day	3:00	6:00	9:00	12:00	15:00	18:00	21:00
Monday								
Tuesday								
Wednesday								
Thursday								
Friday								
Saturday		00:00 – 24:00						
Sunday		00:00 – 24:00						

 Save Changes

All outputs: 

Total current: 0.11 A  
Total power: 18 W  
Total energy: 35 Wh

Voltage: 237.5 V  
Frequency: 50 Hz  
Overall True Power Factor: 0


Energy measured since 2019-08-06 08:07:16

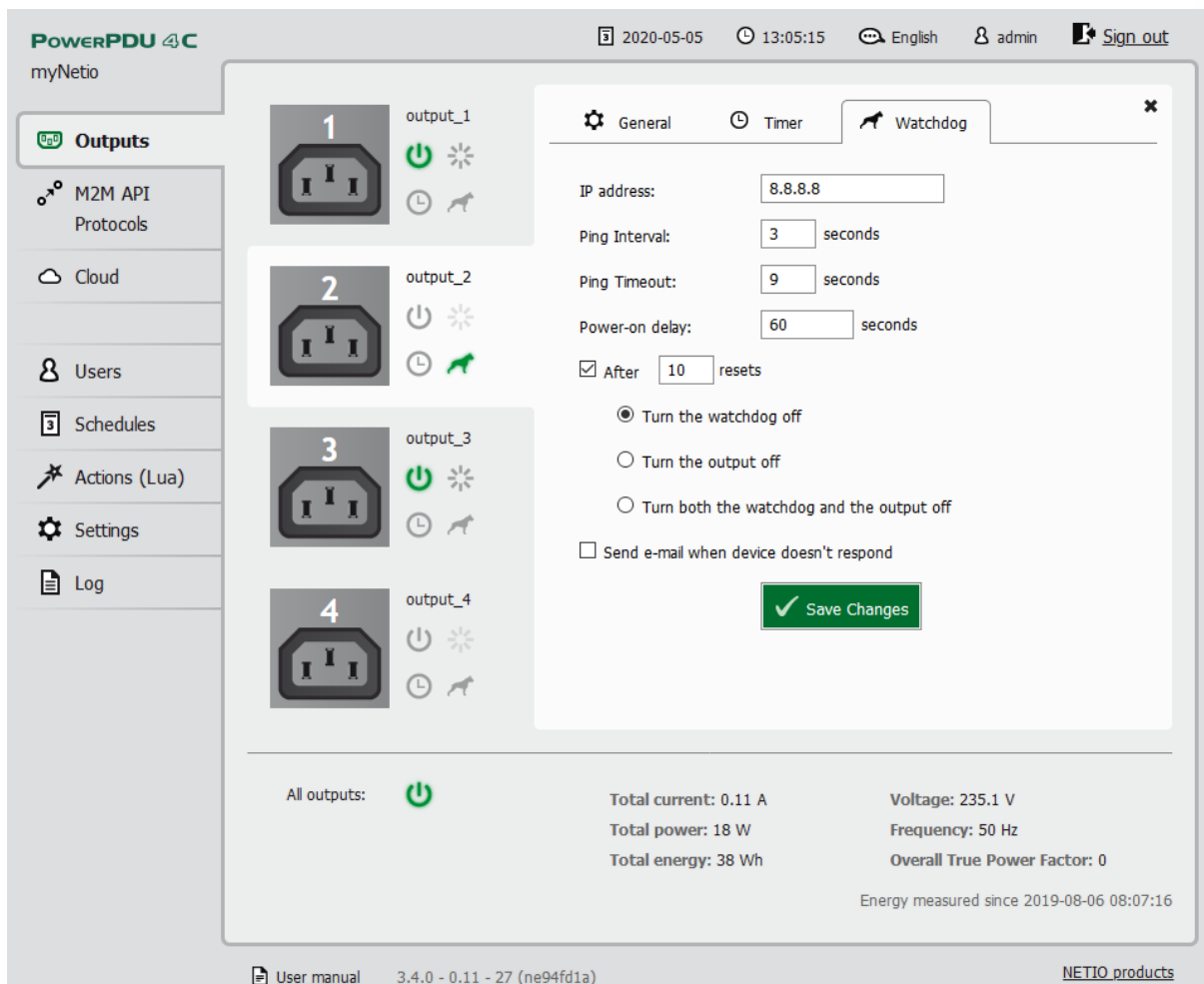
[User manual](#) 3.4.0 - 0.11 - 27 (ne94fd1a) [NETIO products](#)

Figure 7. Configuring the Timer/Scheduler



### 5.1.3 Watchdog function

The  Watchdog function monitors a device connected to an output and restarts it if necessary. The device needs to be accessible over the network using the ping command (it needs to have an IP address assigned). If the monitored device is inaccessible, the specified output is power-cycled. A device is considered inaccessible if it fails to respond to Ping requests within a specified interval. (Ping is a service that allows checking the connection between two network interfaces.) To avoid infinite switching if the monitored device is faulty, the maximum number of restart attempts can be set.

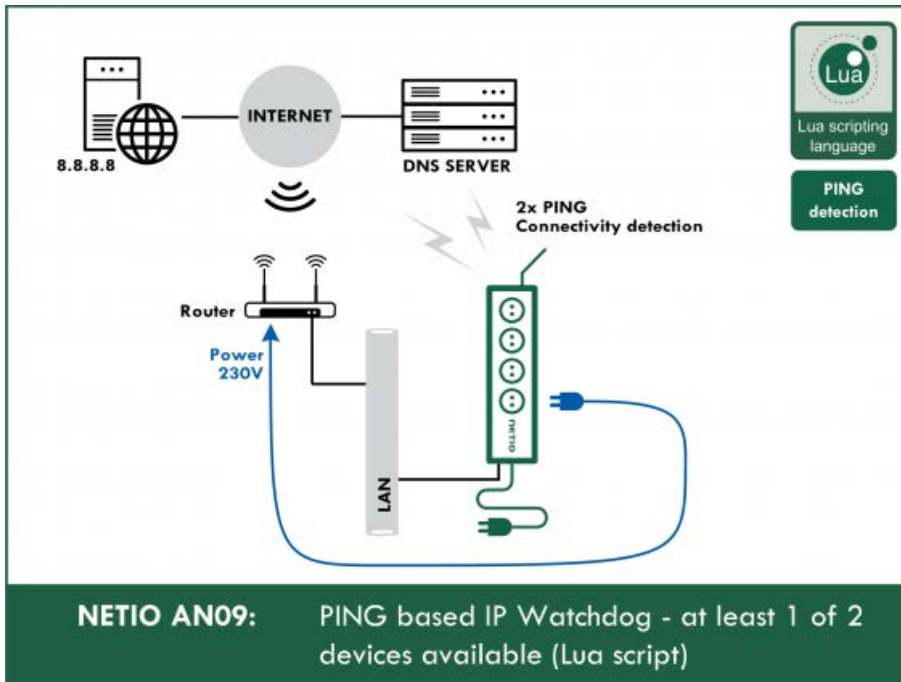


The screenshot displays the NETIO web interface for configuring the Watchdog function. The interface is titled "POWERPDU 4C myNetio" and shows a sidebar with navigation options: Outputs, M2M API Protocols, Cloud, Users, Schedules, Actions (Lua), Settings, and Log. The main content area shows four outputs (output\_1 to output\_4) with their respective status icons. The Watchdog configuration panel is active, showing settings for IP address (8.8.8.8), Ping Interval (3 seconds), Ping Timeout (9 seconds), Power-on delay (60 seconds), and a checkbox for "After 10 resets". Three radio button options are available: "Turn the watchdog off" (selected), "Turn the output off", and "Turn both the watchdog and the output off". There is also a checkbox for "Send e-mail when device doesn't respond" and a "Save Changes" button. At the bottom, system statistics are displayed: All outputs: [power icon], Total current: 0.11 A, Total power: 18 W, Total energy: 38 Wh, Voltage: 235.1 V, Frequency: 50 Hz, Overall True Power Factor: 0, and Energy measured since 2019-08-06 08:07:16.

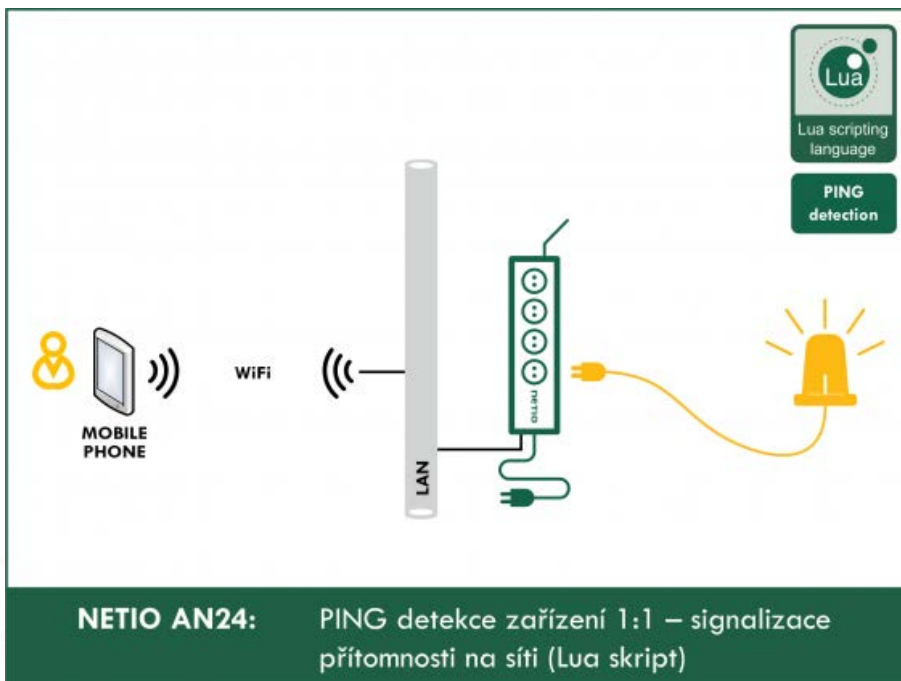
Figure 8. Configuring the Watchdog function

The other options how to use Watchdog are described in following application notes:

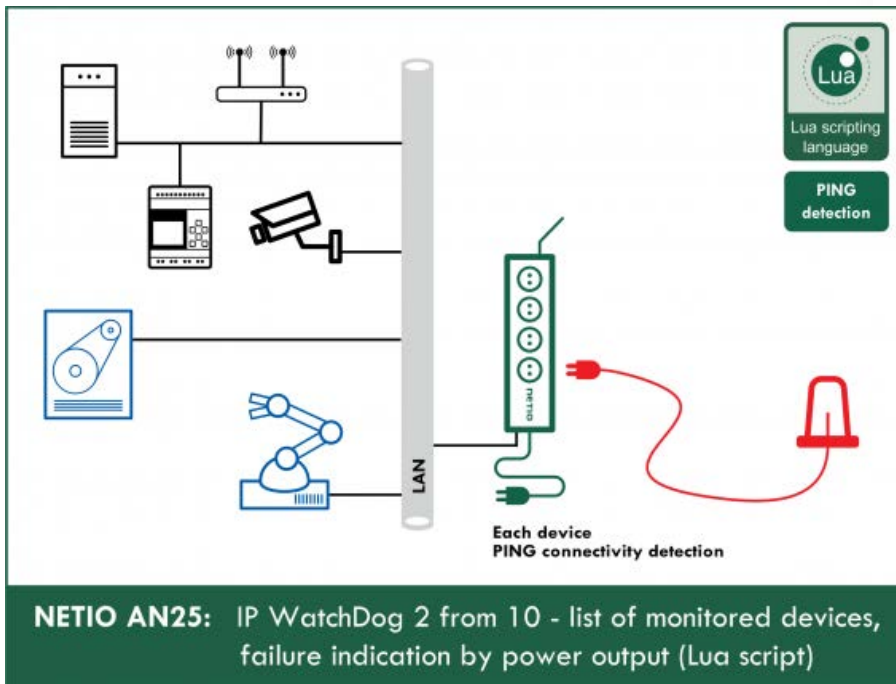
[AN09 IP WatchDog 1 of 2 - PING based failure detection for 1 or 2 devices \(Lua script\)](#)




[AN24 IP WatchDog 1 to 1 - Device LAN connectivity detection and indication \(Lua script\)](#)



AN25 IP WatchDog 2 from 10 - list of monitored devices, failure indication by power output (Lua script)



## 5.1.4 Consumption metering - only NETIO PowerPDU 4C and NETIO 4All

 All power measurement and metering functions are available on the NETIO PowerPDU 4C and NETIO 4All models only.

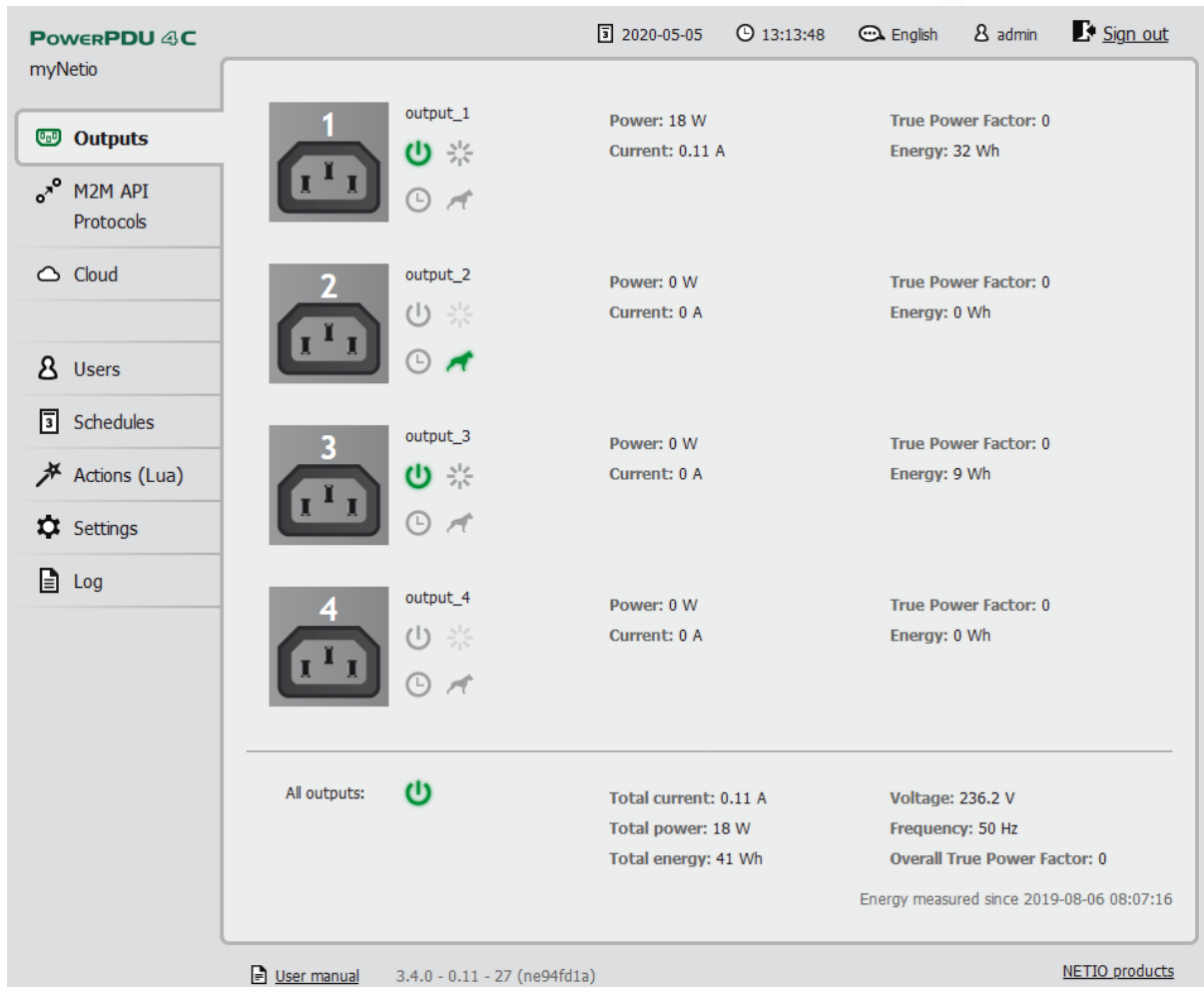


Figure 9. Consumption metering

**Power** in watts [W] is the product of the immediate current and voltage ( $P = U * I$ ).

**Current** in amps [A] shows the immediate current flowing through the given output.

**True Power Factor** expresses the ratio of active and apparent power, or the ratio of resistance and impedance. A value less than 1 means that there is a phase difference between the current and the voltage, i.e. higher energy losses compared to a purely resistive load.

**Energy** in watt-hours (Wh / kWh) is the cumulative energy consumed over a time interval. The value is the total consumption at the given input from the selected date to the present.

By default, the cumulative energy consumption is counted from the time the NETIO PowerPDU 4C or NETIO 4All was first powered on. To reset the counters at all outputs, go to **Settings > System** and click **Reset Energy Consumption Counters** below the line. The energy consumption will be measured from this moment onwards.

**Voltage** in volts [V] is the same for all outputs.

**Frequency** in hertz [Hz] is the same for all outputs.

Other totals are calculated from the values measured at individual inputs.

The screenshot displays the NETIO PowerPDU 4C web interface. The top navigation bar includes the device name 'myNetio', the date '2020-05-05', the time '13:17:18', the language 'English', the user 'admin', and a 'Sign out' link. The left sidebar contains a menu with options: Outputs, M2M API Protocols, Cloud, Users, Schedules, Actions (Lua), Settings (highlighted), and Log. The main content area is titled 'System' and shows various configuration options: Uptime (1 hours 2 minutes 52 sec), Firmware version (3.4.0 with an 'Upgrade' link and a 'Show details' link), Device name (myNetio), HTTP port (80), and Global Outputs PowerUp interval (2000 milliseconds). There are three unchecked checkboxes: 'Disable manual control buttons', 'Disable status LEDs', and 'Log outlet state changes'. A green 'Save Changes' button is visible. Below this, a red box highlights the 'Reset Energy Consumption Counters' button. Other buttons include 'Restore Factory Defaults', 'Export configuration', and 'Import configuration'. The bottom status bar shows 'User manual', version '3.4.0 - 0.11 - 27 (ne94fd1a)', and a link to 'NETIO products'.

Figure 10. Resetting the energy consumption counters [Wh / kWh] for all outputs

## 5.2 M2M API Protocols

All M2M protocols use the same values for the *action* parameter to control the outputs. Allowed values are:

- 0 - turns the output off
- 1 - turns the output on
- 2 - short OFF - turns the output off for a short<sup>1</sup> time (if the output was off, it will be turned on)
- 3 - short ON - turns the output on for a short<sup>2</sup> time (if the output was on, it will be turned off)
- 4 - toggle - toggles the current output state
- 5 - no operation - leaves the output unchanged
- 6 - ignore - ignores the action attribute and only respects the state attribute
- i - power-cycles the output (maintained for backwards compatibility)<sup>3</sup>
- u - leaves the output state unchanged (maintained for backwards compatibility)<sup>4</sup>

---

<sup>1,2</sup> The short-off/short-on time can be also specified directly as a part of the command issued over the respective M2M protocol. If unspecified, the Short ON/OFF delay field value is used.

<sup>3,4</sup> Telnet only

## 5.2.1 SNMP

The screenshot displays the NETIO web interface for configuring SNMP. The top navigation bar includes the 'POWERPDU 4C myNetio' logo, a date of '2020-05-05', a time of '13:51:03', language set to 'English', user 'admin', and a 'Sign out' button. A left sidebar contains menu items: 'Outputs', 'M2M API Protocols' (highlighted), 'Cloud', 'Users', 'Schedules', 'Actions (Lua)', 'Settings', and 'Log'. The main content area is titled 'SNMP' and features a list of protocols on the left: 'Telnet/KSHELL', 'MQTT', 'Serial Console', 'JSON API', 'XML API', 'URL API', and 'Modbus/TCP'. The 'SNMP' configuration panel includes a checked 'Enable SNMP' checkbox, a 'Port' field with '161', an 'SNMP version' dropdown menu set to '1,2c', and a 'Community' field with 'public' (noted as 'READ-ONLY access'). A green 'Save Changes' button is present. A 'Download MIB file' button is located at the bottom right of the configuration area. The footer contains a 'User manual' link, version information '3.4.0 - 0.11 - 27 (ne94fd1a)', and 'NETIO products'.

Figure 11. SNMP configuration for version 1,2c

POWERPDU 4C  
myNetio

2020-05-05 13:52:31 English admin Sign out

Outputs

M2M API Protocols

Cloud

Users

Schedules

Actions (Lua)

Settings

Log

• SNMP

Telnet/KSHELL

MQTT

Serial Console

JSON API

XML API

URL API

Modbus/TCP

Enable SNMP

Port:

SNMP version:

Allow READ-WRITE access

Username:

Password:  [Show password](#)

Security:

Authentication:

Encryption:

Allow READ-ONLY access

Username:

Password:  [Show password](#)

Security:

Authentication:

Encryption:

[User manual](#) 3.4.0 - 0.11 - 27 (ne94fd1a) [NETIO products](#)

Figure 12. SNMP configuration for version 3

**Enable SNMP**

Enables SNMP functions in the system kernel.

**Port**

Port where the device listens for SNMP M2M API commands.

Range: 1 - 65535. The device warns you if you specify a port number that is already in use. However, to be sure, we recommend using either the standard SNMP port 161, or port numbers above 1024.

**SNMP version**

1,2c - unsecured, unencrypted. Generally used to monitor the device.

3 - complex security via encryption and authentication. Can be used to write values / control output states.

**Community**

v1,2c only

Also called the "community string" in SNMP. Similar to a username/password combination. Needed for reading information from NETIO 4x over SNMP.

We recommend to use "pure" ASCII characters (that is, to avoid accented and special characters, such as @, & and so on, if possible).



The following values can be set for SNMP v3 only.

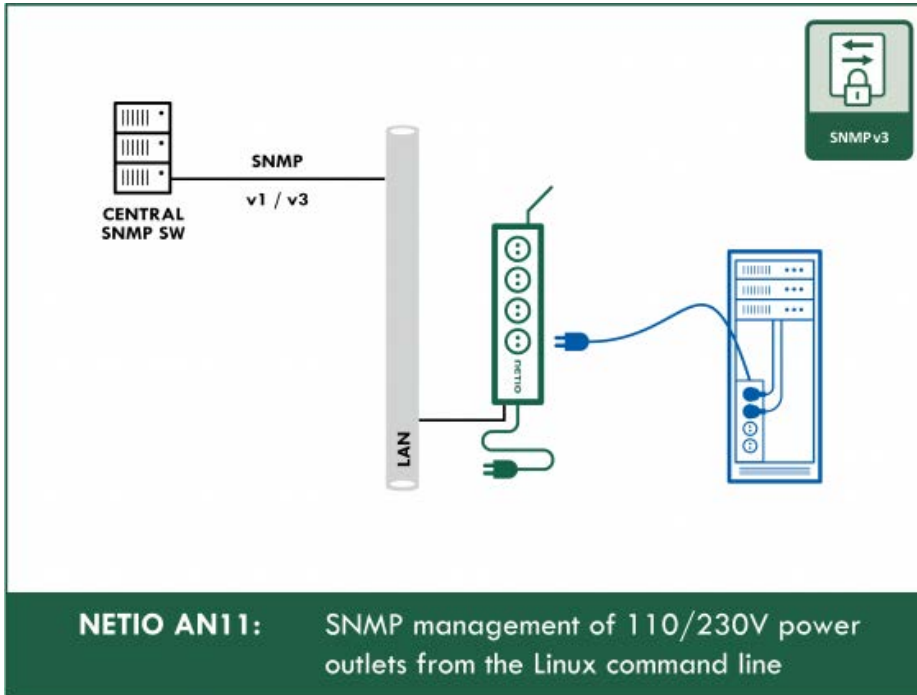
<b><i>Allow READ-WRITE access</i></b>	Select this option if outputs need to be controlled over SNMP (to allow switching the outlets on and off).
<b><i>Allow READ-ONLY access</i></b>	Select this option if outputs should not be controlled and SNMP is only used to read their states (on/off).
<b><i>Username</i></b>	Username for SNMP authentication using the user-based security model.  This is unrelated to the username/password for NETIO 4x administration.
<b><i>Password</i></b>	Password corresponding to the SNMP username. We recommend to follow the usual guidelines for choosing a strong password (avoid easy-to-guess passwords, short passwords, sequences of the same letters or numbers).
<b><i>Security</i></b>	<b>authPriv</b> - when selected, packets are encrypted with the algorithm specified in the <i>Encryption</i> field.  <b>authNoPriv</b> - when selected, SNMP commands are unencrypted and there is a higher risk of eavesdropping.
<b><i>Authentication</i></b>	Authentication algorithm used in the user-based security model. Possible values: [ SHA / MD5 ]
<b><i>Encryption</i></b>	Encryption algorithm for SNMP communication - possible values: [ AES / DES ]
<b><i>Download MIB file</i></b>	Allows downloading the Management Information Base file containing the variable names and data types for the SNMP implementation in NETIO 4x. For more information about the MIB, see the official SNMP website: <a href="http://www.net-snmp.org/">http://www.net-snmp.org/</a>

For details about the M2M SNMP API, visit our website: <http://netio-products.com>, section Support > Download, and download the document titled:

[SNMP - description of NETIO M2M API interface - PDF](#)

For more information and a practical demonstration of using the SNMP protocol with NETIO 4x smart sockets, see the following Application Note:

[AN11 SNMP management of 110/230V power outlets from the command line in Windows and Linux](#)



## 5.2.2 TELNET/KSHELL

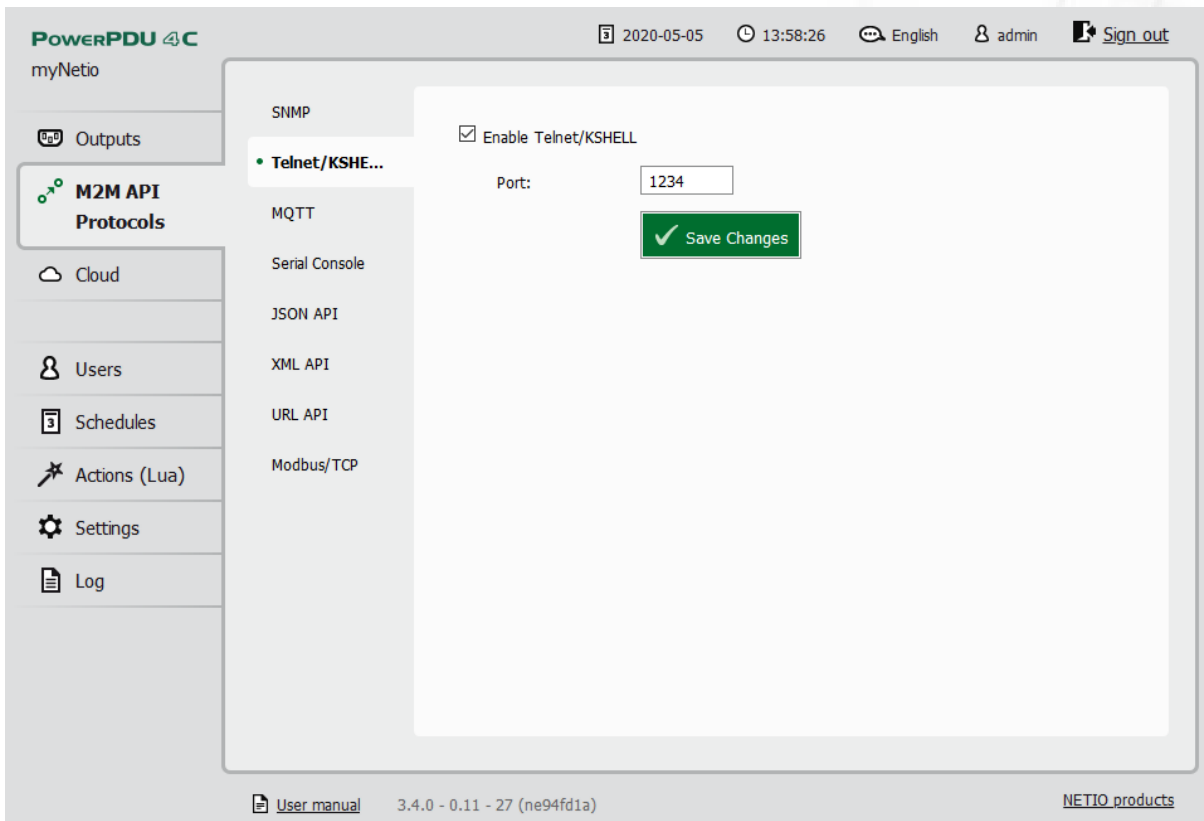


Figure 13. Configuring the TELNET/KSHELL protocol

### *Enable Telnet/KSHELL*

Enables Telnet/KSHELL functions in the system kernel.

### *Port*

Port where the device listens for Telnet/KSHELL M2M API commands.

Range: 1 - 65535. The device warns you if you specify a port number that is already in use. However, to be sure, we recommend to use either the standard Telnet port 23, or port numbers above 1024.

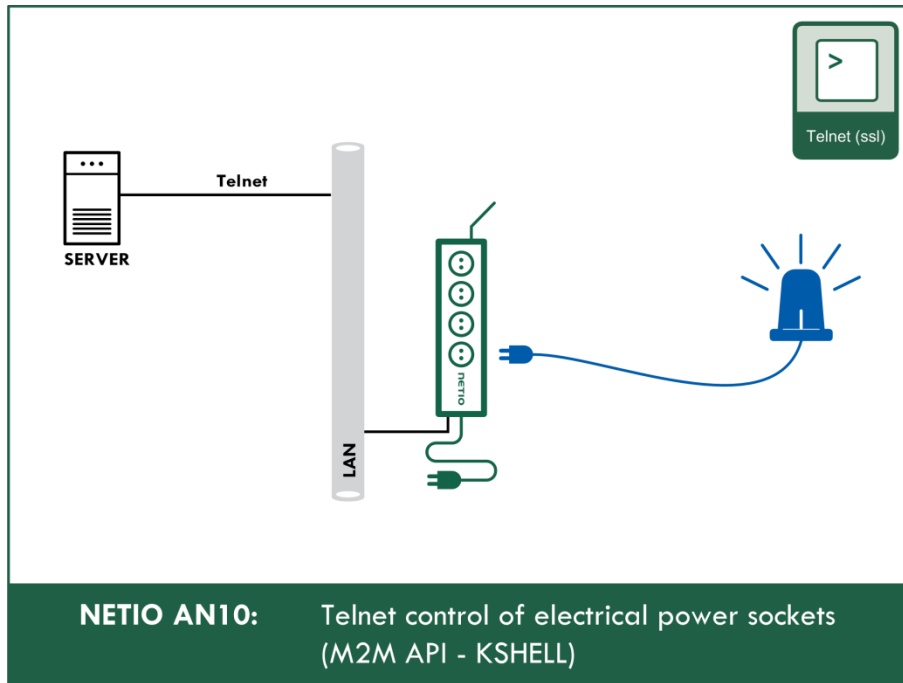
Unlike other M2M API protocols, the Telnet/KSHELL M2M API uses internally-created users and their privileges (admin/user/guest). These users can be configured under the [Users](#) menu entry.

For a list of telnet commands (login, port list, ...) and more details about the M2M Telnet/KSHELL API, see the **Support > Download** section of our website and the following document:

[TELNET - description of NETIO M2M API interface - PDF](#)

For more information and a practical demonstration of using the telnet protocol with NETIO 4x smart sockets, see the following Application Note:

[AN10 Telnet control of electrical power sockets \(M2M API - KSHELL\)](#)



**NETIO AN10:** Telnet control of electrical power sockets (M2M API - KSHELL)

### 5.2.3 MQTT

The screenshot displays the MQTT configuration page in the PowerPDU myNetio web interface. The interface includes a top navigation bar with the date (2020-05-05), time (14:00:40), language (English), user (admin), and a sign out button. A left sidebar contains navigation options: Outputs, M2M API Protocols (selected), Cloud, Users, Schedules, Actions (Lua), Settings, and Log. The main content area is titled 'MQTT' and features a list of protocols on the left: SNMP, Telnet/KSHELL, MQTT (selected), Serial Console, JSON API, XML API, URL API, and Modbus/TCP. The MQTT configuration form includes the following fields and options:

- Enable MQTT
- MQTT status: Publish failed (Disconnected)
- MQTT mode: Generic (dropdown menu)
- Broker Host: example.com
- Broker Port: 1883
- Use credentials
  - Username: [text input]
  - Password: [text input] [Show password](#)
- Use SSL
  - Validate server's SSL certificate
- Client Id: netio
- Default MQTT messages disabled (MQTT allowed only for LUA Actions)
- Update period: 60 seconds
- Current delta [mA]: 200
- TPF delta [mill -]: 50 (e.g. to set TPF delta 0.01 put in value 10)

A green 'Save Changes' button is located at the bottom of the form. At the bottom of the page, there is a footer with a link to the 'User manual', the version number '3.4.0 - 0.11 - 27 (ne94fd1a)', and a link to 'NETIO products'.

Figure 14. MQTT configuration in Generic mode

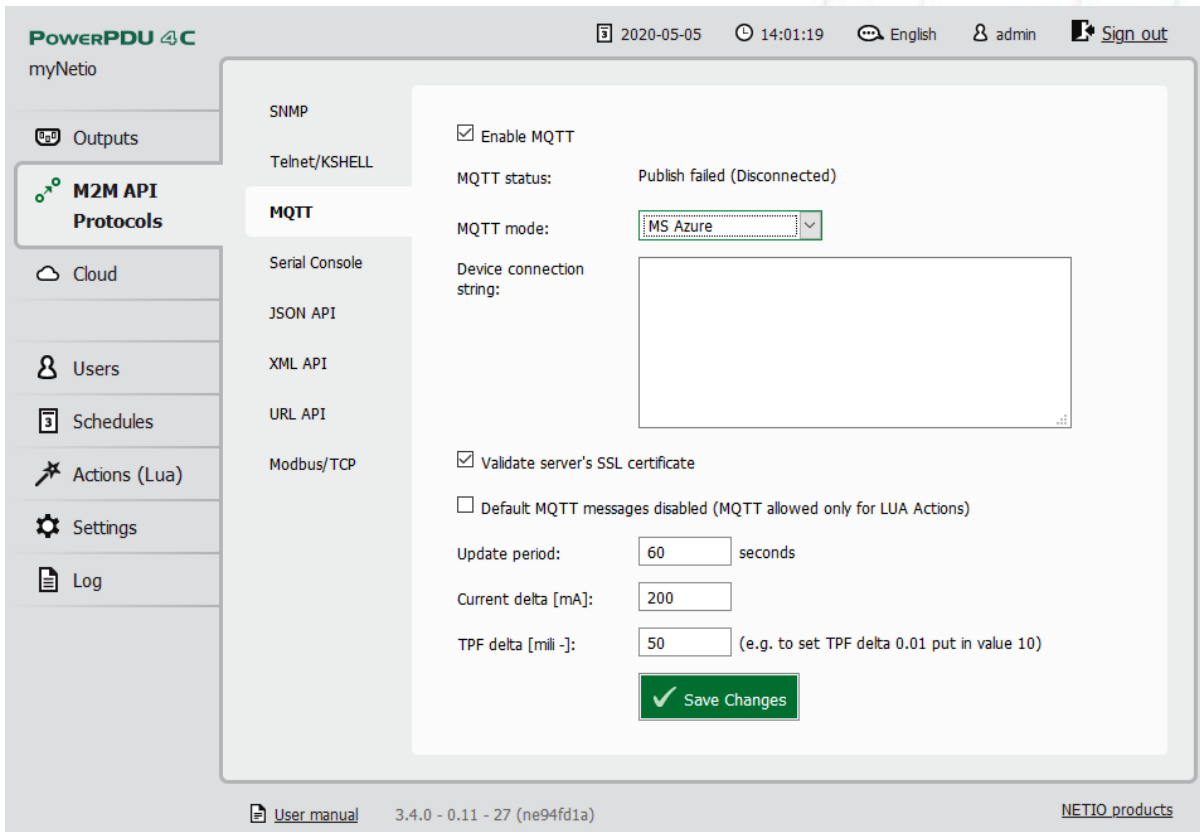


Figure 15. MQTT configuration in MS Azure mode

<b>Enable MQTT</b>	Enables MQTT functions in the system kernel.
<b>MQTT status</b>	Off - MQTT is disabled Connected - NETIO 4x is connected to the MQTT broker Error - error in communication with the MQTT broker
<b>MQTT mode</b>	Generic - generic MQTT broker MS Azure - for connecting to the MS Azure IoT Hub
<b>Broker host</b>	Domain name or IP address of the MQTT broker.
<b>Broker port</b>	Port for the communication with the MQTT broker.
<b>Username</b>	Login name for authenticating with the MQTT broker.
<b>Password</b>	Password for authenticating with the MQTT broker.
<b>Use SSL</b>	Enables secure connection.
<b>Validate server's SSL certificate</b>	When enabled, the server's SSL certificate is validated against verified certification authorities.
<b>Client Id</b>	Unique device ID for distinguishing at the connected broker level (part of MQTT topic).
<b>Update period</b>	Update messages are sent in these intervals.

*Device connection String*

Only MS Azure mode: Device-specific connection string. It is generated in the Device Explorer.

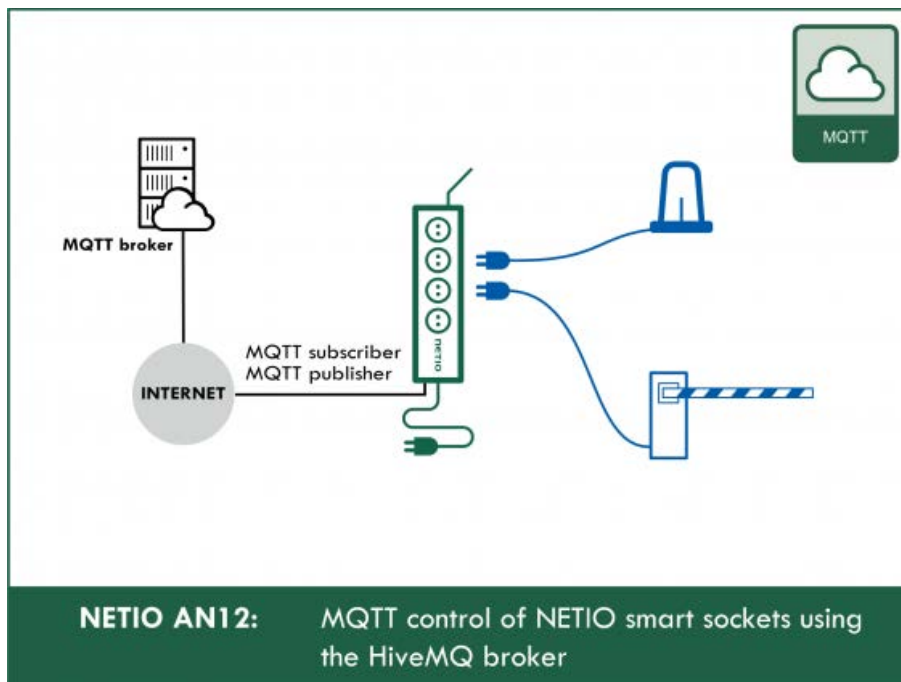
---

For more information about the M2M MQTT API, visit the **Support > Download** section of our website and see the following document:

[MQTT - description of NETIO M2M API interface - PDF](#)

For more information and a practical demonstration of using the MQTT protocol with NETIO 4x smart sockets, see the following Application Notes:

[AN12 MQTT control of NETIO smart sockets using the HiveMQ broker - for an example of the Generic mode](#)



## Serial console - NETIO PowerPDU 4C only

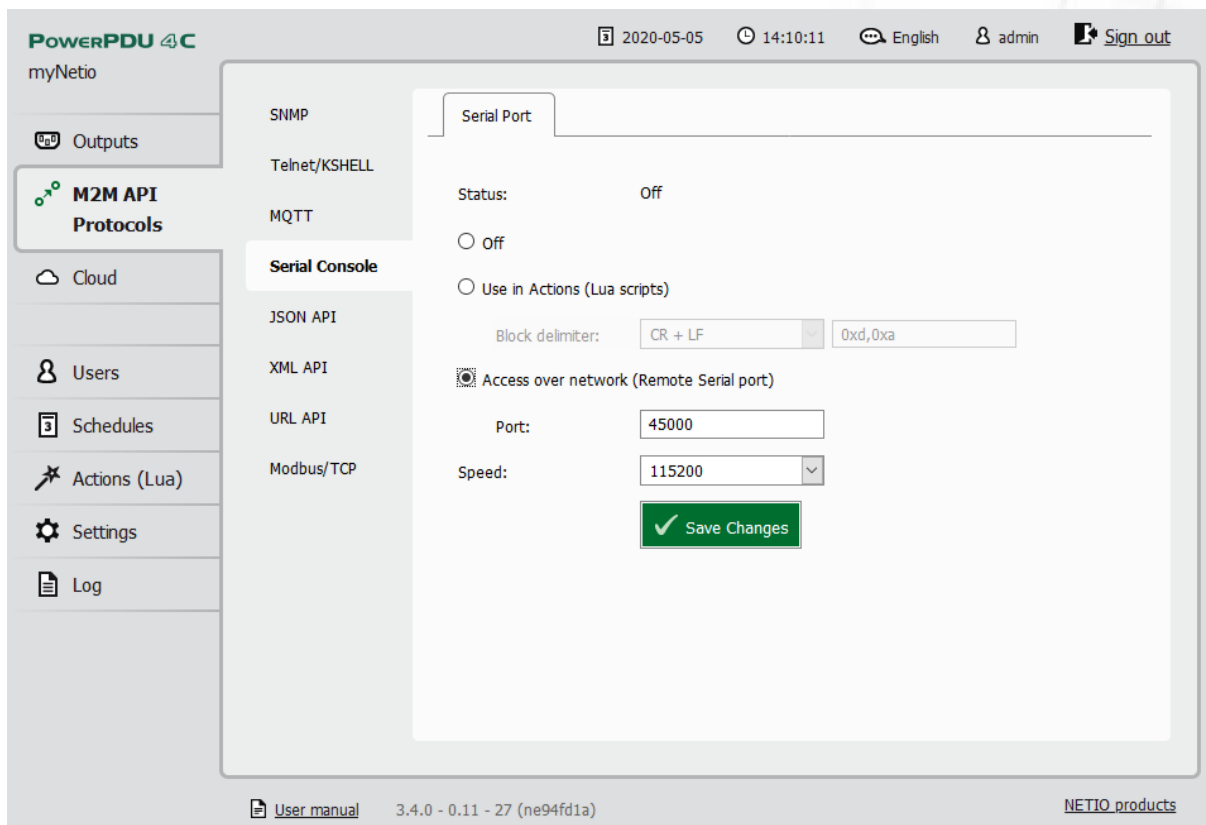


Figure 16. Configuring the serial console

<b>Status</b>	Informative indication of the serial line communication status. Possible states: <ul style="list-style-type: none"> <li>• <i>connected</i></li> <li>• <i>waiting for connection</i></li> <li>• <i>failed to send serial event to actions</i></li> </ul>
---------------	---

<b>Off</b>	The function is turned off ☺
------------	------------------------------

<b>Use in actions</b>	Check this box if you need to react to serial port communication in Lua scripts.
-----------------------	--

<b>Block delimiter</b>	When the specified delimiter is received over the serial line, the TODO variable in the Lua script is filled and the block can be processed. The following delimiters are available for selection: <p>CR+LF</p> <p>CR</p> <p>LF</p> <p>NULL</p> <p><i>custom</i> - in this case, fill in the required delimiter in the next field (comma-separated hexadecimal values)</p>
------------------------	--

<b>Access over network</b>	Makes the serial console accessible over the TCP/IP network.
----------------------------	--



<i>Port</i>	TCP port where NETIO PowerPDU 4C listens for data for the Serial Console M2M API.
<i>Speed</i>	Baudrate (must match the baudrate configured at the remote device).

**AN18 Lua access to RS-232 serial port**



## 5.2.4 JSON API

The screenshot shows the 'JSON API' configuration page in the PowerPDU myNetio web interface. The page is titled 'JSON API' and is part of the 'M2M API Protocols' section. The configuration options include:

- Enable JSON API
- Use custom M2M HTTP(S) port
- Current M2M HTTP(S) port: 80
- Current M2M security protocol: HTTP
- Custom M2M HTTP(S) port: 81
- Enable READ-ONLY
  - Username:
  - Password:  [Show password](#)
- Enable READ-WRITE
  - Username: write
  - Password:  [Show password](#)

A green 'Save Changes' button is located below the configuration options. Below the configuration options, there is a section titled 'Test JSON API:' which includes a link to open the JSON API file (<http://192.168.120.58/netio.json>) and a button to 'Upload JSON file to the device'. There are also links to download example JSON API files: 'Set output 1 to ON', 'Set output 1 to OFF', and 'Toggle output 1'. A note at the bottom states: 'Note: This protocol does not take into account the output start interval - user is responsible for delays between output start. If you send ON action for all outputs together they will be turned on at the same time.'

Figure 17. Configuring the JSON API

<i>Enable JSON API</i>	Enables M2M JSON API functions in the system kernel.
<i>Use custom M2M HTTP(S) port</i>	Check to specify a specific port for M2M JSON API communication.
<i>Current M2M HTTP(S) port</i>	Read-only value. Indicates the port where the device currently listens for M2M JSON API commands.
<i>Current M2M security protocol</i>	Read-only value. Indicates the HTTP or HTTPS used for communication.
<i>Custom M2M HTTP(S) port</i>	Specific port for M2M JSON API only (to fill in the value, enable <i>Use custom port</i> first). Allowed range 1 - 65535. The device warns you if you specify a port number that is already in use. However, to be sure, we recommend to use port numbers above 1024.

<i>Enable READ-ONLY</i>	Enables <b>Read-Only</b> access via M2M JSON API for monitoring. You may also fill in the username and password for this mode. When left empty, the protocol will not require any authentication.
<i>Enable READ-WRITE</i>	Enables <b>Read/Write</b> access for monitoring and output control. You may also fill in the username and password for this mode. When left empty, the protocol will not require any authentication.
<i>Username</i>	Username for the respective access mode (Read-Only/ReadWrite). Note - this is unrelated to the username for accessing the NETIO 4x web administration interface.
<i>Password</i>	Password corresponding to the username (Read-Only/ReadWrite).
<i>Test JSON API</i>	You can use this section to test this M2M API

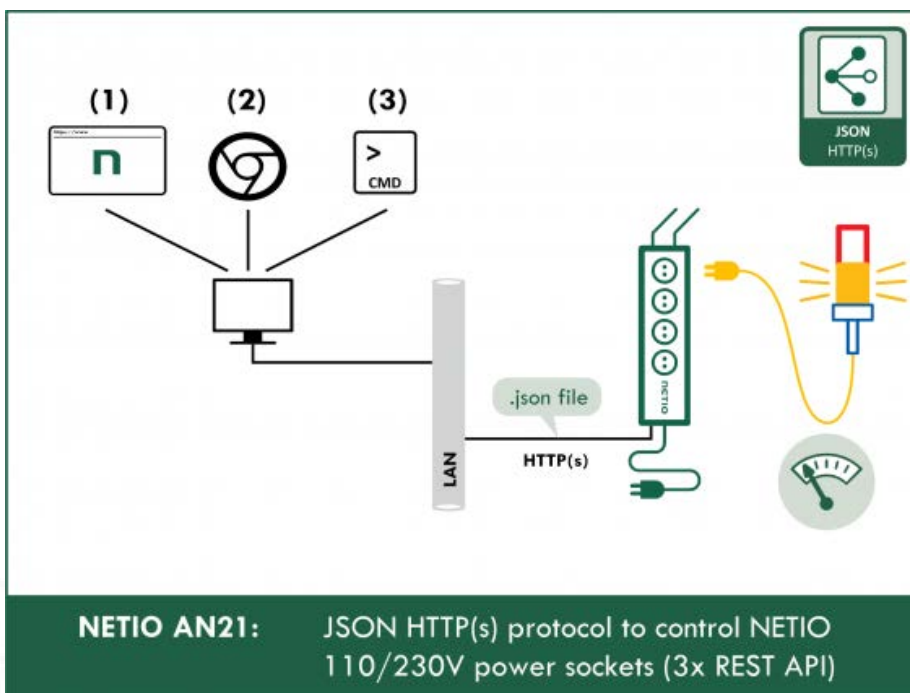
For an example of reading the state of an output using JSON API, click the **Test JSON API** link.

For more information about the M2M JSON API, visit the **Support > Download** section of our website and see the following document:

[JSON - description of NETIO M2M API interface - PDF](#)

For more information and a practical demonstration of using the JSON protocol with NETIO 4x smart sockets, see the following Application Note:

**AN21** [JSON HTTP\(S\) protocol to control NETIO 110/230V power sockets \(3x REST API\)](#)



## 5.2.5 XML API

The screenshot shows the NETIO web interface for configuring the XML API. The left sidebar contains navigation options: Outputs, M2M API Protocols (selected), Cloud, Users, Schedules, Actions (Lua), Settings, and Log. The main content area is titled 'XML API' and includes the following settings:

- Enable XML API
- Use custom M2M HTTP(S) port
  - Current M2M HTTP(S) port: 80
  - Current M2M security protocol: HTTP
  - Custom M2M HTTP(S) port: 81
- Enable READ-ONLY
  - Username:
  - Password:  [Show password](#)
- Enable READ-WRITE
  - Username: write
  - Password:  [Show password](#)

A green 'Save Changes' button is located below the READ-ONLY and READ-WRITE sections. Below the configuration fields is a 'Test XML API' section with the following links and buttons:

- Open XML API file (read password required): <http://192.168.120.58/netio.xml> [Download XML Schema \(XSD\)](#)
- Download example XML API file:
  - [Set output 1 to ON](#)
  - [Set output 1 to OFF](#)
  - [Toggle output 1](#)

A note at the bottom states: "Note: This protocol does not take into account the output start interval - user is responsible for delays between output start. If you send ON action for all outputs together they will be turned on at the same time." The footer of the interface shows 'User manual', '3.4.0 - 0.11 - 27 (ne94fd1a)', and 'NETIO\_products'.

Figure 18. Configuring the XML API

<b>Enable XML API</b>	Enables M2M XML API functions in the system kernel.
<b>Use custom M2M HTTP(S) port</b>	Check to specify a specific port for M2M XML API communication.
<b>Current M2M HTTP(S) port</b>	Read-only value. Indicates the port where the device currently listens for M2M XML API commands.
<b>Current M2M security protocol</b>	Read-only value. Indicates the HTTP or HTTPS used for communication.
<b>Custom M2M HTTP(S) port</b>	Specific port for M2M XML API only (to fill in the value, enable <i>Use custom port</i> first). Allowed range 1 - 65535. The device warns you if you specify a port number that is already in use. However, to be sure, we recommend to use port numbers above 1024.

<i>Enable READ-ONLY</i>	Enables <b>Read-Only</b> access via M2M XML API for monitoring. You may also fill in the username and password for this mode. When left empty, the protocol will not require any authentication.
<i>Enable READ-WRITE</i>	Enables <b>Read/Write</b> access for monitoring and output control. You may also fill in the username and password for this mode. When left empty, the protocol will not require any authentication.
<i>Username</i>	Username for the respective access mode (Read-Only/ReadWrite). <b>Note</b> - this is unrelated to the username for accessing the NETIO 4x web administration interface.
<i>Password</i>	Password corresponding to the username (Read-Only/ReadWrite).
<i>Test XML API</i>	You can use this section to test this M2M API

For an example of reading the state of an output using JSON API, click the **Test XML API** link.

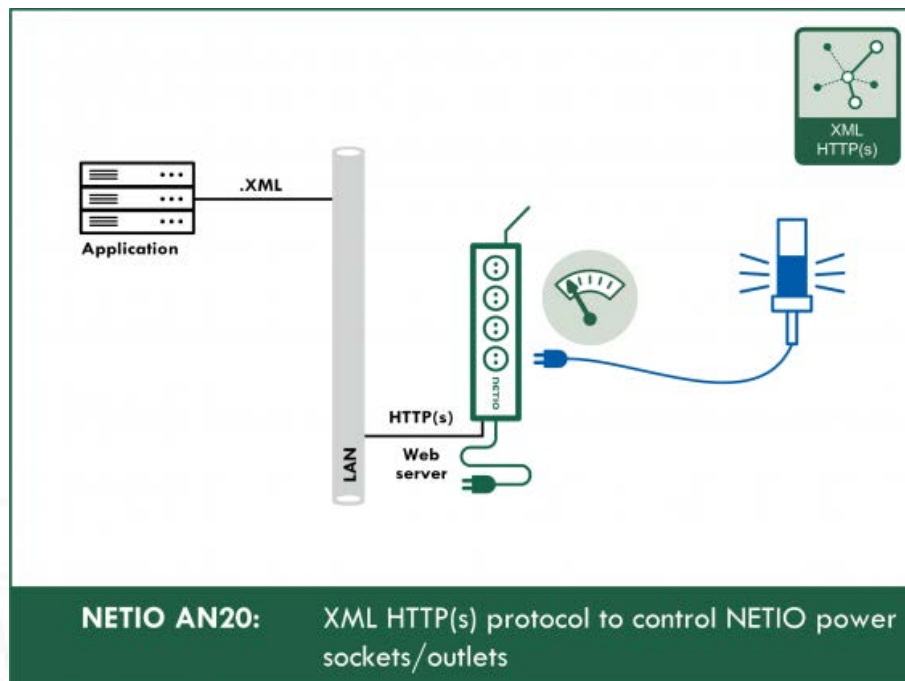
To download the XML style definition (XSD), click the **GET XML Schema (XSD)** button.

For more information about the M2M XML API, visit the **Support > Download** section of our website and see the following document:

[XML - description of NETIO M2M API interface - PDF](#)

For more information and a practical demonstration of using the XML protocol with NETIO 4x smart sockets, see the following Application Note:

[AN20 XML HTTP\(s\) protocol to control NETIO smart power sockets 110/230V](#)



**NETIO AN20:** XML HTTP(s) protocol to control NETIO power sockets/outlets

## 5.2.6 URL API

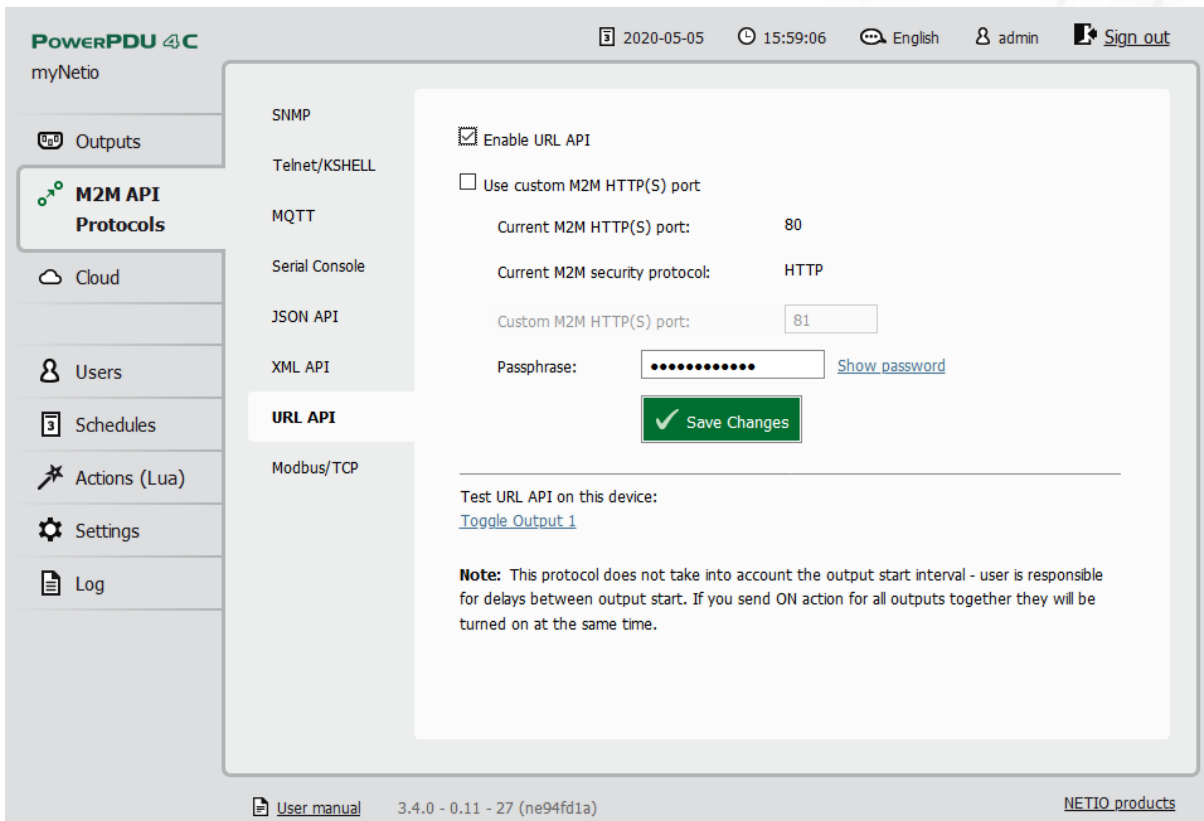


Figure 19. Configuring the URL API

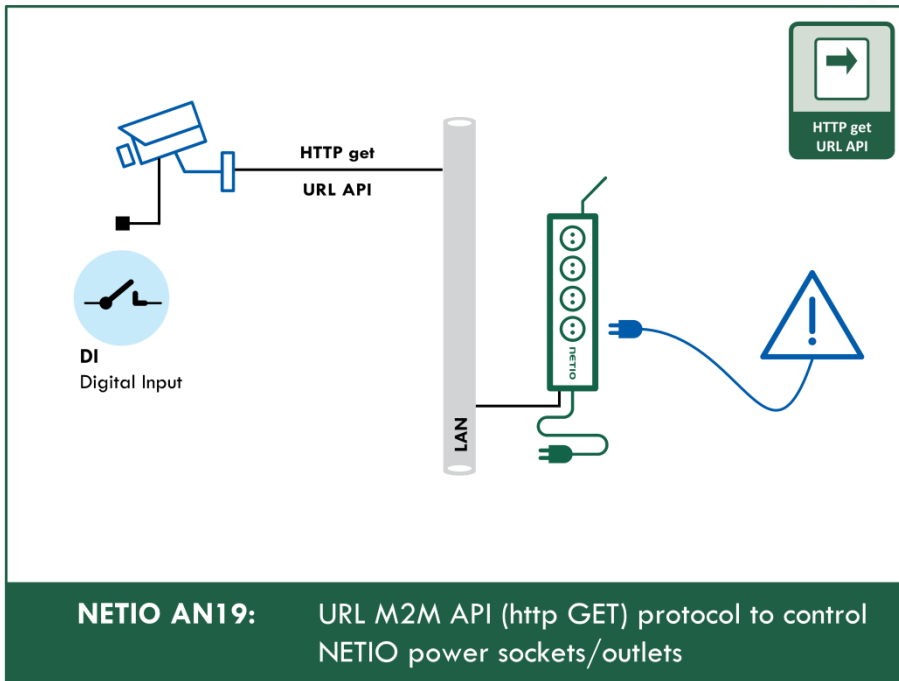
<b>Enable URL API</b>	Enables M2M URL API functions in the system kernel.
<b>Use custom M2M HTTP(S) port</b>	Check to specify a specific port for M2M URL API communication.
<b>Current M2M HTTP(S) port</b>	Read-only value. Indicates the port where the device currently listens for M2M URL API commands.
<b>Current M2M security protocol</b>	Read-only value. Indicates the HTTP or HTTPS used for communication.
<b>Custom M2M HTTP(S) port</b>	Specific port for M2M URL API only (to fill in the value, enable <i>Use custom port</i> first). Allowed range 1 – 65535. The device warns you if you specify a port number that is already in use. However, to be sure, we recommend to use port numbers above 1024.
<b>Passphrase</b>	Password to authenticate HTTP GET communication ( <b>pass</b> attribute in the request).

For an example of toggling output no. 1 using the M2M URL API, click the **Test URL API** link. This opens a new browser tab and invokes the following HTTP GET request:

```
http(s)://<netio.ip.address.here>/netio.cgi?pass=&output1=45
```

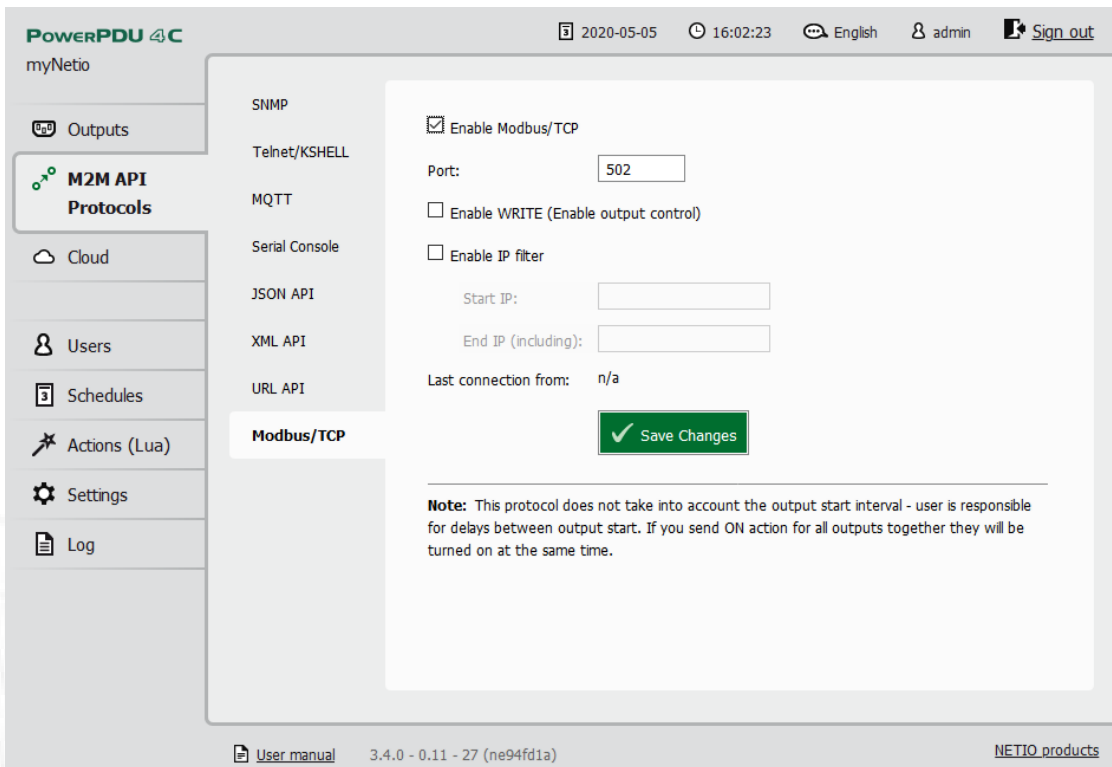
<sup>5</sup> Select http or https depending on whether you use [secure connection](#) (https) or standard unencrypted connection (http).

AN19 URL API (http GET) protocol to control NETIO 4x power sockets/outlets 110 / 230V



**NETIO AN19:** URL M2M API (http GET) protocol to control NETIO power sockets/outlets

5.2.7 Modbus/TCP



- In previous firmware versions (<3.0.0), the *output* parameter was called *outlet* - please take this change into account in your existing implementations.

Figure 20. Configuring the XML API

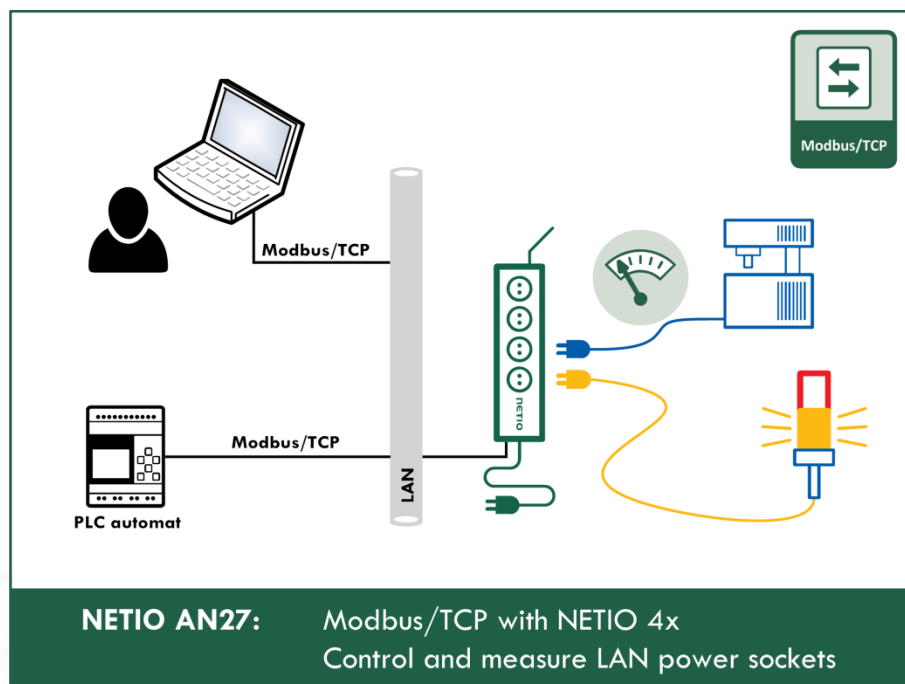
<i>Enable Modbus/TCP</i>	Enables M2M Modbus/TCP API functions in the system kernel.
<i>Port</i>	Specific port for Modbus/TCP only, range 1 - 65535. The device alerts you if you specify a port number that is already occupied. However, to be sure, we recommend using port numbers above 1024.
<i>Enable WRITE (Enable output control)</i>	Enables also output control function.
<i>Enable IP filter</i>	To improve security, the IP filter can be used to specify a range of IP addresses from which Modbus/TCP commands are accepted. Commands from addresses outside of this range will be ignored.
<i>Last connection from</i>	Read-only value indicating the IP address from which the last Modbus/TCP command was received. The "Refresh" button updates this value.

For more information about the M2M Modbus/TCP API, visit the **Support > Download** section of our website and see the following document:

[Modbus / TCP - description of NETIO M2M API interface - PDF](#)

For more information and a practical demonstration of using the XML protocol with NETIO 4x smart sockets, see the following Application Note:

[AN27 Modbus/TCP with NETIO 4x - Control and measure LAN power sockets](#)





## 5.3 Cloud

Netio Cloud is a service provided by NETIO Products a.s. and allow easy central remote control and monitoring of the NETIO devices. What can you do in NETIO Cloud?

### Output control

- On/Off switch.
- Reset button (Short Off for defined time).
- Show power consumption [kWh] per output (metered device only)

### Settings

- Output name can be modified.
- Outputs can be placed to any of groups.
- Short OFF (restart) interval for reset can be set up.

NETIO Cloud is a paid service, but the current customers will gain some free credits with each device added to their NETIO Cloud account.

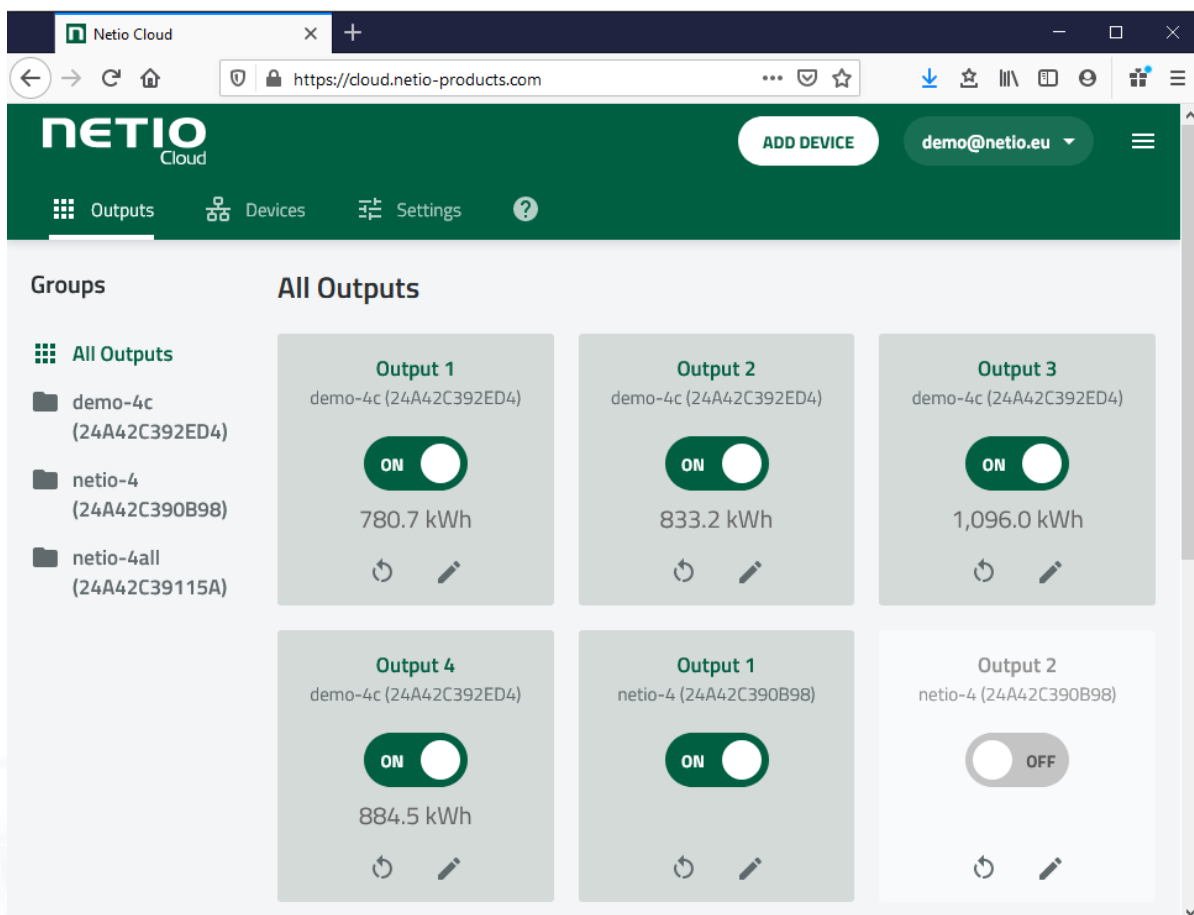


Figure 21 - NETIO Cloud

**Note:** Cloud is available only in Firmware 3.4.0 and newer!

Connection to NETIO Cloud can be configured on Cloud tab.

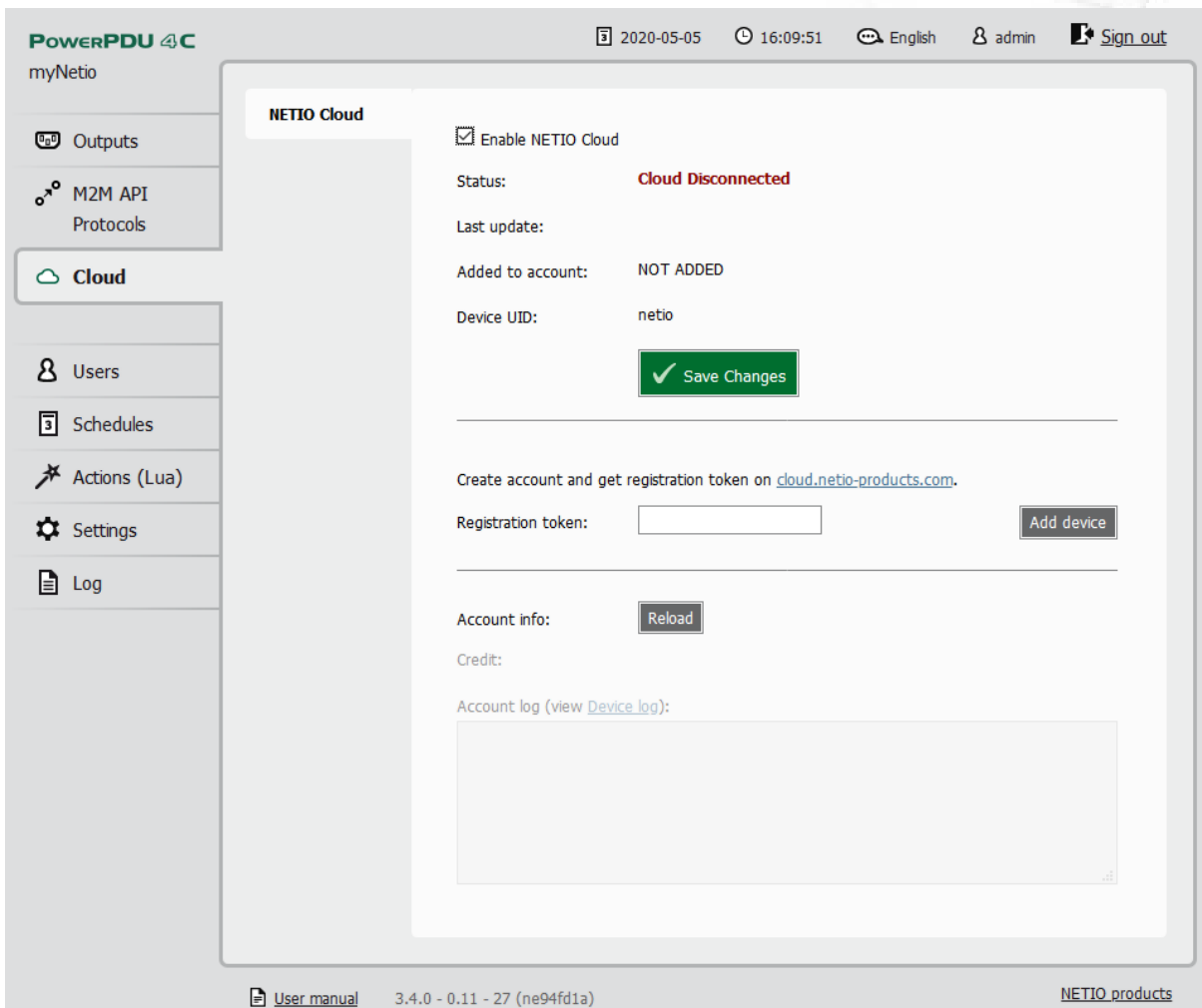


Figure 22. Configuring the Cloud

**Enable NETIO Cloud** Check to enable NETIO Cloud.

**Status**

**Disconnected:** Device not connected to Cloud

**Cloud connect failed...reconnecting:** Device it trying to connect to Cloud.

**Connected:** Device connected to NETIO Cloud.

**Connected and synchronised:** Device connected to NETIO Cloud and account information synchronized.

**Last update**

Date and time of the last update.

**Added to account**

Account name to what this device is assign.

**Device UID**

Unique device ID.

**Save Changes**

Saves the changes.

**Registration token**

Enter the registration token from NETIO Cloud web - shown when "ADD

DEVICE" button used. Then press "Add device" button.

<b>Remove device</b>	If this device is connected to NETIO Cloud use this button to remove/disconnect it from NETIO Cloud.
<b>Account info</b>	Use "Reload" button tu refresh account information.
<b>Credit</b>	Value of the available credit for Cloud account.
<b>Account log</b>	Log of the Cloud account events.

## 5.4 Users

When several users use the NETIO 4x device, it is advisable to assign them different user accounts with the necessary privileges. In the left menu, select **Users**. NETIO 4x distinguishes three basic user categories:

Figure 23. Adding a user

- Administrator**    User with full privileges.
- User**                User that can control the outputs but cannot change system settings.
- Guest**                User that cannot change any settings, may only monitor the current output states.



### Note

NETIO 4x supports up to 5 user accounts. A username must start with a letter and may only contain numbers and letters without accents.

Select one of the above options as required. For a more fine-grained assignment of privileges, click the **more** link to expand the list of privileges:

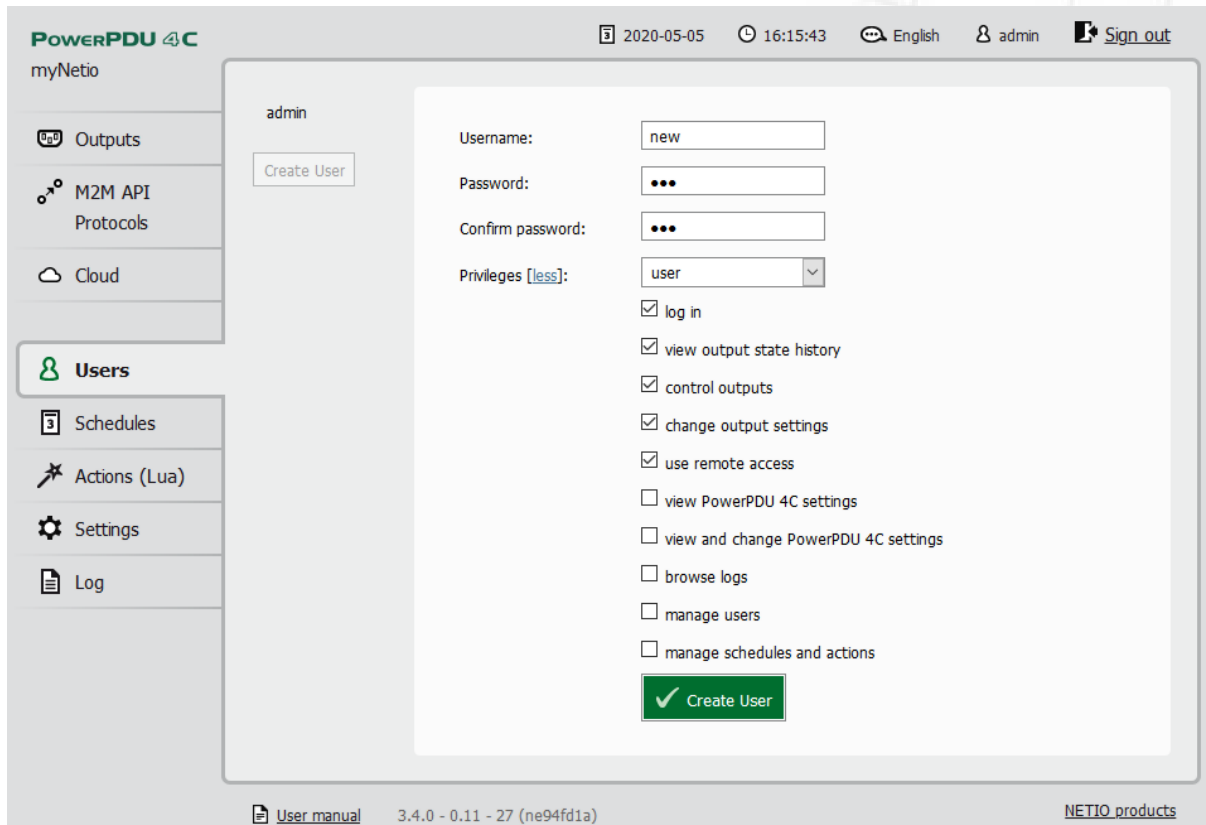


Figure 24. Detailed configuration of user privileges

Confirm by clicking Create user. In a similar way, existing user accounts can be edited.

## Custom privileges

log in	Allows logging in to the NETIO 4x web interface and the M2M Telnet/KSHELL API
control outputs	Allows controlling outputs over the web interface and the M2M Telnet/KSHELL API
change output settings	Allows modifying individual output settings (output name, short on/off delay etc.).
use remote access	Allows remote access
view netio4x settings	Allows viewing the configuration settings ( <i>Settings</i> menu)
view and change netio4x settings	Allows changing the configuration settings (items displayed after clicking the <i>Settings</i> menu - network settings, date&time and so on).
browse logs	Allows viewing the <i>Log</i>
manage users	Enables/disables the <i>Users</i> menu (creating/editing/deleting user accounts)
manage schedules and actions	Enables/disables the <i>Schedules</i> and <i>Actions</i> menu items

## 5.5 Schedules

Schedules can specify when should an output be switched on or when is an action valid. To manage schedules, select **Schedules** in the left menu.

By default, NETIO 4x contains three schedules: **Always**, **Weekend** and **Work days**. To create a new schedule, click **Create schedule**. Specify the schedule name and set the intervals when the output should be switched on.

### Note

For an output to be controlled according to the schedule, the schedule needs to be selected for that particular output at the **Timer** tab in the output configuration.

Figure 25. Adding a custom schedule

A new interval can be quickly created by marking the corresponding slots by clicking and dragging with the **left** mouse button. An existing interval can be shortened or extended by dragging either end of the interval. To delete an interval, click it with the **right** mouse button. To create an interval encompassing an entire day, click the box next to that day in the **All day** column. By clicking and dragging, intervals (including all-day ones) can be created for several days at the same time.

In this quick way, intervals can be created with a granularity of 10 minutes. To specify the times for an existing interval more precisely, click it with the left mouse button to open the start and end time settings. There it is possible to specify the time up to the second. To save the new values, click **OK**. To create a new interval by entering the precise times, click an empty space within the respective day with the **left** mouse button.

The screenshot shows the 'Schedules' management interface. A modal window titled 'Edit interval' is displayed, with the following fields and options:

- Name:** new
- From:** 08:30:10
- To:** 14:30:20
- Buttons:** OK, Cancel, Delete Interval

The background interface includes a sidebar with navigation options: Outputs, M2M API Protocols, Cloud, Users, Schedules (selected), Actions (Lua), Settings, and Log. The main area shows a calendar grid with columns for days of the week and time slots (3:00, 6:00, 9:00, 12:00, 15:00, 18:00, 21:00). Existing intervals are shown as green blocks on the grid.


Figure 26. Adding an interval with a precision to the second



#### Deleting a schedule

By deleting a schedule, all timers with this schedule are automatically disabled. The states of the respective outputs are unchanged; however, from that moment on, the outputs are under manual control.

## 5.6 Actions (Lua scripts)

Actions are used to create rules that automate NETIO 4x functions. To manage actions, select  **Actions** in the left menu. To add a new action, click **Create Rule**. In the top part, choose the **Trigger** to start the action, and the **Schedule** that determines when the trigger is active. To enable the action, check **Enabled**.

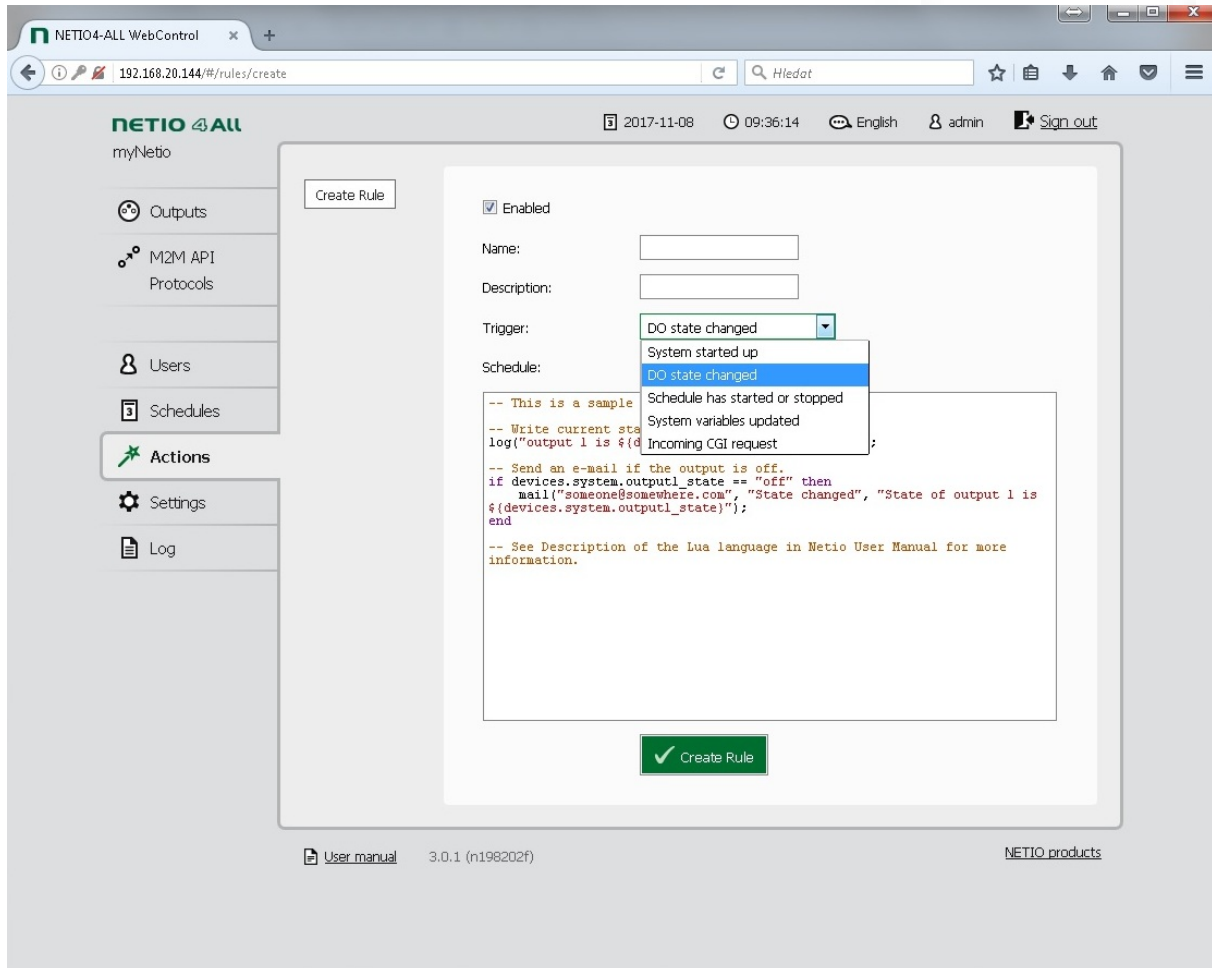


Figure 27. Setting the rule trigger

The trigger can be set to one of the following events:

<i>System started up</i>	The action is started after the system boots.
<i>DO state changed</i>	The action is started whenever the state of an output changes.
<i>Schedule has started or stopped</i>	The action is started whenever a schedule activates an output or another function.
<i>System variables updated</i>	The action is started at least once in every 2 seconds.
<i>Incoming CGI request</i>	The action is activated whenever the device receives a CGI request.



The action itself, i.e. what should be done when the trigger event occurs, is written as source code in the Lua programming language. Syntax highlighting is supported to make writing Lua code simpler and more user-friendly.

For a basic description of the Lua language and its syntax, see NETIO Lua Programming section at <https://wiki.netio-products.com>

The pre-filled action periodically activates and deactivates 2 outputs.

To activate this action, check *Enabled* at the top and restart your NETIO 4x device.

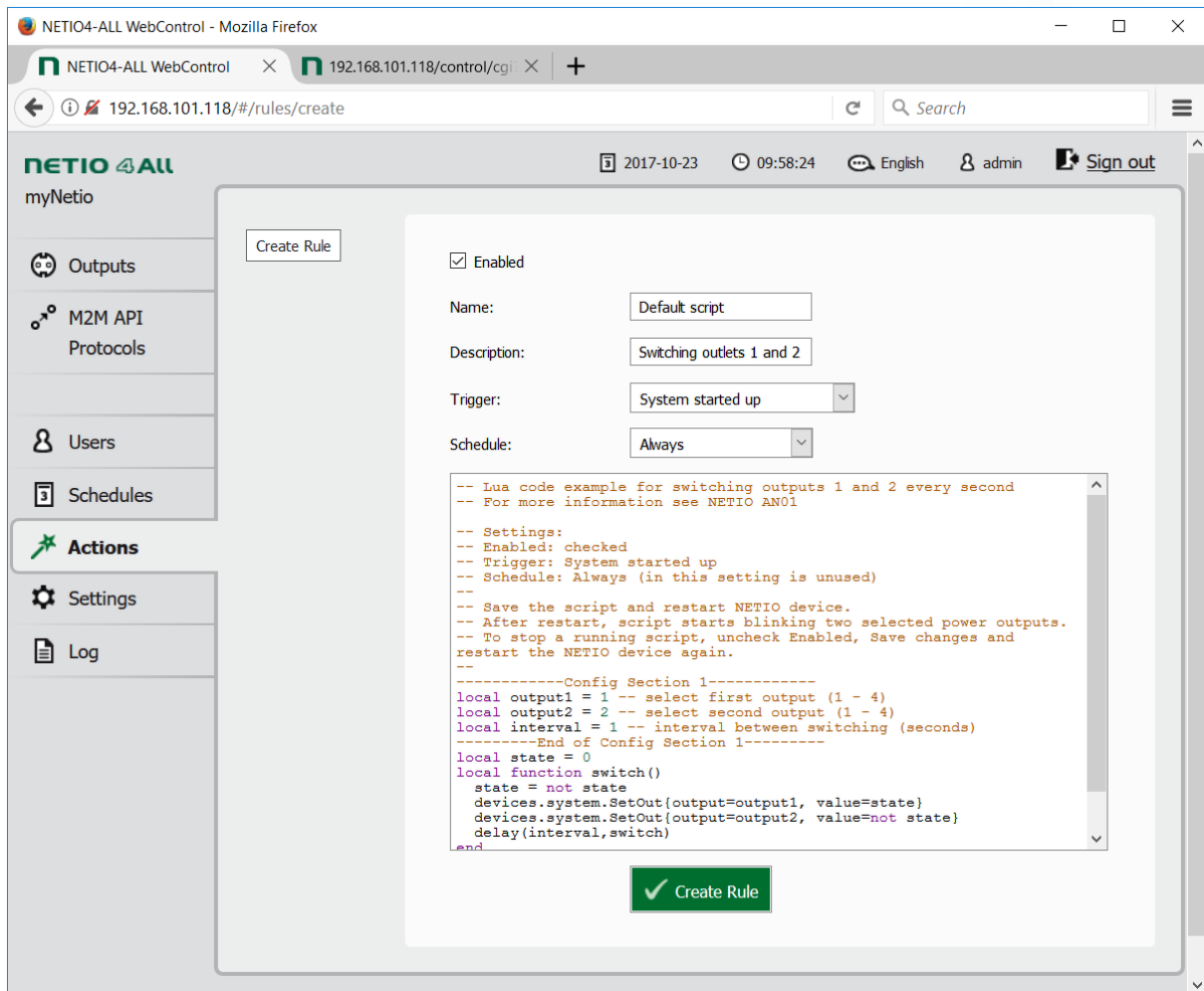


Figure 28. Activating the pre-filled action

#### Description of the pre-filled “Default script”:

Introductory comments (in Lua, comments start with double dashes: “--”)

```
-- Lua code example for switching outputs 1 and 2 every second
-- For more information see NETIO4x AN01

-- Settings:
-- Enabled: checked
-- Trigger: System started up
```

```
-- Schedule: Always (in this setting is unused)
--
-- Save the script and restart NETIO4x device.
-- After restart, script starts blinking two selected power outputs.
-- To stop a running script, uncheck Enabled, Save changes and restart the
NETIO4x device again.
```

### Configuration of the script:

```
-----Config Section 1-----
local output1 = 1
local output2 = 2
local interval = 1
-----End of Config Section 1-----
local state = 0
```

<b>output1</b>	Select the first NETIO 4x output to switch (possible values: 1, 2, 3 or 4)
<b>output2</b>	Select the second NETIO 4x output to switch (possible values: 1, 2, 3 or 4)
<b>interval</b>	Delay (in seconds) between the switching of the 1 <sup>st</sup> and the 2 <sup>nd</sup> output
<b>state</b>	Initial state of the variable controlling the flasher (0 - switched off)

### Definition of the function that actually switches the outputs (flashing):

```
local function switch()
  state = not state
  devices.system.SetOut{output=output1, value=state} --
  devices.system.SetOut{output=output2, value=not state}
  delay(interval,switch)
end
```

<b>function switch()</b>	Function definition/name
<b>state = not state</b>	Inverts the <b>state</b> variable (if the state was 0, after this command it will be 1 and vice versa)
<b>devices.system.SetOut{output,value}</b>	Sets the output state (socket ON/OFF)
<b>delay</b>	Invokes the same function after a delay

Main body of the Lua script:

```
log("Flasher started")
switch()
```

log()

Writes the "Flasher started" message to the system log

Switch()

First invocation of the switch function



### Terminating an active action

An active action cannot be terminated by simply unchecking *Enabled*; this only prevents future activations. If the action is already active and it is not programmed to stop by itself, NETIO 4x needs to be restarted, too.

The screenshot shows the NETIO 4 ALL WebControl interface. The main content area displays an event log with the following data:


Time	Type	Message
2017-11-08 10:02:08	INFO	outlet 1 was set to state off due to request of off, source actin
2017-11-08 10:02:08	INFO	outlet 1 was scheduled to change state to off due to request of off, source actin
2017-11-08 10:02:08	INFO	outlet 2 was set to state on due to request of on, source actin
2017-11-08 10:02:08	INFO	outlet 2 was scheduled to change state to on due to request of on, source actin
2017-11-08 10:02:07	INFO	outlet 1 was set to state on due to request of on, source actin
2017-11-08 10:02:07	INFO	outlet 1 was scheduled to change state to on due to request of on, source actin
2017-11-08 10:02:07	INFO	outlet 2 was set to state off due to request of off, source actin
2017-11-08 10:02:07	INFO	outlet 2 was scheduled to change state to off due to request of off, source actin
2017-11-08 10:02:06	INFO	outlet 1 was set to state off due to request of off, source actin
2017-11-08 10:02:06	INFO	outlet 1 was scheduled to change state to off due to request of off, source actin
2017-11-08 10:02:06	INFO	outlet 2 request of on succeeded with no change, source actin
2017-11-08 10:02:06	INFO	outlet 3 request of off succeeded with no change, source schedule
2017-11-08 10:02:06	INFO	Flasher started
2017-11-08 10:02:05	INFO	Application started, version: 3.0.1 nextgen1 (rev.n198202f/2017-10-23@netio4-all)
2017-11-08 10:01:34	INFO	outlet 2 was set to state on due to request of on, source system
2017-11-08 10:01:32	INFO	outlet 4 request of off succeeded with no change, source system
2017-11-08 10:01:32	INFO	outlet 2 was scheduled to change state to on due to request of on, source system
2017-11-08 10:01:32	INFO	outlet 1 was set to state on due to request of on, source system
2017-11-08 10:01:32	INFO	outlet 1 was scheduled to change state to on due to request of on, source system
2017-11-08 09:25:20	INFO	outlet 3 was set to state off due to request of off, source schedule
2017-11-08 09:25:20	INFO	outlet 3 was scheduled to change state to off due to request of off, source schedule

Below the log table are buttons for 'Refresh', 'Export to file', and 'Clear log'. At the bottom of the interface, there are links for 'User manual 3.0.1 (n198202f)' and 'NETIO products'.


Figure 29. Event log containing information from the action


For more examples of Lua actions, see <http://www.netio-products.com/en/lua-scripts> and the [Application notes](#) section.

## 5.7 Settings

To ensure correct operation as intended, the device settings need to be properly configured. Select  **Settings** in the left menu to display a sub-menu with the product settings.

### 5.7.1 Network mode

 Does not apply to NETIO PowerPDU 4C.

To configure the network interfaces of NETIO 4ALL and NETIO 4, first click  **Network Mode** in the Settings menu. Four network modes are available, depending on the desired use of the Ethernet and/or Wi-Fi network interface. Depending on the chosen network mode, you will be directed to set the corresponding parameters for the primary network interface and, if necessary, for Wi-Fi. In order to correctly configure the chosen network mode, all required parameters need to be set.

A warning about a possible connection loss is displayed before a network mode change:

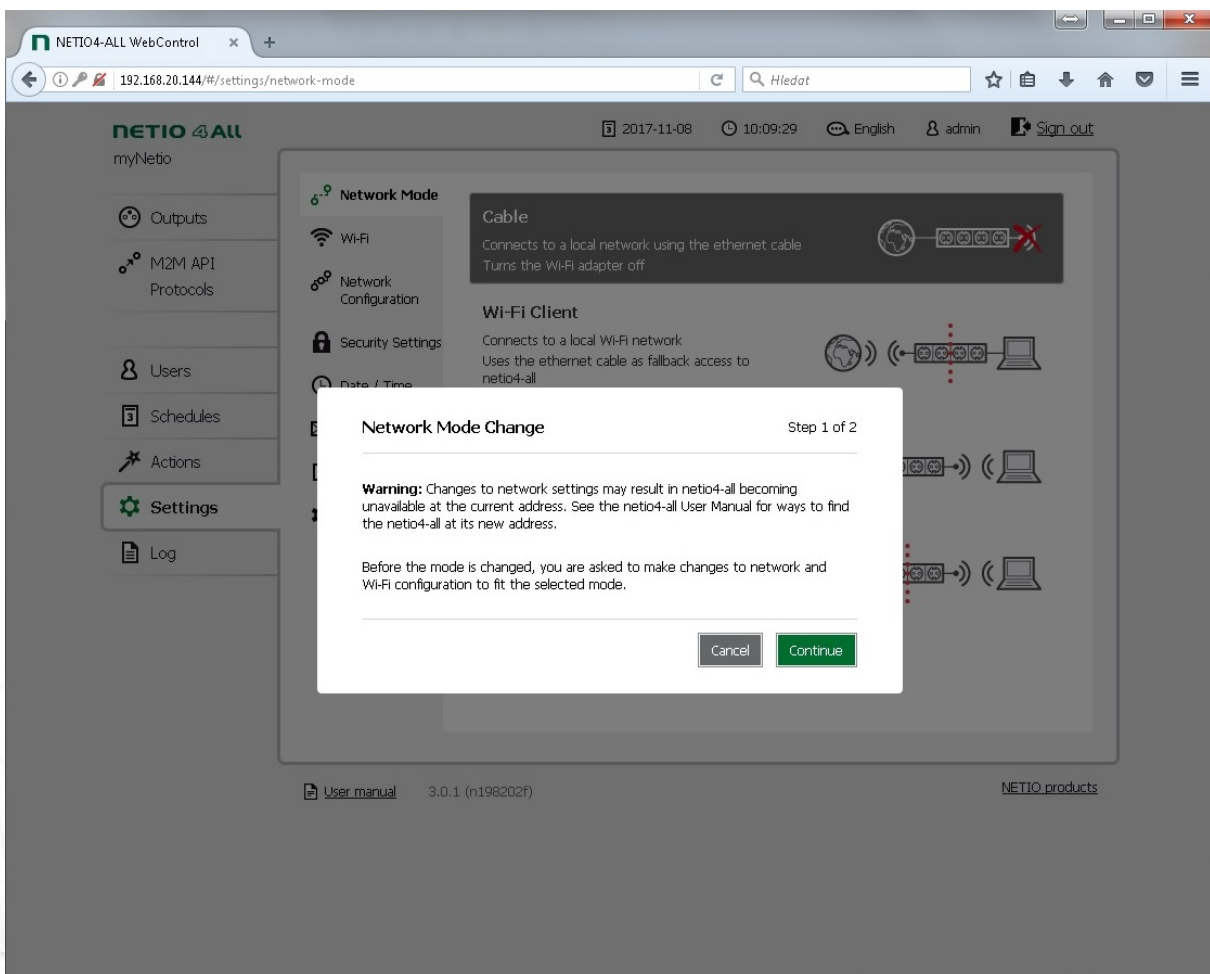


Figure 30. Network mode change warning

If you do not want to use the Wi-Fi interface of your NETIO 4/4All at all, choose the **Cable** mode. In this mode, NETIO 4/4All is connected via the Ethernet (as the primary network interface) and Wi-Fi is switched off.

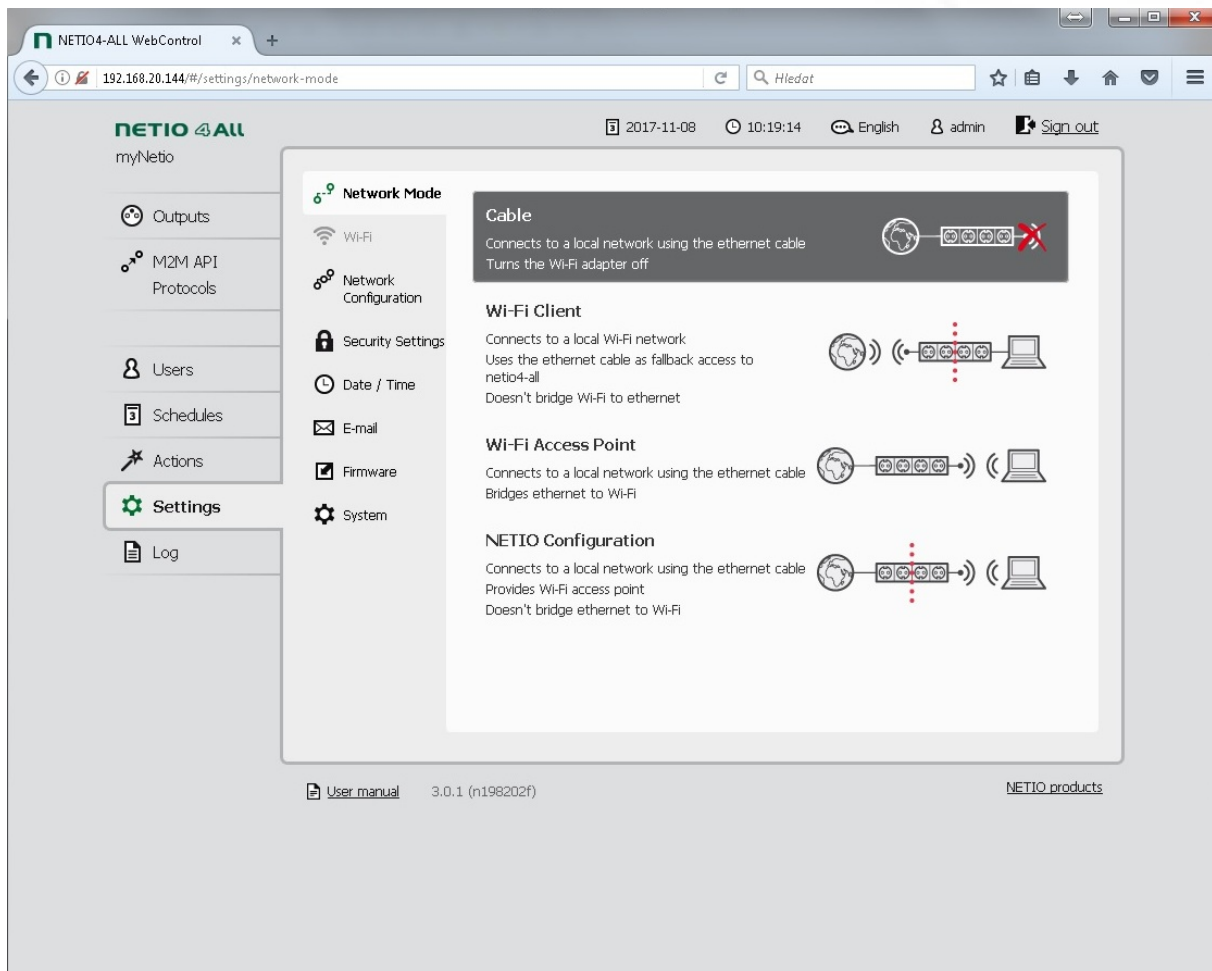


Figure 31. Configuring the Cable network mode

If you prefer to access NETIO 4/4All via a local Wi-Fi network, choose the **Wi-Fi Client** mode. In this mode, NETIO 4/4All connects over Wi-Fi (as the primary network interface) to the Wi-Fi access point of your network. Ethernet (as the secondary network interface) is assigned a separate network address range and only serves as a back-up method for accessing your device. The IP address at the Ethernet interface will be 192.168.2.78, and a DHCP server is active at this interface and assigning addresses from the 192.168.2.0/24 range.

In the Wi-Fi Client mode, NETIO 4/4All should NOT be simultaneously connected to a network with another active primary DHCP server. This will avoid a situation where other network devices would receive IP addresses from NETIO 4/4All (if your primary DHCP server works with a different IP range, such devices could become inaccessible from other network elements).

There is no routing or bridging between the Ethernet and Wi-Fi interfaces. Any device connected to the Ethernet interface of your NETIO 4/4All is not accessible via the Wi-Fi network to which your NETIO 4/4All is connected.

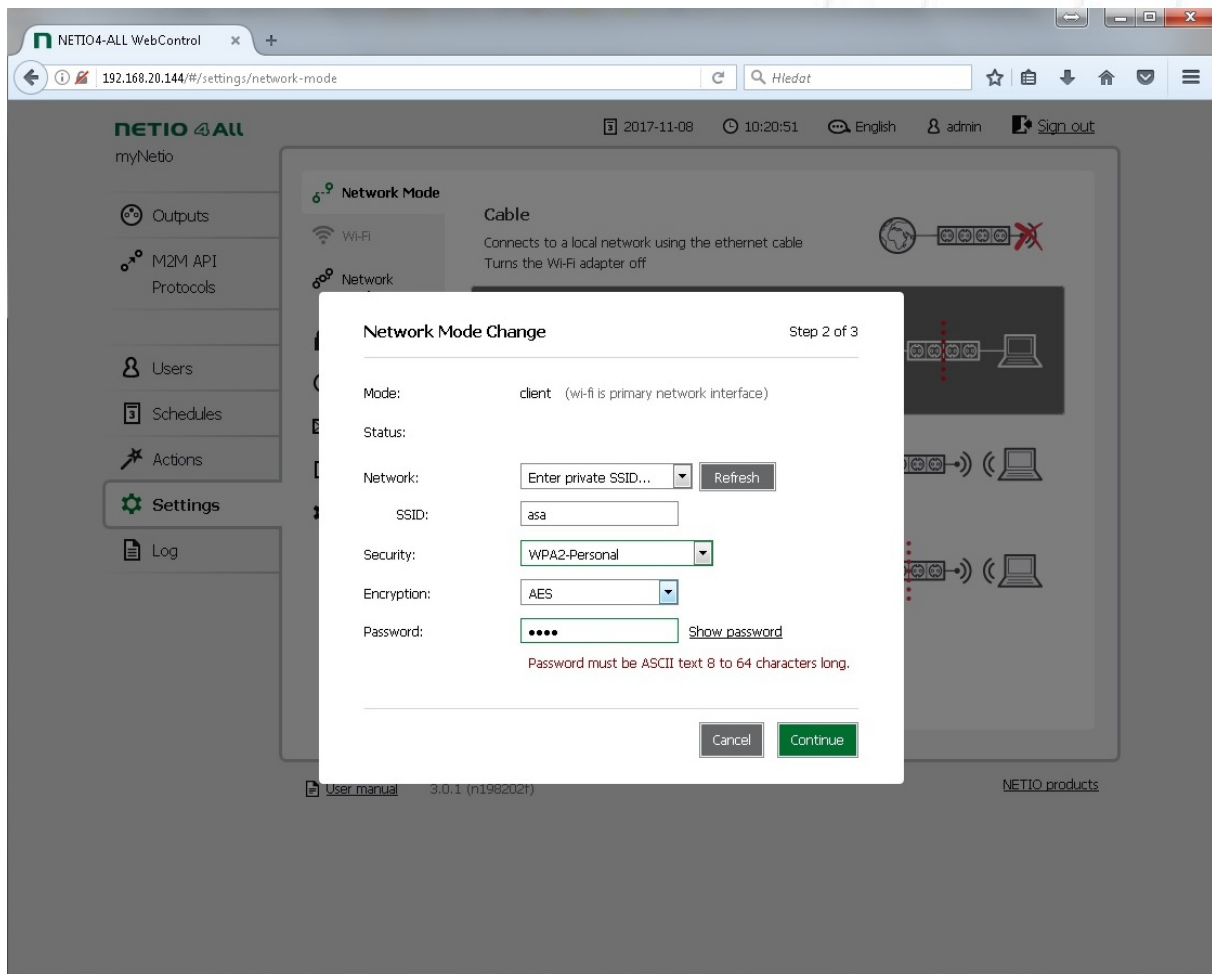


Figure 32. Selecting a Wi-Fi network in the Wi-Fi Client mode

In this mode, a dialog with Wi-Fi configuration appears. Select your network from the list of available networks and fill in the password. If your network is currently inactive (undetected), fill in the SSID, select the security and encryption mode and fill in the password. After the configuration is finished, these settings are accessible in the Settings/Wi-Fi submenu. For a detailed description of these parameters, see the [Settings/Wi-Fi settings](#) section.

After confirming the Wi-Fi network selection, the IP parameters need to be configured:

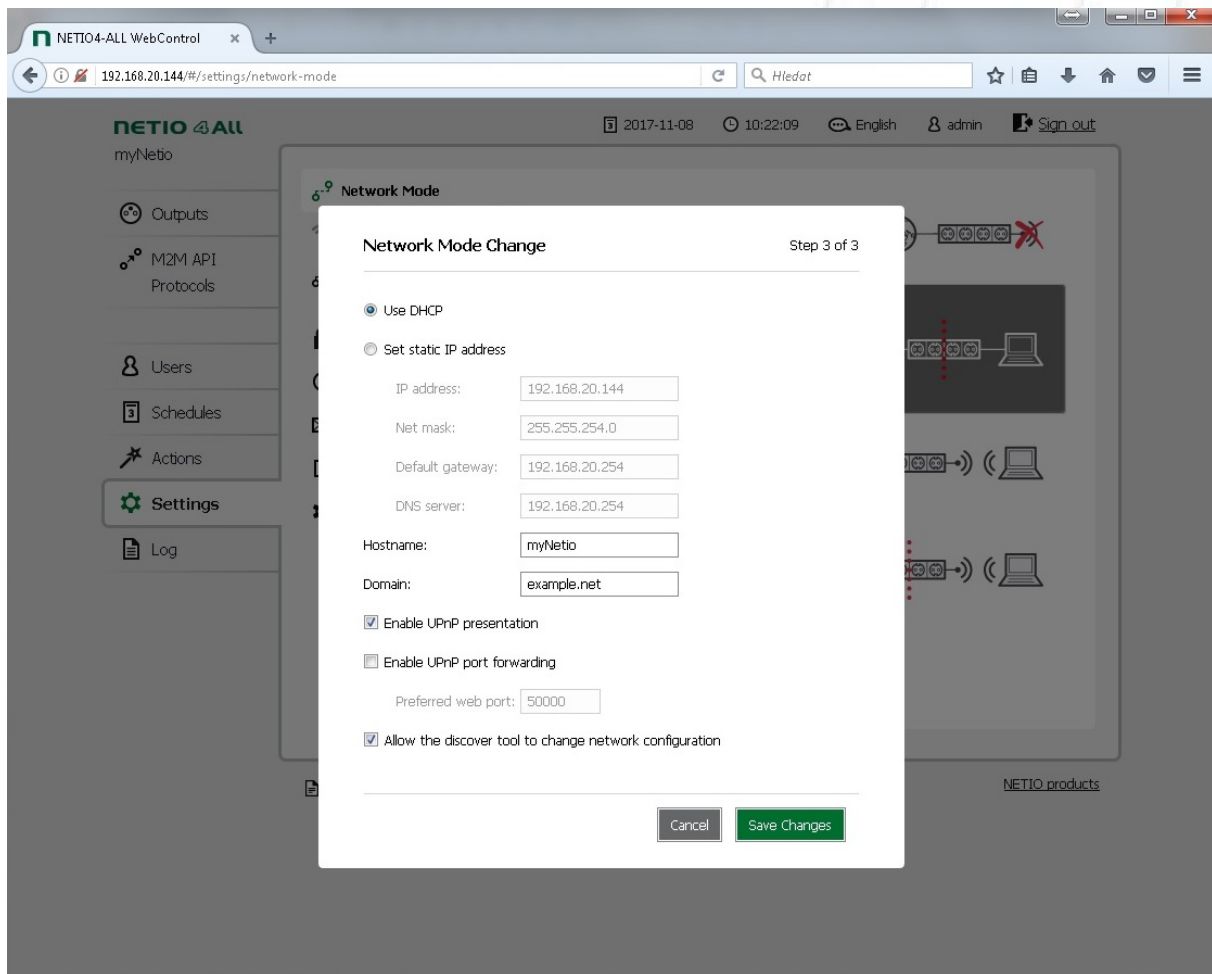


Figure 33. Configuring the Wi-Fi network IP parameters in the Wi-Fi Client mode

Select DHCP, if it is available in your network. Otherwise, set an IP address manually. After the configuration is finished, these settings are accessible in the Settings/Network Configuration submenu. For a detailed description of these parameters, see the [Settings/Network configuration](#) section.



#### Problems with saving the configuration?

If you have problems saving the configuration and continuing to the next step, make sure that your Wi-Fi access point is configured with WPA2-personal security and AES or TKIP encryption.



#### Caution

After changing the network configuration, it may be necessary to re-discover NETIO 4/4All at its new address. The discovery procedure is described in the [Detecting and configuring the IP address](#) section.

In the *Wi-Fi Access point* mode, your NETIO 4/4All is connected via the Ethernet (primary network interface) just as in the Cable mode. However, it also serves as a Wi-Fi access point. Other devices can connect to your NETIO 4/4All over Wi-Fi (secondary network interface) and gain access to the Ethernet network (bridging between Ethernet and Wi-Fi takes place). In this mode, NETIO 4/4All does NOT provide DHCP on any network interface.

After selecting this mode, the IP parameters need to be configured. NETIO 4/4All will be accessible at this address from both the Ethernet and the Wi-Fi network. After the configuration is finished, these settings are accessible in the Settings/Network Configuration submenu. For a detailed description of these parameters, see the [Settings/Network configuration](#) section.

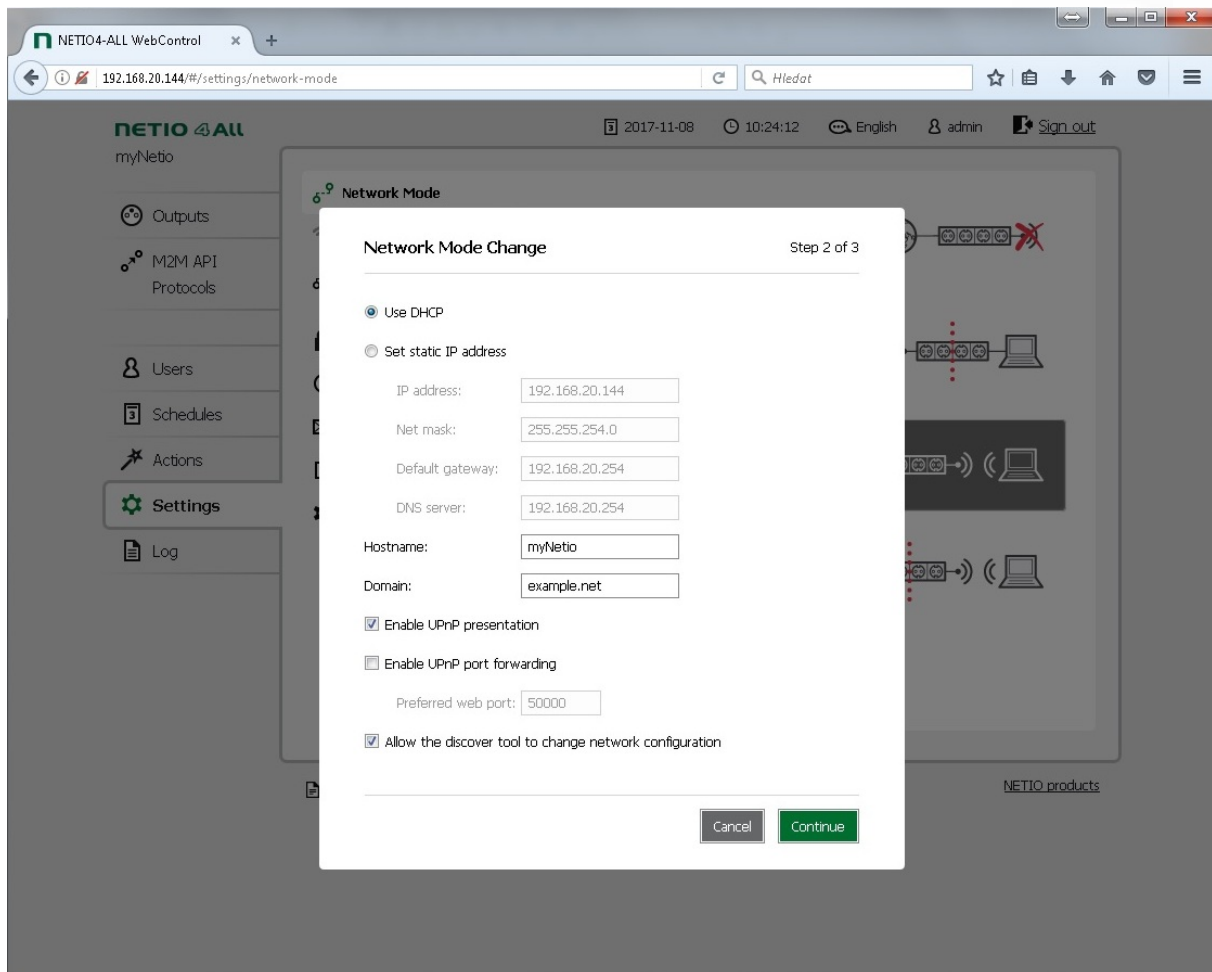


Figure 34. Configuring the Wi-Fi network IP parameters in the Wi-Fi Access Point mode

After saving the IP configuration, the parameters for the Wi-Fi Access Point mode need to be set.

After the configuration is finished, these settings are accessible in the Settings/Wi-Fi submenu. For a detailed description of these parameters, see the [Settings/Wi-Fi settings](#) section.



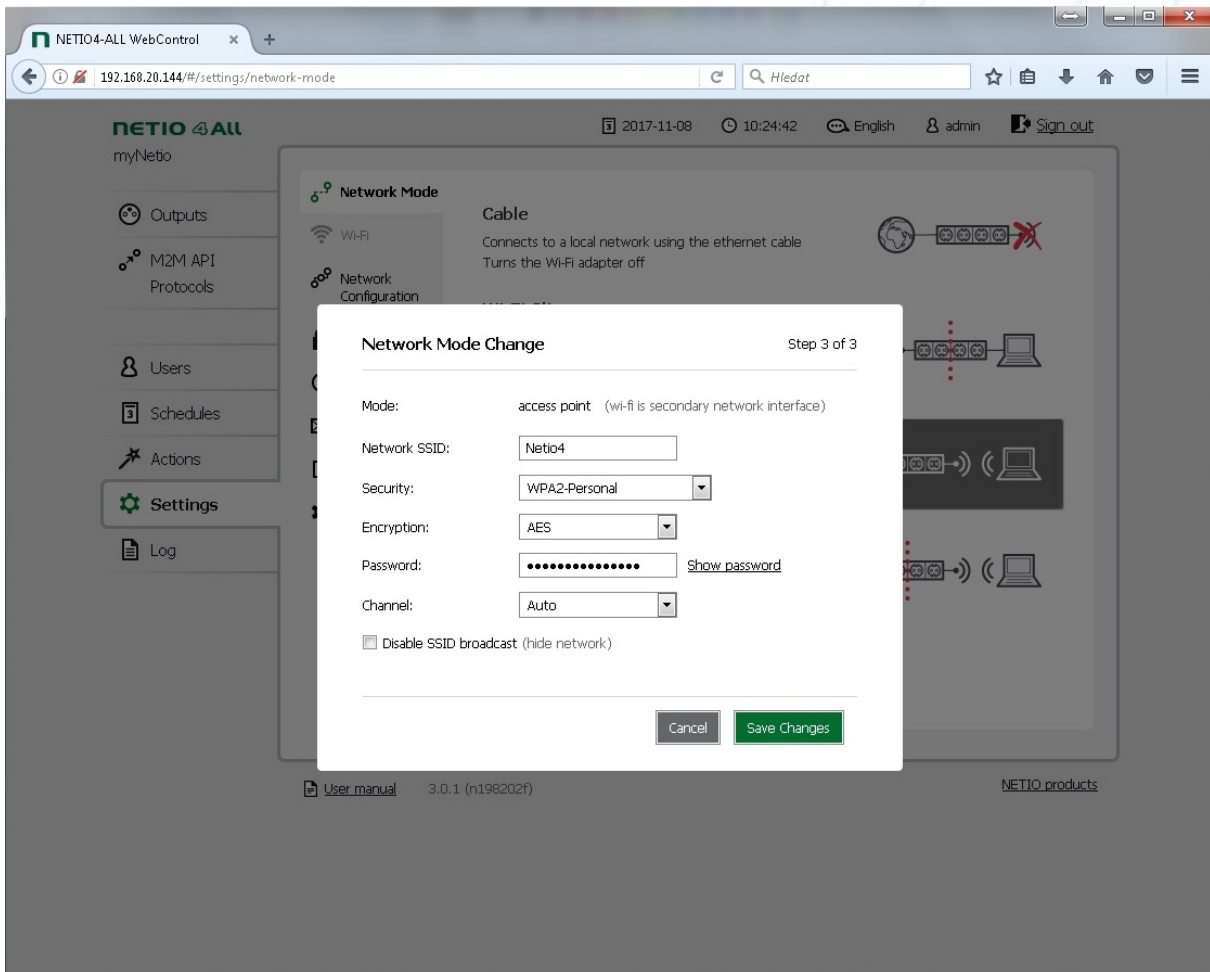


Figure 35. Configuring the Wi-Fi network parameters in the Wi-Fi Access Point mode




### Caution


After changing the network configuration, it may be necessary to re-discover NETIO 4/4All at its new address. The discovery procedure is described in the [Detecting and configuring the IP address](#) section.

The **NETIO Configuration** mode is similar to the Wi-Fi Access Point mode; however, Ethernet and Wi-Fi networks are isolated (no routing or bridging between Ethernet and Wi-Fi). The Wi-Fi interface has a static IP 192.168.2.78 and an active DHCP server assigning addresses from the 192.168.2.0/24 range.

The configuration takes place in the same way as in the previous mode.

## 5.7.2 Wi-Fi settings

 Does not apply to NETIO PowerPDU 4C.

In the Settings menu, select  **Wi-Fi**. Depending on the selected network mode (see [Settings/Network mode](#)), the Wi-Fi interface can be in the following modes:

**Inactive** - in the Cable network mode.

**Access point** - in the NETIO Configuration network mode (no bridging between Ethernet and Wi-Fi) and the **Wi-Fi Access Point** network mode (active bridging between Ethernet and Wi-Fi).

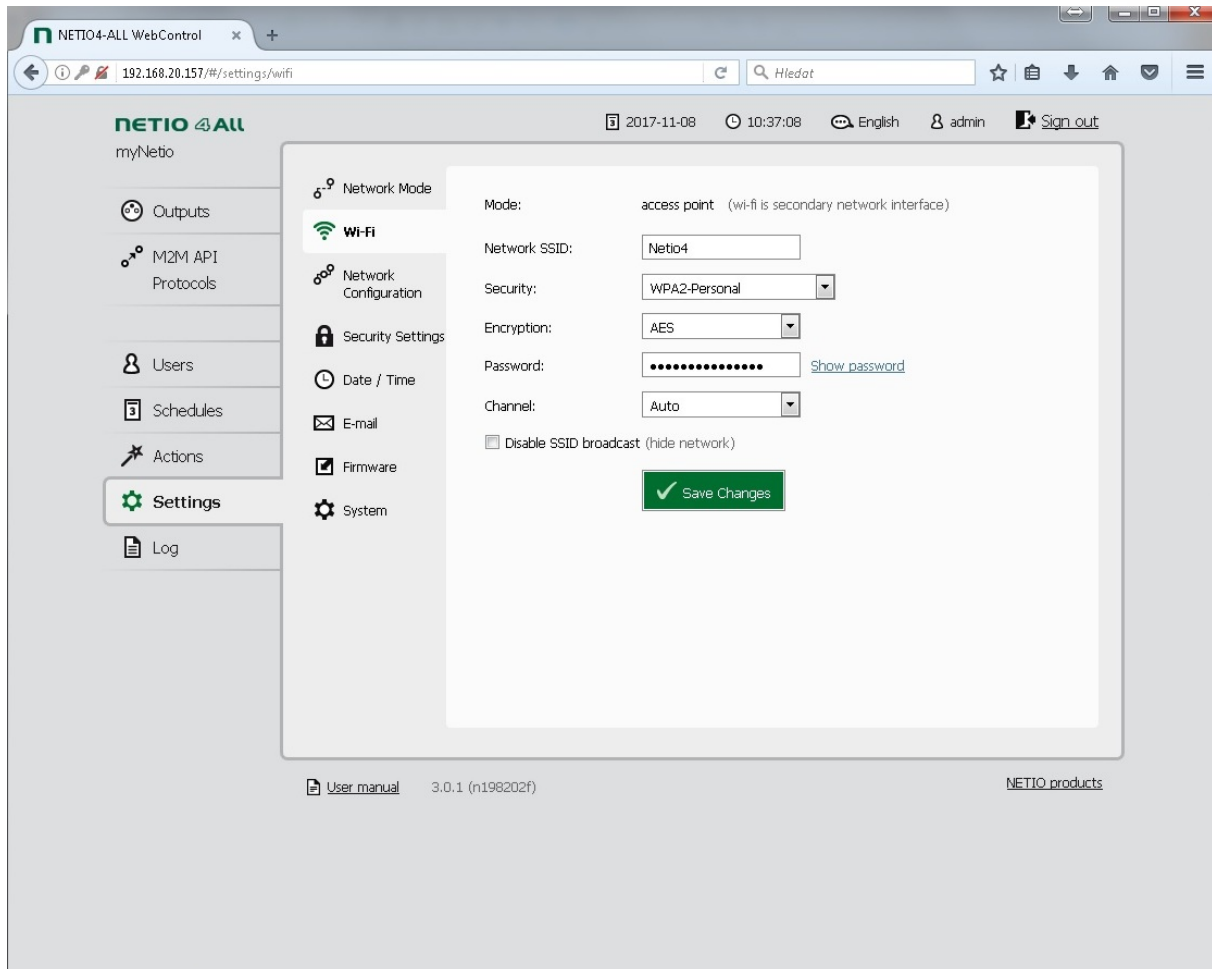


Figure 36. Wi-Fi configuration in the Access Point mode

<b>Mode</b>	Shows the Wi-Fi mode.
<b>Network SSID</b>	Enter the SSID identifier of your wireless network.
<b>Security</b>	Security mode of the wireless network, if any.
<b>Encryption</b>	Encryption method in the wireless network, if it is secured.
<b>Password</b>	Password for accessing the wireless network, if it is secured.

**Channel** Select the radio channel for your Wi-Fi network.

**Disable SSID broadcast** Only check if you want to hide your Wi-Fi network.

IP parameters of the interface are configured in the Settings/Network configuration menu.

In the Wi-Fi Client network mode, NETIO 4/4All uses its Wi-Fi interface to connect to a Wi-Fi network access point. In this mode, Ethernet is a back-up connection, without bridging to Wi-Fi.

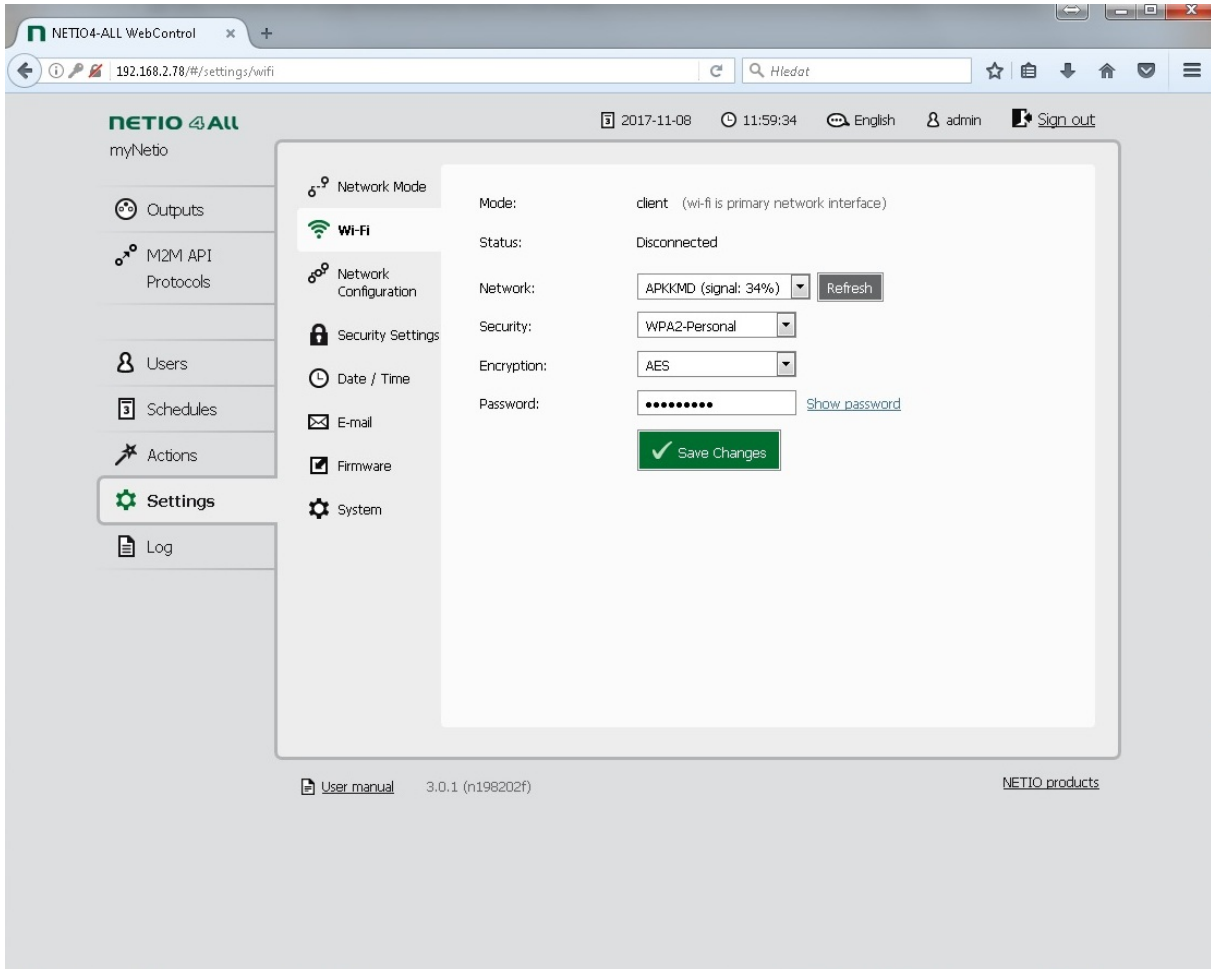


Figure 37. Wi-Fi configuration in the Wi-Fi Client mode

**Mode** Shows the Wi-Fi mode.

**Status** Shows whether or not NETIO 4/4All is connected to the selected network.

**Network** Select an existing network or enter the name of the network to connect to.

**Security** Security mode of the wireless network, if any.

**Encryption** Encryption method in the wireless network, if it is secured.

Password

Password for accessing the wireless network, if it is secured.


---



---

 IP parameters of the interface are configured in the Settings/Network configuration menu.

### 5.7.3 Network configuration

In the Settings menu on the left, select  **Network**. This menu allows configuring the IP parameters of the Ethernet and Wi-Fi interfaces; the parameters displayed depend on the selected network mode.

Network mode	Ethernet	Wi-Fi
Cable	Complete configuration accessible, default IP 192.168.1.78	Off
Wi-Fi client	Fixed IP 192.168.2.78 + DHCP server active at the interface	Complete configuration accessible, default IP 192.168.1.78
Wi-Fi Access Point	Complete configuration common to both interfaces is accessible, default IP 192.168.1.78	
NETIO Configuration (default)	Complete configuration accessible, default IP 192.168.1.78	Fixed IP 192.168.2.78 + DHCP server active at the interface

The simplest option is to **Use DHCP** — if a DHCP server is in your network, the device automatically sets its IP address and other parameters. However, to make sure that the IP address does not change, it may be more suitable to **Set static IP address** and the remaining parameters.

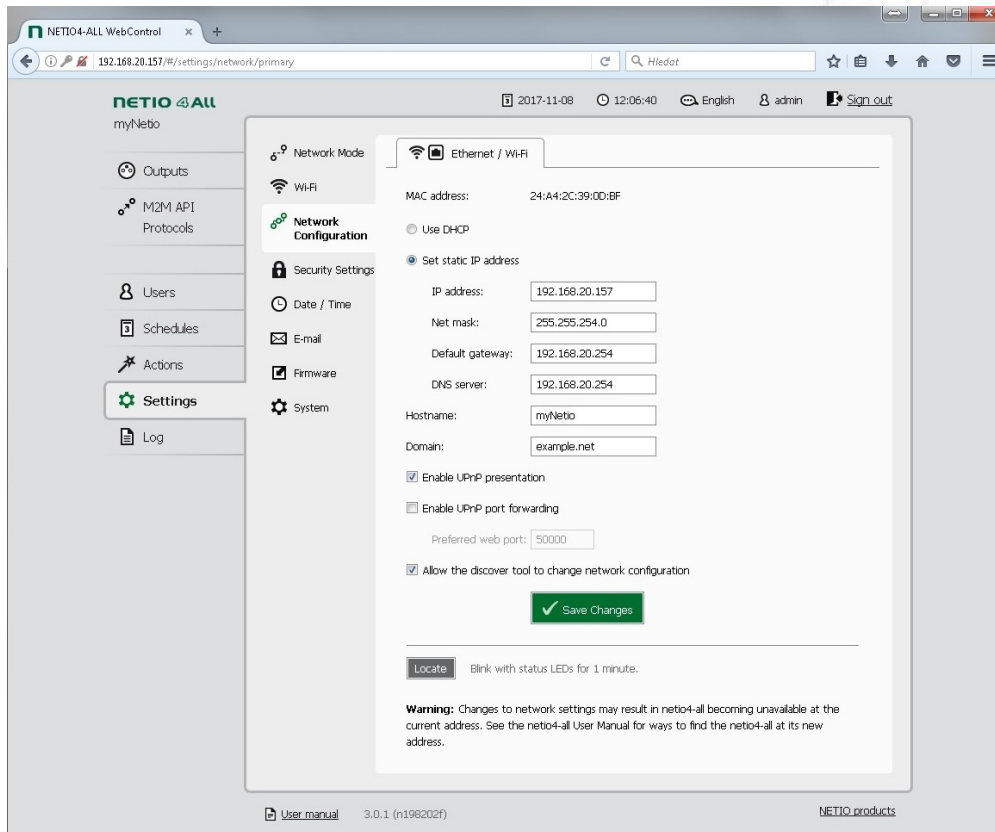


Figure 38. Network configuration

When setting the static IP address, use an IP address and a network mask corresponding to the network to which NETIO 4x is connected. As the gateway, use the Ethernet port of your router. If your router provides DNS service, use it as the DNS server, too. If unsure, use a public DNS server such as 8.8.8.8.

Pay attention to the domain configuration. Enter the *Hostname*, which identifies the device in your network, and your *Domain* name.

When **Enable UPnP presentation** is checked, the device can present itself in the network using UPnP protocols, and appear e.g. in the My Network Places folder in Windows. To enable remote access to your NETIO 4x, check **Enable UPnP port forwarding** and specify the preferred web port. After configuration changes are saved, a link with the remote access address appears next to this option.



#### Caution

For the remote access to work, your router must support UPnP protocols and the corresponding parameters need to be properly configured. If you are unsure with the settings, consult your network administrator or your Internet service provider. If UPnP does not work, it is also possible to statically forward an external port to the NETIO 4x http port; for details, consult the documentation for your router.


After discovering the device in the network and performing the initial network configuration, we recommend to disable the **Allow the discover tool to change network configuration** option. When the *Locate* button is clicked, the red LED of output no. 1 starts flashing, making it easier to physically identify the specific NETIO 4x device if you have more of them in your network.

**Caution**

After changing the network configuration, it may be necessary to re-discover NETIO 4x at its new address. The discovery procedure is described in section 2.1 Detecting and configuring the IP address.

Click **Save Changes** to save the settings.

## 5.7.4 Security settings

In the Settings menu on the left, select  **Security settings**. NETIO 4x supports secure connection over HTTPS. To use it, simply check **Turn on secure connection (HTTPS)** and confirm by clicking **Save Changes**. A self-signed certificate is generated and you will be automatically redirected to the device web interface. All active connections are terminated and re-established. It is also necessary to confirm a permanent security exception for the certificate in your browser.

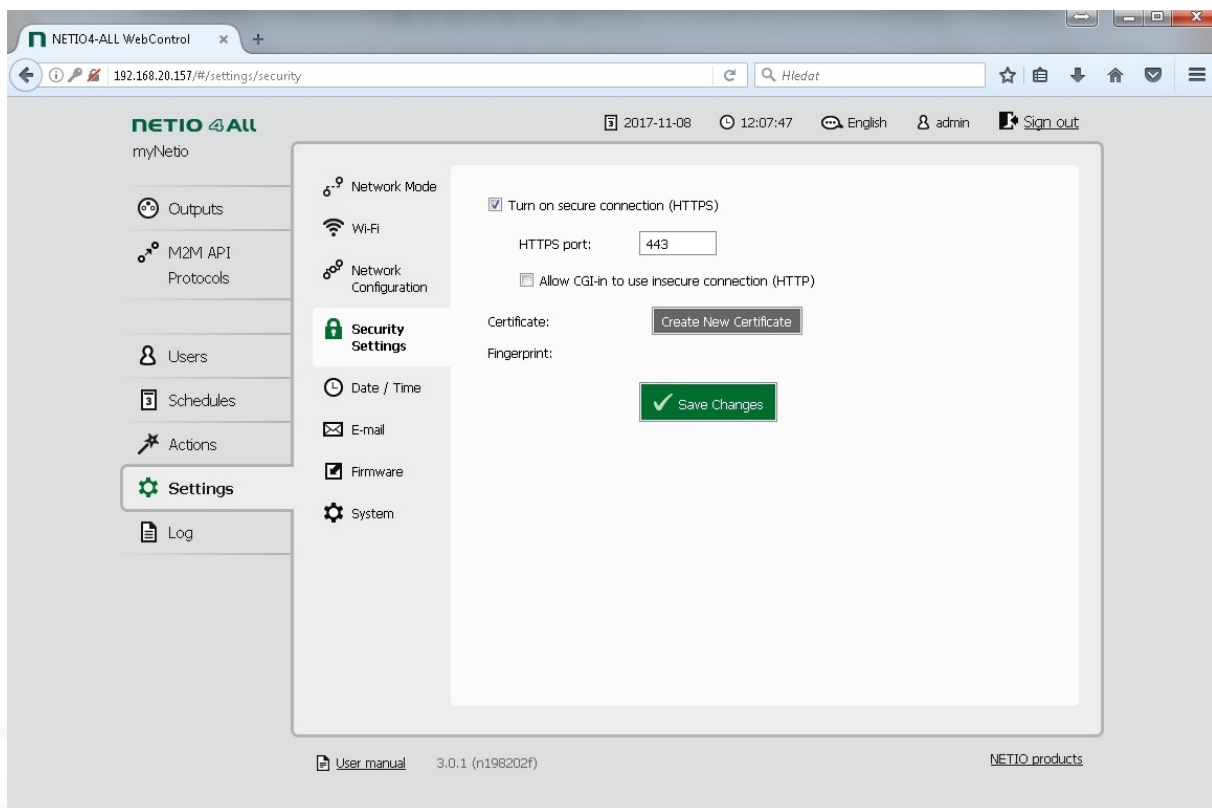


Figure 39. Security settings

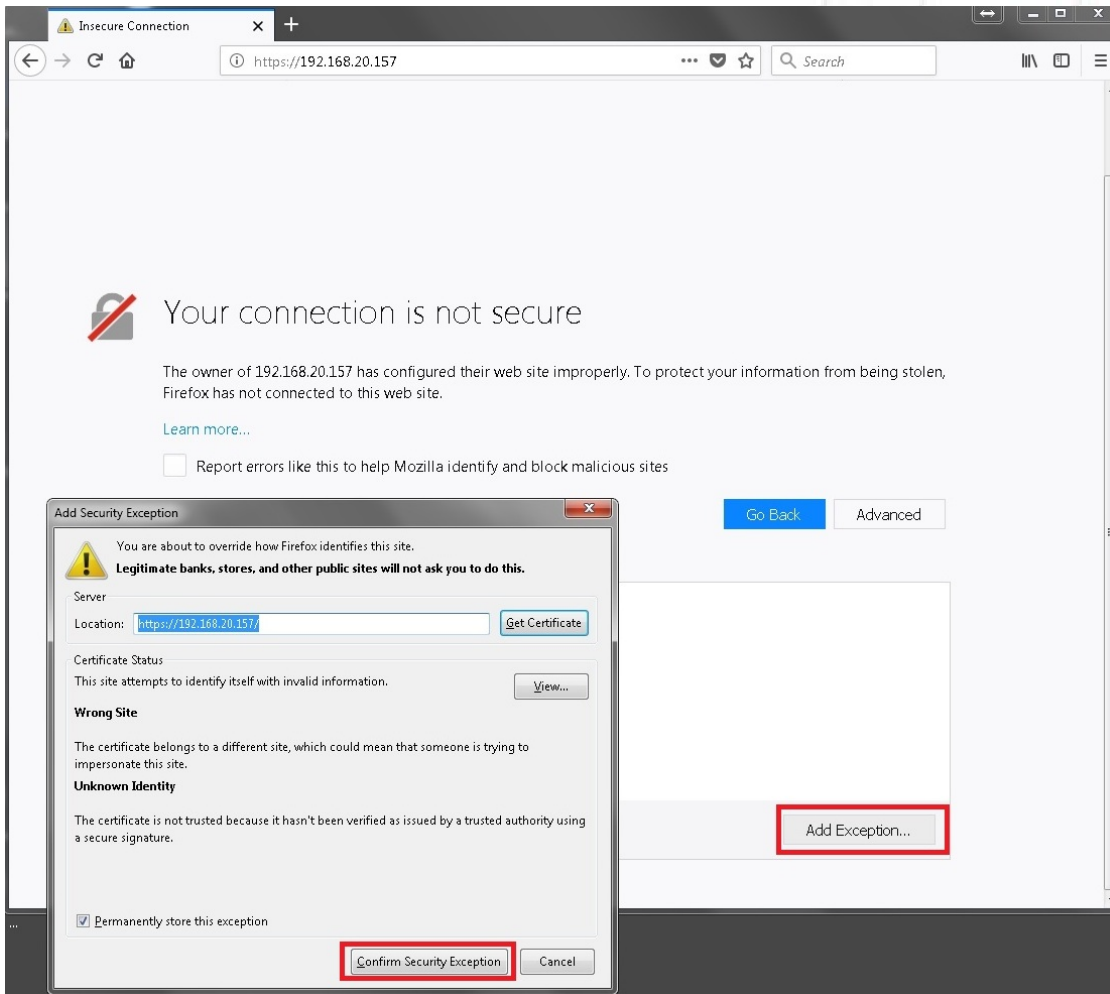



Figure 40. Confirming the security exception in your browser (Mozilla Firefox shown)

The menu displays information about the certificate validity and key fingerprint. If necessary, a new certificate can be created by clicking **Create New Certificate**. If you wish to keep using the insecure http protocol for incoming CGI requests for Lua actions, check **Allow CGI-in to use insecure connection (HTTP)**. In the settings, it is also possible to change the default HTTPS port to a different one. In that case, the port needs to be specified together with the IP address when connecting to NETIO 4x, e.g. `https://192.168.1.50:555` (where 555 is the new port).

## 5.7.5 Date / Time

In the Settings menu on the left, select  **Date / Time**. First, choose your city to specify the timezone where NETIO 4x is operated (may be different from the local time on the computer used to connect to NETIO 4x).

The device supports two methods for setting its clock. The time can be either specified manually or synchronized automatically with a NTP server. For the automatic synchronization, it is sufficient to specify the NTP server. If you are unsure, keep the default server, pool.ntp.org. When setting the date/time manually, enter the date in the YYYY-MM-DD format and the time in the HH:MM:SS format. You can also use the option to synchronize the time with your local computer.



### Note

If your NETIO 4x has Internet access when it is powered on for the first time, it automatically sets its clock to the current date and time in the GMT timezone. To display the correct time, this timezone needs to be changed to match your location.

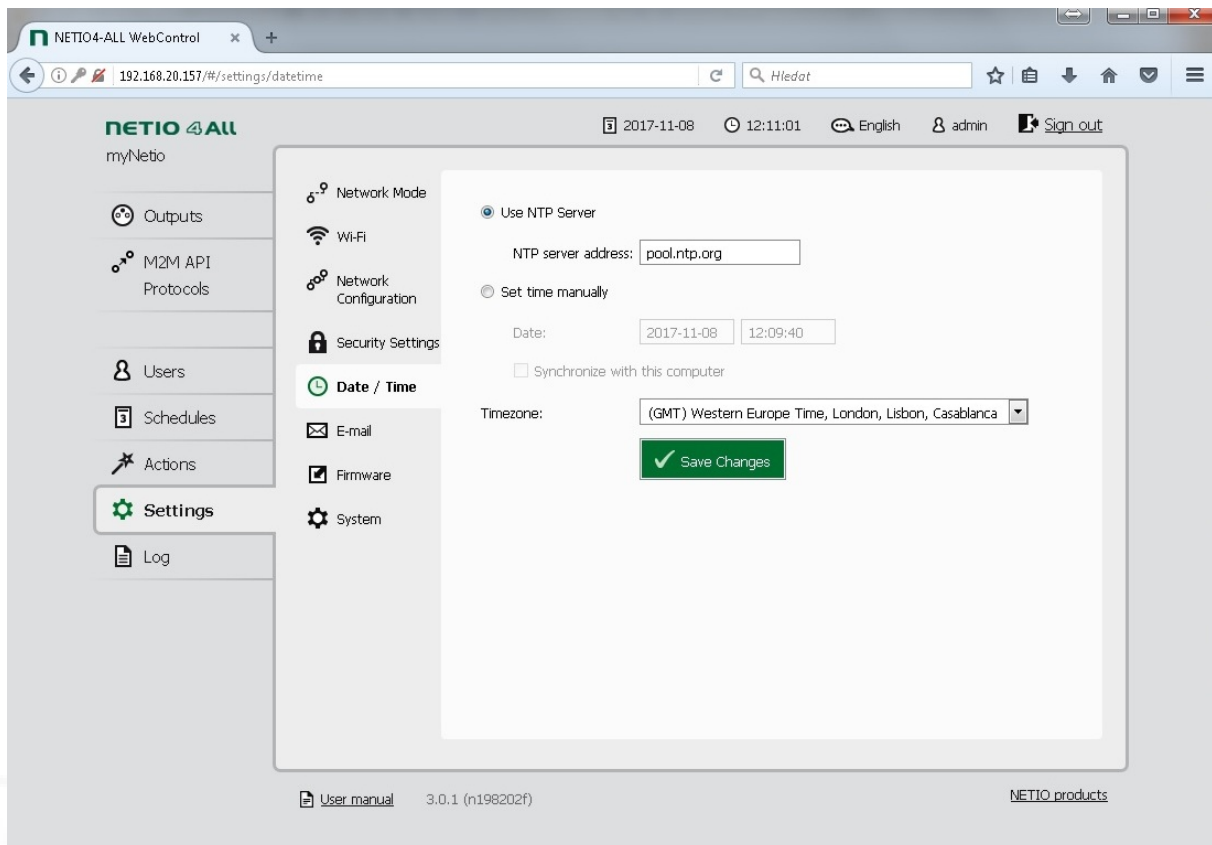



Figure 41. Date / time settings

Click **Save Changes** to save the settings.



## 5.7.6 E-mail

In the Settings menu on the left, select  E-mail. This setting applies to the sending of e-mail alerts.

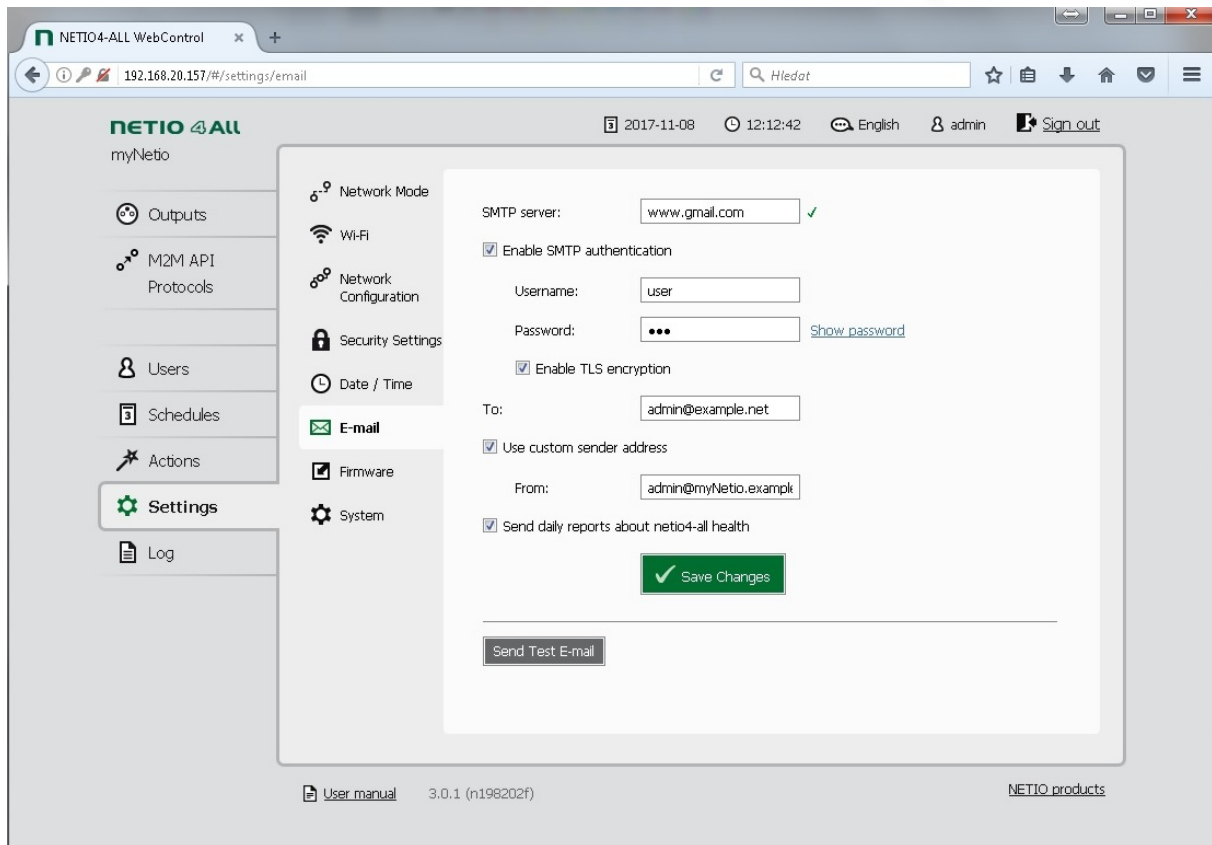


Figure 42. E-mail settings

### SMTP server

Mail server for sending messages. When a different port than the default (25) is used, enter the server address followed by a colon and the port number, e.g. *smtp.netio.eu:8025*

### Enable SMTP authentication

Check this option if the SMTP server requires authentication. Then, fill in the **Username** and **Password** for logging in to the SMTP server.

### Enable TLS encryption

Check this option if the SMTP server requires TLS encryption for logging in.

### To

Enter the e-mail addresses of e-mail recipients. Separate multiple addresses with commas.

### Use custom sender address

Check this option if you want to use a non-default sender's e-mail address for all e-mails sent by your device.


### From

The custom sender's address.

To receive a periodic report with the device status, check **Send daily reports about NETIO 4x health**. The report, sent every day after midnight, contains a summary of the device load in time and events logged during the previous day.

Click **Save Changes** to save the settings. To test the configuration, click **Send Test E-mail**.

### 5.7.7 Firmware update

The  **Firmware** section allows updating the firmware of your device. For details about the installed firmware version, click **Show details**.

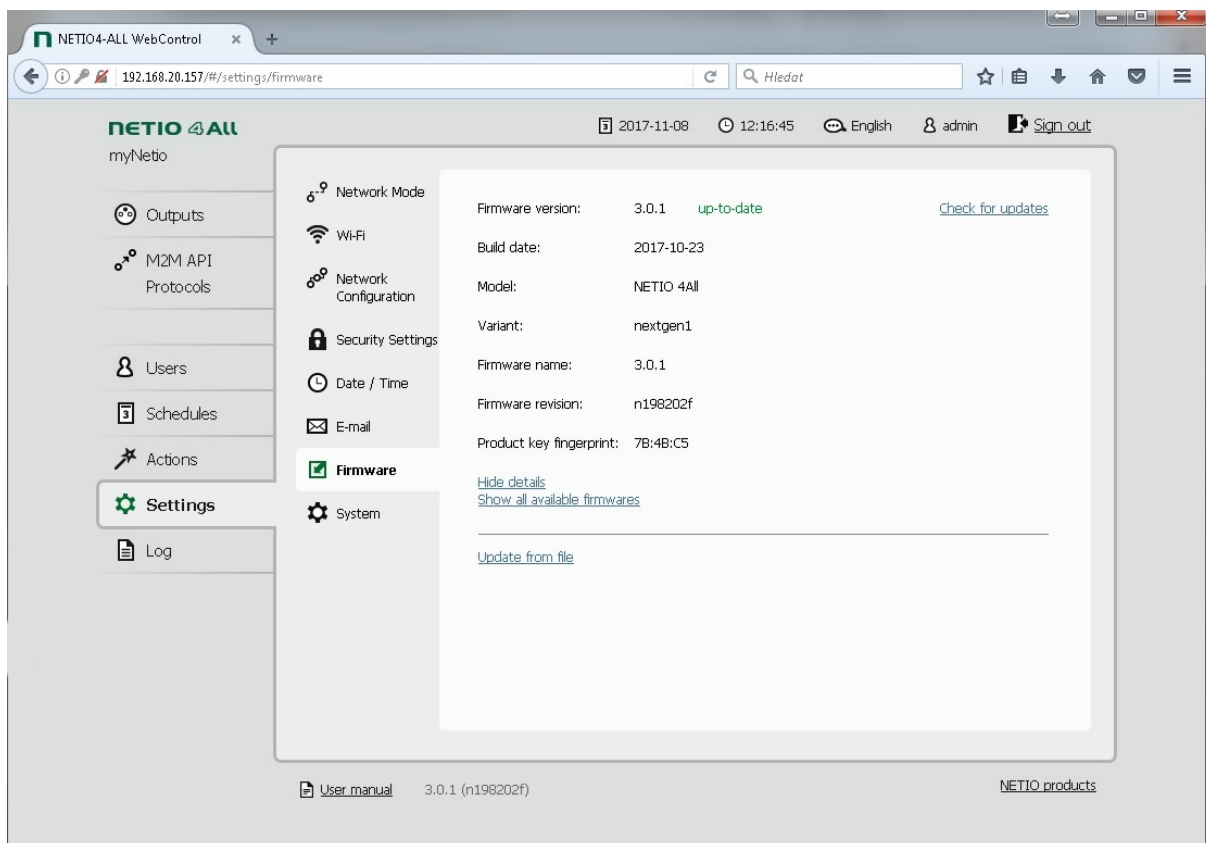



Figure 43. Details about the installed firmware

NETIO 4x can be easily updated from the device web interface, as long as it is connected to the internet and its network interface is correctly configured. To display the list of available firmware versions, click **Show all available firmwares**. Then click the **Check for updates** button to update the list. Read carefully the **Release notes** for each firmware version, which describe the bugfixes and new improvements.

To download the selected firmware, click the  button with its version. Firmware that is already downloaded is shown in green. To install the downloaded firmware, click the button with the firmware name and confirm by clicking **Start the update**. To install the firmware automatically after downloading, check the **Install automatically after download** option.

Besides installing the firmware from the web, it is equally possible to **Update from file**. Firmware files are available at <http://www.netio-products.com/en/firmware-archive>. After downloading,

click **Browse** to select the file with the new firmware and then click **Install firmware** to start the installation.

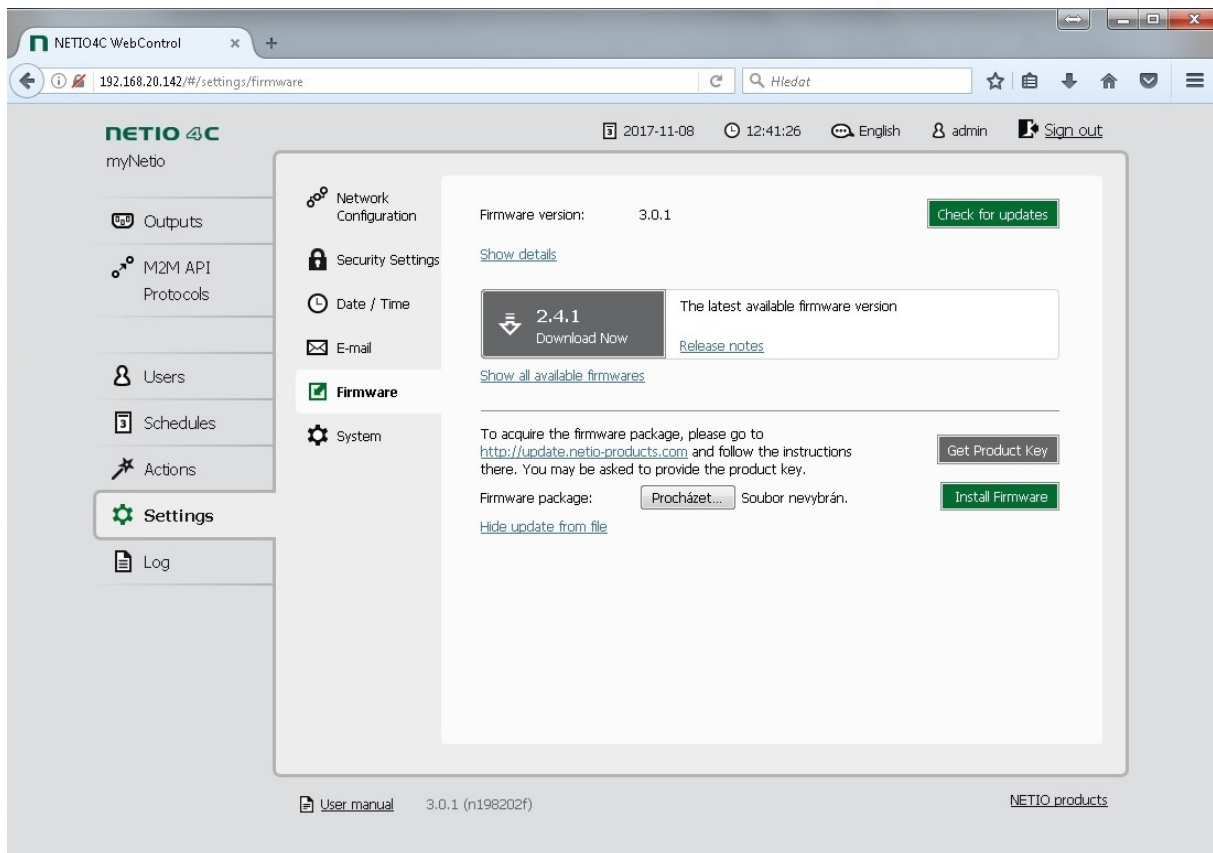


Figure 44. Firmware update from a file

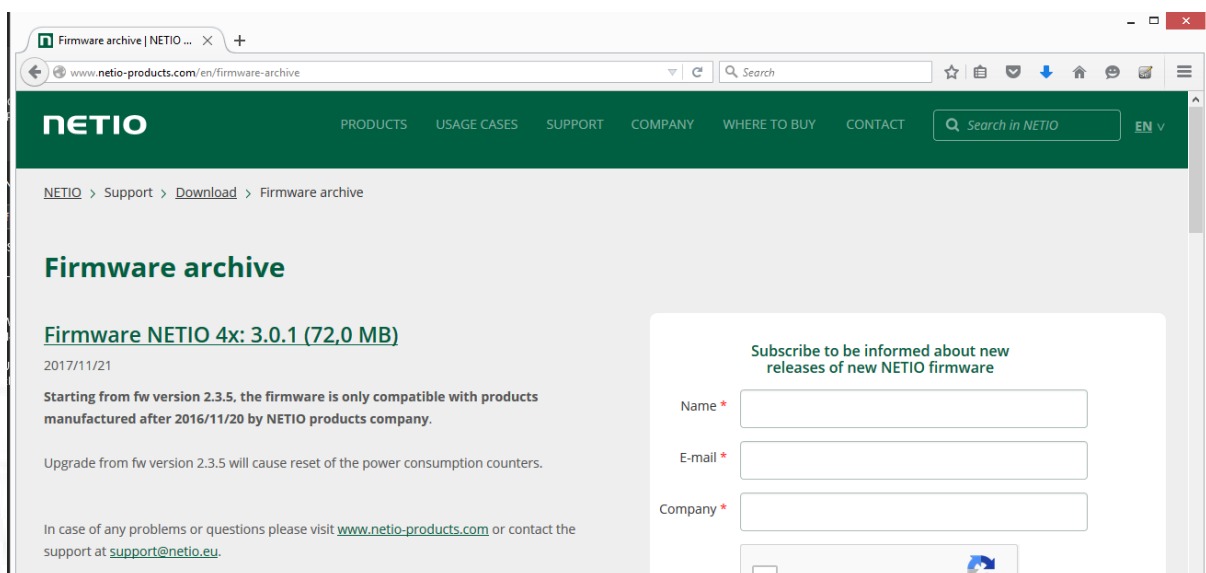


Figure 45 - Firmware download link at the netio-products.com website



**Caution**

Firmware installation causes the NETIO 4x to restart. Do not power off or restart the device during the process; otherwise, the firmware could be damaged, rendering the device inoperable.

The progress of the installation is indicated. When the installation finishes, a message about a successful update appears and you will be taken back to the login screen. During the firmware update, the red LED of output 3 flashes and the red LED of output 4 is lit (see chapter [LED indicators](#)).

## 5.7.8 System

This section allows performing basic settings and viewing basic parameters.

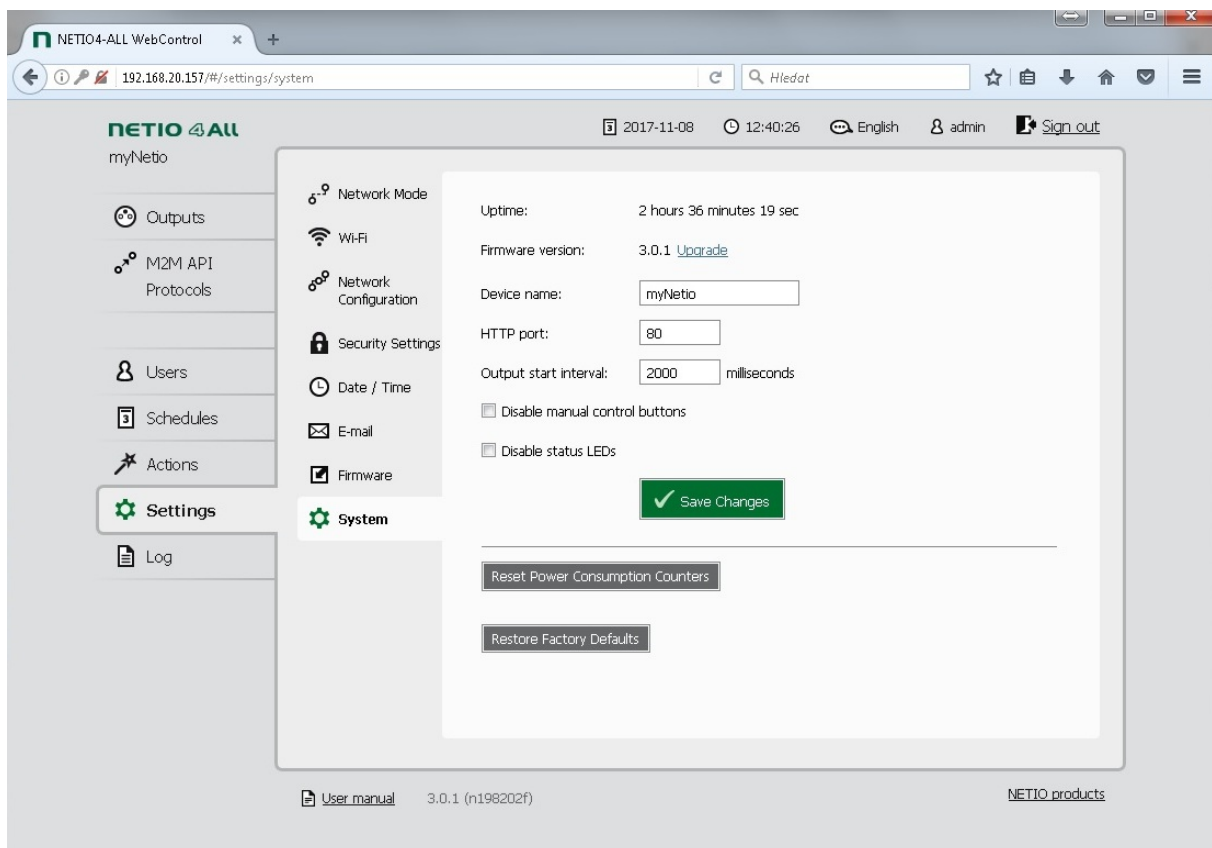


Figure 46. System settings and status

<b>Uptime</b>	Time since the last restart of the device.
<b>Firmware version</b>	Version currently installed and a link to <b>Upgrade</b> to a newer version.
<b>Device name</b>	Shown in NETIO Discover and under the device logo in the web administration (in the figure above: <i>myNetio</i> under the NETIO 4 All logo in the top left corner)
	<b>CAUTION:</b> This value is NOT propagated into the <i>hostname</i> parameter in

the *Network Configuration* section.

<i>HTTP port</i>	Current http port. If the port number is different from 80, it has to be specified in the web browser after the NETIO 4x IP address, e.g.: 192.168.0.99:888
<i>Output start interval</i>	Delay in seconds before the NETIO 4x outputs are switched on when the device is powering up. Starting from the second output, the <b>PowerUp Interval</b> must elapse after the previous output was switched on.
<i>Disable manual control buttons</i>	When checked, outputs cannot be controlled with the hardware buttons.
<i>Disable status LEDs</i>	When checked, LED status indicators on the device are disabled.

Click **Save Changes** to save the settings. The **Reset energy consumption counters** option resets all measured values for individual outputs as well as for the device as a whole.


To reset the NETIO 4x to the factory settings, click **Restore Factory Defaults**. This erases all settings and restores the default values. The **Keep network settings** (to preserve the network configuration) and **Keep SSL CA certificates** options in the confirmation dialog make it easier to re-discover the device after the reset. To start the process of restoring factory defaults, click **Reset settings**.



#### Caution

During the reset to factory defaults, NETIO 4x restarts. After the restart, the login credentials will be reset to admin / admin. If you do not know the device's new IP address, follow the steps in the [Detecting and configuring the IP address](#) section.

## 5.8 Log

In the left menu, select  *Log*.

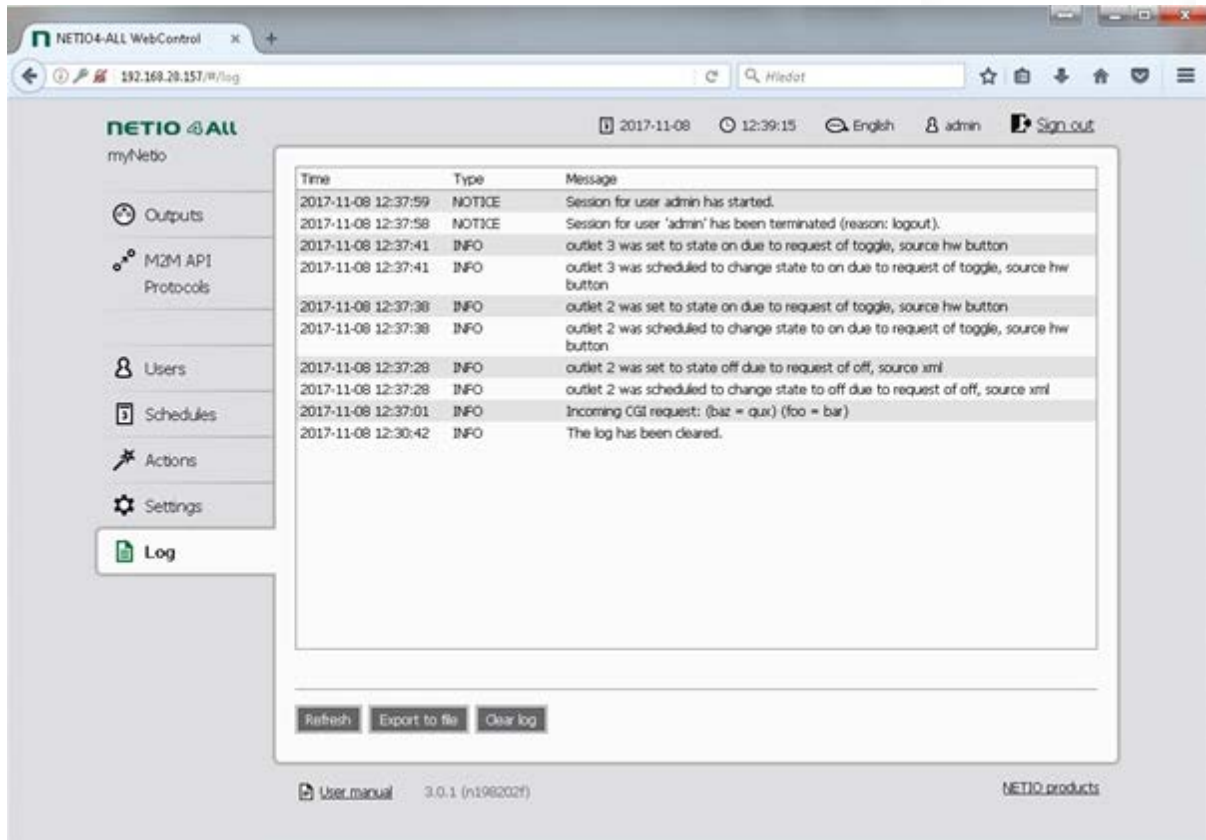


Figure 47. Log

The log contains information about the operation of the device. The most recent entries are shown at the top. To see older entries, scroll down. Click *Refresh* to reload the log to show the most recent entries. The log can be exported to HTML; to do that, click the *Export to file* button and save the file. To clear all log entries, click the *Clear log* button.

The log contains four types of entries.

- Info** Messages for information that document normal device operation, such as an application being started or automatic database maintenance taking place. Custom messages from user-defined actions also fall into this category (see the [Configuring actions](#) section).
- Notice** Notifications concerning the device operation, such as the beginning and end of a user session.
- Warning** This category contains warning messages, such as failed login attempts due to incorrect username or password.
- Error** These messages indicate abnormal and potentially faulty device operation.

## 6 Controlling the outputs manually

In addition to being controlled by a computer, the device can be also controlled using the four buttons on the front panel. The buttons (from left to right) correspond to outputs 1 to 4. To switch the corresponding output on or off, press and hold the button for two seconds. If the output was switched off, it is switched on, and vice versa. In case the buttons do not react, verify the *Settings > System > Disable manual control buttons* setting to make sure that the buttons are not disabled.

## 7 LED status indicators


LED indicators on the device inform the user about the output states and the state of the device.

There is a green-red LED for each output, located above the corresponding manual control button. The green color for each output indicates the current output state. If the LED is lit green, the output is on; otherwise, the output is off.

The LED indicators can be switched off completely in the user interface: *Settings > System > Disable status LEDs*.

The red<sup>6</sup> color indicates different states of the device as a whole. The following states are possible:

LED indicator for output 1 flashes red	Locate function was activated
LED indicator for output 2 flashes red	Device is waiting for an address to be assigned by a DHCP server
LED indicator for output 2 is lit red	DHCP request failure (90-second timeout - the device did not receive an IP address from a DHCP server)
LED indicator for output 3 flashes red	Firmware update is in progress
LED indicator for output 4 is lit red	The device is in service mode
LED indicators for all outputs are lit red	The device is powering up
LED indicators for all outputs flash red	Factory defaults are being restored

The remaining two LEDs on the front panel show the Wi-Fi and Bluetooth status (NETIO 4 and NETIO 4All only). If the  Wi-Fi LED is green, Wi-Fi is active. If the LED is off, Wi-Fi is inactive. The LED flashes in case of problems with Wi-Fi.

---

<sup>6</sup> The LED indicators may also be orange. For example when output 2 is active (its green LED is on) and at the same time, the DHCP server did not assign an address, the green and red colors of the LED will be lit at the same time → the LED appears orange.

## 8 Acoustic indication

Acoustic indication has the following meanings:

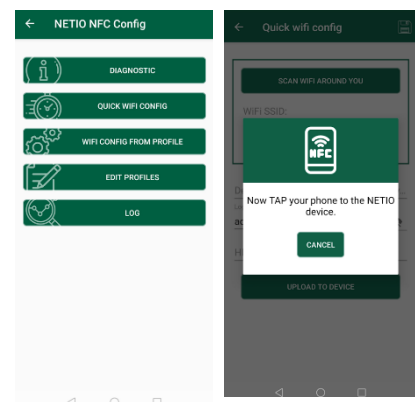
1x beep	The device is starting.
2x beep	The device is entering service mode. The process of restoring factory defaults has finished.
3x beep	The process of restoring factory defaults has started.

## 9 NETIO Mobile2 for Android

NETIO Mobile2 application is for control NETIO devices produced after 2016.

Features:

- Switch On / Off each power socket on local network.
- Show power consumption on each power output (if supported).
- Searching NETIO devices in local network
- Install NFC enabled NETIO devices



<https://play.google.com/store/apps/details?id=cz.netio.netio>



## 10 Troubleshooting

### 10.1 Forgotten password, restoring factory defaults

See [Restoring factory defaults](#).

### 10.2 Firmware upgrade problems

If a problem occurs during firmware upgrade (e.g. a network outage, device powered off during the upgrade), the device can be forced to start in service mode. To do so, press and hold the button for output 4 when powering up the device. Hold the button until the device beeps 2 times. Then, open the device's IP address in your browser and click the **Firmware** button in the top menu. Continue by uploading a firmware file according to the procedure for [upgrading the firmware from a file](#).

### 10.3 Fuse reset - only NETIO 4 and NETIO 4All

If the NETIO 4 / NETIO 4All device stops working and no LEDs light up, it may be the case that the maximum rated current was exceeded and the device has shut itself down. To prevent damage to the device, a resettable safety fuse interrupts the power. The fuse button at the right side of the device pops out to indicate that. Disconnect all appliances connected to the outputs.

Before switching NETIO 4 / NETIO 4All on again, let it cool down; this may take several minutes. To power up the device again, press the fuse button. If the button cannot be pressed, let the device cool down for a while. Before reconnecting appliances, make sure that the excessive current that has led to the power interruption was not caused by a faulty appliance.

## 11 Application Notes (ANxx)

As a demonstration of practical uses of NETIO 4x, we have prepared examples of use for all technologies implemented in the current firmware, such as Lua scripts, M2M API protocols, methods of connecting NETIOx with other sensors, devices or cloud services, and more.

See the following section of our website:

NETIO > Support > Application Notes

Application Notes (ANxx) |... X +

www.netio-products.com/en/application-notes

NETIO PRODUCTS USAGE CASES SUPPORT COMPANY WHERE TO BUY CONTACT Search in NETIO EN v

NETIO > Support > Application Notes (ANxx)

### Application Notes (ANxx)

Application notes provide a better understanding of products NETIO use in your applications.

Filtering cancel filters

NETIO 4All  NETIO 4  NETIO 4C  Lua tutorials  User library  M2M API  3rd party HW-SW

NETIO AN13: STE2 thermometer (HW group) switches NETIO sockets based on temperature

**AN13 STE2 thermometer (HW group) switches NETIO sockets based on temperature**

23.10.2017

STE2 by HW group is a LAN/WiFi thermometer/humidity meter. The NETIO AN13 Application Note presents a **Lua script** that connects NETIO 4x **smart sockets** with the STE2 **thermometer**. **230V electrical sockets** are switched on or off according to the **temperature** (or humidity) measured by STE2. The devices communicate over a LAN, values are transferred in a **.xml file**. The script can be modified for other devices and **xml data**.

NETIO AN10: Telnet control of electrical power sockets (M2M API - KSHHELL)

**AN10 Telnet control of electrical power sockets (M2M API - KSHHELL)**

16.10.2017

NETIO 4x **electrical sockets** (NETIO 4 / 4All / 4C) can be **controlled over a LAN** using the **Telnet M2M API protocol and KSHHELL commands**. The NETIO AN10 Application Note demonstrates how to use **Telnet** to control individual sockets, gives a brief description of the **KSHHELL** protocol and shows examples how to test the communication in Windows using several utilities.

Figure 48 - Application Notes at the NETIO website

## Conclusion

The manufacturer assumes no responsibility for any technical or printing errors and reserves the right to modify the product or this manual without prior notice. Such changes are announced at the manufacturer's website, <http://netio-products.com>.

The manufacturer disclaims all warranties of any kind with respect to the contents of this manual, as well as all implied warranties of merchantability or fitness for a particular purpose.

In particular, the manufacturer disclaims all responsibility for any damages caused by incorrect use of the product, failure to comply with instructions and recommendations in the user manual, and/or unprofessional actions of third parties not authorized by the manufacturer to perform warranty service.

We trust that you will be satisfied with our product. If you have any questions or comments regarding NETIO 4x functions, please contact us.

Your NETIO products team

NETIO products a.s.

U Pily 103/3  
143 00 Praha 4 - Modrany  
Czech Republic  
[info@netio.eu](mailto:info@netio.eu)

© 2017 NETIO products a.s. All rights reserved.

## Declaration of conformity

**Manufacturer/Importer:** NETIO products a.s.

**Address:** U Pily 3/103  
143 00 Praha 4, Czech Republic

**Product:** NETIO 4 DE, NETIO 4AII DE,  
NETIO 4 FR, NETIO 4AII FR



### RTTED:

We hereby declare that the above-mentioned product(s) comply with essential requirements of the Government Regulations No. 483/2002 Sb. and No. 251/2003 Sb. (Directive 1999/5/EC) on radio equipment and telecommunications terminal equipment and the mutual recognition of their conformity.

The following standards were used in the conformity assessment:

EN 55022:2010	ETSI EN 301489-1 V1.9.2:2011
EN 61000-3-2:2006+A1:2009+A2:2009	ETSI EN 301489 17 V2.2.1:2012
EN 61000-3-3:2013	ETSI EN 300 328 V1.8.1
EN 55024:2010	

### LVD:

We hereby declare that the above-mentioned product(s) comply with essential requirements of the Government Regulation No. 17/2003 Sb. (Directive 2006/95/EC) on electrical equipment designed for use within certain voltage limits.

The following standards were used in the conformity assessment:

EN 60950-1:2006+A11:2009+A1:2010+A12:2011+A2:2013

### RoHS:

We hereby declare that the above-mentioned product(s) comply with essential requirements of the Government Regulation No. 481/2012 Sb. (Directive 2011/65/EU) on the restriction of the use of certain hazardous substances in electrical and electronic equipment.

The following standards were used in the conformity assessment:

EN 50581:2012

Praha, November 16, 2016

Jan Řehák, Chairman of the Board

## Declaration of conformity

**Manufacturer/Importer:** NETIO products a.s.

**Address:** U Pily 3/103  
143 00 Praha 4, Czech Republic



**Product:** NETIO PowerPDU 4C

### RTTED:

We hereby declare that the above-mentioned product(s) comply with essential requirements of the Government Regulations No. 483/2002 Sb. and No. 251/2003 Sb. (Directive 1999/5/EC) on radio equipment and telecommunications terminal equipment and the mutual recognition of their conformity.

The following standards were used in the conformity assessment:

EN 55011, ed.3:2010

EN 61326-1, ed.2:2013

### LVD:

We hereby declare that the above-mentioned product(s) comply with essential requirements of the Government Regulation No. 17/2003 Sb. (Directive 2006/95/EC) on electrical equipment designed for use within certain voltage limits.

The following standards were used in the conformity assessment:

EN 61010-1, ed.2:2011

### RoHS:

We hereby declare that the above-mentioned product(s) comply with essential requirements of the Government Regulation No. 481/2012 Sb. (Directive 2011/65/EU) on the restriction of the use of certain hazardous substances in electrical and electronic equipment.

The following standards were used in the conformity assessment:

EN 50581:2012

Praha, April 1, 2019

Jan Řehák, Chairman of the Board

## NETIO products overview

	LAN (RJ45)	WiFi / Antenna	IQRF 868MHz	Output Type	Switched outputs	Metered outputs	DI Inputs	Power	Button(s) / LED	NFC	Industrial features	Power-Up state
PowerPDU 4C	2x	-	-	C13	4	4	-	110/230V / 10A	4 / 4	-	ZCS, RS232	Yes
PowerPDU 4PS	1	-	-	C13	4	-	-	110/230V / 10A	1 / 4	-	ZVS	Yes
PowerDIN 4PZ	1	Int.	-	Terminal block	4	2	2x (S0)	110/230V / 16A	1 / 4	Yes	ZCS	Yes
PowerBOX 3PE	1	-	-	FR	3	-	-	230V / 16A	1 / 0	-	ZVS	Yes
PowerBOX 3PF	1	-	-	DE	3	-	-	230V / 16A	1 / 0	-	ZVS	Yes
PowerBOX 3PG	1	-	-	UK	3	-	-	230V / 16A	1 / 0	-	ZVS	Yes
PowerCable Modbus 101E	-	Int.	-	FR	1	1	-	230V / 16A	1 / 1	Yes	ZCS	Yes
PowerCable Modbus 101F	-	Int.	-	DE	1	1	-	230V / 16A	1 / 1	Yes	ZCS	Yes
PowerCable Modbus 101J	-	Int.	-	CH	1	1	-	230V / 10A	1 / 1	Yes	ZCS	Yes
PowerCable Modbus 101G	-	Int.	-	UK	1	1	-	230V / 13A	1 / 1	Yes	ZCS	Yes
PowerCable Modbus 101S	-	Int.	-	C13	1	1	-	110/230V / 10A	1 / 1	Yes	ZCS	Yes
PowerCable IQRF 901E	-	-	Yes	FR	1	1	-	230V / 16A	1 / 1	-	ZCS	-
PowerCable IQRF 901F	-	-	Yes	DE	1	1	-	230V / 16A	1 / 1	-	ZCS	-
NETIO 4 DE	1	Fixed	-	DE	4	-	-	230V/15A	4 / 4	-	-	-
NETIO 4 FR	1	Fixed	-	FR	4	-	-	230V/15A	4 / 4	-	-	-
NETIO 4AII DE	1	Ext.	-	DE	4	4	-	230V/15A	4 / 4	-	-	-
NETIO 4AII FR	1	Ext.	-	FR	4	4	-	230V/15A	4 / 4	-	-	-

## NETIO products - features

	Web	HTTPS	Scheduler function	IP Watadog function	Lua scripting	Telnet	Modbus/TCP	XML	JSON	URL-API	SNMP	Http(s) push XML / JSON	MQTT	MQTT-flex
PowerPDU 4C	✓	✓	Yes	Yes	Yes	✓	✓	✓	✓	✓	v1/v3	-	✓	-
PowerPDU 4PS	✓	-	planned	-	-	✓	✓	✓	✓	✓	v1	-	-	✓
PowerDIN 4PZ	✓	-	planned	-	-	✓	✓	✓	✓	✓	v1	✓	-	✓
PowerBox 3Px (E,F,G)	✓	-	planned	-	-	✓	✓	✓	✓	✓	v1	-	-	✓
PowerCable Modbus 101x	✓	-	planned	-	-	✓	✓	-	-	-	v1	-	-	-
PowerCable REST 101x	✓	-	planned	-	-	-	-	✓	✓	✓	v1	-	-	-
PowerCable MQTT 101x	✓	-	planned	-	-	-	-	-	-	-	v1	✓	-	✓
PowerCable IQRF 901x	-	-	-	-	-	-	-	-	-	-	-	-	-	-
NETIO 4 DE	✓	✓	Yes	Yes	Yes	✓	✓	✓	✓	✓	v1/v3	-	✓	-
NETIO 4 FR	✓	✓	Yes	Yes	Yes	✓	✓	✓	✓	✓	v1/v3	-	✓	-
NETIO 4AII DE	✓	✓	Yes	Yes	Yes	✓	✓	✓	✓	✓	v1/v3	-	✓	-
NETIO 4AII FR	✓	✓	Yes	Yes	Yes	✓	✓	✓	✓	✓	v1/v3	-	✓	-