User manual

RFID TAB



Dear Customer!

Thank you very much for choosing our product. At the same time, please read this manual carefully before using it, as it indicates the most appropriate ways to handle this appliance, taking into account basic safety and maintenance principles. Please also retain the manual for future reference.

Remember!

The manufacturer will not be held liable for any damage caused by improper use of the device or improper handling, nor for any malfunctions of the controller resulting from improper operation.

Table of contents

Table of contents 3
1 Introductory information 5
2 Warranty and liability 5
3 Safety in use 6
3.1 Storage, operating conditions 6
3.2 Installation and use of the module 6
3.3 Disposal and decommissioning 6
4 Purpose of the device 6
5 Module design 7
5.1 Specifications 7
5.2 General features 9
6 Configuring the unit 10
6.1 Changing basic network settings - Discoverer programme 10
6.2 Changing the subnet of the computer to be configured 11
6.3 Configuring LAN and Wi-Fi network settings 12
7 Module functions 13
7.1 Device status - Home tab 14
7.2 Operation modes of the reader 15
7.3 Card management 17
7.3.1 Adding and removing cards from the browser level 17
7.3.2 Adding a card using the Modbus protocol 19
7.3.3 Customised API (http GET) 20
7.4 Writing data to Mifare and ICODE cards - HTTP GET command 21
7.4.1 Recording data on the ICODE card 21
7.4.2 Data recording on Mifare Classic 1k/4k 23
7.5 Reading Mifare and ICODE blocks 25
7.5.1 Reading the contents of an ICODE tag - Block Reader 25
7.5.2 Reading the contents of an ICODE tag - HTTP GET command 26
7.5.3 Reading a Mifare Classic 1k/4k block - Block Reader 27
7.5.4 Reading the contents of a Mifare Classic tag - HTTP GET command 28
7.6 Access – card groups 30
7.7 Event logging 32
7.7.1 Viewing events via the Web page 33
7.7.2 Viewing events via XML file 34
7.7.3 Downloading recorded logs in CSV format 35
7.8 Responses to events (inputs/outputs) - I/O Settings 36
7.9 Long open – I/O Settings 38
7.10 Relay activation on another Inveo device – I/O Settings 39

7.11 Audio-visual signalling – I/O Settings 40
7.12 Real-time clock (RTC) 41
7.13 Configuring the SNMP server 42
7.14 Communication protocols and administration 42
8 Communication with the module 44
8.1 Modbus 44
8.2 Integration with proprietary software 48
8.3 HTTP GET – status reading 49
8.4 HTTP GET – control 50
8.5 Editing cards via HTTP POST 52
8.6 HTTP Client 53
8.7 Communication with the module from an external network 56
9 Description of connectors 57
10 DHCP 58
11 Restoring factory settings 58
12 Software update 58

1 Introductory information

Before working with the controller, read the User Manual and follow the instructions contained therein!

Description of symbols used in this manual:

Warning	This symbol indicates that it is necessary to read a specific section of the User Manual that contains important information and warnings. Ignoring these warnings may lead to injury or damage to the equipment.
Tip	Important instructions and information. Observing the texts marked with this sign facilitates operation.

The screenshots shown in this manual may differ from their actual appearance. Due to the continuous software development of the modules, some functions may differ from those described in the manual. The manufacturer is not responsible for any undesired effects resulting from software differences.

2 Warranty and liability

Warning	The manufacturer provides a two-year warranty for the device and a post-warranty service for a period of 10 years from the date the device is placed on the market. The warranty covers all defects in materials and workmanship.

The manufacturer undertakes to honour the warranty agreement if the following conditions are met:

- all repairs, modifications, extensions and calibrations of the appliance are carried out by the manufacturer or an authorised service centre,
- the mains power supply system complies with the applicable standards in this respect,
- the appliance is operated in accordance with the instructions given in
- in this manual,
- the appliance is used in accordance with its intended use..

The manufacturer shall not be held liable for any consequences resulting consequences of incorrect installation, improper use of the appliance, non-compliance with the operating instructions and repairs carried out by unqualified persons.

Warning	There are no user-serviceable parts in the appliance.

3 Safety in use

The module was constructed using modern electronic components in line with the latest trends in world electronics. Particular emphasis was placed on ensuring optimum operational safety and control reliability. The unit has a housing made of high-quality plastic.

3.1 Storage, operating conditions

The device should be stored indoors where the atmosphere is free of vapours and corrosive agents and:

- an ambient temperature of -30°C to +60°C,
- humidity between 25% and 90% (no condensation allowed)
- an atmospheric pressure of 700 to 1060hPa.

The unit is designed to operate under the following conditions:

- ambient temperature of -10°C to +55°C,
- humidity between 30% and 75%,
- atmospheric pressure of 700 to 1060hPa.

Recommended transport conditions:

- ambient temperature from -40°C to +85°C,
- humidity 5% to 95%,
- atmospheric pressure 700 to 1060hPa.

3.2 Installation and use of the module

The module should be operated as recommended later in this manual.

3.3 Disposal and decommissioning

In the event that it becomes necessary to dispose of the unit (e.g. at the end of its useful life), contact the manufacturer or the manufacturer's representative, who is obliged to respond appropriately, i.e. to collect the unit from the user. The user may also contact companies dealing with the disposal and/or decommissioning of electrical or computer equipment. Under no circumstances should the appliance be placed with other waste.

4 Purpose of the device

The RFID TAB device is an industrial RFID reader designed for broad use across various sectors such as industry, commerce, logistics, and services. Due to its functionality, it can be utilized for tasks such as authorizing access to industrial machines, recording employee work hours, managing room access, and in systems that control the operation of forklifts. Additionally, the reader is well-suited for contactless payment systems and monitoring large venues, such as stadiums or exhibition halls.

RFID TAB supports various transponder standards, including Mifare Classic®, Mifare DESFire®, Unique EM4100, HID iClass®, and HITAG 2. With integrated communication protocols (Modbus TCP, Modbus RTU, HTTP, SNMP), the device can be seamlessly integrated into automation systems.

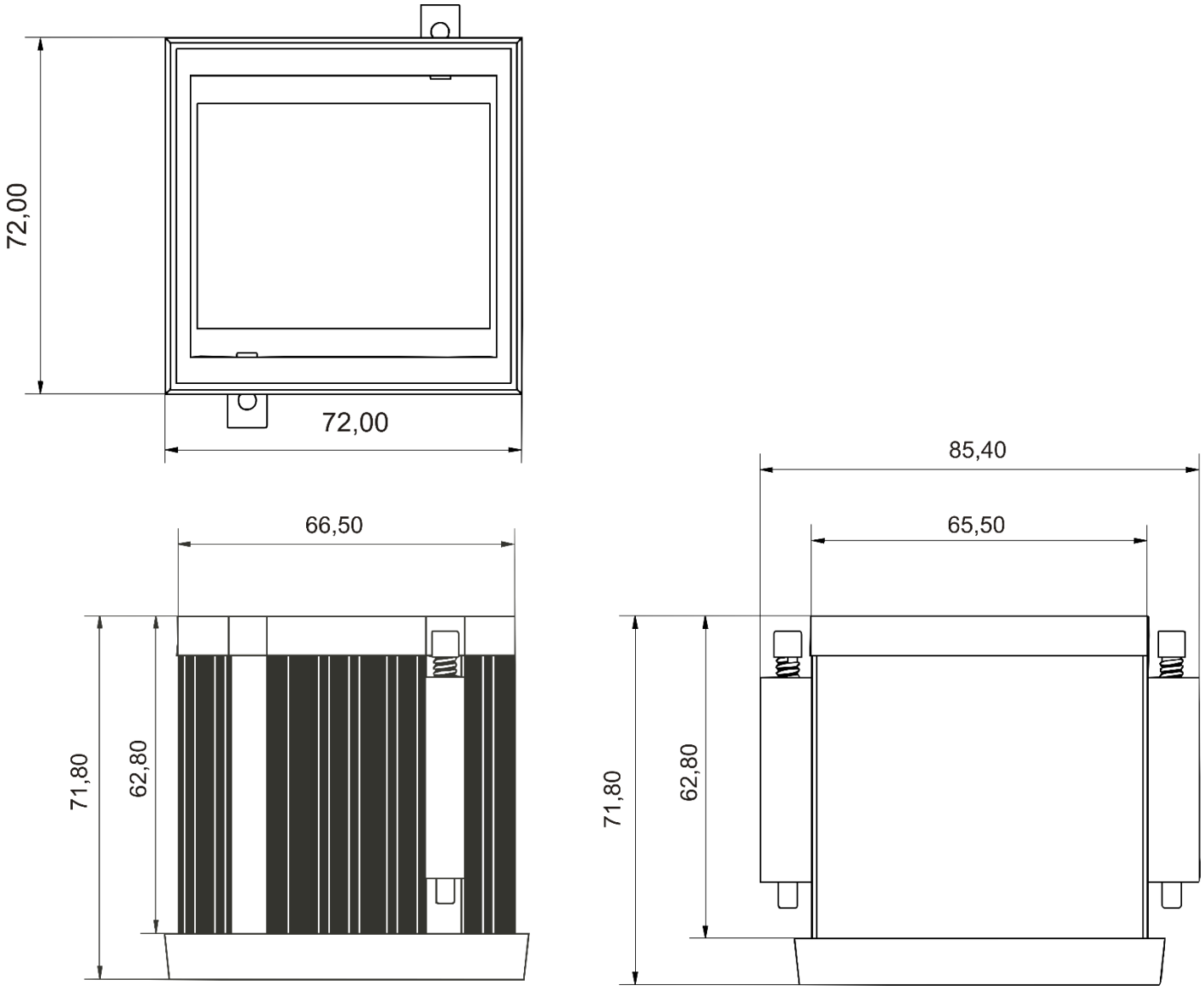
Moreover, RFID TAB offers the ability to control two relay outputs and manage two digital inputs. Thanks to built-in memory, the user has access to event logs and stored identifiers, making this device an excellent tool for access management and monitoring in industrial environments. A robust IP60-rated enclosure and the option for panel mounting ensure durability and reliable performance.

5 Module design

5.1 Specifications

Power supply	**RFID TAB Passive PoE**	Two optional power supply types are available: 10-24VDC via a 3.5mm disconnect connector Passive PoE 10-24VDC
	RFID TAB PoE 802,3af	Two optional power supply types are available: 10-24VDC via a 3.5mm disconnect connector PoE: 33-57V PoE IEEE 802.3af
Power consumption	max 2,5W (~200mA@12V)	
Transponders	**RFID TAB Mif**	Supported transponder standard: Mifare Classic®
	RFID TAB Uni	Supported transponder standard: Unique EM4100 EM4102
	RFID TAB Ico	Supported transponder standard: ICODE® (ISO 15693)
	RFID TAB HT2	Standard of supported transponders: HITAG (HITAG 2)
	tag reading distance: up to 8cm built-in memory: 1000 tags, 30000 events	
Inputs	Number of inputs: 2, Input type: Normally Open (NO)	
Outputs	Number of outputs: 2, Output type: NO relay, maximum load: 1A @ 30V DC..	
Communication	1 Ethernet port, speed up to 10Mbps, 1 RS485 port, modbus RTU.	
Enclosure class	IP65	
Operating temperature	-10°C to +55°C	

Dimensions:



5.2 General features

A general view of the RFID TAB:



Communication with the module is via LAN or RS485.

The following options are available for accessing the RFID tag readout:

- via the built-in web server, using a standard web browser (preferred browsers are MOZILLA FIREFOX, OPERA, CHROME),
- over HTTP protocol in server mode
- over HTTP protocol in client mode
- via MODBUS TCP protocol
- via MODBUS RTU protocol (RS485)
- via SNMP protocol

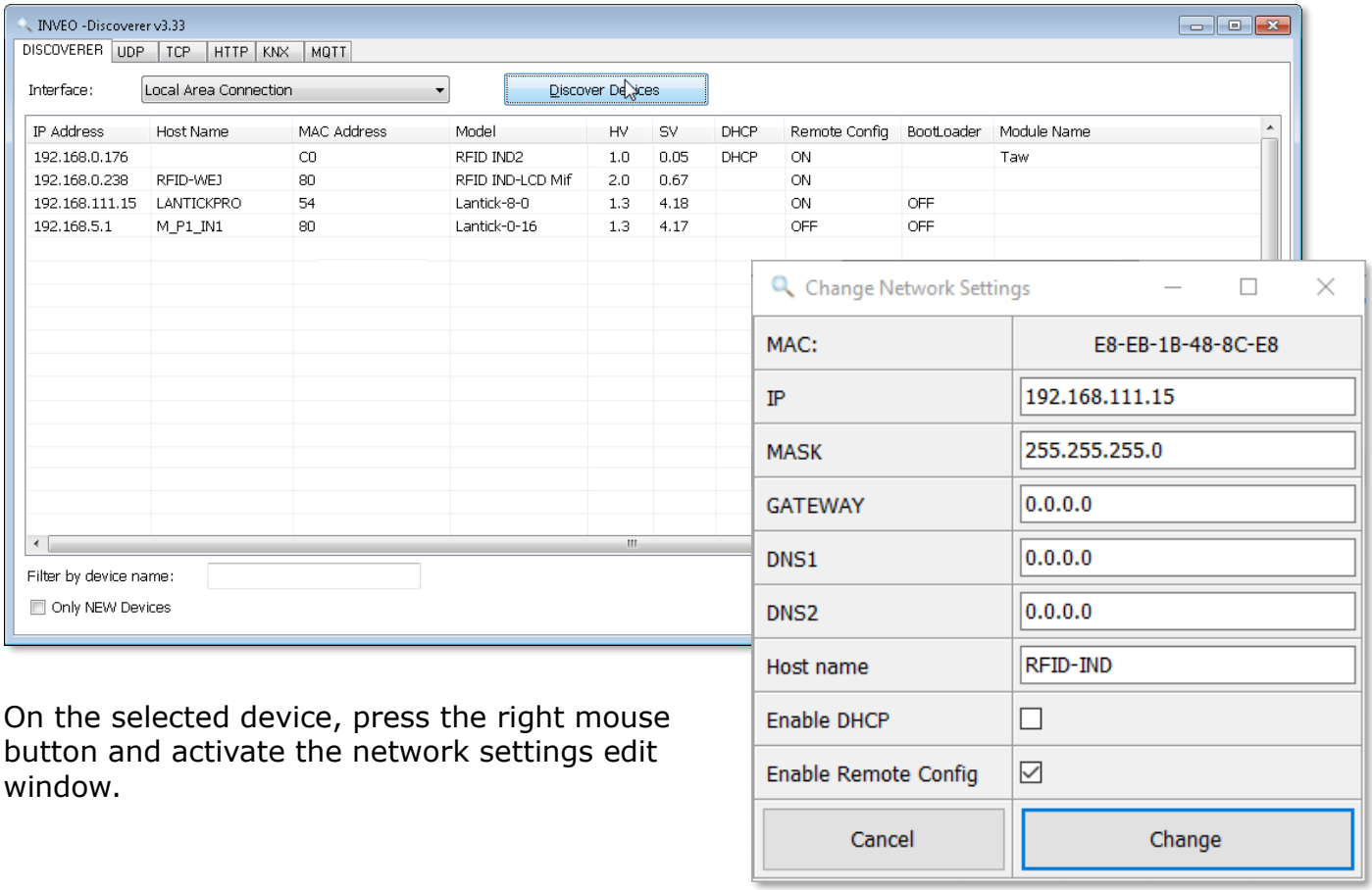
The module is equipped with LEDs that indicate the power supply and the current status of the device.

6 Configuring the unit

On first start-up, it is necessary to configure the device.
This can be done in two ways. The simplest method is to use the Inveo Discoverer program.

6.1 Changing basic network settings - Discoverer programme

Download and install the Discoverer programme. After launching the application, select the appropriate network interface and press the Discover Devices button - all available Inveo devices will be displayed.



On the selected device, press the right mouse button and activate the network settings edit window.

If Remote Config is disabled (enabled by default), it is necessary to configure the device, via the built-in web server. This operation requires changing the subnet of the computer (chapter 6.2).

The Remote Config setting can be edited from the web server under the Administration tab. Simply check/uncheck the Enable remote config option.

After making changes, confirm them with Save button.

6.2 Changing the subnet of the computer to be configured

When configuring the device bypassing the Discoverer application, the subnet address of the computer connected to the same network must first be changed.

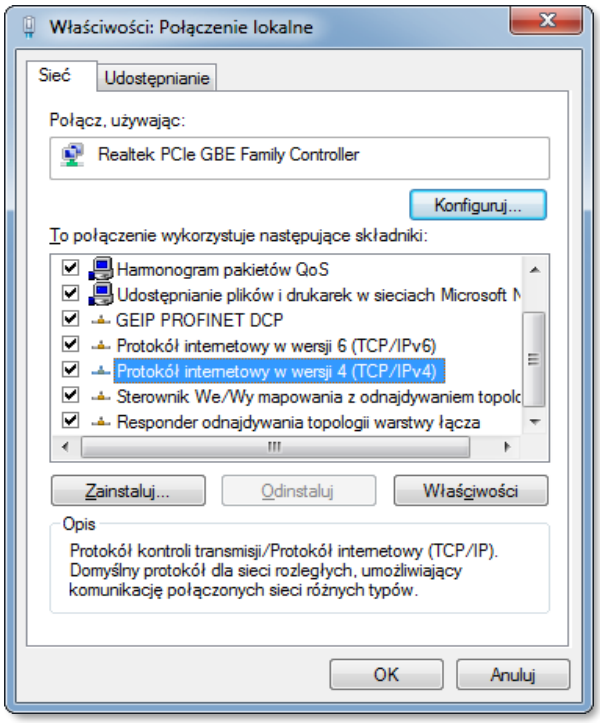
To do this, go to the network configuration of the computer:

Press Win + R, type ncpa.cpl and press Enter,

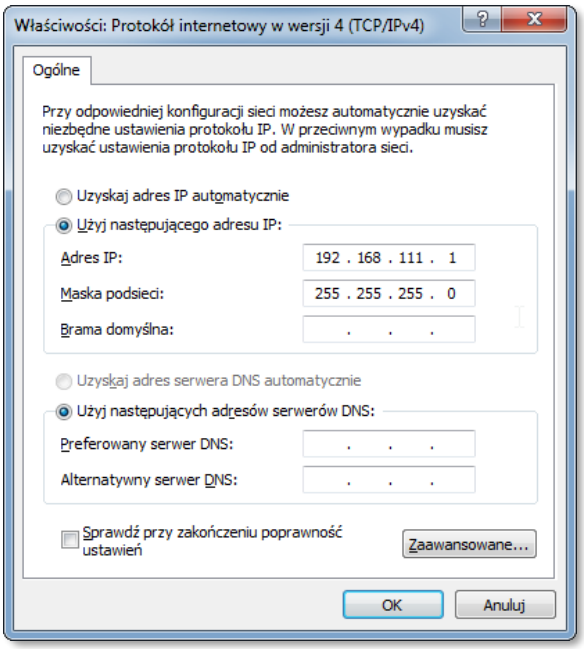
OR

Start → Control Panel → Network and Internet → Network and Sharing Centre → Change network adapter settings.

Select your network connection, press the right mouse button and click Properties.
Once selected, the configuration screen will appear:



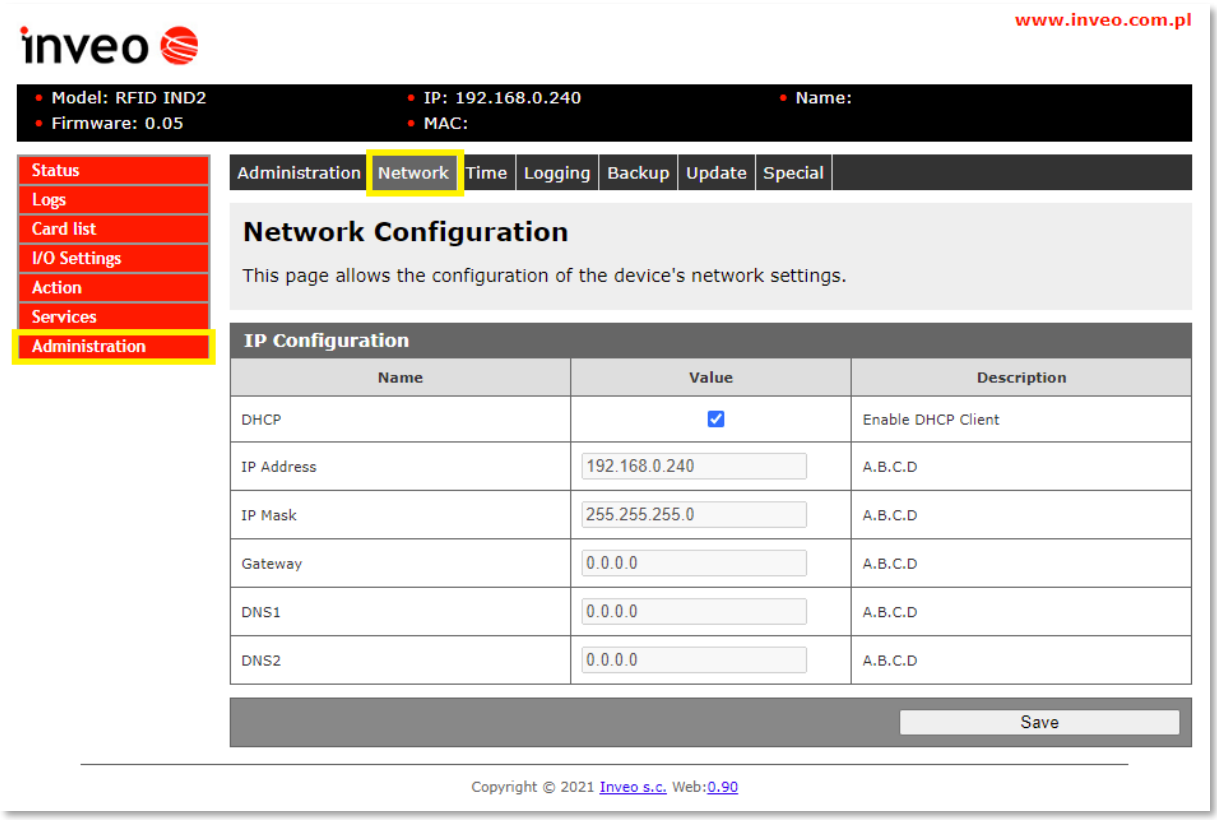
Then select the "Internet Protocol (TCP/IP)" setting and enter the following parameters:



After accepting the settings with the OK button, start your web browser and enter the address: 192.168.111.15. (Default user and password: admin/admin00).

6.3 Configuring LAN and Wi-Fi network settings

In the Network tab it is possible to change the LAN parameters.

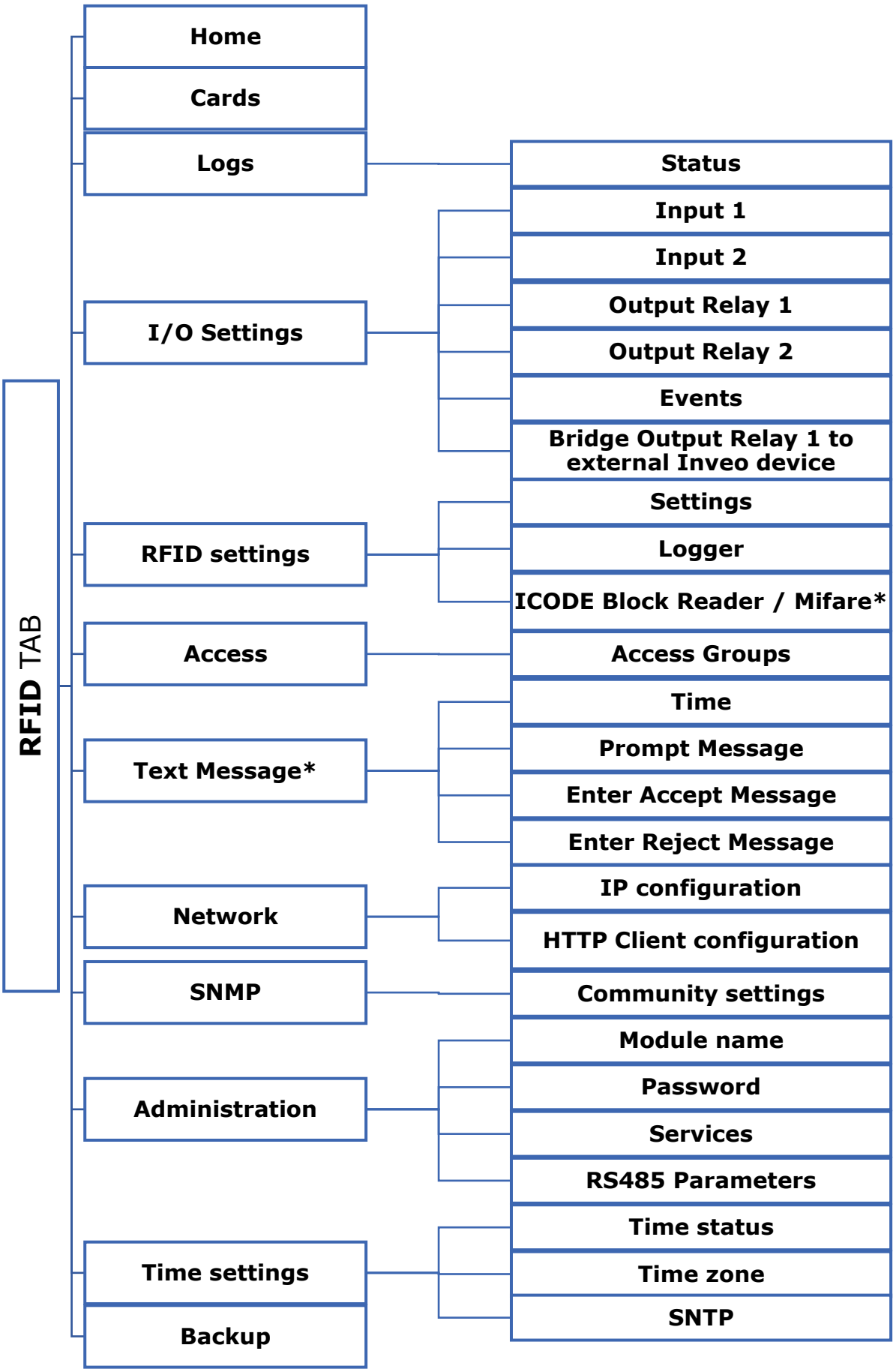


Use the fields to change the network settings of the device:

- **DHCP** - selecting this field forces the use of the IP address assigned by the DHCP server,
- **IP Address** - module's IP address (the module will be visible in the network at this address),
- **IP Mask** - IP subnet mask,
- **Gateway** - network gateway,
- **DNS1, DNS2** - addresses of DNS servers.

After making changes, click Save.

7 Module functions



* depending on software version

7.1 Device status - Home tab

The status of the device is available via the web page (factory default 192.168.111.15).

Status	
Name	**Value**
Input 1:	Off
Input 2:	Off
Relay 1:	Off
Relay 2:	Off
Last read ID:	0000000000
ID known as:	None
Number of read ID:	0
Active Groups:	1---
Other info:	

When the HOME tab is selected, the Status table is displayed:

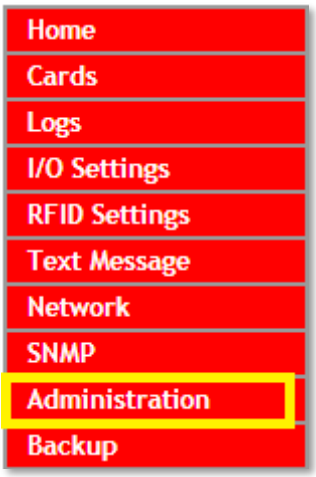
- **Input 1** - current status of input number 1 (binary input),
- **Input 2** - current status of input number 2 (binary input),
- **Relay 1** - current status of relay output number 1,
- **Relay 2** - current status of relay output number 2,
- **Last read ID** - last TAG read in HEX format,
- **ID known as** - type of tag read (unknown/user),
- **Number of read ID** - number of tags read since device reset,
- **Active Groups** - numbers of groups that are active at the moment,
- **Other info** - this field informs about the timeout of the response from the server (message: Protocol Timeout!).

Tip

Displaying the flag (LOCK!) in the Last ID: 8500c2b4a8 item means that reading of the next tags is blocked until deletion with the releaseId command (HTTP GET) or in the case of Modbus, the value 0 should be sent to the address 1 Holding Reg or the value 0 to the address 1004 Single Coil.

7.2 Operation modes of the reader

The device allows a choice of operating modes. The extensive control options allow for stand-alone operation and for software control

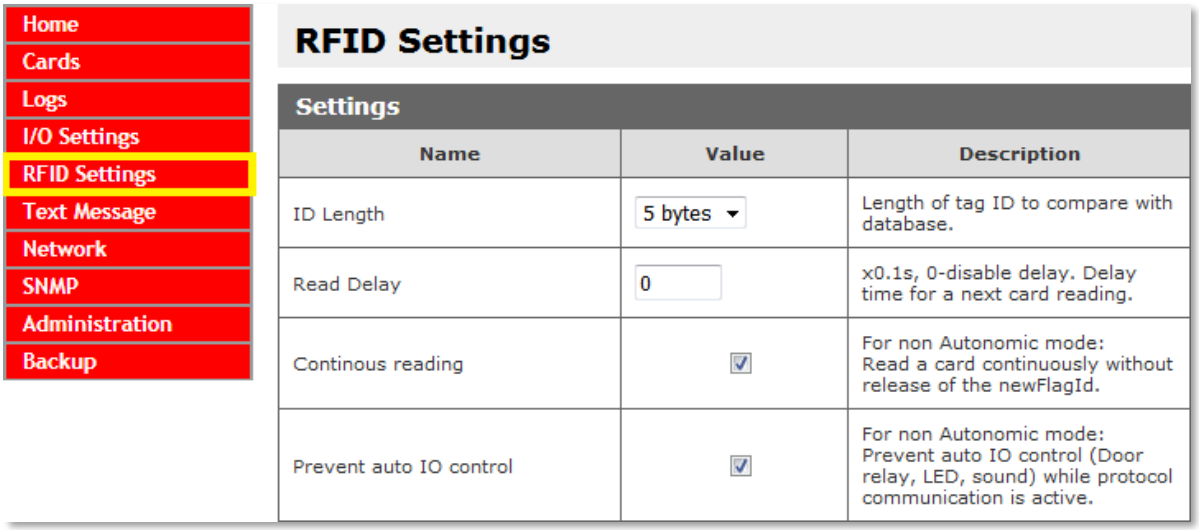


Autonomic	☐	
Enable MODBUS TCP	☐	
Enable MODBUS RTU	☐	
Enable SNMP	☐	
Enable HTTP GET	☑	
Enable HTTP Client	☐	
Timeout	0	(x0,1s) a reader communication loss timeout after which the reader switches to Autonomic mode.

Autonomic mode
Stand-alone mode, i.e. the reader compares the tags with the internal memory, opens the lock for the application of a stored card, etc.

Software control mode
The module has the option of being controlled by protocols. In this setting, it is possible to define the time after which, in the event of a lack of communication with the server, the reader will switch to autonomous mode.
Note that when communication is regained, the reader will return to control by protocols mode.

Advanced mode settings

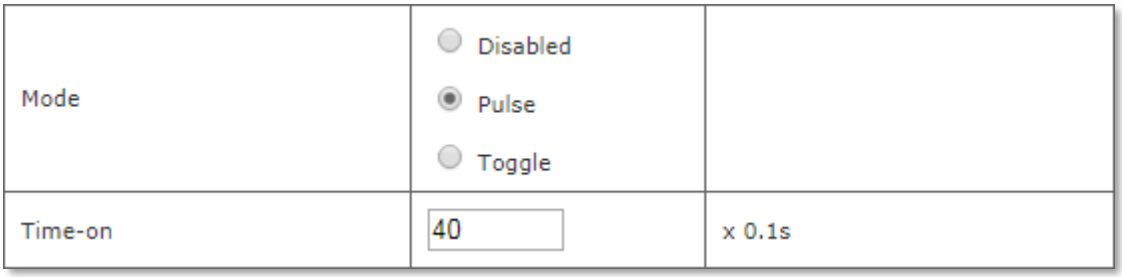


- **ID Length** – length of the ID code for comparison with the card database
- **Read Delay** – value of the delay in reading consecutive tags multiplied by 0.1s (150 = 15 seconds)

The following settings apply to protocol control only:

- **Continous reading –** so-called hotel mode - if this option is selected, the card will only be visible when it is applied to the reader, and the ID code is replaced by zeros when it is taken away.

In addition, on the I/O Settings tab in the Output Relay 1 table, the output can be set to Pulse mode. This means that after the card has been taken away, the output will still be on for a certain period of time (e.g.: 40 * 0.1s = 4 seconds).



Most commonly, such an option is used to provide power to a machine that is only activated when an authorised employee inserts a card into the reader. Similarly, in hotel systems. Power is provided to the room when the card is inserted into the reader (special-purpose enclosure).

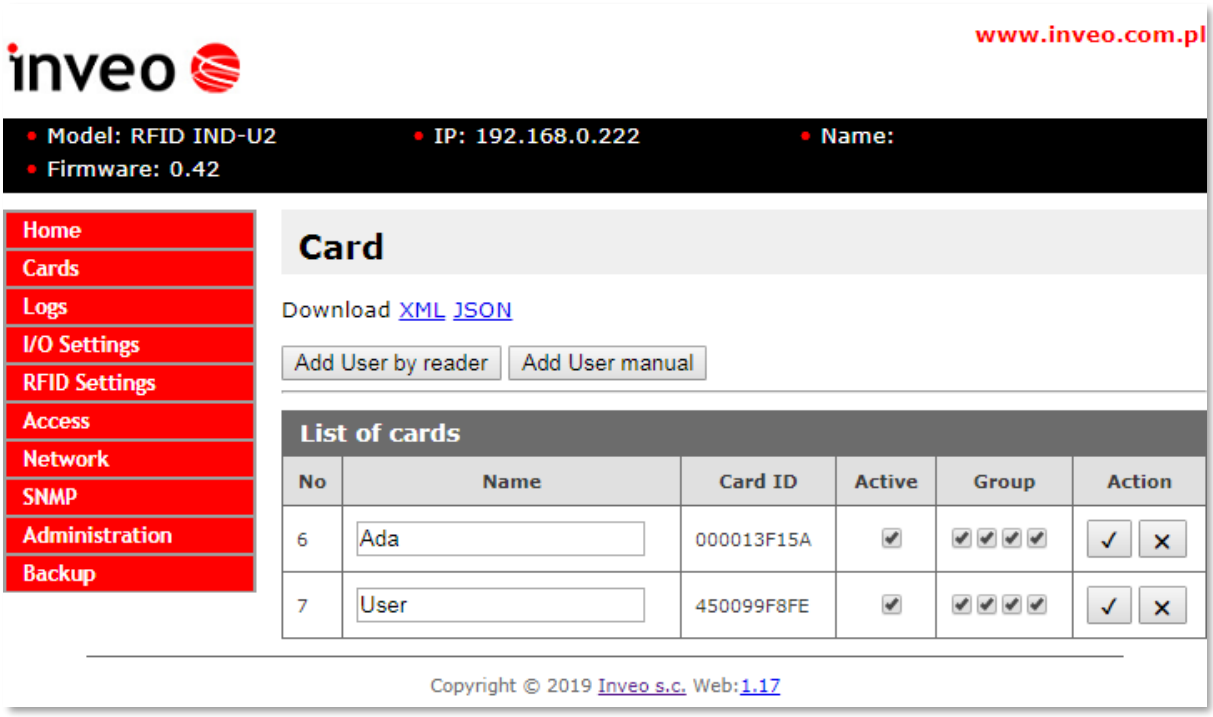
- **Prevent auto IO control –** forces software control of the outputs, LEDs and display when the reader is controlled by the protocol. This option is useful together with the setting of communication override, when the reader automatically enters autonomous mode. As a result, during software control, the server/controller must take care of user interaction. Conversely, when communication is lost and the reader automatically enters autonomous mode, the reader will respond as set in the I/O Settings tab.

7.3 Card management

The reader offers the possibility to manage cards both through the implemented communication protocols and using external software.

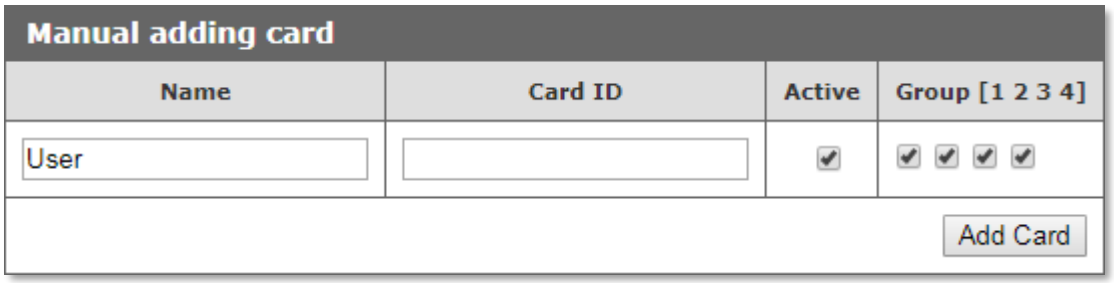
7.3.1 Adding and removing cards from the browser level

In the Cards tab, it is possible to manually assign TAGs that the reader will recognise.



To add a new user card, click on the Add User by reader button in the Cards tab and then bring the card close to the reader. The assignment of the card will be signalled by an appropriate beep.

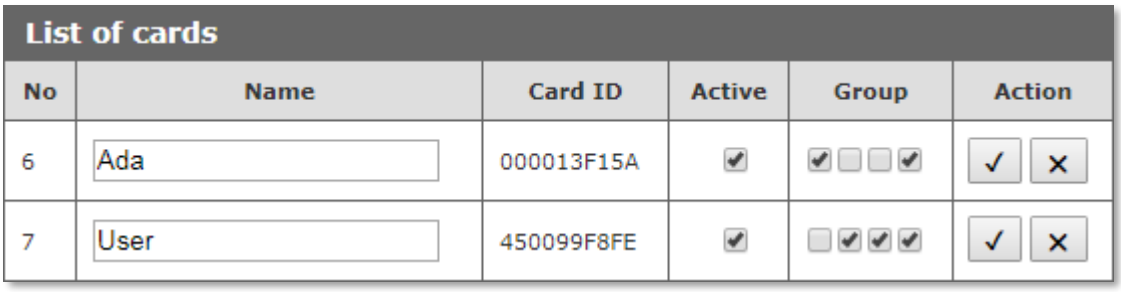
If the user does not have a physical TAG and there is a need to add a new user, click on the Add User manual button. The following window will be displayed:



In the window above, all you need to do is assign a user name, enter the card ID and specify membership of the relevant groups. The data entered should be confirmed with the Add Card button.

After the card has been correctly added to the reader, another entry will appear in the List of cards table.

The List of cards table allows editing cards stored in the module's memory.



List of cards

No	Name	Card ID	Active	Group	Action
6	Ada	000013F15A	☑	☑ ☐ ☐ ☑	✓ ✕
7	User	450099F8FE	☑	☐ ☑ ☑ ☑	✓ ✕

The Active field activates or deactivates the selected card, so that a blocked user will not be recognised when the card is applied.

The Group option is responsible for activating the card at a specific time (section 7.6 Access – card groups).

The buton ✓ is responsible for saving the current settings.

The buton ✕ allows you to remove a card from the reader's memory.

You can download all cards in the reader by referring to the resource cardList.xml (e.g.: http://192.168.111.15/cardList.xml).

Example card file:

```
<cardList>
  <cardItem>
     <no>5</no>
     <name>Ada</name>
     <active>1</active>
     <group>1001</group>
      <cardId>000013F15A</cardId>
</cardItem>
<cardItem>
      <no>6</no>
      <name>User</name>
      <active>1</active>
      <group>0111</group>
      <cardId>450099F8FE</cardId>
  </cardItem>
</cardList>
```

Cards can also be added via the RfidIndManager software.
To save cards to the reader's memory, simply select the CSV file with the card number, user name and card ID. The RfidIndManager program will establish communication with the module and save the cards to the device memory.

7.3.2 Adding a card using the Modbus protocol

The module has a built-in Modbus protocol. Using this protocol, a card and the user name of this card can be stored in the reader's memory.

In chapter 8.1 Modbus, all Modbus addresses for RFID devices are stored in a table. In the following example, the values sent to the Holding Registers addresses are decimal values.

Example for writing a Mifare card:
card number 8
card id: 4923267D
user name: John

First enable the Modbus TCP or RTU function on the module. Then connect the module to the computer using a USB to RS485 converter or TCP connection to provide Modbus communication.
The next step is to run the Modbus software and set the computer as the master station. For easier configuration, set the Holding Registers display points from address 1100 to 1149 (50 points).
After configuration, open the address editing window. To store the card number, send the value minus 1 to address 1100 Holding Registers. For number 8, send the value 7.
The next step is to select the card settings (such as card activation, group membership) by sending the appropriate value to address 1101 Holding Registers. (see chapter 8.1).

Divide the card id into bytes and convert to a decimal value:
49 (hex)=73 (dec)
23 (hex)=35 (dec)
26 (hex)=38 (dec)
7D (hex)=125 (dec)
4 bytes for Mifare cards, 5 bytes for Unique cards

Sending further card id's is done by sending further bytes to the 1102-1105 Holding Registers addresses.

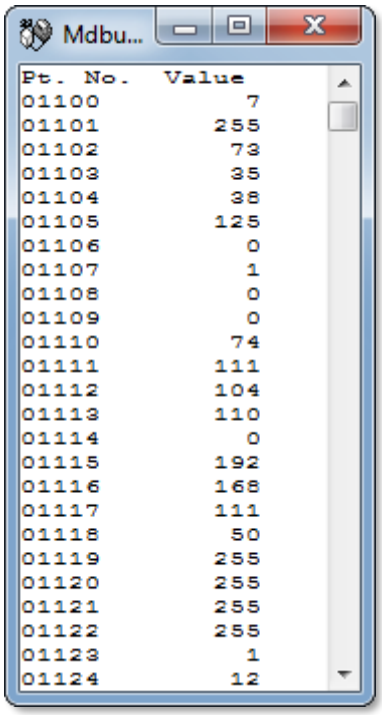
Finally, a name must be given to the card user by converting the name into an ASCII code:
J=74
o=111
h=104
n=110

After converting the user name into an ASCII code, you need to send the code values to addresses from 1110 Holding Registers (in the example 1110-1113). The name must be terminated with NULL, i.e. after the last ASCII code a 0 must be sent to the next address. (in the example to address 1114).

Example
In the example, we store a Mifare card with the number 8, card ID 4923267D and user name John. The values for the Holding Registers addresses will be as follows:

Address 1100: 7 (card number - 1)
Addresses 1102-1105: 73, 35, 38, 125 (card ID)
Addresses 1110-1113: 74, 111, 104, 110 (user name)
Address 1114: 0 (NULL)

The screenshot below shows the Holding Registers addresses with values from the card write example.



7.3.3 Customised API (http GET)

With HTTP GET protocol commands, you can add a card to the reader's memory.

To add a card, enter one of the following addresses into the browser:

- In hexadecimal notation:

" addressIPdevice"/msg.php?addId="idcard"

Example:
http://192.168.111.15/msg.php?addId=0600ADDA62

- In decimal notation:

" addressIPdevice"/msg.php?addDecId="idcard"

Example:
http://192.168.111.15/msg.php?addDecId=10000000098

The device returns the number under which it has stored the card in question:

```
Return: 8
```

Using this number, give the user name of the card by typing:
„addressIPdevice"/msg.php?changeName=*" card number"*!*" username"*

Example"
http://192.168.111.15/msg.php?changeName=8!John

The card can also be added with the command:
"addressIPdevice"/msg.php?addCard.
First read the card then enter the above command into the browser. The device saves the last tag read.
After applying this command, the number under which the card was stored will be returned.
To assign a name, use the command:
"addressIPdevice"/msg.php?changeName="card number"!"username".

More of the HTTP GET command is described in 8.4 HTTP GET – control.

7.4 Writing data to Mifare and ICODE cards - HTTP GET command

The reader has the ability to write memory blocks of Mifare Classic 1k/4k and ICODE cards. Data recording can be for any memory blocks.

7.4.1 Recording data on the ICODE card

ICODE cards are equipped with an EEPROM. They have the ability to both read and write data. The tag memory is divided into blocks, each 4 bytes long.

> Tip
>
> In order for the read/write function via HTTP GET to work correctly, it is necessary that the Read Block Module function in the RFID Settings tab is OFF!

ICODE Block Reader

Name	Value	Description
Enable	☐	Enable Read Block Module. (Only for advanced user!)
Block address	0	Address of block to read.
Overwrite UID	☐	Overwrite UID by read data.
Offset for UID	0	Start copy offset for override UID.
Add data to HTTPClient	☐	Add read data to HTTPClient.

The user is able to write and read the contents of the tag.
By sending the appropriate parameters, 16 bytes of tag memory (4 data blocks) can be written once. If fewer bytes than a multiple of 4 (the length of a single ICODE block) are written, the remaining bytes in the block will be overwritten with zeros - example no. 1.

Syntax addressIP/block.xml?block=nrbloku&data=data
block - block number for writing/reading data,
data - up to 16 bytes of data in hex format.

After sending the command, apply the card to the reader.

The reader's response will be as follows:

```
<cardContent>
  <result>0</result>
  <uid>9D720004000004E0</uid>
  <block>33333333444444445555555566666666</block>
</cardContent>
```

If the result value is different from 0, the write operation fails.

> Tip
>
> If the date field is present, a write will be performed. If it is not present then a read will occur.

Example no. 1 - sending 7 bytes of data:
http://192.168.111.15/block.xml?block=**3**&data=**1A999999111118**

Explanation:
The reader returned the following data:

```
<cardContent>
  <result>0</result>
  <uid>9D720004000004E0</uid>
  <block>1A999999111118098765421AF251536</block>
</cardContent>
```

The reader is assigned an IP address: 192.168.111.15. The data parameter accepts up to 16 bytes, while 7 bytes were sent. So, from block 3, the writing of data to the card will start:

- block 3 - 1A999999,
- block 4 - 11111800, (the remaining byte has been overwritten with zeros)
- block 5 - 98765421, (block unchanged)
- block 6 - AF251536. (block unchanged)

Example #2 - full write of 4 blocks of memory:
http://192.168.111.15/block.xml?block=4&data=BBB44444555CCC556666AAA688888FFF

Explanation:
The reader returned the following data:

```
<cardContent>
  <result>0</result>
  <uid>9D720004000004E0</uid>
  <block>BBB44444555CCC556666AAA688888FFF</block>
</cardContent>
```

The reader is assigned an IP address: 192.168.111.15. From block no. 4, data will start to be written to the card. The following data will be written to the card:

- block 4 - BBB44444,
- block 5 - 555CCC55,
- block 6 - 6666AAA6,
- block 7 - 88888FFF.

Example 3 - failed write operation:
http://192.168.111.15/block.xml?block=4&data=BBB44444555CCC556666AAA688888FFF

The reader returned the following data:

```
<cardContent>
  <result>10</result>
  <uid>9D720004000004E0</uid>
  <block>BBB44444555CCC556666AAA688888FFF</block>
</cardContent>
```

Explanation:
The data write failed because the value for result is different from 0.

7.4.2 Data recording on Mifare Classic 1k/4k

Mifare cards are equipped with an EEPROM. They have the ability to both read as well as writing data. The tag memory is divided into blocks, each 16 bytes long.

> Tip
>
> In order for the read/write function via HTTP GET to work correctly, it is necessary that the Read Block Module function is OFF in the RFID Settings tab!

Mifare

Name	Value	Description
Enable	☐	Enable Read Block Module. (Only for advanced user!)
Block address		Address of block to read.
Auth key		Key in hex format (12 characters). Leave empty when not change. Default key is: FFFFFFFFFFFF.
Auth key B	☐	Authenticate by key B. Default is key A.
Overwrite UID	☐	Overwrite UID by read data.
Offset for UID	0	Start copy offset for override UID.
Add data to HTTPClient	☐	Add read data to HTTPClient.

The user is able to write and read the contents of the tag.
By sending the appropriate parameters, 16 bytes, i.e. one whole block of tag memory, can be written once. If fewer bytes (less than 16) are written, the remaining bytes of the block will be overwritten with zeros.

Syntax:
adresIP/block.xml?block=**nrbloku**&data=**dane**&keyType=**AlubB**&keyIdx=**0-7**

- **block** - block number for writing/reading data,
- **data** - up to 16 bytes of data in hex format (if the parameter is present - writing will take place, if the parameter is absent - reading will take place),
- **keyType** - authorization type: key A or key B,
- **keyIdx** - key index from memory (0-7) - description below.

After the command is sent, the card should be inserted into the reader.

The user can set up to 8 Mifare keys. A specific key must be referenced by keyIdx. By default, all keys (0-7) have the value FFFFFFFFFFFF.
We recommend that the key with index '0' remains unchanged. Mifare tags have the default keys A and B set to the value FFFFFFFFFFFFFF.

To set the Mifare key it is necessary to send 2 parameters:

- **keyIdx –** location number,
- **key –** key in hex format (6 bytes → 12 hex characters).

Syntax:
adresIP/block.xml?keyIdx=**nrlokacji**&key=**key**

Once the command has been sent, apply the card to the reader.

Example of key writing:
http://192.168.111.15/block.xml?keyIdx=2&key=FFB7FFB7FFB7

Explanation:
The reader returned the following data:

```
<cardContent>
  <result>12</result>
  <uid>00000000000000</uid>
  <block>00000000000000000000000000000000</block>
</cardContent>
```

Note (for key writing).
If the result value is equal to 12 - key write operation performed successfully.
If the result value is equal to 11 - key writing operation failed.

Example no. 1 - writing to block no. 34:
http://192.168.111.15/block.xml?block=34&data=39383736353433AA&keyType=A&keyIdx=1

Explanation:
The reader returned the following data:

```
<cardContent>
  <result>0</result>
  <uid>0000001CC60D0D</uid>
  <block>39383736353433AA0000000000000000</block>
</cardContent>
```

The result value is equal to '0', so the write operation performed successfully.

Authentication with key 'A' (keyType=A) from location No. '1' (keyIdx=1) was used.
The data parameter accepts up to 16 bytes. As 8 bytes were sent, the remaining 8 bytes in block no. 34 were overwritten with zeros.

Example no. 2 - failed write operation:
http://192.168.111.15/block.xml?block=34&data=ABA00D&keyType=B&keyIdx=1

Explanation:
The reader returned the following data:

```
<cardContent>
  <result>10</result>
  <uid>0000001CC60D0D</uid>
  <block>ABA00D0000000000000000000000000000</block>
</cardContent>
```

The data write failed because the value for result is different from 0.

Authentication with key 'B' (keyType=B) from location number '1' (keyIdx=1) was used.

7.5 Reading Mifare and ICODE blocks

The reader has the ability to read Mifare Classic 1k/4k and ICODE card memory blocks. Any block can be read from memory. The read data is available via the HTTPClient protocol and Modbus (from address 100 Holding Registers). In HTTP Client protocol mode, a "block" field with the hexadecimal value of the read data (16 bytes) is added to the GET request. In addition, it is possible to force the assignment of the read block as the UID number of the card. In this case, as many bytes of the block are copied as the card's UID number (e.g. 8 bytes for ICODE tags).

7.5.1 Reading the contents of an ICODE tag - Block Reader

In the RFID Settings tab of the ICODE Block Reader table, you can configure the block reading of the ICODE card - HTTP Client, Modbus, standalone operation.

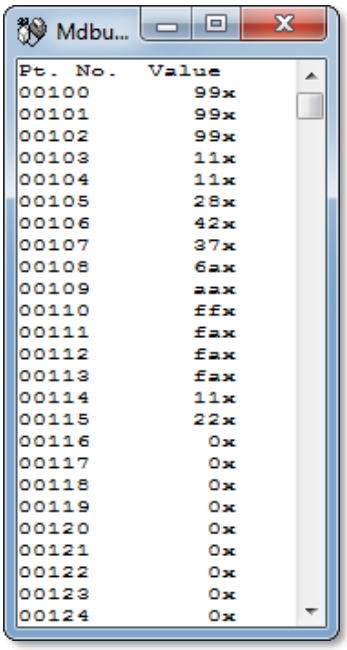
ICODE Block Reader

Name	Value	Description
Enable	☑	Enable Read Block Module. (Only for advanced user!)
Block address	3	Address of block to read.
Overwrite UID	☑	Overwrite UID by read data.
Offset for UID	0	Start copy offset for override UID.
Add data to HTTPClient	☐	Add read data to HTTPClient.

Meaning of the individual fields:

- **Enable** - activation of the block reading function,
- **Block address** - address of the block to be read,
- **Override UID** - overwriting of the card identifier by the read data,
- **Offset for UID** - number of byte in the block from which the reading of data starts,
- **Add data to HTTPClient** - the read block will be added to the GET request in HTTP Client mode.

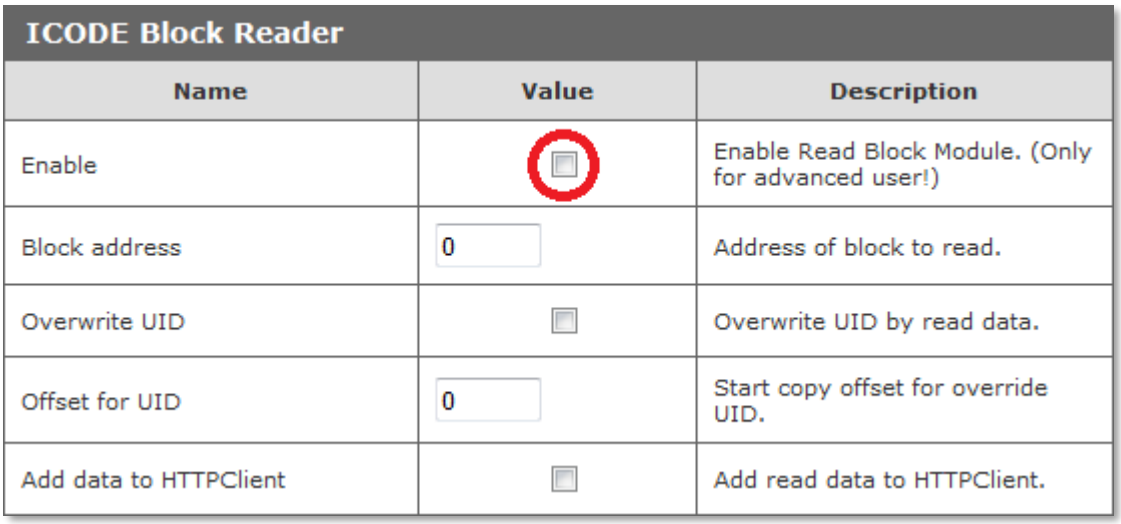
Block reading via MODBUS protocol - data contains Hold registers. Reg. 100 - 115:



7.5.2 Reading the contents of an ICODE tag - HTTP GET command

ICODE cards are equipped with an EEPROM. They have the ability to both read and write data. There are 1024 bits of memory (128B) available to the user, divided into 32 blocks, each block being 4 bytes long.

> **Tip** In order for the read/write function via HTTP GET to work correctly, it is necessary that the Read Block Module function is OFF in the RFID Settings tab!



ICODE Block Reader

Name	Value	Description
Enable	☐	Enable Read Block Module. (Only for advanced user!)
Block address	0	Address of block to read.
Overwrite UID	☐	Overwrite UID by read data.
Offset for UID	0	Start copy offset for override UID.
Add data to HTTPClient	☐	Add read data to HTTPClient.

The user is able to write and read the contents of the tag. In order to read the contents of the card, it is sufficient to use the command: addressIP/block.xml?block=blocknumber.

After sending the command, apply the card to the reader.

The response of the reader will be as follows:

```
<cardContent>
  <result>0</result>
  <uid>9D720004000004E0</uid>
  <block>33333333444444445555555566666666</block>
</cardContent>
```

If the result value is different from 0 - the read operation has failed.

7.5.3 Reading a Mifare Classic 1k/4k block - Block Reader

In the RFID Settings tab of the Mifare Block Reader table, you can configure the block reading of the Mifare card - HTTP Client, Modbus, standalone operation.

Mifare		
Name	**Value**	**Description**
Enable	☑	Enable Read Block Module. (Only for advanced user!)
Block address	34	Address of block to read.
Auth key		Key in hex format (12 characters). Leave empty when not change. Default key is: FFFFFFFFFFFF.
Auth key B	☐	Authenticate by key B. Default is key A.
Overwrite UID	☑	Overwrite UID by read data.
Offset for UID	0	Start copy offset for override UID.
Add data to HTTPClient	☐	Add read data to HTTPClient.

Meaning of the individual fields:

- **Enable** - to enable the block reading function,
- **Block address** - address of the block to be read,
- **Auth key** - authentication key in HEX format - the key is stored in secure memory (write-only). Only to be entered if a change is required. Default key 6xff (FFFFFFFFFFFF).
- **Auth key B** - key selection. Default authentication key is key "A", If key "B" must be used this option must be checked.
- **Override UID** - overwrite the card identifier with the read data,
- **Offset for UID** - the character from which the data reading starts,
- **Add data to HTTPClient** - the read block will be added to the GET request in HTTP Client mode.

Tip

Entering an incorrect key results in the reader not beeping. Such a state is treated as a card not being read.

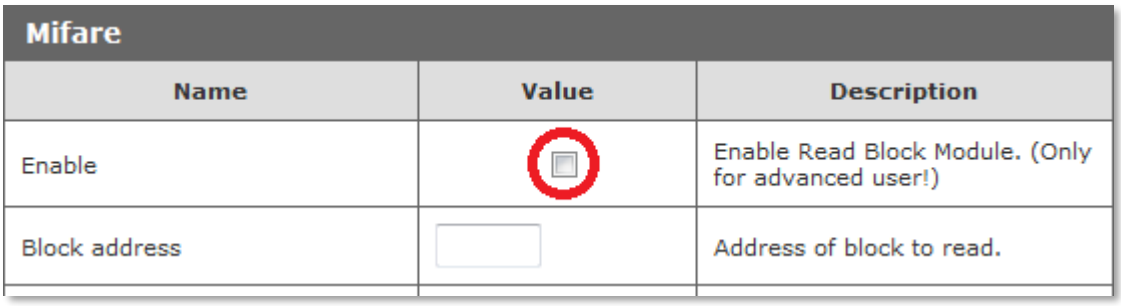
Example of reading a block via HTTPClient protocol:
GET?mac=801F127B8E7B&ip=192.168.0.236&name=&id=697C0004000004E0&inout=0&**block**=99999911112842376AAAFFFAFAFA1122&io=0&put=1

Block reading via MODBUS protocol - data contains registers Hold. Reg. 100 - 115.

7.5.4 Reading the contents of a Mifare Classic tag - HTTP GET command

Tip

In order for the read/write function via HTTP GET to work correctly, it is necessary that the Mifare Block Reader function is OFF in the RFID Settings tab!



Mifare		
Name	**Value**	**Description**
Enable		Enable Read Block Module. (Only for advanced user!)
Block address		Address of block to read.

The user has the option to write and read the contents of the tag. By sending the appropriate parameters, 16 bytes, i.e. one whole block of tag memory, can be read out at a time.

Syntax:
adresIP/block.xml?block=**nrbloku**&keyType=**AlubB**&keyIdx=**0-7**

- **block –** block number for reading data,
- **keyType –** authorization type: key A or key B,
- **keyIdx –** key index from memory (0-7) - description below.

After sending the command, apply the card to the reader.

Tip

Entering an incorrect key results in the reader not beeping. Such a condition is treated as a card not being read.

The response of the reader will be as follows:

```
<cardContent>
  <result>0</result>
  <uid>0000001CC60D0D</uid>
  <block>ABA00D0000000000000000000000000</block>
</cardContent>
```

Note: If the result value is different from 0 - the reading operation has failed.

The user can set up to 8 Mifare keys. A specific key should be referenced by keyIdx. By default, all keys (0-7) have the value FFFFFFFFFFFF.
We recommend that the key with index '0' remains unchanged. Mifare tags have the default keys A and B set to the value FFFFFFFFFFFFFF.

To set the Mifare key it is necessary to send 2 parameters:

- keyIdx - location number,
- key - key in hex format (6 bytes → 12 hex characters).

Syntax:
addressIP/block.xml?keyIdx=location-number&key=key

After sending the command, apply the card to the reader.

Example of a key entry:

http://192.168.111.15/block.xml?keyIdx=2&key=FFB7FFB7FFB7

Explanation:
The reader returned the following data:

```
<cardContent>
  <result>12</result>
  <uid>00000000000000</uid>
  <block>00000000000000000000000000000000</block>
</cardContent>
```

Note (for key writing).
If the result value is equal to 12 - key write operation performed successfully.
If the result value is 11 - key writing operation has failed.

Example no. 1 - reading block no. 14:
http://192.168.111.15/block.xml?block=14&keyType=A&keyIdx=3

Explanation:
The reader returned the following data:

```
<cardContent>
  <result>0</result>
  <uid>0000001CC60D0D</uid>
  <block>99000000000000000000000000000000</block>
</cardContent>
```

The result value is equal to '0', so the read operation performed successfully.
Authentication with key 'A' (keyType=A) from location '3' (keyIdx=3) was used.

Example no. 2 - unsuccessful data read operation:
http://192.168.111.15/block.xml?block=14&keyType=B&keyIdx=3

Explanation:
The reader returned the following data:

```
<cardContent>
  <result>10</result>
  <uid>0000001CC60D0D</uid>
  <block>00000000000000000000000000000000</block>
</cardContent>
```

The result value is different from '0', so the data read operation failed.

Authentication with key 'B' (keyType=B) from location number '3' (keyIdx=3) was used.

7.6 Access – card groups

The reader allows selected cards to be activated at a specific time. This means that you must set the card activation time from which the user will be recognised in the system. Groups are configured in the Access tab.

Access Groups

No	Enable rule	Start time	Group 1	Group 2	Group 3	Group 4	Save
1	☑	07:00	☑	☐	☐	☐	✓
2	☑	15:00	☑	☑	☐	☐	✓
3	☑	21:00	☐	☐	☑	☑	✓
4	☐	06:00	☐	☐	☐	☐	✓

In the Access Groups table, 4 conditions can be defined which can be assigned to 4 groups. The relevant fields in the table mean:

- **enable rule** - activation of the selected condition,
- **start time** - the time from which the groups are activated and the previous ones deactivated (at 21:00 groups 1 and 2 are deactivated and groups 3 and 4 are activated),
- **group 1-4** - assignment of the condition to the respective groups.

The configuration of each condition should be saved using the Save button.

Tip

If the Enable rule box is not checked, assigning a group to this condition will not unlock the card.

If several groups with different card activation conditions are assigned to a card, the condition that causes the card to be activated will take precedence.

Failure to assign the card to any of the groups is equivalent to deactivation.

Example no. 1:

Only conditions No. 1 and No. 2 are attached.

Access Groups

No	Enable rule	Start time	Group 1	Group 2	Group 3	Group 4	Save
1	☑	07:00	☑	☐	☐	☐	✓
2	☑	14:00	☑	☑	☑	☑	✓

Cards with group no. 1 ticked will be active all day.
Cards with groups No. 2, No. 3 and No. 4 ticked will be active from 14:00 to 7:00.

Example No. 2:

Only conditions No. 1, No. 3 and No. 4 are included.

Access Groups

No	Enable rule	Start time	Group 1	Group 2	Group 3	Group 4	Save
1	☑	07:00	☑	☐	☐	☐	✓
2	☐	14:00	☑	☑	☐	☐	✓
3	☑	15:00	☐	☐	☑	☑	✓
4	☑	21:00	☐	☐	☐	☑	✓

List of cards

No	Name	Card ID	Active	Group	Action
6	Olaf	0600A4638A	☑	☐☑☑☐	✓ ✕
7	John	450099F8FE	☑	☑☐☐☑	✓ ✕
8	Rex	04001BB7BC	☑	☑☑☑☐	✓ ✕

Groups:
Before 7:00 a.m. group 4 is activated from 9:00 p.m. At 7:00 a.m. group 4 is deactivated and group 1 is activated. At 2:00 p.m. condition no. 2 starts, which should delete condition no. 1. The Enable rule field is not checked, so the condition is not taken into account by the reader. At 3 p.m. group 1 is deactivated, while groups 3 and 4 are activated. From 9 p.m. to 7 a.m. only group 4 is active. group 2 is inactive all day.

Users:
To the user Olaf are assigned groups No. 2 and No. 3. The user will be recognised in the system between 15:00 and 21:00. Only then is group 3 active.

User John is assigned groups 1 and 4. This means that the card will be active all day. From 7 a.m. the condition for group 1 is fulfilled. At 3 p.m. group 1 is deactivated, but group 4 is activated, which is deactivated at 7 a.m.

User Rex is assigned groups 1, 2 and 3. This user's card will be active from 7 a.m. to 9 p.m. At 7 a.m., the condition for group 1 is fulfilled. At 3 p.m., group 1 is deactivated, but group 3 is simultaneously activated. At 9 p.m., the condition for group 3 is deactivated.

7.7 Event logging

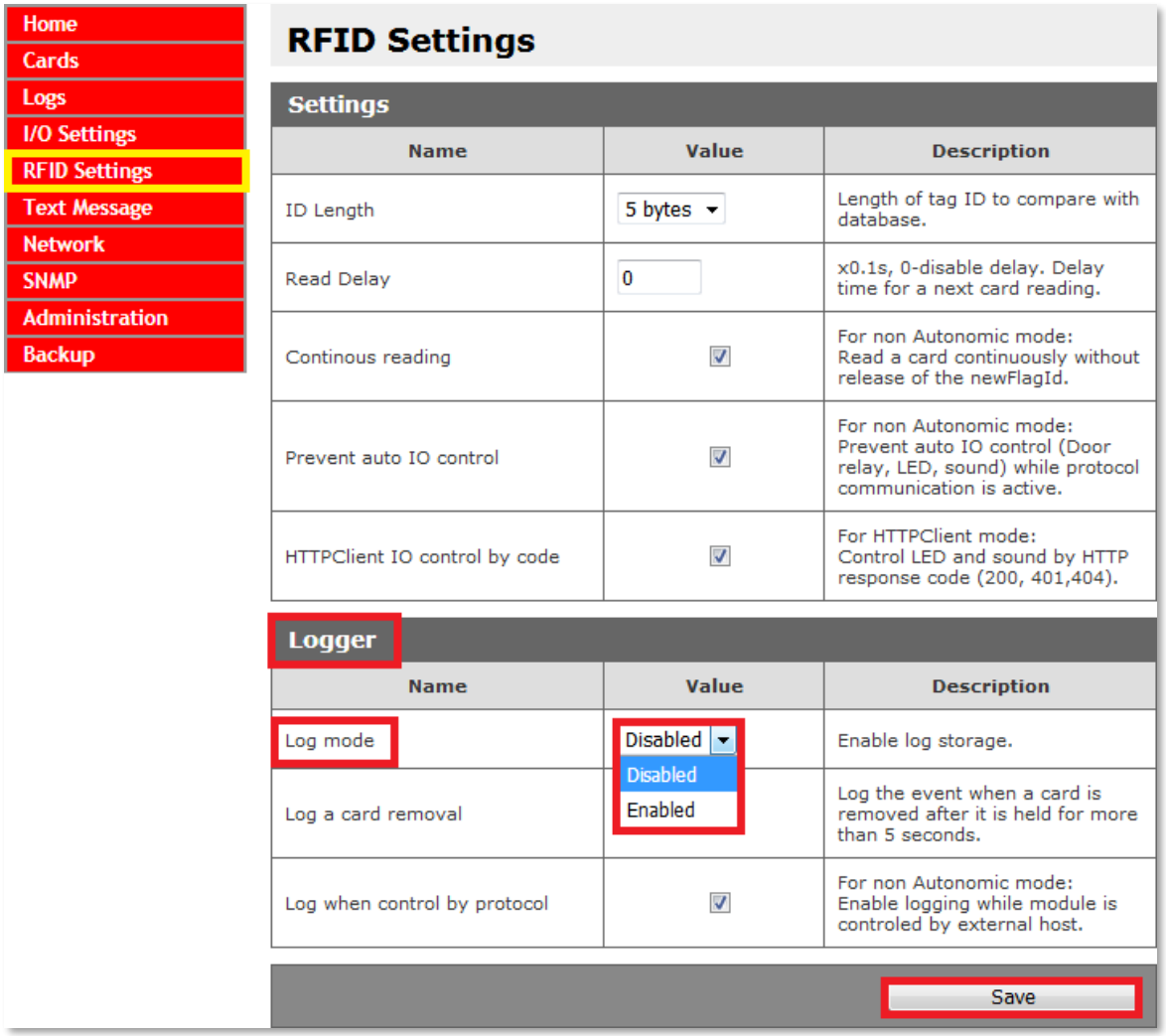
The reader has a possibility to log events of tag assignments.

In order to have the reader log RFID tag assignments in the memory, the appropriate Log mode should be selected in the RFID Settings→ Logger menu.

Preview of events is available through:

- WWW page (Logs tab)
- XML file
- modbus - reading from appropriate addresses (chapter 8.1 Modbus)
- raw file

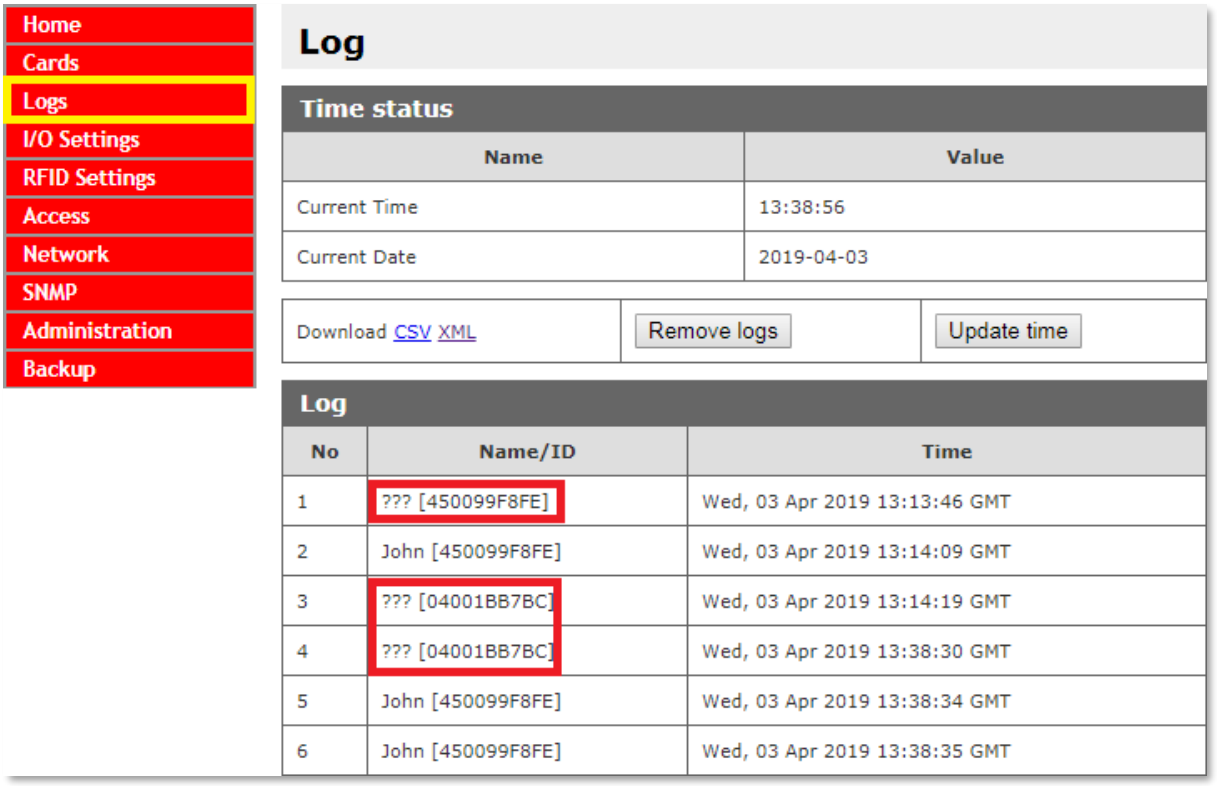
Configuration of events



- **Log mode: Disabled** – logging of logs disabled,
- **Log mode: Enabled** – logging of events to internal memory enabled,
- **Log a card removal** – ticking this option will additionally cause logging of an event of tag removal from a reader, if it was applied longer than 5sec,
- Option available only in 'Continuos reading' mode,
- **Log when controlled by protocol** - this option allows logging events to be recorded even when the reader is controlled by protocol. By default, when control by protocol is selected, the reader does not log events.

7.7.1 Viewing events via the Web page

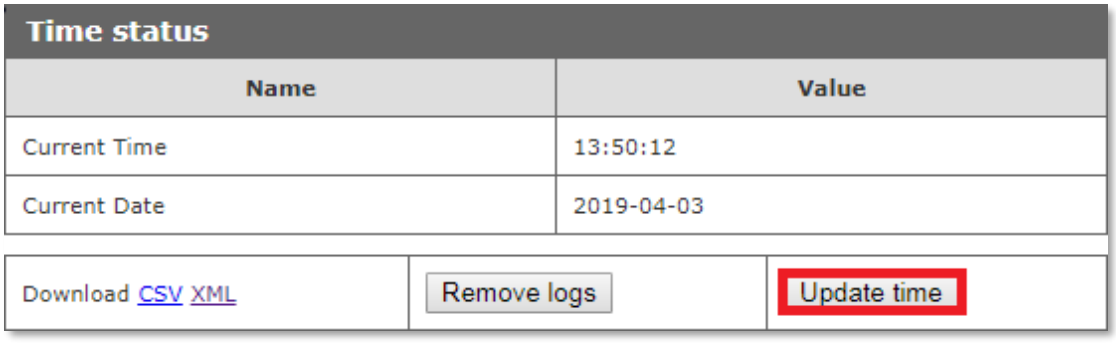
Log table contains history of card readings.



Marked cards mean attempt to read a card which is not stored in reader's memory or is inactive.

Clicking Remove logs button will remove all logs from reader's memory.

In addition, the reader has a built-in real-time clock. Clicking on the Update time button will synchronise the internal clock with the current time set in the computer.



7.7.2 Viewing events via XML file

The user can download all the logs present in the reader by referring to the resource addressIP/logList.xml or in the Logs tab press the Download XML line - example below**:**

```
<logList>
  <logItem>
      <no>1</no>
      <id>0</id>
      <name/>
      <cardId>450099F8FE</cardId>
      <state>00000110</state>
      <time>1554297226</time>
  </logItem>
  <logItem>
      <no>2</no>
      <id>7</id>
      <name>John</name>
      <cardId>450099F8FE</cardId>
      <state>10000110</state>
      <time>1554297249</time>
  </logItem>
  <logItem>
      <no>3</no>
      <id>0</id>
      <name/>
      <cardId>04001BB7BC</cardId>
      <state>00000110</state>
      <time>1554297259</time>
  </logItem>
</logList>
```

Meaning of individual tags:

- **no** - read number,
- **id** - the id number of the defined card from the Cards tab minus 1,
- **name** - the name assigned by the user. For unsaved or inactive cards this tag is empty,
- **cardID** - the UID number of the card,
- **state** - applies only to RFID RCP device with keys - a flag indicating which function keys have been selected:

BIT	Decimal value	Description
0	1	Function key no. 1 (e.g. private)
1	2	Function key 2 (e.g. business)
2	4	Illuminated key 1 (e.g. entrance)
3	8	Illuminated key no. 2 (e.g. exit)
4	16	Reserved
5	32	Reserved
6	64	Reserved
7	128	Card recognition by the reader

- **time** – time expressed in Unix format.

7.7.3 Downloading recorded logs in CSV format

The option to download the logs recorded by the RFID TAB in CSV format is available under the Logs tab. By clicking on the CSV icon, a CSV file containing the logged events will be downloaded.

No	UserId	CardId	Name	Date	Time	Flags	Timestamp
1	0	???	40005AEF4B	24.02.2012	08:59:41	68	1707728381
2	0	???	5500D08509	24.02.2012	08:59:48	68	1707728388
3	0	???	5500D08509	24.02.2012	09:00:17	68	1707728417
4	0	???	40005AEF4B	24.02.2012	09:00:36	68	1707728436
5	7	Marek Nowak	40005AEF4B	24.02.2012	09:00:54	132	1707728454
6	6	Jan Kowalski	5500D08509	24.02.2012	09:00:59	1160	1707728459
7	6	Jan Kowalski	5500D08509	24.02.2012	09:01:01	1162	1707728461
8	7	Marek Nowak	40005AEF4B	24.02.2012	09:01:03	1161	1707728463
9	7	Marek Nowak	40005AEF4B	24.02.2012	09:01:05	1156	1707728465
10	7	Marek Nowak	40005AEF4B	24.02.2012	09:01:06	1157	1707728466
11	6	Jan Kowalski	5500D08509	24.02.2012	09:01:08	1156	1707728468
12	0	???	5400EEA1EF	24.02.2012	10:21:11	68	1707733271

The CSV file contains event information such as sequence number, card user ID (if defined), card ID number, card user name (if defined), date and time of the event and a flag representing the card application mode. In addition, for accuracy, it also includes a timestamp in Unix time format.

A flag translation table can be used to interpret the data in the CSV file.

		binar	dec	hex
LOG_CARD_ON	Card insertion	0100 0000	64	40
LOG_CARD_AWAY	Card withdrawal	1100 0000	192	C0
LOG_CARD_KNOWN_ON	Touching a known card	1000 0000	128	80
LOG_POWER_ON	Log from external reader	1111 1111	255	FF
LOG_LONG_OPEN	Long card hold	0001 0000 0000	256	100
LOG_EXT_READER		0010 0000 0000	512	200
LOG_OUT1	Current state of output 1	0100 0000 0000	1024	400
LOG_OUT2	Current state of output 2	1000 0000 0000	2048	800
LOG_IN1	Current state of output 1	01 0000 0000 0000	4096	1000
LOG_IN2	Current state of output 2	10 0000 0000 0000	8192	2000

For example, if the value of the flag is 128, this means that a card has been applied and the flag 1024 indicates the current state of output 1. By summing these values, e.g. for flag 1152, we can conclude that the application of the card has also triggered a change in the state of output 1.

With this option, the user can easily download and interpret the logs recorded by the RFID device in CSV format for further analysis and data management.

7.8 Responses to events (inputs/outputs) - I/O Settings

The reader can automatically control the audible signalling, the optical signalling, lock relay and alarm status. These functions are described in the tables below.

> **Tip** If the reader is set up for software control (section 7.2 Operation modes of the reader), additional functions controlling the outputs as well as the audible and visual signalling must be configured.

In the Input 1 table, selecting Door unlock forces the output relay 1 (deadbolt) to be automatically activated when input 1 is shorted.

Input 1		
Name	**Value**	**Description**
Door unlock	☑	Trigger an Open state for Output 1.

.

Table Input 2 allows you to set the response to the opening of Input 2.

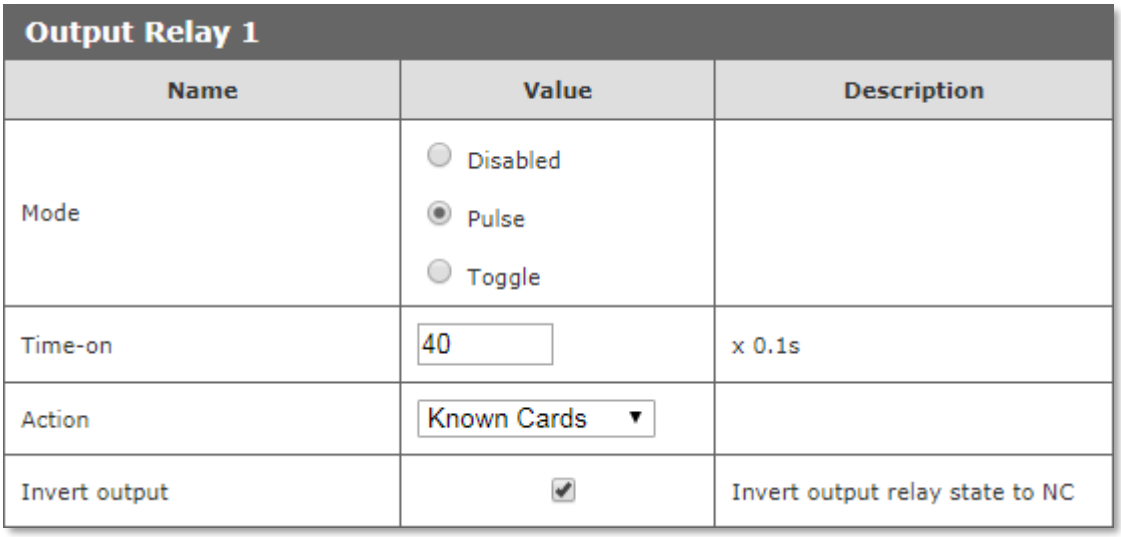
Input 2		
Name	**Value**	**Description**
Open Timeout Alarm	☑	Override 'Output 2' for alarm state when door is open for long time.
Max Time	5	x 1s

Checking the box Open Timeout Alarm activates the function of automatic activation of relay No. 2 when input No. 2 is open for longer than the time specified in the Max Time field.

Using this function, a door open sensor can be connected to input 2 and an alarm siren to output 2. This way, if someone opens the door and it is not closed within the specified time or is not closed, the alarm siren will be activated.

Save all changes by clicking Save.

In the table Output Relay 1, Output Relay 2 you can configure the relay outputs of the module.



The following table explains the individual fields:

no	Name	Description
1	Mode	**Disabled –** disables the relay control **1-Pulse** – when the output is activated, the relay is switched on for a defined period of time (e.g. control of a bolt), **Toggle** – the TAG reading event changes the status of the output to the opposite.
2	Time-on	Time of activation of relay in 1-Pulse mode given in unit x*0.1s (value 20 is 2 seconds).
3	Action	Action that causes activation of the output: **None** (control by protocol) - control is via HTTP, SMNP, MODBUS protocol, **All Cards** - activation of the output each time an RFID tag is applied, regardless of whether the TAG is stored in the module's memory, **Known Cards** - activation of the output only after an active tag is applied (stored in the reader's memory). **Unknown Cards** - activation of the output only after an unknown or inactive tag has been applied (for example: for warning purposes).
4	Invert output	Changes the relay output type from NO to NC output.

The relay outputs can be configured separately.

7.9 Long open – I/O Settings

The Long Open option is only available for relay output 1 and allows it to be switched on for a longer period of time, up to a maximum of 38912 seconds.

Long Open	150	Enable relay for given time (x 1s) when card is hold for 5 seconds.

Activation of the Option

- The option is only effective when the relay output is set to **Pulse mode**.
- The user card must be active and assigned to **group 4**.
- Group 4 does not have to be active, but the card must belong to it.

> Tip
>
> If the card is not assigned to group 4, the Long Open option will not be activated.

Example:

- Setting the output on the **I/O Settings** tab:

Output Relay 1		
Name	**Value**	**Description**
Mode	○ Disabled ◉ Pulse ○ Toggle	
Time-on	40	x 0.1s
Action	Known Cards ▾	
Invert output	☐	Invert output relay state to NC
Long Open	10	Enable relay for given time (x 1s) when card is hold for 5 seconds.

- **User2 card:** Assigned to group no. 2 and group no. 4 (on the Cards tab). Groups 1 and 2 are currently active.

No	Name	Card ID	Active	Group	Action
7	User2	B4F1A3FA	☑	☐ ☑ ☐ ☑	✓ X

When the User2 card is brought close to the reader, output No. 1 will be activated for 4 seconds (0.1*40). If the card is held against the reader for more than 5 seconds, the output will be activated for an additional 10 seconds. The operation will be as follows:

- Output 1 activates for 4 seconds.
- Output switches off for 1 second.
- Output turns on again for 10 seconds.

Audible signalling

- When a card is approached, the user will hear a standard reading tone.

- If the card is held against the reader for more than 5 seconds, the device will beep in a single tone.
- The last, high tone signals the activation of the **Long Open** function.

With this option, relay output 1 can be configured for a longer period of time, which can be useful in a variety of applications requiring a longer opening.

7.10 Relay activation on another Inveo device – I/O Settings

The Bridge Output Relay 1 to external Inveo device function allows a status change command to be sent to the output of another Inveo device. This makes it possible, for example: to open an electric door strike not connected directly to the reader. Communication takes place via the LAN.

Bridge Output Relay 1 to external Inveo device

Name	Value	Description
Enable	☑	
Server IP	192.168.0.239	A.B.C.D
Port	9761	Default 9761 (UDP)
Channel	1	

Save

Meaning of individual fields:

- **Enable** – enabling the relay status transfer service,
- **Server IP** – IP address of the Inveo device serving the output to be controlled by the reader,
- **Port** – port on which the remote device is listening - default 9761 (UDP),
- **Channel** – the channel number of the remote device to be controlled.

This service transfers the status of relay output 1 at the reader to a remote device (e.g. LanTick). The relay at the remote device is activated when relay output 1 at the reader is activated, so it is important what is set in the Output Relay 1 table.

Output Relay 1

Name	Value	Description
Mode	○ Disabled ◉ Pulse ○ Toggle	
Time-on	40	x 0.1s
Action	All Cards	

Tip

If Relay 1 at the reader is set to Pulse mode, it is necessary to configure the output at the remote device accordingly (Time Based mode), so that the output at this device works in the same way as at the reader. Applying a card will only turn the output on. The remote device must automatically deactivate the relay after a certain time has elapsed.

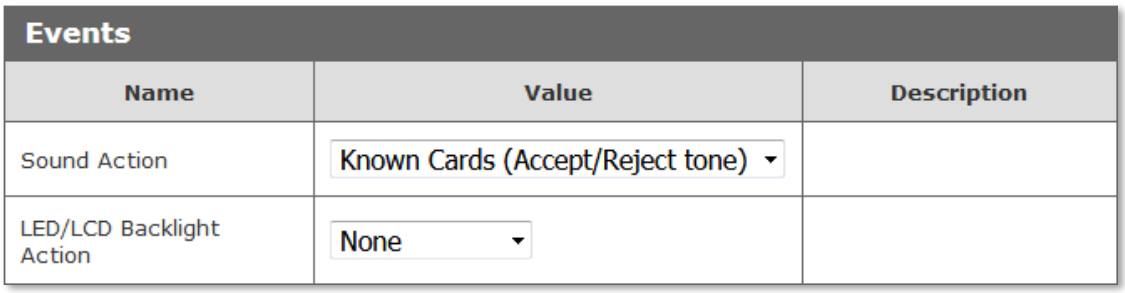
Configuration of remote modules
To enable the service, go to the Administration tab and select Enable Program Access, then confirm with Save Config.

7.11 Audio-visual signalling – I/O Settings

The reader has 3 types of audible signal:

- **Accept tone** – two tones in succession: the first a short low tone and the second a higher and longer tone. Positive tone,
- **Reject tone** – two tones following each other: the first tone as for the accept tone (short tone) and the second tone lower and longer than the first tone. Negative sound,
- **Short beep** – only a short tone from the accept/reject sound.

In the Events table, the reader behaviour can be programmed when an RFID tag is applied.



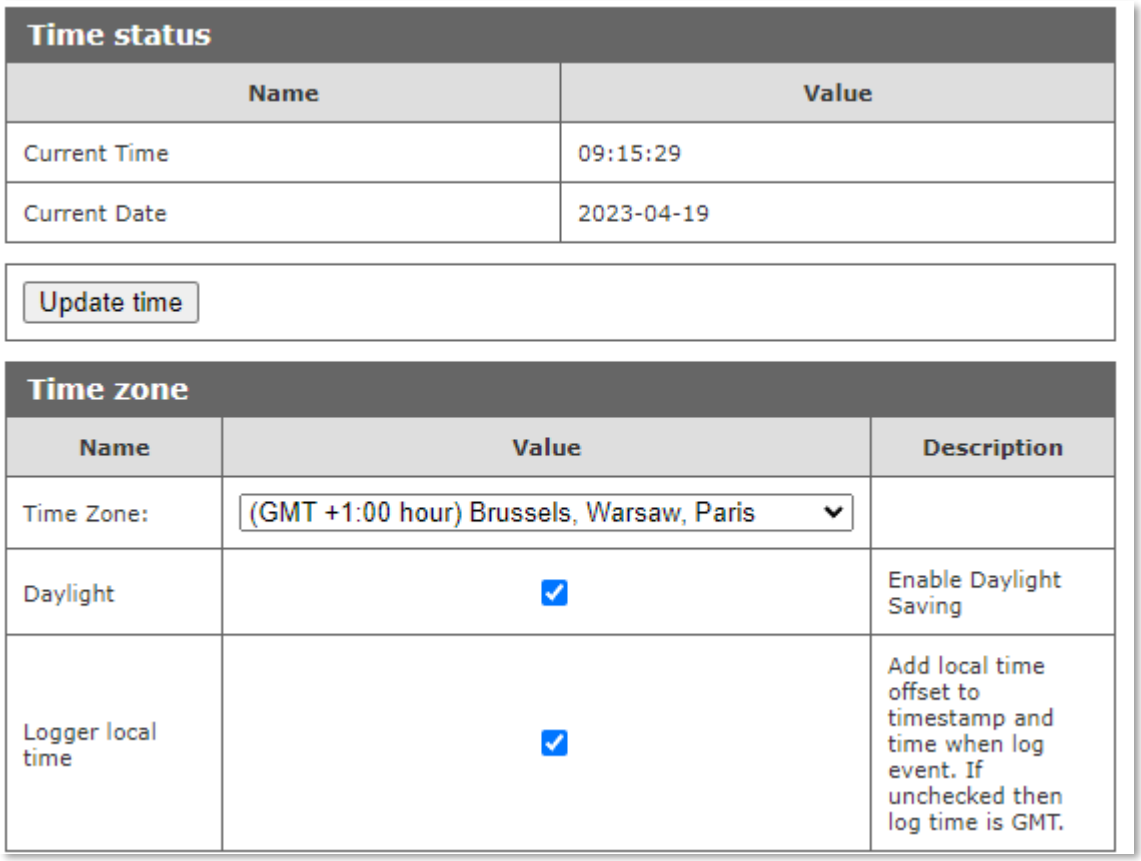
No	Name	Dscription
1	Sound Action	Action that activates the beep. **None** – no beep. **All Cards** (Accept tone) – the module emits an accept tone each time an rfid tag (known and unknown) is applied. **All Cards** (Short beep) – short beep for each tag*. **Known Cards** – generates a positive beep for known tags (stored in memory) and a negative beep for unknown tags
2	LED Backlight Action	**None** – control is via HTTP, SMNP, MODBUS protocol. **All Cards** – signalling when each rfid tag is applied **Known Cards** – signals when a known tag is applied (stored in the reader's memory).

* A short beep can be used to communicate between the reader and the server.
In such a case, the reader with a short signal indicates that it has read the tag, but only the server should signal the corresponding processing of the information.

7.12 Real-time clock (RTC)

The readers are equipped with an internal real-time clock (RTC).

The time can be set manually on the Time settings tab, under Time status using the Update time button. The time will be downloaded from the system. The time zone can be set from the drop-down list. If the Daylight saving option is selected, summer and winter time will be enforced.



Time status

Name	Value
Current Time	09:15:29
Current Date	2023-04-19

Update time

Time zone

Name	Value	Description
Time Zone:	(GMT +1:00 hour) Brussels, Warsaw, Paris	
Daylight	☑	Enable Daylight Saving
Logger local time	☑	Add local time offset to timestamp and time when log event. If unchecked then log time is GMT.

In addition, the module has the option of synchronising the internal clock with a time server using the SNTP protocol. The settings can be configured in the NETWORK → SNTP tab.
To enable synchronisation with an SNTP server, enter the relevant SNTP server address in the Server field. Entering the address 0.0.0.0 disables time synchronisation.



SNTP

Server	0.0.0.0	A.B.C.D (0.0.0.0 to disable)

7.13 Configuring the SNMP server

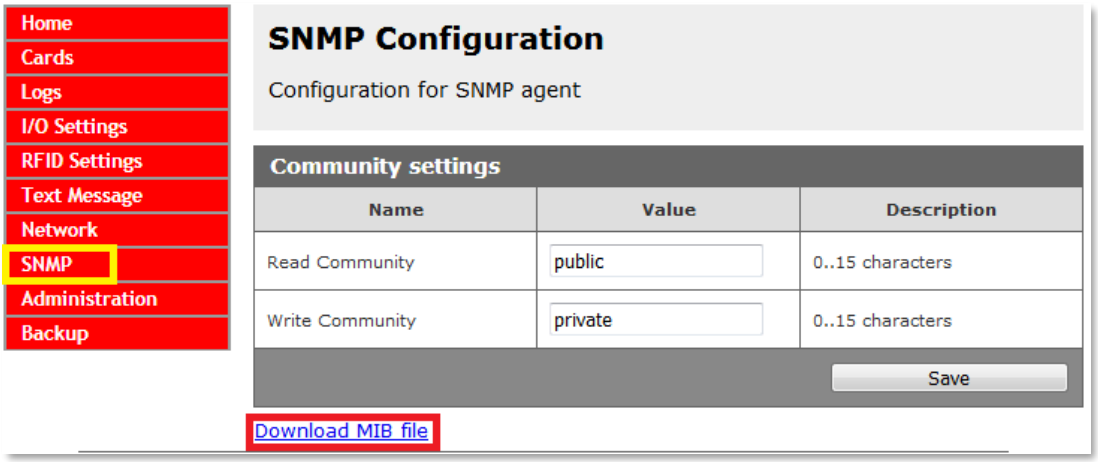
The module is equipped with an SNMP v2c server. To enable this function, go to the Administration tab → Services window and select Enable SNMP.



The SNMP protocol makes it possible to:

- Retrieving input states
- Setting output states
- Retrieving read tag IDs

The MIB file, which describes the SNMP structure, can be downloaded from the SNMP tab by clicking Download MIB file.

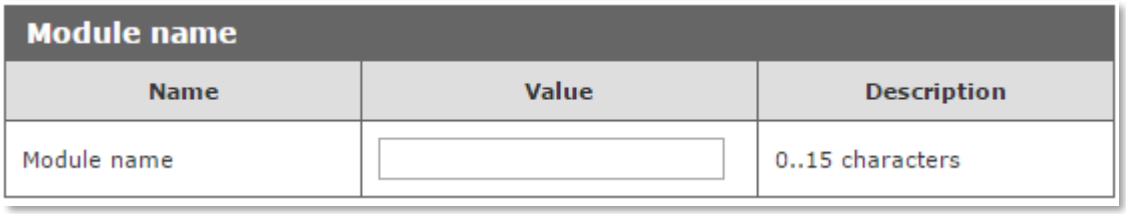


7.14 Communication protocols and administration

The Administration menu allows active services to be configured and access passwords to be changed.

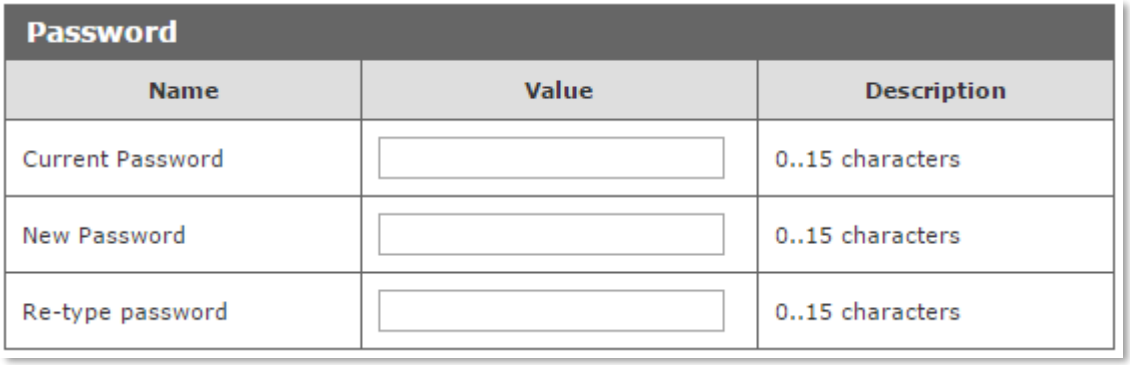
Module name

Each reader can be given a unique name for identification purposes.



To change the password:

1. enter the current password in the Current Password field.
2. enter the new password in the New Password and Re-type Password fields.
3. confirm the change by clicking Save Config..



Setting up services

You can select which services you want to make available by ticking the corresponding check boxes:

- **Autonomic** – reader standalone mode
- **Enable MODBUS TCP** – enable MODBUS TCP service
- **Enable MODBUS RTU** – switching on the MODBUS RTU service
- **Enable SNMP** – switching on the SNMP protocol service
- **Enable HTTP GET** – mode of module operation as HTTP server
- **Enable HTTP Client** – mode of module operation as HTTP client
- **Timeout** – time after which the module will enter autonomous mode if it does not receive information from the server
- **Enable Remote Network Config** – enable remote module configuration (for Discoverer)
- **Enable TFTP Bootloader** – enable the bootloader.

Services

Name	Value	Description
Autonomic	☑	
Enable MODBUS TCP	☐	
Enable MODBUS RTU	☐	
Enable SNMP	☐	
Enable HTTP GET	☐	
Enable HTTP Client	☐	
Timeout	0	(x0,1s) a reader communication loss timeout after which the reader switches to Autonomic mode.
Enable Remote Network Config	☑	Enable Remote Network Config by Inveo Discover Software.
Enable TFTP Bootloader	☐	Allow remote upgrade firmware by TFTP. For safety reasons, the option should be disabled.

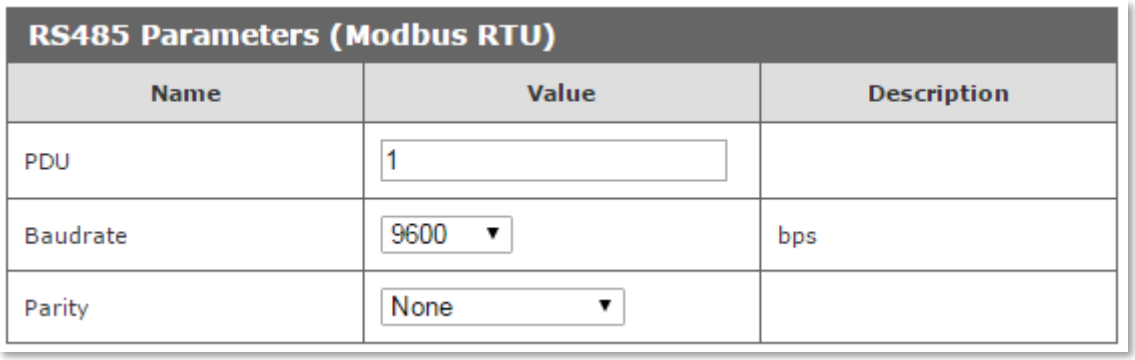
Tip

For security reasons, the TFTP Bootloader and Remote Network Config options should be disabled during normal operation. Only enable them before a software update..

8 Communication with the module

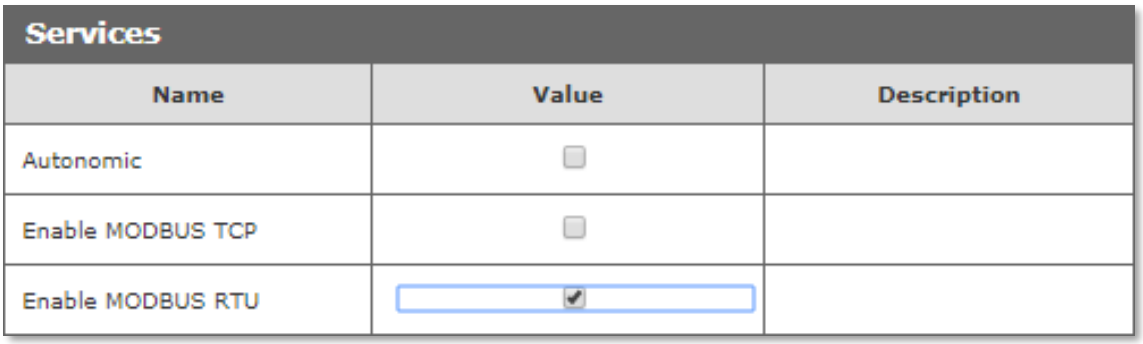
8.1 Modbus

Modbus is available either via RS485 (Modbus RTU) or via TCP (Modbus TCP). The RS485 port parameter settings for Modbus RTU communication are found on the Administration tab.



No	Name	Description
1	PDU	Modbus address of the device
2	Baudrate	The baud rate of the data transmission. Available baud rates: 1200, 2400, 4800, 9600, 19200, 34800, 57600, 115200
3	Parity	Parity setting. Options available: None, None and 2 Stop, Even, Odd, Mark, Space

After setting the appropriate parameters, activate the Modbus RTU service and confirm with Save.



The device supports the following Modbus RTU functions:

- 0x01 Read Coils
- 0x03 Read Holding Register
- 0x05 Write Single Coil
- 0x06 Write Single Register
- 0x0F Write Multiple Coils
- 0x10 Write Multiple Registers

Addressing Holding Registers for Modbus RTU and Modbus TCP:

Control RFID			
No	Address	R/W	Description
1	1	R/W	Read flag Read: 1 - new transponder read Write: 0 - reset read flag
2	2	R	ID_LEN
3	3	R	ID CARD/UID/CSN [0]
4	4	R	ID CARD /UID/CSN [1]
5	5	R	ID CARD /UID/CSN [2]
6	6	R	ID CARD /UID/CSN [3]
7	7	R	ID CARD /UID/CSN [4]
8	8	R	ID CARD /UID/CSN [5]
9	9	R	ID CARD /UID/CSN [6]
10	10	R	ID CARD /UID/CSN [7]
11	11	R	ID CARD /UID/CSN [8]
12	12	R	ID karty/UID/CSN [9]
13	13	R	Tag type
14	14	R	ID_MODEL
15	15	R	ID_SW
16	16	R	ID_HW
17	17	R	Type of TAG read 0- none 1- unknown TAG 2- USER 3- MASTER
18	100	R	Only Mifare-enabled RFID modules - block read 16 consecutive bytes
19	1000	R	ID CARD /UID/CSN [0]
20	1001	R	ID CARD /UID/CSN [1]
21	1002	R	ID CARD /UID/CSN [2]
22	1003	R	ID CARD /UID/CSN [3]
23	1004	R	ID CARD /UID/CSN [4]
24	1005	R	Model urządzenia (IND-U4 → 0x5534)
25	1006	R	Software version
26	1007	R	Hardware version

Holding Registers addresses 1000-1007 are arranged in this way for reasons of backward compatibility.

Editing card entries			
No	Address	R/W	Description
27	1100	R/W	Address of the card record to be edited
28	1101	R/W	Card settings: see below
29	1102	R/W	ID CARD /UID/CSN [0]
30	1103	R/W	ID CARD /UID/CSN [1]
31	1104	R/W	ID CARD /UID/CSN [2]
32	1105	R/W	ID CARD /UID/CSN [3]
33	1106	R/W	ID CARD /UID/CSN [4]
34	1107	R/W	ID CARD /UID/CSN [5]
35	1108	R/W	ID CARD /UID/CSN [6]
36	1109	R/W	ID CARD /UID/CSN [7]
37	1110-1149	R/W	User name, encoding of Polish characters ISO8859-2, name must be terminated with NULL (0x00)

Card settings (address 1101 Holding Registers):
When modifying a user card, address 1101 Holding Register is responsible for whether the card is stored in the reader's memory, whether it is active and which groups are assigned to the card.

Register 1101		
Bit	Decimal value	Function
0	1	Determines whether the card is stored in the reader's memory.
1	2	Determines whether the card is active: 0 - card inactive, 1 - card active.
2	4	Assignment to group 1
3	8	Assignment to group 2
4	16	Assignment to group 3
5	32	Assignment to group 4
6	64	Reserved
7	128	Reserved

Examples of sent values with explanation:

- Deleting a card from the reader's memory - value 0.
- Deactivating a card and assigning it to group 1 at the same time - value 5.
- Assigning a card to all groups together with its activation - value 63.

Log records				
No	Address	**Type**	**R/W**	**Desciption**
38	1200	Holding Reg	R/W	Number of records [MSB], sending 0 deletes all logs
39	1201	Holding Reg	R	Number of records [LSB].
40	1202	Holding Reg	R	Selected record for reading [MSB]
41	1203	Holding Reg	R	Selected record for reading [LSB]
42	1204	Holding Reg	R	Timestamp [LSB]
43	1205	Holding Reg	R	Timestamp
44	1206	Holding Reg	R	Timestamp
45	1207	Holding Reg	R	Timestamp [MSB]
46	1208	Holding Reg	R	Holding Reg R Event event 0x0080 - card known event 0x0040 - card withdrawn
47	1210	Holding Reg	R	User ID (according to card table)
48	1212	Holding Reg	R	Card ID /UID/CSN [0]
49	1213	Holding Reg	R	Card ID /UID/CSN [1]
50	1214	Holding Reg	R	Card ID /UID/CSN [2]
51	1215	Holding Reg	R	Card ID /UID/CSN [3]
52	1216	Holding Reg	R	Card ID /UID/CSN [4]
53	1217	Holding Reg	R	Card ID /UID/CSN [5]
54	1218	Holding Reg	R	Card ID /UID/CSN [6]
55	1219	Holding Reg	R	Card ID /UID/CSN [7]

Control of the read card				
No	Address	**Type**	**R/W**	**Desciption**
56	1300	Holding Reg	R	Card record address
57	1301	Holding Reg	R	Card settings
58	1302	Holding Reg	R	Card ID /UID/CSN [0]
59	1303	Holding Reg	R	Card ID /UID/CSN [1]
60	1304	Holding Reg	R	Card ID /UID/CSN [2]

61	1305	Holding Reg	R	Card ID /UID/CSN [3]
62	1306	Holding Reg	R	Card ID /UID/CSN [4]
63	1307	Holding Reg	R	Card ID /UID/CSN [5]
64	1308	Holding Reg	R	Card ID /UID/CSN [6]
65	1309	Holding Reg	R	Card ID /UID/CSN [7]
66	1310-1349	Holding Reg	R	User name, encoding ISO8859-2, name must be NULL (0x00)


Tip

Addresses 1300-1349 Holding Registers contain the same data as addresses 1100-1149 Holding Registers. The difference is that addresses 1300-1349 are read-only.

IO INPUT OUTPUT				
No	**Address**	**Type**	**R/W**	**Description**
1	1	Single Coil	R/W	State/setting of output 1
2	2	Single Coil	R/W	State/setting of output 2
3	3	Single Coil	R	State of output 1
4	4	Single Coil	R	State of output no. 2
5	11	Single Coil	R	State of input 1
6	12	Single Coil	R	Input 2 state
7	13	Single Coil	R/W	LED control
8	14	Single Coil	R/W	LED control
9	15	Single Coil	W	1 - "Accept" sound
10	16	Single Coil	W	1 - "Reject" sound
11	17	Single Coil	R/W	Reading flag Reading: 1 - new transponder read Write: 0 - reset read flag
12	18	Single Coil	R/W	Reset status Read: 1 - reader restart occurred (e.g. by power failure) Write: 1 - device reset forced 0 - resetting the restart flag
13	1000	Single Coil	R	Output status no. 1 - lock (door)
14	1001	Single Coil	R	Input 1 status
15	1002	Single Coil	R	State of zone no. 2
16	1003	Single Coil	R	Reserve
17	1004	Single Coil	R/W	Read flag Read: 1 - new transponder read Write: 0 - reset read flag
18	1005	Single Coil	R/W	Reset status Read: 1 - a reader restart has occurred (e.g. by power failure) Write: 1 - device reset forced 0 - resetting the restart flag
19	1010	Single Coil	W	1 - activation of door lock relay
20	1011	Single Coil	W	1 - "Accept" sound
21	1012	Single Coil	W	1 - "Reject" sound
22	1013	Single Coil	R/W	LED control
23	1014	Single Coil	R/W	LED control

Tip

After the tag has been read correctly, the Single Coil register 1004 stores the value 1 - a new tag has been read. Reading the next transponder is only possible after resetting the read flag. (Writing a value of zero to the Single Coil 1004 register). Resetting the read flag is not necessary if the module is set in Continuous Reading mode.

8.2 Integration with proprietary software

Inveo's RFID readers can be integrated with custom software, operating in either server mode (selected option Administration → Enable HTTP GET) or client mode (selected option Administration → **Enable HTTP Client**).

Server Mode (HTTP Server):
In this mode, the reader operates as a server. An external host (client) connects to the module and manages it via the HTTP protocol using the GET method. To enable this mode, select the option Administration → Enable HTTP GET.

- **Status Reading**: The client periodically reads the status.xml file, decodes the relevant XML tags (each tag is explained in section 8.3), and sends any commands back to the reader. The status.xml tag contains all necessary information that can be recorded in a database, such as the reader's MAC address, the status of inputs and outputs, and the card ID.
- **User Interaction**: After processing the data, the client can send commands to the reader, such as acceptance/rejection sounds, LED blinking, door latch release, or text display. The reader blocks the ability to read subsequent tags until the client invokes the releaseId=1 function.
- **Audio Signal**: If communication does not work in real-time, the reader can emit a short beep after reading a tag, informing the user that the tag has been read. This is useful so the user does not need to hold the card near the reader but only until the beep is heard.

Client Mode (HTTP Client):
In this mode, the reader operates similarly to a web browser. It sends requests to a server and waits for responses. The server – the resource-providing entity – waits for requests, and upon receiving one, processes it and sends the appropriate response. To enable this mode, select the option Administration → **Enable HTTP Client**.

- **Data Transmission**: The module automatically connects to the server and sends data about the read tag using the HTTP GET method. The server processes the request and sends a response in the form of an XML file with commands that dictate the module's actions, such as controlling the LEDs, audio signals.
- **Direct Integration**: The module sends the code to the server or control application immediately after reading the card. The client-server mode enables easy integration with PHP servers, Node JS, MySQL, etc.

The client-server mode allows seamless integration of RFID readers with various servers and applications, providing flexibility and efficiency in managing data and user interactions.

8.3 HTTP GET – status reading

RFID modules can be controlled using the HTTP protocol on port 80. To read the current status of the module, follow these steps:

1. **Open a web browser.**
2. **Enter the address: http://adresIPurzadzenia/status.xml.**
    - Replace deviceIPAddress with the actual IP address of the device.

After completing these steps, an XML resource with basic information about the reader will be displayed in the browser:

```
<status>
  <name/>
  <mac>00:00:00:00:00:00</mac>
  <id>0000000000</id>
  <newId>0</newId>
  <known>2</known>
  <cnt>87</cnt>
  <out0>Off</out0>
  <out1>Off</out1>
  <in0>Off</in0>
  <in1>Off</in1>
  <resetFlag>1</resetFlag>
  <enable>1</enable>
  <group>1234</group>
  <httpClientStatus>0</httpClientStatus>
  <n_logs>301</n_logs>
  <timeout>0</timeout>
  <fw>0.42</fw>
  <hw>1.0</hw>
</status>
```

Section	Description
<name></name>	Module name
<mac>00:00:00:00:00:00**</mac>**	MAC address of the module
<id>0600ADDA8E**</id>**	Last read code from RFID tag in hex format
<newId>1</newId>	In Control only by HTTP GET mode 1 - new RFID tag read 0 - no new RFID tag read
<known>2</known>	Read tag type 1 - tag unknown 2 - user 3 - master
<cnt>1</cnt>	Number of read RFID tags since device reset
<out0>Off</out0>	Current state of relay output 1
<out1>Off</out1>	Current state of relay output 2
<in0>Off</in0>	Current state of input 1
<in1>Off</in1>	Current state of input 2
<resetFlag>1</resetFlag>	1 - module reset occurred

Section	Description
<enable>1</enable>	1 - radio module switched on 0 - radio module switched off
<group>1234</group>	Numbers of currently active groups
<httpClientStatus>0 </httpClientStatus>	Current status of TCP connection in Control only by HTTP Client mode 1-Connected to server -socket open 2- received data from server 3-connection terminated 100-no connection to server
<n_logs>0</n_logs>	Current number of logs in the reader
<timeout>0</timeout>	Time since last connection to server
<fw>0.42</fw>	Software version
<hw>1.0</hw>	Current hardware version

8.4 HTTP GET – control

The control of the module in Enable HTTP GET mode consists in sending an appropriate command to the module via the HTTP protocol.

http://adresIPurzadzenia/status.xml?			
No	**Command**	**Name**	**Description**
1	enable	Enable RFID	Activation of the antenna in the RFID module http://192.168.111.15/status.xml?enable=1 Antenna deactivation in the RFID module http://192.168.111.15/status.xml?enable=0
2	resetFlag	Reset Flag	After module reset flag set to 1 Reset flag reset http://192.168.111.15/status.xml?resetFlag=0
3	releaseId	Release ID	Clearing read flag and waiting for RFID tag proximity http://192.168.111.15/status.xml?releaseId=1
4	Ledr	Red LED	Activation of signalling diode led=TimeOn,TimeOff,Cnt TimeOn*0.1 seconds, TimeOff*0.1 seconds http://192.168.111.15/status.xml?ledr=5,3,4 Switching on led for 0.5 seconds, switching off for 0.3 seconds and repeating the sequence 4 times Cnt=255 - endless repetition Cnt=0 - LED off
5	Ledg	Green LED	Switching on the led led=TimeOn,TimeOff,Cnt TimeOn*0.1 seconds, TimeOff*0.1 seconds http://192.168.111.15/status.xml?ledg=5,3,4 Switching on led for 0.5 seconds, switching off for 0.3 seconds and repeating the sequence 4 times Cnt=255 - endless repetition Cnt=0 - LED off

http://adresIPurzadzenia/status.xml?		
No	**Command**	**Description**
6	buzz	Control of the sound generator REJECT sound generation http://192.168.111.15/status.xml?buzz=r ACCEPT sound generation http://192.168.111.15/status.xml?buzz=a
7	open	Activation of relay output 1 in the mode defined in the I/O Settings tab http://192.168.111.15/status.xml?open=1
8	open2	Activation of relay output 2 in a mode defined in the I/O Settings tab http://192.168.111.15/status.xml?open2=1
9	out0	Control of relay output 1 0 - deactivate relay output 1 - activation of the relay output http://192.168.111.15/status.xml?out0=1
10	out1	Controlling relay output 2 0 - switching off the relay output 1 - activation of relay output http://192.168.111.15/status.xml?out1=1

http://adresIPurzadzenia/msg.php?		
No	**Command**	**Description**
1	setLog	Sets the current log indicator http://192.168.111.15/msg.php?setLog=x x – number of logs
2	removeCard	Deletes the selected card from the memory x - card number minus 1 http://192.168.111.14/msg.php?removeCard=x
3	removeAllCards	Removal of all RFID tags from the reader http://192.168.111.15/msg.php?removeAllCards=1
4	removeLog	Deletion of all logs from the reader http://192.168.111.15/msg.php?removeLog=1
5	removeNLog	Removal of oldest x (number) logs from reader memory http://192.168.111.15/msg.php?removeNLog=x x= number of logs to be deleted
6	clkY	Setting the year in the RTC x=[0-99] http://192.168.111.15/msg.php?clkY=x
7	clkM	RTC month setting x=[0-11] 0-January, 1-February ..... http://192.168.111.15/msg.php?clkM=x
8	clkD	RTC day setting x=[1-31] http://192.168.111.15/msg.php?clkD=x
9	clkH	RTC time setting x=[0-23] http://192.168.111.15/msg.php?clkH=x
10	clkm	RTC minutes setting x=[0-59] http://192.168.111.15/msg.php?clkm=x
11	clkS	RTC seconds setting x=[0-59]

		http://192.168.111.15/msg.php?clkS=x
12	clkd	RTC day of the week setting x=[0-6] 0-Sunday, 1-Monday http://192.168.111.15/msg.php?clkd=x
13	factory	Return to factory settings http://192.168.111.15/msg.php?factory=1
14	addId	Manual addition of card to memory (ID in HEX format) http://192.168.111.15/msg.php?addId=1122334455
15	changeName	Manual change of card name x - card number from Cards tab minus 1 http://192.168.111.15/msg.php?changeName=x!Jan
16	flag	Manual editing of card permissions x - card number from Cards tab decreased by 1 y - sum of assigned rights: 1 - determination whether the card is stored in the reader's memory, 2 - card active (2) or inactive (0), 4 - card assigned to group 1, 8 - card assigned to group 2, 16 - card assigned to group 3, 32 - card assigned to group 4. The sum of the assigned rights determines the value of y.

8.5 Editing cards via HTTP POST

The reader allows cards to be edited using the HTTP protocol via POST.

The data structure can be found in the edit.xml file.
id: 6
card: 00
name: User2
act: 1
grp: 0100

Indications:
id – card number from the Cards tab minus 1,
card – TAG,
name – user name,
act – determines whether the Active option is selected for the given card (value 0 or 1),
grp card belonging to relevant groups (value 0 means the card is not assigned to a group, value 1 means the card is assigned to a selected group), e.g:
0001 – card assigned to group 4 only,
0111 – card assigned to group no. 2, no. 3 and no. 4.

The data structure of the POST method has the following form:
id=**value**&card=00&name=**value**&act=**value**&grp=**value**

Example:
id=6&card=00&name=User2&act=1&grp=0100

By changing the relevant parameters, the card can be edited.
HTTP POST also allows you to add a new card by entering id=-1. The reader will add the card to the first free position.


Tip

It is not possible to edit the UID number of a card stored in the reader's memory.

Example of editing a tab:
To edit a card, use cURL. cURL allows HTTP requests to be sent from the Windows command line. The syntax of a card edit request is as follows:

curl -u admin:admin00 -d "id=6&card=00&name=John&act=1&grp=0111" -X POST http://192.168.0.222/edit.html

-u admin:admin00 – command is responsible for user authentication, login:password,
-d "id=6&card=00&name=John&act=1&grp=0111" -X POST – command to edit the card:
id=6 – card number 7,
card=0 – when editing the card its UID number cannot be changed,
name=John – giving a new user name,
act=1 – forcing activation of the selected card,
grp=0111 – assigning the card to groups no. 2, no. 3 and no. 4.
http://192.168.0.222/edit.html – module IP address and resource to be referenced.

```
d:\curl>curl -u admin:admin00 -d "id=6&card=00&name=John&act=1&grp=0111" -X POST
 http://192.168.0.222/edit.html
```

Example of adding a new card:
To add a card to the reader's memory, use cURL. cURL allows HTTP requests to be sent from the Windows command line. The syntax for adding a new card is as follows:
curl -u admin:admin00 -d "id=-1&card=0600A4638D&name=Ahr&act=1&grp=1001" -X POST http://192.168.0.222/edit.html

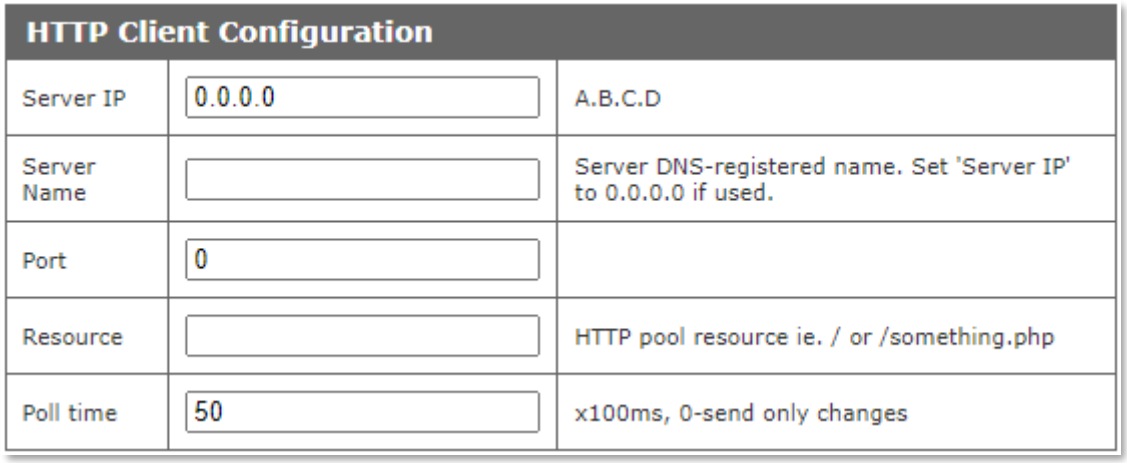
The above request will add another card at the next free position in the reader memory.

id=-1 – forces the addition of a new card.

```
d:\curl>curl -u admin:admin00 -d "id=-1&card=0600A4638D&name=Ahr&act=1&grp=1001"
 -X POST http://192.168.0.222/edit.html
```

8.6 HTTP Client

The module can be controlled in HTTP Client mode by selecting the Enable HTTP Client option on the Administration tab.
If the reader is set to Enable HTTP Client mode, the address to which data is sent must be configured. The table Network → HTTP Client Configuration is intended for this purpose.



Name	Description
Server IP	The IP address of the server to which the reader will send data
Server name	The name of the domain address - if registered on the DNS server. If you enter a domain address, enter 0.0.0.0 in the Server IP field

Port	The port on which the server is listening
Resource	The resource referred to by the reader, e.g. /somefile.php
Poll time	Period for sending data to the server Poll time=0 - data is only sent when an rfid tag is applied Poll time>0 - data is sent cyclically and after the tag is applied e.g. Poll time=50 - data will be sent every 5 seconds and after the tag is applied

Sending data to the HTTP Client server
The module sends data to the corresponding server resource depending on whether the reader has read the TAG or because of a cyclic exchange of information.

When the reader has read the RFID TAG, the data is sent as:
- **A** – MAC address,
- **B** – IP address,
- **C** – UID,
- **D** – applies only to RFID T&A with keys - a flag indicating the function keys (e.g. whether a private/service entrance/exit has taken place),
- **E** – information on the inputs and outputs activated,
- **F** – status indicating whether a card has been inserted or withdrawn.

?mac=**A**&ip=**B**&id=**C**&inout=**D**&io=**E**&**F**

During the cyclic information exchange, data is sent to the server in the form of:
- **A** – MAC address,
- **B** – IP address,
- **E** – information on switched inputs and outputs.

?mac=**A**&ip=**B**&io=**E**

D – flag indicating which function keys have been selected (e.g. whether a private/service entrance/exit has taken place). The value is sent as a sum of the components:

Examples	Private	Duty	Input	Output	D
	0 or 1	0 or 2	0 or 4	0 or 8	
1.	1	0	4	0	5
2.	0	2	4	0	6
3.	1	0	0	8	9
4.	0	2	0	8	10

Examples from the table:
1. private input. Value inout=5.
2. business input. Value inout=6.
3. private output. Value inout=9.
4. business output. Value inout=10.

E – the status of the switched inputs and outputs is sent as the sum of the component values:

Examples	In 1	In2	Out1	Out2	E
	0 or 1	0 or 2	0 or 4	0 or 8	0 - 15
1.	1	0	4	0	5
2.	0	0	4	8	12
3.	1	2	4	8	15

Przykłady z tabeli:
1. Wejście nr 1 oraz wyjście nr 1 są załączone. Wartość io=5.

2. Wyjście nr 1 i nr 2 są załączone. Wartość io=12.
3. input no. 1 and no. 2, and output no. 1 and no. 2 are switched on. Value of io=15.

F – signalling that a card has been applied or taken away. It may take the following values:
- put=1 - signalling card insertion,
- away=1 - signalling card withdrawal (Continuous reading mode only),
- put=1&away=1 - signalling card insertion and immediate withdrawal (Continuous reading mode only).

Examples of data sent to the server after card insertion:
?mac=0000000&ip=10.10.0.1&id=00000000&inout=6&io=8&put=1
?mac=0000000&ip=10.10.0.1&id=00000000&inout=6&io=12&put=1&away=1

xamples of cyclically sent data (poll time):
?mac=0000000&ip=10.10.0.1&io=0
?mac=0000000&ip=10.10.0.1&io=7
Only for RFID modules supporting the Mifare standard:
If the module has block reading enabled, a message may be added with the block read, e.g.: block=1234567890ABCDEF1234567890ABCDEF.

In response, the server may send nothing back or send back an XML resource with tags:

No	Command	Desription
5	<ledg>	Switch on the green indicator light Syntax: <ledg>TimeOn,TimeOff,Cnt</ledg>. (as in HTTP GET)
6	<ledr>	Activation of the red indication LED Syntax: <ledr>TimeOn,TimeOff,Cnt</ledr>. (as in HTTP GET)
7	<open>	1 - switching on relay no. 1 (e.g. bolt) in such a mode as defined in the I/O Settings tab
8	<open2>	1 - switching on of relay no. 2 in such a mode as defined in the I/O Settings tab
9	<out0>	0 - relay 1 deactivated 1 - activation of relay 1
10	<out1>	0 - switching off relay 2 1 - switching on relay 2
11	<buzz>	1 - sound signal on (ACCEPT sound) 2 - sound signal on (REJECT tone)
12	<ring>	Applies only to T&A with keys 1 - switching off the backlighting of the function keys 2/3 - illumination of the left/right function key 4/5 - flashing left/right function key illumination 6/7 - flashing "in the loop" illumination of the left/right function key 8 - flashing "in the loop" illumination of the function keys

Using the reader's operation in HTTP Client mode, it is possible to use the returned HTTP code (Response code) to control the interaction with the user.

In the RFID Settings tab, you can enable the option for the reader to respond automatically to the basic 3 HTTP response codes (200, 401, 404).



With this configuration, the reader will respond as follows depending on the code number:

- code 200: the module will generate an "accept" tone and the green LED will blink,
- code 401: the module will generate a "reject" tone and the red LED will blink,
- code 404: the module will generate the "reject" sound and two LEDs will blink.

```
<buzz>1</buzz>
<clear>0</clear>
<text>Enter please</text>
<open>1</open>
```

Switching on the green LED for 2 seconds:

```
<ledg>20,0,1</ledg>
```

Flashing red 2 times with a timeout of 0.5/0.5 seconds:

```
<ledr>5,5,2</ledr>
```

The syntax of the XML file is not checked. Only information between known tags is searched.

Example of php server operation:

```
<?php

if( $_GET["id"] ) // module send id and MAC - $_GET["mac"]
{
     $who=$_GET["id"];
     // you can check id in DB and do some action
     echo "<buzz>1</buzz>"; // sound signal
     echo "<clear>1</clear>"; // clear LCD
     echo "<text>Card ID: $who</text>" // print ID on LCD
     echo "<open>1</open>"; // door open
}
else // no id - default state polling
{
     echo "<clear>1</clear>"; // clear LCD
     echo "<text>Hello</text>"; //print prompt text
}
```

8.7 Communication with the module from an external network

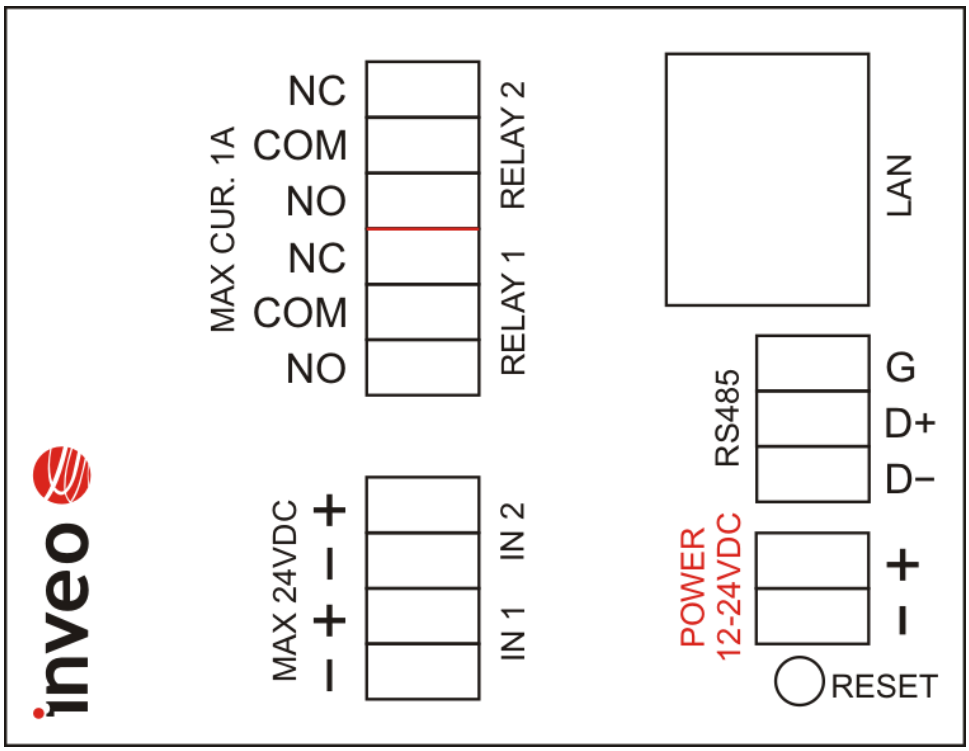
If the module is on a different LAN from the computer connecting to it, port forwarding is required. Depending on the method of communication with the module used, it is necessary to contact the Network Administrator and redirect the ports:

Support via WWW and HTTP protocol:
- TCP/IP port 80

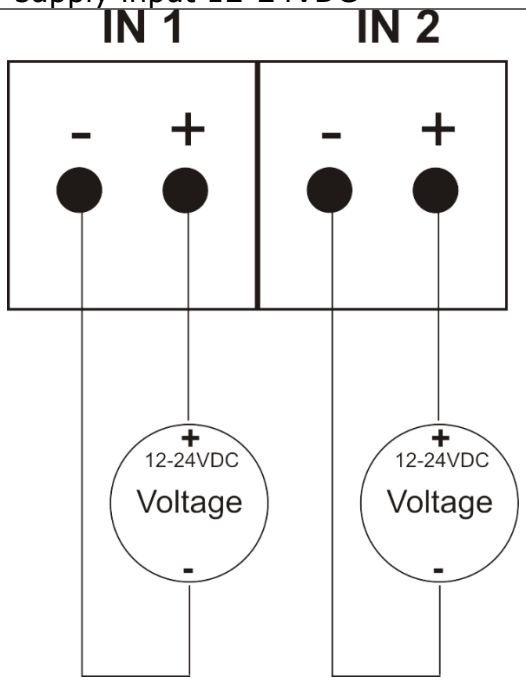
Support via MODBUS TCP:
- TCP/IP port 502

Support via SNMP:
- UDP port 161

9 Description of connectors



No	Name	Description
1	LAN	LAN connection socket
2	Reset	RESET Short-circuiting the pins for between 10-15 seconds returns the unit to factory settings.
3	RS485	RS485 connector for Modbus RTU protocol
4	RELAY 1	Connector for relay no. 1
5	RELAY 2	Relay connector no. 2
6	IN1	General purpose input
7	IN2	General purpose input
8	+12V-	Power supply input 12-24VDC



10 DHCP

The DHCP service can be enabled in the Network tab or via the Discoverer programme. The Discoverer program can be downloaded from www.inveo.com.pl.

11 Restoring factory settings

To restore the factory settings of the device, please:

- Switch on the device.
- Shorten the RESET jumper for between 10 and 15 seconds.
- During the beep, open the RESET jumper.

After performing the above steps, the device will set the following parameters:

- IP address: 192.168.111.15
- IP mask: 255.255.255.0
- User: admin
- Password: admin00

12 Software update

The module is equipped with the possibility of updating the software. The program is supplied as a file with the extension .bin

> Warning Incorrect use of the software update function may damage the module.

To carry out the software update operation, you must:

- check the Enable TFTP Bootloader option, which can be found on the device page under Administration,
- run the Windows command line (Start->Run, type 'cmd' and confirm with Enter),
- go to the directory where the .bin file is located,
- enter the command:

tftp -i <address_ip_module> PUT file.bin

where: <address_ip_module> is the IP address of the module,
file.bin is the file with the update program.
Programming takes approximately 1 minute. Completion of programming is confirmed by the 'File Transferred' message.
After the software update operation it is necessary to disable the Enable TFTP Bootloader option in order for the device to operate correctly.

CE

EC DECLARATION OF CONFORMITY

Manufacturer	Inveo spółka z ograniczona odpowiedzialnością Rzemieślnicza 21 43-340 Kozy
Product	**RFID TAB**
Model	**13,56 MHz, 125kHz,**

The product is compliance with the requirements of the following European directives:

2014/53/EU	Radio Equipment
2011/65/EU	Restriction of the use of certain hazardous substances in electrical and electronic equipment (RoHS II).

The compliance with the requirements of the European Directive was proved by the application of the following harmonized standards:

Safety:	EN 60950-1:2006 + A11:2009 + A1:2010 + A12:2011 + AC:2011 + A2:2013, EN 62479:2010
EMC:	EN 301 489-1 /V2.1.1 EN 301 489-3 V2.1.1 EN 55032:2015 + AC:20016 Class B EN 55024:2010
Radio	EN 300 330 V2.1.1
RoHS II	EN 50581:2012

The object of the declaration described above is in conformity with Directive 2011/65/EU of the European Parliament ad of the Council of 8 June 2011 on the restriction of the use of certain hazardous substances in electrical and electronic equipment.

mgr inż. Sławomir Darmofał
inveo
mgr inż. Sławomir Darmofał

Kozy, 19.12.2022 r.



www.inveo.com.pl

tel.: +48 33 444 65 87
kom.: +48 785 552 252

ul. Rzemieślnicza 21
43-340 Kozy

serwis@inveo.com.pl

