
SNMP

NETIO M2M API protocols docs

Protocol version: NETIO-MIB 1.08

Short summary

SNMP is implemented as M2M API protocol, where NETIO device can use SNMP protocol allowing monitoring and control of NETIO power outputs (power sockets 230V or power outlets IEC-320 110/230V).

SNMP M2M API protocol must be enabled first in the WEB configuration of the respective device. For details, see the “NETIO WEB configuration” chapter.

Supported devices

- **v1,2c - SNMP v1 / v2c** allow status monitoring and output control (Read & Write)
- **v3 - SNMP v3** uses authorization and encryption, allows monitoring and control of each output (**Read & Write**).

- **NETIO PowerCable REST 101x** (*Energy metering*)
- **NETIO PowerBOX 3Px**
- **NETIO PowerBOX 4Kx**
- **NETIO PowerDIN 4PZ**

- **NETIO PowerPDU 4C** (*Linux based core, can behave a bit differently*)
- **NETIO PowerPDU 4PS**
- **NETIO PowerPDU 8QS** (*Energy metering*)

NOTE: *This document provides basic info about the M2M API protocol.
Other device functions are described in the product manual.*

Quick start with SNMP & NETIO

NETIO MIB is available for download at the web administration – “M2M API Protocols” / “SNMP” tab.

Following standard MIBs are also required:

- SNMPv2-SMI
- SNMPv2-TC

SNMP v3 – control of outputs

Object OID x – output number (1 to 8)	Type	Value	Action
netioOutputAction.x.0 1.3.6.1.4.1.47952.1.1.1.5.x.0	INTEGER (i)	0 1 2 3 4 5	Turn OFF Turn ON Short OFF delay (restart) Short ON delay Toggle (invert the state) No change

General examples:

- Output **2** = **ON** SET 1.3.6.1.4.1.47952.1.1.1.5.**2**.0 = **1**
- **Toggle** Output **1** SET 1.3.6.1.4.1.47952.1.1.1.5.**1**.0 = **4**
- Output **8** = **OFF** SET 1.3.6.1.4.1.47952.1.1.1.5.**8**.0 = **0**

OFF

Examples of using snmpset in Linux command line:

- Turn output **1 ON** (with use of MIB):
`snmpset -m NETIO-PRODUCTS-NETIO-MIB -M /usr/share/snmp/mibs/ -v 3 -a SHA -A a1234567 -l authPriv -u admin -x AES -X a1234567 192.168.2.78 netioOutputAction.1.0 i 1`
- Turn output **5 OFF** (with use of OID):
`snmpset -v 3 -a SHA -A netiopass -l authPriv -u netio -x AES -X netiopass 192.168.101.159 1.3.6.1.4.1.47952.1.1.1.5.5.0 i 0`
- **Toggle** output **8** (with use of OID):
`snmpset -v 3 -a SHA -A netiopass -l authPriv -u netio -x AES -X netiopass 192.168.101.159 1.3.6.1.4.1.47952.1.1.1.5.8.0 i 4`

General NETIO output functions

Output status – “read” function

- **0** – Power **OFF**
- **1** – Power **ON**

Output actions – “write” function

- **0** – Turn **OFF**
- **1** – Turn **ON**
- **2** – Short OFF delay (restart)
- **3** – Short ON delay
- **4** – Toggle (invert the state)
- **5** – No change
- **6** – *Ignored (do not use, only for reading)*

Short ON / OFF delay

This command switches a power output On / Off for a defined time. It is useful for example to power-cycle a server with a defined switch-off time, or to switch on a pump for a defined time.

This “short” delay is protected: the power output will remain in the defined state regardless of any other M2M requests received. During this time, the output state can only be changed by pressing the button on the NETIO device and this action cancel M2M short ON/OFF command for the particular output. Other requests to control the particular output are simply ignored.

The short ON / OFF delay interval can be defined in the device web administration. It is specified in ms (milliseconds) and rounded up to hundreds of milliseconds (0,1s).

This interval can be also defined using some M2M API protocol commands. In that case, it is valid only for a single protocol session (the following short ON / Short OFF command). When the connection is closed or restarted, the interval is reset to the device default value (defined in the web administration for each output).

Security issues

Do not use default usernames and passwords! Keep your Ethernet and WiFi networks secured.

Power-Up outputs state

After connecting the device to power input, all outputs are set to the selected state based on its individual settings:

- **LAST state**

After a power outage, the NETIO device sets each power output to the last stored state of this one output.

- **ON**

The output is turned ON.

- **OFF**

The output stays OFF.

Note: **Function Scheduler** is checked in Power-Up initialization. When enabled, it can affect one or more power output states based on current time and date.

Custom based **Lua scripts** can affect output states too.

General NETIO input read features

Input status

- **0** – “open” / **ON**
- **1** – “closed” / **OFF**

S0 counters

- Number of S0 impulses / “ON” pulses

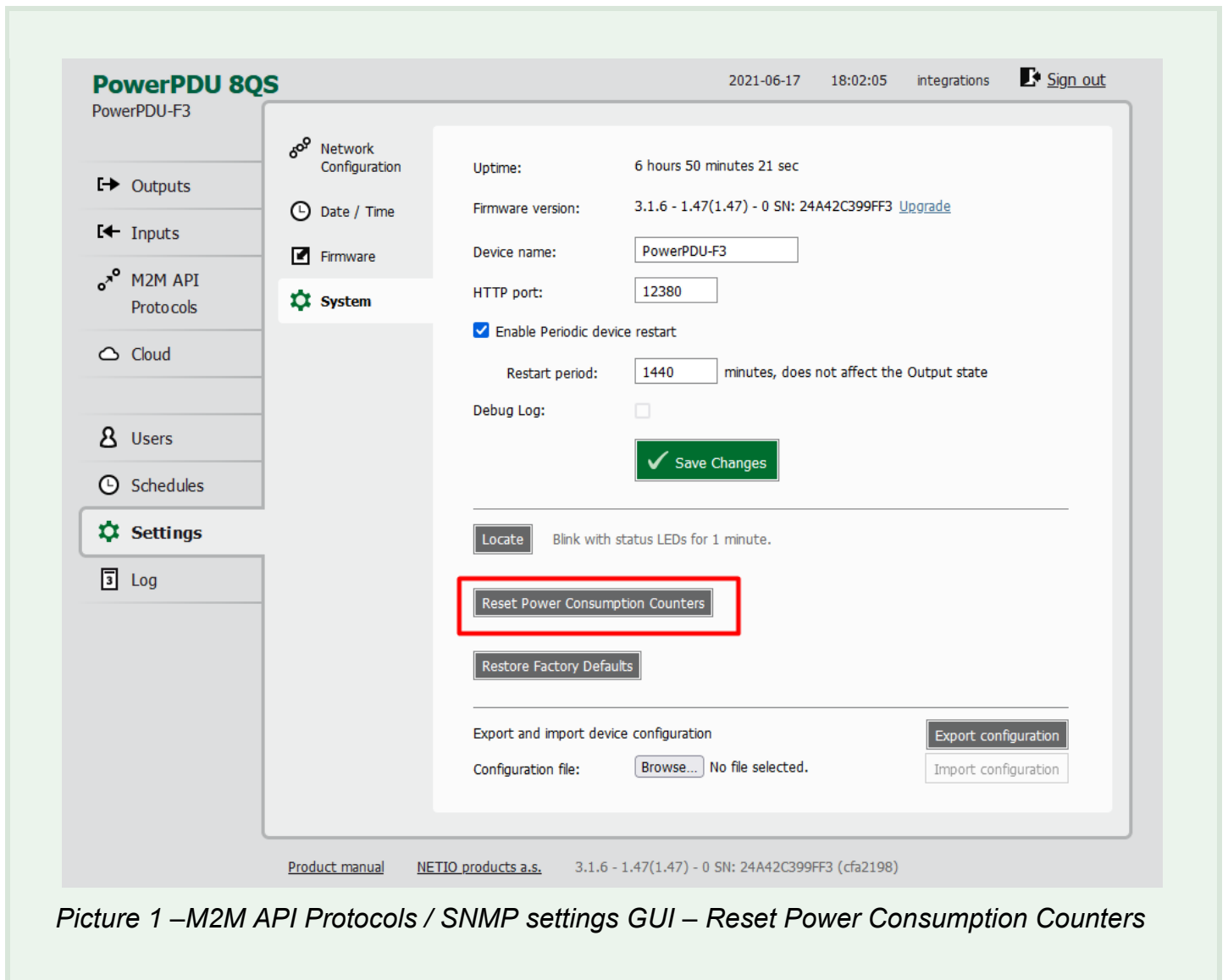
General NETIO Counters

- Power consumption counters can be reset to 0 manually
- Power consumption NR counters are Not Resettable
- S0 pulse counters on DI (Digital Inputs) can be reset to 0 manually

- All counters are not affected by power outage.

How to reset counters to 0

Click to button in the device settings. It will reset the counters.



Picture 1 –M2M API Protocols / SNMP settings GUI – Reset Power Consumption Counters

Energy metering variables

Energy metering is available for:

- NETIO PowerPDU 4C
- PowerCable REST
- PowerBox 4Kx
- PowerDIN 4PZ
- PowerPDU 8QS

Parameters for the **whole NETIO device**:

Variable	Unit	Description
Voltage	V	Instantaneous voltage
Frequency	Hz	Instantaneous frequency
Total Current	mA	Instantaneous total current through all power outputs
Total Power Factor	-	Instantaneous True Power Factor – weighted average from all meters
Total Phase	°	Instantaneous True Phase – weighted average from all meters
Total Load	W	Total load (power) of all power outputs (device's own internal consumption is not included)
Total Energy	Wh	Total Counter of consumed Energy (resettable)
Total Energy NR	Wh	Not Resettable counter of total consumed Energy
Total Reverse Energy	Wh	Counter of Reversed (produced) Energy (resettable)
Energy Start	-	Date and time of the last reset of all energy counters
<i>Overall True Power Factor</i>		<i>Historical compatibility only, do not use it.</i>
<i>Overall Phase</i>		<i>Historical compatibility only, do not use it.</i>

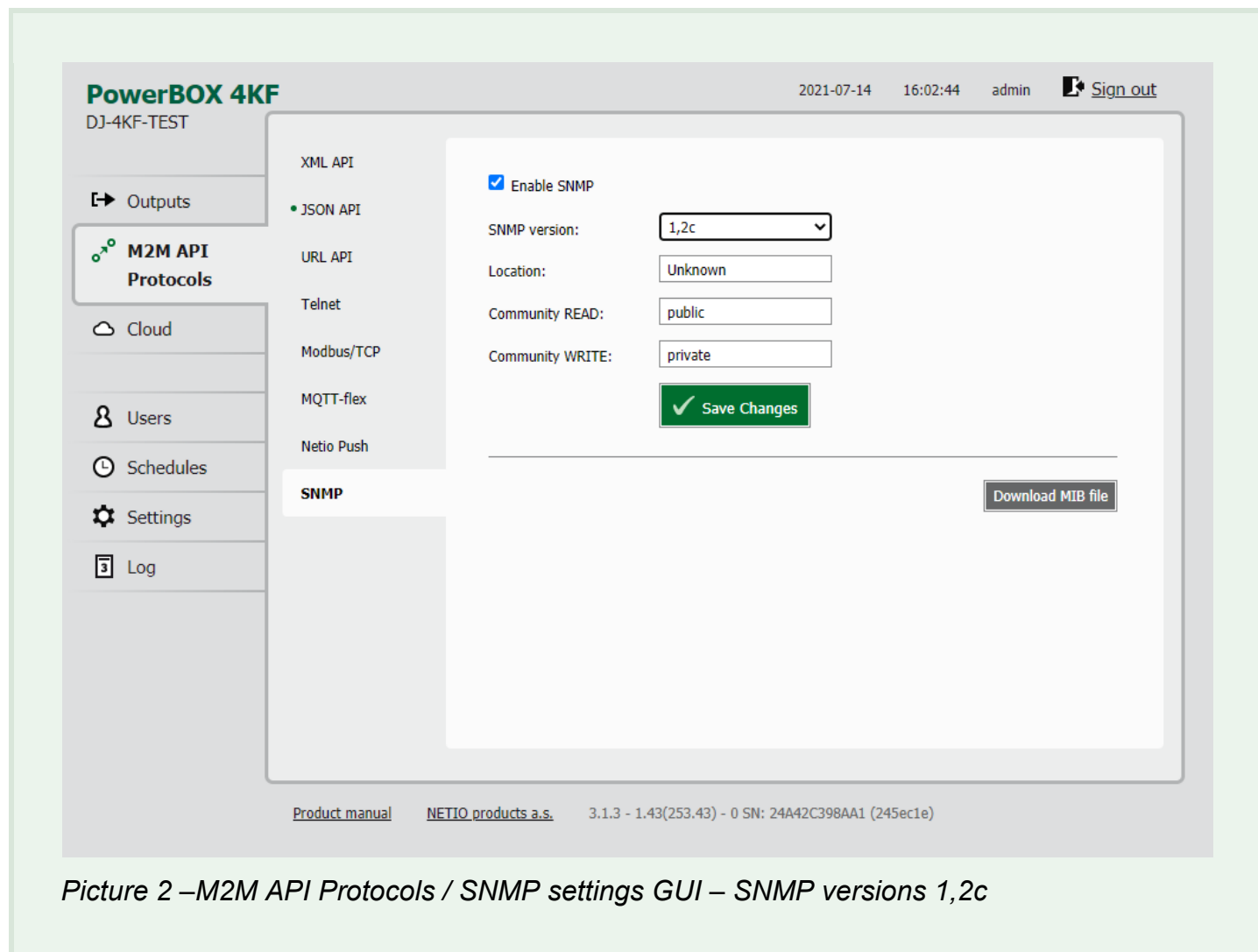
Parameters for **each power output**:

Variable	Unit	Description
Current	mA	Electric current for the output
Power Factor	-	TPF True Power Factor for the output
Phase	°	Phase for the specific power output
Energy	Wh	Counter of Energy consumed per output (resettable)
Energy NR	Wh	Not Resettable counter of output consumed Energy
Reverse Energy	Wh	Counter of Energy produced per output (resettable)
Reverse Energy NR	Wh	Not Resettable counter of Reversed (produced) Energy
Load	W	Instantaneous load (power) for the specific power output.

NETIO WEB configuration

M2M API protocols can be enabled and configured only over the web administration – select “M2M API Protocols” in the left-hand side menu and then select the “SNMP” tab.

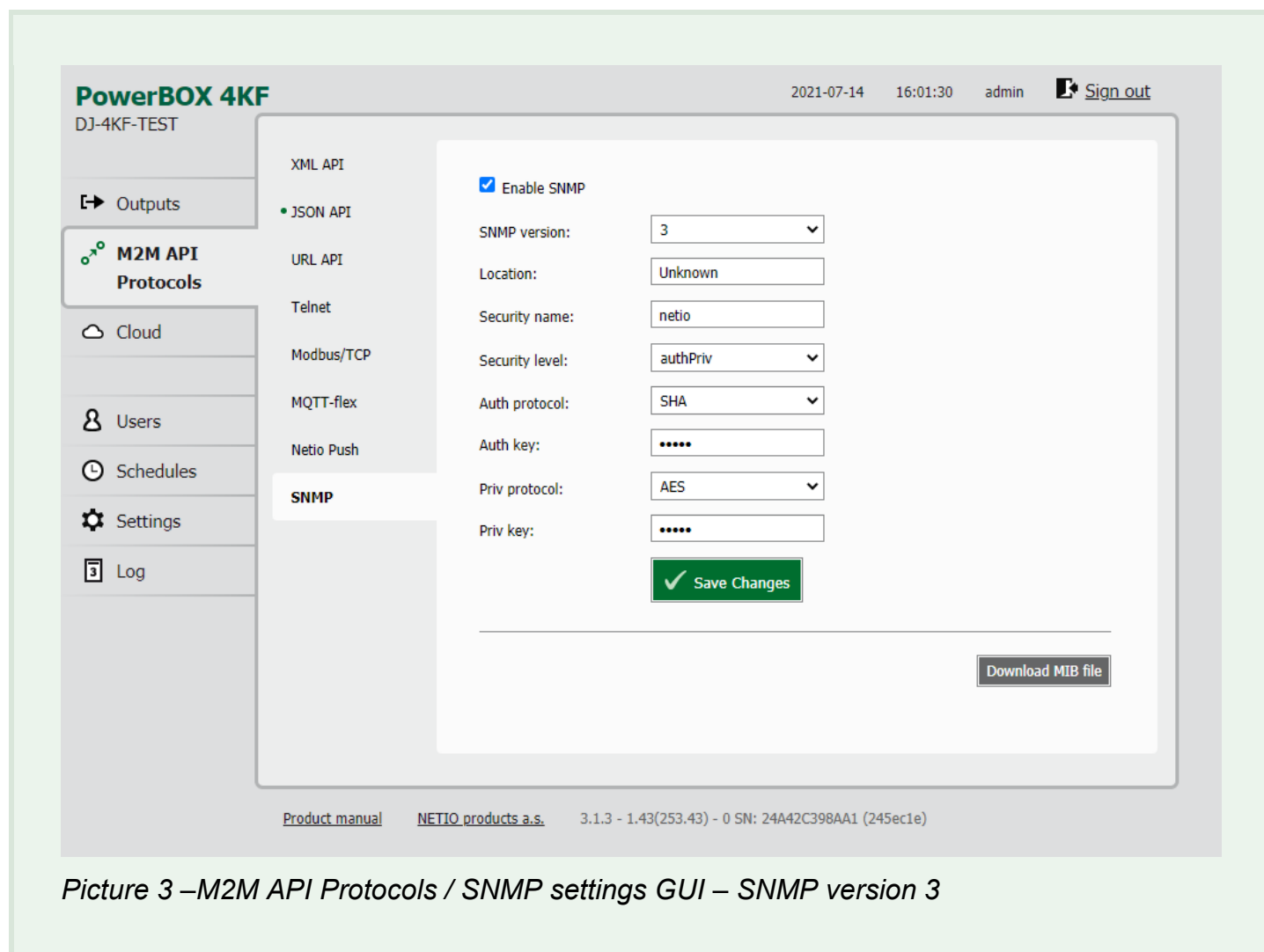
SNMP version 1,2c



Picture 2 –M2M API Protocols / SNMP settings GUI – SNMP versions 1,2c

- **Enable SNMP** – Enable/disable M2M API protocol
- **SNMP version**
 - **1,2c** – SNMP v1 / v2c uses basic authorization and no encryption, allows monitoring and control
- **Location** – Optional, description of NETIO device physical location
- **Community READ** – authorization for READ
- **Community WRITE** – authorization for WRITE
- SNMP port is 161, can't be edited.
- **Download MIB file (button)** – download of NETIO MIB file

SNMP version 3



Picture 3 –M2M API Protocols / SNMP settings GUI – SNMP version 3

- **Enable SNMP** – Enable/disable M2M API protocol
- **SNMP version**
 - **3** – SNMP v3 uses authorization and encryption, allows monitoring and control
- **Location** – Optional, description of NETIO device physical location
- **Security name** – username used for authentication
- **Security level** – security levels supported by SNMP agent
 - **authPriv** - Communication with authentication and privacy. Protocols used are MD5 and SHA for authentication and AES (**AES-128**) for privacy.
 - **authNoPriv** - Communication with authentication and without privacy. The protocols used for Authentication are MD5 and SHA.
 - **noAuthnoPriv** - Communication without authentication and privacy.
- **Auth protocol** – Authorization protocol used
- **Auth key** – Password used for authorization

- **Priv protocol** – Privacy protocol used
- **Priv key** – Password used for privacy
- SNMP port is 161, can't be edited.
- **Download MIB file (button)** – download of NETIO MIB file

Note: NETIO SNMP M2M Protocol is using “Password” as “authentication protocol pass phrase” and “privacy protocol pass phrase”.

NETIO SNMP protocol structure

NETIO MIB is available for download at the web administration – “M2M API Protocols” / “SNMP” tab.

Following standard MIBs are also required:

- SNMPv2-SMI
- SNMPv2-TC

Note: Linux-based devices (PowerPDU 4C, NETIO 4/4all) use Objects and OID notation without trailing zeroes, in contrast to format mentioned in table below.

E.g.: For accessing certain Output ID, following format is used:

- netioOutputID.x
- 1.3.6.1.4.1.47952.1.1.1.1.x

Monitoring (read)

Object OID x – output number (1 - 8)	Type	Value example	Note
netioOutputID.x.0 1.3.6.1.4.1.47952.1.1.1.1.x.0	INTEGER	1	
netioOutputName.x.0 1.3.6.1.4.1.47952.1.1.1.2.x.0	STRING	output_1	Based on user defined name
netioOutputState.x.0 1.3.6.1.4.1.47952.1.1.1.3.x.0	INTEGER	off(0), on(1)	
netioOutputStateString.x.0 1.3.6.1.4.1.47952.1.1.1.4.x.0	STRING	"off", "on"	
netioOutputLoad.x.0 1.3.6.1.4.1.47952.1.1.1.25.x.0	INTEGER	24	[W]
netioOutputEnergy.x.0 1.3.6.1.4.1.47952.1.1.1.26.x.0	INTEGER	13	[Wh]
netioOutputEnergyStart.x.0 1.3.6.1.4.1.47952.1.1.1.27.x.0	DateTime	2017-6-23,5:47:3.0,+0:0	Initial date and time. UTC based*

netioOutputCurrent.x.0 1.3.6.1.4.1.47952.1.1.1.28.x.0	INTEGER	195	[mA]
netioOutputPowerFactor.x.0 1.3.6.1.4.1.47952.1.1.1.29.x.0	INTEGER	534	Current power factor * 1000
netioOutputPhase.x.0 1.3.6.1.4.1.47952.1.1.1.30.x.0	INTEGER	120	Phase shift in range 0-360 [°]
netioOutputEnergyNR.x.0 1.3.6.1.4.1.47952.1.1.1.31.x.0	INTEGER	255	[Wh]
netioOutputReverseEnergy.x.0 1.3.6.1.4.1.47952.1.1.1.32.x.0	INTEGER	140	[Wh]
netioOutputReverseEnergyNR.x.0 1.3.6.1.4.1.47952.1.1.1.33.x.0	INTEGER	620	[Wh]
netioVoltage 1.3.6.1.4.1.47952.1.2.1.0	INTEGER	239100	Voltage in the power grid [mV]
netioFrequency 1.3.6.1.4.1.47952.1.2.2.0	INTEGER	49900	Frequency in the power grid [mHz]
netioTotalCurrent 1.3.6.1.4.1.47952.1.2.3.0	INTEGER	195	[mA]
netioOverallPowerFactor 1.3.6.1.4.1.47952.1.2.4.0	INTEGER	534	Current power factor * 1000
netioTotalLoad 1.3.6.1.4.1.47952.1.2.5.0	INTEGER	24	[W]
netioTotalEnergy 1.3.6.1.4.1.47952.1.2.6.0	INTEGER	13	[Wh]
netioEnergyStart 1.3.6.1.4.1.47952.1.2.7.0	DateTime	2017-6-23,5:47:3.0,+0:0	Initial date and time. UTC based*
netioTotalPhase 1.3.6.1.4.1.47952.1.2.8.0	INTEGER	120	[°]
netioTotalReverseEnergy 1.3.6.1.4.1.47952.1.2.9.0	INTEGER	420	[Wh]
netioTotalEnergyNR 1.3.6.1.4.1.47952.1.2.10.0	INTEGER	3200	[Wh]
netioTotalReverseEnergyNR 1.3.6.1.4.1.47952.1.2.11.0	INTEGER	1020	[Wh]
netioInputID.x.0 1.3.6.1.4.1.47952.1.3.1.1.x.0	INTEGER	1	
netioInputName.x.0 1.3.6.1.4.1.47952.1.3.1.2.x.0	STRING	Input 1	Based on user defined name
netioInputState.x.0 1.3.6.1.4.1.47952.1.3.1.3.x.0	INTEGER	off(0), on(1)	
netioInputStateString.x.0	STRING	"off", "on"	

1.3.6.1.4.1.47952.1.3.1.4.x.0			
netioInputS0Counter.x.0 1.3.6.1.4.1.47952.1.3.1.5.x.0	INTEGER	30	Number of S0 pulses

*: The start time that Energy (cumulated consumption) is counted from. The value is the same for all outputs.

Note: Metered values are available only for device with metering support. Other devices return value "0".

Control (write)

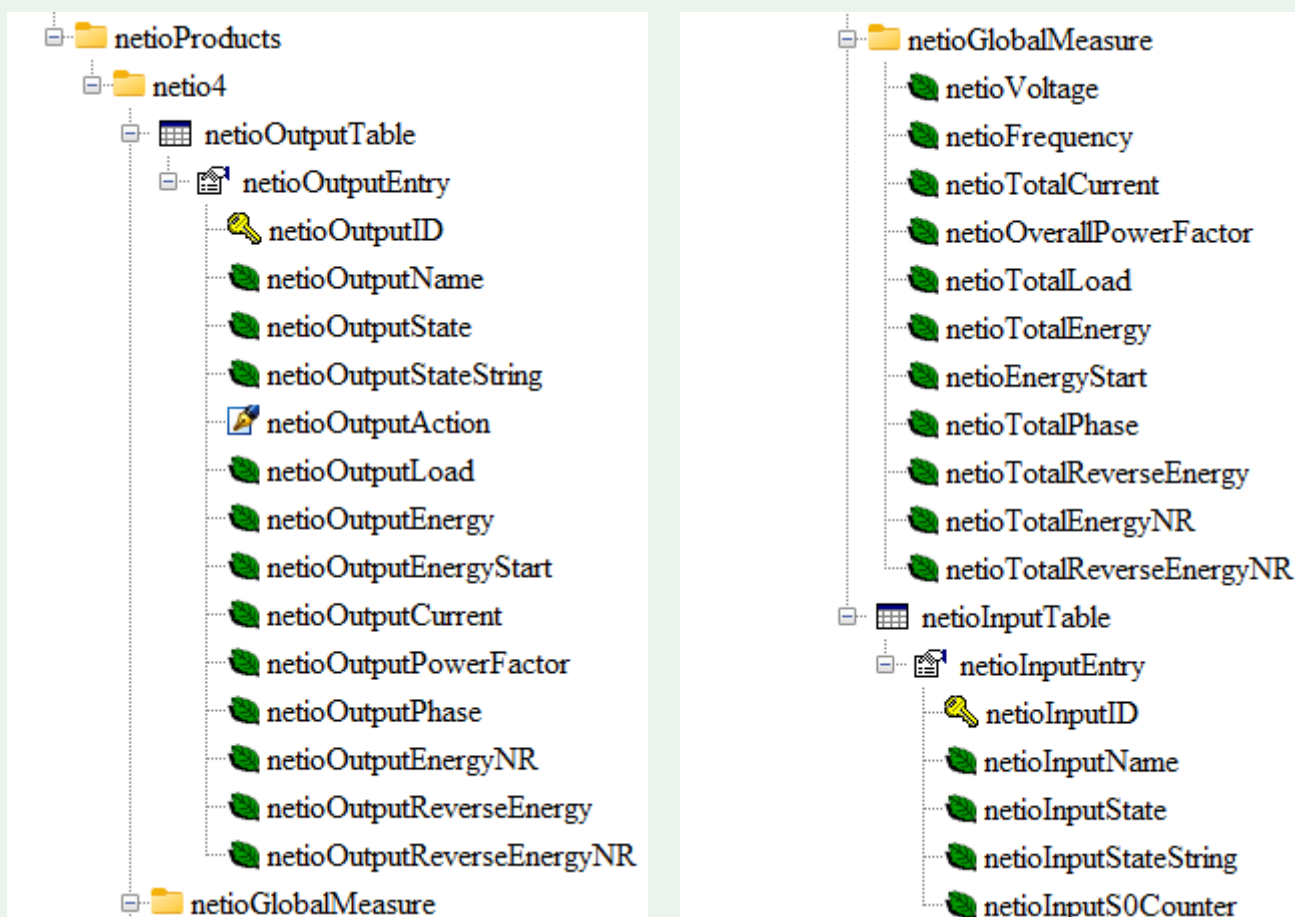
Object OID x – output number (1 - 8)	Type	Value	Action
netioOutputAction.x.0 1.3.6.1.4.1.47952.1.1.1.5.x.0	INTEGER	0 1 2 3 4 5	Turn OFF Turn ON Short OFF delay (restart) Short ON delay Toggle (invert the state) No change

NETIO MIB (version 1.07):

Version available for all standard NETIO devices (all except NETIO PowerPDU 4C as mentioned earlier)

Note: Linux-based devices have latest MIB version 1.05 with no inputs, NR counters and Phase metering support.

NETIO MIB Tree structure



Picture 4 –M2M API Protocols / MIB structure

NETIO MIB file content

```
NETIO-PRODUCTS-NETIO-MIB DEFINITIONS ::= BEGIN

--      NETIO-MIB 1.07
--
--      History:
```

```

--      1.00      13.7.2016  Bretislav Bakala - Created
--      1.01      27.3.2017  Petr Kristal - MODULE-IDENTITY capitals change
--      1.02      27.5.2017  Jiri Zouhar - Added voltage, frequency, current and
power factor.
--      1.03      8.12.2017  Jiri Zouhar - Fixed outlet power factor range & naming.
netioGlobalMeasure and netioOutletTable format definition fixed.
--      1.04      8.12.2017  Jiri Zouhar - Synchronized naming conventions with other
protocols
--      1.05      6.1.2018   Jiri Zouhar - Change outlet to output in all names
--      1.06      7.5.2020   David Foldyna/Ignac Jebetyc - added Inputs support
--      1.07      1.4.2020 (april) Frantisek Vochcampadlo - NR counters added
--
-- This is an MIB file for Netio product family - programable and remote
controlable LAN/WiFi power sockets
-- www.netio-products.com

IMPORTS
    MODULE-IDENTITY, enterprises, OBJECT-TYPE      FROM SNMPv2-SMI
    DateAndTime                                     FROM SNMPv2-TC

;

netioProducts MODULE-IDENTITY
    LAST-UPDATED "201703270000Z"
    ORGANIZATION "www.netio-products.com"
    CONTACT-INFO
        "postal: NETIO products a.s.
        Mezi vodami 1955/19
        CZ 14300, Praha 4, Modrany

        email: info@netio.eu"
    DESCRIPTION "Netio specific data structures"
    REVISION    "201703270000Z"
    DESCRIPTION "NETIO products module"
    ::= { enterprises 47952 }

--
-- top level structure
--
netio4      OBJECT IDENTIFIER ::= { netioProducts 1 }

```

```

netioOutputTable OBJECT-TYPE
    SYNTAX      SEQUENCE OF NetioOutputEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "This table contains current status of Netio outputs."

    ::= { netio4 1 }

netioOutputEntry OBJECT-TYPE
    SYNTAX      NetioOutputEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "A row describing one netio output."
    INDEX      { netioOutputID }
    ::= { netioOutputTable 1 }

NetioOutputEntry ::= SEQUENCE {
    netioOutputID INTEGER(1..8),
    netioOutputName  OCTET STRING,
    netioOutputState  INTEGER(0..1),
    netioOutputStateString  OCTET STRING,
    netioOutputAction  INTEGER(0..5),
    netioOutputLoad  INTEGER(0..2147483647), -- Integer32
    netioOutputEnergy  INTEGER (0..9223372036854775807), -- Integer64
    netioOutputEnergyStart  DateAndTime,
    netioOutputCurrent  INTEGER(0..2147483647),
    netioOutputPowerFactor  INTEGER(0..1000),
    netioOutputPhase  INTEGER(0..1000),
    netioOutputEnergyNR  INTEGER (0..9223372036854775807), -- Integer64
    netioOutputReverseEnergy  INTEGER (0..9223372036854775807) -- Integer64
}

netioOutputID OBJECT-TYPE
    SYNTAX      INTEGER(1..8)
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Output number."

```

```
::= { netioOutputEntry 1 }
```

```
netioOutputName OBJECT-TYPE
```

```
SYNTAX      OCTET STRING (SIZE(0..100))
```

```
MAX-ACCESS  read-only
```

```
STATUS      current
```

```
DESCRIPTION
```

```
"Description of the output given by user."
```

```
::= { netioOutputEntry 2 }
```

```
netioOutputState OBJECT-TYPE
```

```
SYNTAX      INTEGER {
```

```
    off(0),
```

```
    on(1)
```

```
}
```

```
MAX-ACCESS  read-only
```

```
STATUS      current
```

```
DESCRIPTION
```

```
"Current output state."
```

```
::= { netioOutputEntry 3 }
```

```
netioOutputStateString OBJECT-TYPE
```

```
SYNTAX      OCTET STRING (SIZE(0..20))
```

```
MAX-ACCESS  read-only
```

```
STATUS      current
```

```
DESCRIPTION
```

```
"Current output state (string)."
```

```
::= { netioOutputEntry 4 }
```

```
netioOutputAction OBJECT-TYPE
```

```
SYNTAX      INTEGER {
```

```
    off(0),
```

```
    on(1),
```

```
    reset(2),
```

```
    shortOn(3),
```

```
    switch(4),
```

```
    idle(5)
```

```
}
```

```
MAX-ACCESS  read-write
```

```
STATUS      current
```

```
DESCRIPTION
```


"Command to control output."

::= { netioOutputEntry 5 }

netioOutputLoad OBJECT-TYPE

SYNTAX INTEGER(0..2147483647)

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Current power consumption in Watts. 0 when power meters are not available"

::= { netioOutputEntry 25 }

netioOutputEnergy OBJECT-TYPE

SYNTAX INTEGER(0..9223372036854775807)

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Cumulated consumption from netioOutputEnergyStart. In Wh. 0 when power meters are not available."

::= { netioOutputEntry 26 }

netioOutputEnergyStart OBJECT-TYPE

SYNTAX DateAndTime

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Start time from which energy is counted. UTC based. for backward compatibility only, is the same for all outputs. use netioGlobalMeasure.netioEnergyStart instead."

::= { netioOutputEntry 27 }

netioOutputCurrent OBJECT-TYPE

SYNTAX INTEGER(0..2147483647)

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Current current in mA. 0 when power meters are not available"

::= { netioOutputEntry 28 }

netioOutputPowerFactor OBJECT-TYPE

SYNTAX INTEGER(0..1000)

MAX-ACCESS read-only

```
STATUS      current
DESCRIPTION
"Current power factor * 1000. 0 when power meters are not available"
::= { netioOutputEntry 29 }
```

```
netioOutputPhase OBJECT-TYPE
SYNTAX      INTEGER(0..1000)
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
"Current phase * 100. 999 when power meters are not available"
::= { netioOutputEntry 30 }
```

```
netioOutputEnergyNR OBJECT-TYPE
SYNTAX      INTEGER(0..9223372036854775807)
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
"Cumulated non-resettable consumption from the born of device. In Wh. 0 when
power meters are not available."
::= { netioOutputEntry 31 }
```

```
netioOutputReverseEnergy OBJECT-TYPE
SYNTAX      INTEGER(0..9223372036854775807)
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
"Cumulated delivered energy from netioOutputEnergyStart. In Wh. 0 when power
meters are not available."
::= { netioOutputEntry 32 }
```

```
netioOutputReverseEnergyNR OBJECT-TYPE
SYNTAX      INTEGER(0..9223372036854775807)
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
"Cumulated non-resettable delivered energy from netioOutputEnergyStart. In Wh.
0 when power meters are not available."
::= { netioOutputEntry 33 }
```

```
netioGlobalMeasure OBJECT IDENTIFIER ::= { netio4 2 }
```

```
netioVoltage OBJECT-TYPE
```

```
SYNTAX      INTEGER(0..2147483647)
```

```
MAX-ACCESS  read-only
```

```
STATUS      current
```

```
DESCRIPTION
```

```
"Voltage in the power grid in mV. 0 when power meters are not available"
```

```
::= { netioGlobalMeasure 1 }
```

```
netioFrequency OBJECT-TYPE
```

```
SYNTAX      INTEGER(0..2147483647)
```

```
MAX-ACCESS  read-only
```

```
STATUS      current
```

```
DESCRIPTION
```

```
"Frequency in the power grid in mHz. 0 when power meters are not available"
```

```
::= { netioGlobalMeasure 2 }
```

```
netioTotalCurrent OBJECT-TYPE
```

```
SYNTAX      INTEGER(0..2147483647)
```

```
MAX-ACCESS  read-only
```

```
STATUS      current
```

```
DESCRIPTION
```

```
"Total current for all outputs in mA. 0 when power meters are not available"
```

```
::= { netioGlobalMeasure 3 }
```

```
netioOverallPowerFactor OBJECT-TYPE
```

```
SYNTAX      INTEGER(0..1000)
```

```
MAX-ACCESS  read-only
```

```
STATUS      current
```

```
DESCRIPTION
```

```
"Overall power factor across all outputs. 0 when power meters are not available"
```

```
::= { netioGlobalMeasure 4 }
```

```
netioTotalLoad OBJECT-TYPE
```

```
SYNTAX      INTEGER(0..2147483647)
```

```
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"Total load for all outputs in W. 0 when power meters are not available"
```

```
::= { netioGlobalMeasure 5 }
```

```
netioTotalEnergy OBJECT-TYPE
```

```
SYNTAX INTEGER(0..2147483647)
```

```
MAX-ACCESS read-only
```

```
STATUS current
```

```
DESCRIPTION
```

```
"Total energy for all outputs in Wh starting from netioEnergyStart. 0 when
power meters are not available"
```

```
::= { netioGlobalMeasure 6 }
```

```
netioEnergyStart OBJECT-TYPE
```

```
SYNTAX DateAndTime
```

```
MAX-ACCESS read-only
```

```
STATUS current
```

```
DESCRIPTION
```

```
"Start time from which energy is counted. UTC based."
```

```
::= { netioGlobalMeasure 7 }
```

```
netioTotalPhase OBJECT-TYPE
```

```
SYNTAX INTEGER(0..2147483647)
```

```
MAX-ACCESS read-only
```

```
STATUS current
```

```
DESCRIPTION
```

```
"Total phase for all outputs. 0 or 999 when this value is not available"
```

```
::= { netioGlobalMeasure 8 }
```

```
netioTotalReverseEnergy OBJECT-TYPE
```

```
SYNTAX INTEGER(0..2147483647)
```

```
MAX-ACCESS read-only
```

```
STATUS current
```

```
DESCRIPTION
```

```
"Total Reverse Energy for all outputs."
```

```
::= { netioGlobalMeasure 9 }
```

```
netioTotalEnergyNR OBJECT-TYPE
```

```
SYNTAX      INTEGER(0..2147483647)
```

```
MAX-ACCESS  read-only
```

```
STATUS      current
```

```
DESCRIPTION
```

```
"Total Energy Non Ressetable for all outputs."
```

```
::= { netioGlobalMeasure 10 }
```

```
netioTotalReverseEnergyNR OBJECT-TYPE
```

```
SYNTAX      INTEGER(0..2147483647)
```

```
MAX-ACCESS  read-only
```

```
STATUS      current
```

```
DESCRIPTION
```

```
"Total Reverse Energy Non Ressetable for all outputs."
```

```
::= { netioGlobalMeasure 11 }
```

```
netioInputTable OBJECT-TYPE
```

```
SYNTAX      SEQUENCE OF NetioInputEntry
```

```
MAX-ACCESS  not-accessible
```

```
STATUS      current
```

```
DESCRIPTION
```

```
"This table contains current status of Netio Inputs."
```

```
::= { netio4 3 }
```

```
netioInputEntry OBJECT-TYPE
```

```
SYNTAX      NetioInputEntry
```

```
MAX-ACCESS  not-accessible
```

```
STATUS      current
```

```
DESCRIPTION
```

```
"A row describing one netio input."
```

```
INDEX      { netioInputID }
```

```
::= {netioInputTable 1 }
```

```
NetioInputEntry ::= SEQUENCE {
```

```
  netioInputID INTEGER(1..8),
```

```
netioInputName      OCTET STRING,
netioInputState     INTEGER(0..1),
netioInputStateString  OCTET STRING,
netioInputS0Counter INTEGER (0..9223372036854775807) -- Integer64
}
```

```
netioInputID OBJECT-TYPE
    SYNTAX      INTEGER(1..8)
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Input number."
    ::= { netioInputEntry 1 }
```

```
netioInputName OBJECT-TYPE
    SYNTAX      OCTET STRING (SIZE(0..100))
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Description of the input given by user."
    ::= { netioInputEntry 2 }
```

```
netioInputState OBJECT-TYPE
    SYNTAX      INTEGER {
        off(0),
        on(1)
    }
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Current input state."
    ::= { netioInputEntry 3 }
```

```
netioInputStateString OBJECT-TYPE
    SYNTAX      OCTET STRING (SIZE(0..20))
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Current input state (string)."
    ::= { netioInputEntry 4 }
```

```
netioInputS0Counter OBJECT-TYPE
    SYNTAX      INTEGER(0..9223372036854775807)
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "S0Counter."
        ::= { netioInputEntry 5 }

END
```

NETIO PowerBOX 3Px – snmpwalk listing

Note: In PowerBOX 3Px devices there are 3 outputs and no metered channels.

```
NETIO-PRODUCTS-NETIO-MIB::netioOutputID.1.0 = INTEGER: 1
NETIO-PRODUCTS-NETIO-MIB::netioOutputID.2.0 = INTEGER: 2
NETIO-PRODUCTS-NETIO-MIB::netioOutputID.3.0 = INTEGER: 3
NETIO-PRODUCTS-NETIO-MIB::netioOutputID.4.0 = INTEGER: 4
NETIO-PRODUCTS-NETIO-MIB::netioOutputName.1.0 = STRING: "Power output 1"
NETIO-PRODUCTS-NETIO-MIB::netioOutputName.2.0 = STRING: "Power output 2"
NETIO-PRODUCTS-NETIO-MIB::netioOutputName.3.0 = STRING: "Power output 3"
NETIO-PRODUCTS-NETIO-MIB::netioOutputName.4.0 = ""
NETIO-PRODUCTS-NETIO-MIB::netioOutputState.1.0 = INTEGER: on(1)
NETIO-PRODUCTS-NETIO-MIB::netioOutputState.2.0 = INTEGER: on(1)
NETIO-PRODUCTS-NETIO-MIB::netioOutputState.3.0 = INTEGER: on(1)
NETIO-PRODUCTS-NETIO-MIB::netioOutputState.4.0 = INTEGER: -1
NETIO-PRODUCTS-NETIO-MIB::netioOutputStateString.1.0 = STRING: "on"
NETIO-PRODUCTS-NETIO-MIB::netioOutputStateString.2.0 = STRING: "on"
NETIO-PRODUCTS-NETIO-MIB::netioOutputStateString.3.0 = STRING: "on"
NETIO-PRODUCTS-NETIO-MIB::netioOutputStateString.4.0 = ""
NETIO-PRODUCTS-NETIO-MIB::netioOutputAction.1.0 = INTEGER: on(1)
NETIO-PRODUCTS-NETIO-MIB::netioOutputAction.2.0 = INTEGER: on(1)
NETIO-PRODUCTS-NETIO-MIB::netioOutputAction.3.0 = INTEGER: on(1)
NETIO-PRODUCTS-NETIO-MIB::netioOutputAction.4.0 = INTEGER: -1
NETIO-PRODUCTS-NETIO-MIB::netioOutputLoad.1.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputLoad.2.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputLoad.3.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputLoad.4.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputEnergy.1.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputEnergy.2.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputEnergy.3.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputEnergy.4.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputEnergyStart.1.0 = STRING: 1970-1-1,0:0:0.0,.2
NETIO-PRODUCTS-NETIO-MIB::netioOutputEnergyStart.2.0 = STRING: 1970-1-1,0:0:0.0,.2
NETIO-PRODUCTS-NETIO-MIB::netioOutputEnergyStart.3.0 = STRING: 1970-1-1,0:0:0.0,.2
NETIO-PRODUCTS-NETIO-MIB::netioOutputEnergyStart.4.0 = STRING: 1970-1-1,0:0:0.0,.2
NETIO-PRODUCTS-NETIO-MIB::netioOutputCurrent.1.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputCurrent.2.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputCurrent.3.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputCurrent.4.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputPowerFactor.1.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputPowerFactor.2.0 = INTEGER: 0
```



```
NETIO-PRODUCTS-NETIO-MIB::netioOutputPowerFactor.3.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputPowerFactor.4.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputPhase.1.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputPhase.2.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputPhase.3.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputPhase.4.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputEnergyNR.1.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputEnergyNR.2.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputEnergyNR.3.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputEnergyNR.4.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputReverseEnergy.1.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputReverseEnergy.2.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputReverseEnergy.3.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputReverseEnergy.4.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputReverseEnergyNR.1.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputReverseEnergyNR.2.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputReverseEnergyNR.3.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputReverseEnergyNR.4.0 = INTEGER: 0
```

NETIO PowerPDU 8QS – snmpwalk listing

Note: In PowerPDU 8QS there are 2 metered channels (Output 1 + all)

```
NETIO-PRODUCTS-NETIO-MIB::netioOutputID.1.0 = INTEGER: 1
NETIO-PRODUCTS-NETIO-MIB::netioOutputID.2.0 = INTEGER: 2
NETIO-PRODUCTS-NETIO-MIB::netioOutputID.3.0 = INTEGER: 3
NETIO-PRODUCTS-NETIO-MIB::netioOutputID.4.0 = INTEGER: 4
NETIO-PRODUCTS-NETIO-MIB::netioOutputID.5.0 = INTEGER: 5
NETIO-PRODUCTS-NETIO-MIB::netioOutputID.6.0 = INTEGER: 6
NETIO-PRODUCTS-NETIO-MIB::netioOutputID.7.0 = INTEGER: 7
NETIO-PRODUCTS-NETIO-MIB::netioOutputID.8.0 = INTEGER: 8
NETIO-PRODUCTS-NETIO-MIB::netioOutputName.1.0 = STRING: "Power output 1"
NETIO-PRODUCTS-NETIO-MIB::netioOutputName.2.0 = STRING: "Power output 2"
NETIO-PRODUCTS-NETIO-MIB::netioOutputName.3.0 = STRING: "Power output 3"
NETIO-PRODUCTS-NETIO-MIB::netioOutputName.4.0 = STRING: "Power output 4"
NETIO-PRODUCTS-NETIO-MIB::netioOutputName.5.0 = STRING: "Power output 5"
NETIO-PRODUCTS-NETIO-MIB::netioOutputName.6.0 = STRING: "Power output 6"
NETIO-PRODUCTS-NETIO-MIB::netioOutputName.7.0 = STRING: "Power output 7"
NETIO-PRODUCTS-NETIO-MIB::netioOutputName.8.0 = STRING: "Power output 8"
```

NETIO-PRODUCTS-NETIO-MIB::netioOutputState.1.0 = INTEGER: on(1)
NETIO-PRODUCTS-NETIO-MIB::netioOutputState.2.0 = INTEGER: on(1)
NETIO-PRODUCTS-NETIO-MIB::netioOutputState.3.0 = INTEGER: off(0)
NETIO-PRODUCTS-NETIO-MIB::netioOutputState.4.0 = INTEGER: on(1)
NETIO-PRODUCTS-NETIO-MIB::netioOutputState.5.0 = INTEGER: on(1)
NETIO-PRODUCTS-NETIO-MIB::netioOutputState.6.0 = INTEGER: off(0)
NETIO-PRODUCTS-NETIO-MIB::netioOutputState.7.0 = INTEGER: on(1)
NETIO-PRODUCTS-NETIO-MIB::netioOutputState.8.0 = INTEGER: on(1)
NETIO-PRODUCTS-NETIO-MIB::netioOutputAction.1.0 = INTEGER: on(1)
NETIO-PRODUCTS-NETIO-MIB::netioOutputAction.2.0 = INTEGER: on(1)
NETIO-PRODUCTS-NETIO-MIB::netioOutputAction.3.0 = INTEGER: off(0)
NETIO-PRODUCTS-NETIO-MIB::netioOutputAction.4.0 = INTEGER: on(1)
NETIO-PRODUCTS-NETIO-MIB::netioOutputAction.5.0 = INTEGER: on(1)
NETIO-PRODUCTS-NETIO-MIB::netioOutputAction.6.0 = INTEGER: off(0)
NETIO-PRODUCTS-NETIO-MIB::netioOutputAction.7.0 = INTEGER: on(1)
NETIO-PRODUCTS-NETIO-MIB::netioOutputAction.8.0 = INTEGER: on(1)
NETIO-PRODUCTS-NETIO-MIB::netioOutputEntry.8.1.0 = STRING: "on"
NETIO-PRODUCTS-NETIO-MIB::netioOutputEntry.8.2.0 = STRING: "on"
NETIO-PRODUCTS-NETIO-MIB::netioOutputEntry.8.3.0 = STRING: "off"
NETIO-PRODUCTS-NETIO-MIB::netioOutputEntry.8.4.0 = STRING: "on"
NETIO-PRODUCTS-NETIO-MIB::netioOutputEntry.8.5.0 = STRING: "on"
NETIO-PRODUCTS-NETIO-MIB::netioOutputEntry.8.6.0 = STRING: "off"
NETIO-PRODUCTS-NETIO-MIB::netioOutputEntry.8.7.0 = STRING: "on"
NETIO-PRODUCTS-NETIO-MIB::netioOutputEntry.8.8.0 = STRING: "on"
NETIO-PRODUCTS-NETIO-MIB::netioOutputLoad.1.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputLoad.2.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputLoad.3.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputLoad.4.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputLoad.5.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputLoad.6.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputLoad.7.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputLoad.8.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputEnergy.1.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputEnergy.2.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputEnergy.3.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputEnergy.4.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputEnergy.5.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputEnergy.6.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputEnergy.7.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputEnergy.8.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputEnergyStart.1.0 = STRING: 1970-1-1,0:0:0.0,.1

NETIO-PRODUCTS-NETIO-MIB::netioOutputEnergyStart.2.0 = STRING: 1970-1-1,0:0:0.0,.1
NETIO-PRODUCTS-NETIO-MIB::netioOutputEnergyStart.3.0 = STRING: 1970-1-1,0:0:0.0,.1
NETIO-PRODUCTS-NETIO-MIB::netioOutputEnergyStart.4.0 = STRING: 1970-1-1,0:0:0.0,.1
NETIO-PRODUCTS-NETIO-MIB::netioOutputEnergyStart.5.0 = STRING: 1970-1-1,0:0:0.0,.1
NETIO-PRODUCTS-NETIO-MIB::netioOutputEnergyStart.6.0 = STRING: 1970-1-1,0:0:0.0,.1
NETIO-PRODUCTS-NETIO-MIB::netioOutputEnergyStart.7.0 = STRING: 1970-1-1,0:0:0.0,.1
NETIO-PRODUCTS-NETIO-MIB::netioOutputEnergyStart.8.0 = STRING: 1970-1-1,0:0:0.0,.1
NETIO-PRODUCTS-NETIO-MIB::netioOutputCurrent.1.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputCurrent.2.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputCurrent.3.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputCurrent.4.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputCurrent.5.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputCurrent.6.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputCurrent.7.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputCurrent.8.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputPowerFactor.1.0 = INTEGER: 100
NETIO-PRODUCTS-NETIO-MIB::netioOutputPowerFactor.2.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputPowerFactor.3.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputPowerFactor.4.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputPowerFactor.5.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputPowerFactor.6.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputPowerFactor.7.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputPowerFactor.8.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputPhase.1.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputPhase.2.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputPhase.3.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputPhase.4.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputPhase.5.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputPhase.6.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputPhase.7.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputPhase.8.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputEnergyNR.1.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputEnergyNR.2.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputEnergyNR.3.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputEnergyNR.4.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputEnergyNR.5.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputEnergyNR.6.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputEnergyNR.7.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputEnergyNR.8.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputReverseEnergy.1.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputReverseEnergy.2.0 = INTEGER: 0

```
NETIO-PRODUCTS-NETIO-MIB::netioOutputReverseEnergy.3.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputReverseEnergy.4.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputReverseEnergy.5.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputReverseEnergy.6.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputReverseEnergy.7.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputReverseEnergy.8.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputReverseEnergyNR.1.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputReverseEnergyNR.2.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputReverseEnergyNR.3.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputReverseEnergyNR.4.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputReverseEnergyNR.5.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputReverseEnergyNR.6.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputReverseEnergyNR.7.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputReverseEnergyNR.8.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioVoltage.0 = INTEGER: 233540
NETIO-PRODUCTS-NETIO-MIB::netioFrequency.0 = INTEGER: 49980
NETIO-PRODUCTS-NETIO-MIB::netioTotalCurrent.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOverallPowerFactor.0 = INTEGER: 100
NETIO-PRODUCTS-NETIO-MIB::netioTotalLoad.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioTotalEnergy.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioEnergyStart.0 = STRING: 1970-1-1,0:0:0.0,.1
NETIO-PRODUCTS-NETIO-MIB::netioTotalPhase.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioTotalReverseEnergy.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioTotalEnergyNR.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioTotalReverseEnergyNR.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioInputID.1.0 = INTEGER: 1
NETIO-PRODUCTS-NETIO-MIB::netioInputID.2.0 = INTEGER: 2
NETIO-PRODUCTS-NETIO-MIB::netioInputName.1.0 = STRING: "Input 1"
NETIO-PRODUCTS-NETIO-MIB::netioInputName.2.0 = STRING: "Input 2"
NETIO-PRODUCTS-NETIO-MIB::netioInputState.1.0 = INTEGER: off(0)
NETIO-PRODUCTS-NETIO-MIB::netioInputState.2.0 = INTEGER: off(0)
NETIO-PRODUCTS-NETIO-MIB::netioInputStateString.1.0 = STRING: "off"
NETIO-PRODUCTS-NETIO-MIB::netioInputStateString.2.0 = STRING: "off"
NETIO-PRODUCTS-NETIO-MIB::netioInputs0Counter.1.0 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioInputs0Counter.2.0 = INTEGER: -1
```

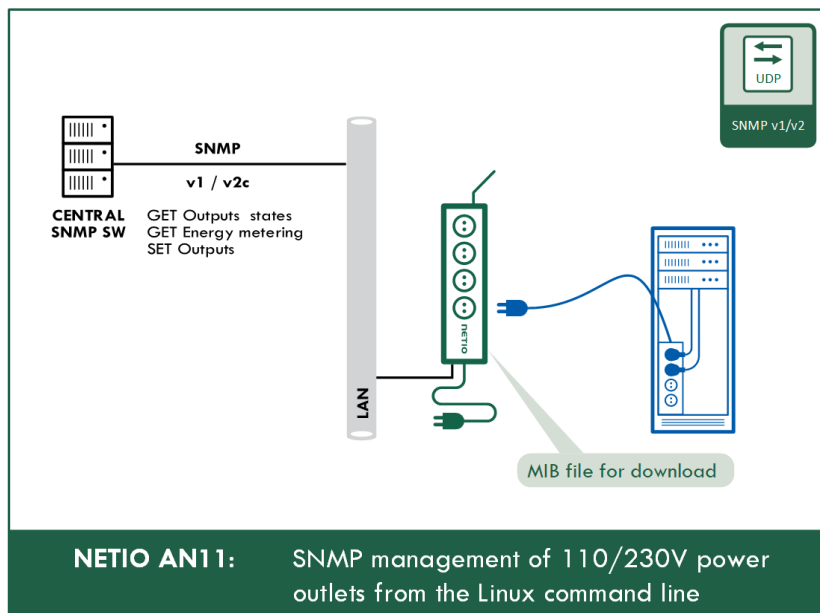
NETIO 4C – snmpwalk listing

Note: In the NETIO PowerPDU 4C model, there are 4 metered outputs

```
NETIO-PRODUCTS-NETIO-MIB::netioOutputID.1 = INTEGER: 1
NETIO-PRODUCTS-NETIO-MIB::netioOutputID.2 = INTEGER: 2
NETIO-PRODUCTS-NETIO-MIB::netioOutputID.3 = INTEGER: 3
NETIO-PRODUCTS-NETIO-MIB::netioOutputID.4 = INTEGER: 4
NETIO-PRODUCTS-NETIO-MIB::netioOutputName.1 = STRING: "output_1"
NETIO-PRODUCTS-NETIO-MIB::netioOutputName.2 = STRING: "output_2"
NETIO-PRODUCTS-NETIO-MIB::netioOutputName.3 = STRING: "output_3"
NETIO-PRODUCTS-NETIO-MIB::netioOutputName.4 = STRING: "output_4"
NETIO-PRODUCTS-NETIO-MIB::netioOutputState.1 = INTEGER: off(0)
NETIO-PRODUCTS-NETIO-MIB::netioOutputState.2 = INTEGER: off(0)
NETIO-PRODUCTS-NETIO-MIB::netioOutputState.3 = INTEGER: off(0)
NETIO-PRODUCTS-NETIO-MIB::netioOutputState.4 = INTEGER: off(0)
NETIO-PRODUCTS-NETIO-MIB::netioOutputStateString.1 = STRING: "off"
NETIO-PRODUCTS-NETIO-MIB::netioOutputStateString.2 = STRING: "off"
NETIO-PRODUCTS-NETIO-MIB::netioOutputStateString.3 = STRING: "off"
NETIO-PRODUCTS-NETIO-MIB::netioOutputStateString.4 = STRING: "off"
NETIO-PRODUCTS-NETIO-MIB::netioOutputLoad.1 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputLoad.2 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputLoad.3 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputLoad.4 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputEnergy.1 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputEnergy.2 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputEnergy.3 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputEnergy.4 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputEnergyStart.1 = STRING: 2021-8-2,11:54:49.0,+0:0
NETIO-PRODUCTS-NETIO-MIB::netioOutputEnergyStart.2 = STRING: 2021-8-2,11:54:49.0,+0:0
NETIO-PRODUCTS-NETIO-MIB::netioOutputEnergyStart.3 = STRING: 2021-8-2,11:54:49.0,+0:0
NETIO-PRODUCTS-NETIO-MIB::netioOutputEnergyStart.4 = STRING: 2021-8-2,11:54:49.0,+0:0
NETIO-PRODUCTS-NETIO-MIB::netioOutputCurrent.1 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputCurrent.2 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputCurrent.3 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputCurrent.4 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputPowerFactor.1 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputPowerFactor.2 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputPowerFactor.3 = INTEGER: 0
NETIO-PRODUCTS-NETIO-MIB::netioOutputPowerFactor.4 = INTEGER: 0
```

AN11 SNMPv1/2c management of 110/230V power outlets from the command line in Windows and Linux

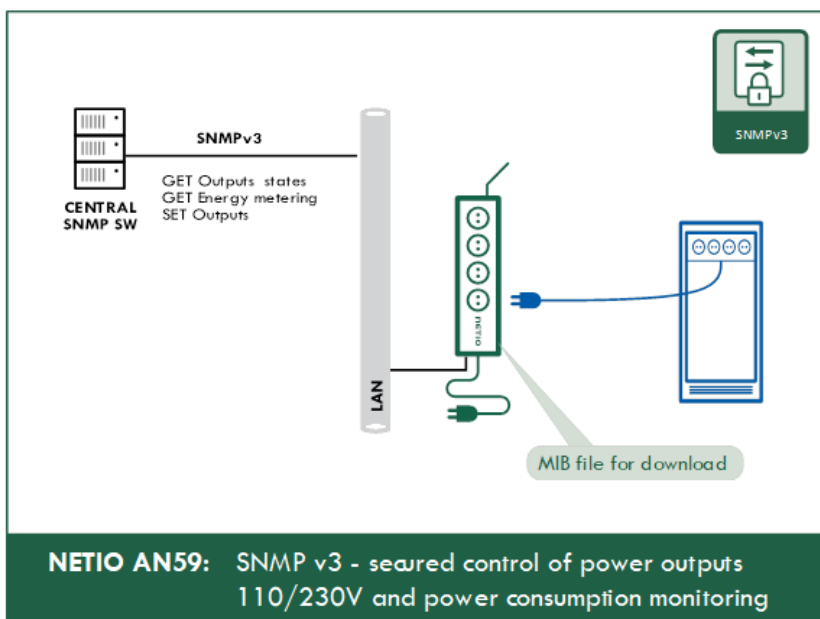
NETIO devices can be monitored and controlled over LAN using SNMP. In this way, it is possible to measure the immediate consumption or to switch individual outputs on or off. This AN11 Application Note shows how to perform SNMPv1/2c read and write operations in MS Windows and Linux.



>> Read the AN11 on www.netio-products.com

AN59 SNMPv3 management of PDU Outputs (Win and Linux CMD)

NETIO devices allow reading of outputs states and values of electrical measurements via SNMPv3 (SNMP get) and control of PDU outputs (SNMP set). AN59 describes how to implement reading and writing using SNMP v3 in MS Windows and Linux.



>> Read the AN11 on www.netio-products.com

Software tools for SNMP reading and control (Windows only)

iReasoning MIB Browser

Free tool for browsing MIB files.

Supports basic devices Read/Write with SNMPv1/2. SNMPv3 is not supported at all.

<https://ireasoning.com/mibbrowser.shtml>

SysUpTime Network Monitor

Tool for SNMP-based monitoring. Personal edition is free to use.

Supports basic devices Read/Write with SNMPv1/2. SNMPv3 is not supported at all.

<http://www.ireasoning.com/network-monitoring-software.php>

PowerSNMP Free Manager

Free lightweight tool that can be used to control and monitor devices.

Supports basic devices Read/Write with SNMPv1/2. SNMPv3 is supported for reading only.

<https://www.dart.com/products/powersnmp-for-net#free-manager>

Document history

Document Revision	Publication Date	Description
1.0	11.2.2017	Initial release for FW 3.1.0, NETIO-MIB 1.05
1.1	1.12.2018	AN11 description added
1.2	9.8.2021	MIB v1.07, protocol structure updated
1.3	14.9.2021	SNMP v3 update, MIB tree, 8QS

Keywords

Title:

M2M SNMP - Protocol API for NETIO power sockets

Author:

NETIO products: networked power sockets

Subject:

SNMP v1 & SNMP v3 protocol documentation and M2M API for NETIO 4x products

Keywords:

SNMP, snmpv3, ssl, snmp power socket, snmp power strip, snmp based api, snmp power socket monitoring, snmp WIFI power socket, NETIO, M2M API, M2M API, power socket, control, power sockets, power outlets, power outlet, output, power control, kwh, voltage, networked power sockets, power outlets 110V, power output 230V, LAN controller power socket, NETIO 4All, NETIO 4C, NETIO 4, truePF, true power factor, TPF power factor