nVent SCHROFF
Guardian Management Gateway

User Manual

Release 1.0

06.04.2020

Doc-No.: 63972-383

Schroff GmbH
Langenalber Str. 96-100
75334 Straubenhardt/Germany
schroff.nVent.com

The Schroff Guardian Management Gateway uses an implementation of the MD5 Message-Digest algorithm that is derived from the RSA Data Security, Inc. MD5 Message-Digest algorithm.

Table of contents

1 Safety

	Read hardware manual and quick start guides The nVent SCHROFF Guardian Management Gateway & IPDU are intended to be installed and maintained by qualified and trained personnel in compliance with local and national electrical codes and safety regulations. The hardware description and the corresponding safety instructions are not scope of this manual. Before initial operation, read the resp. manuals.

⚠	Before using the devices, check connectors and electrical cables. All connectors and cables must be designed and rated in accordance with the technical data.

1.1 Intended Use

The nVent SCHROFF Guardian Management Gateway is an environmental monitoring platform designed to sense, track, store and alarm health and security parameters in an IT-datacenter infrastructure.

The heart of the platform is a compact control unit with just 1U in height/depth and 250 mm in width, it can be installed as 19" unit or into any available space in a data center rack.

The Guardian Management Gateway offers three sensor management ports with each port being able to monitor up to 16 sensor devices with a total cable length of 40 meters per port, allowing a single Guardian Management Gateway unit to monitor multiple racks or complete rack aisles.

Besides monitoring physical parameters like temperature, humidity, smoke, door status or water intrusion, the Guardian Management Gateway can also monitor Schroff RackChiller and In-Row Coolers – with an easy plug and play installation.

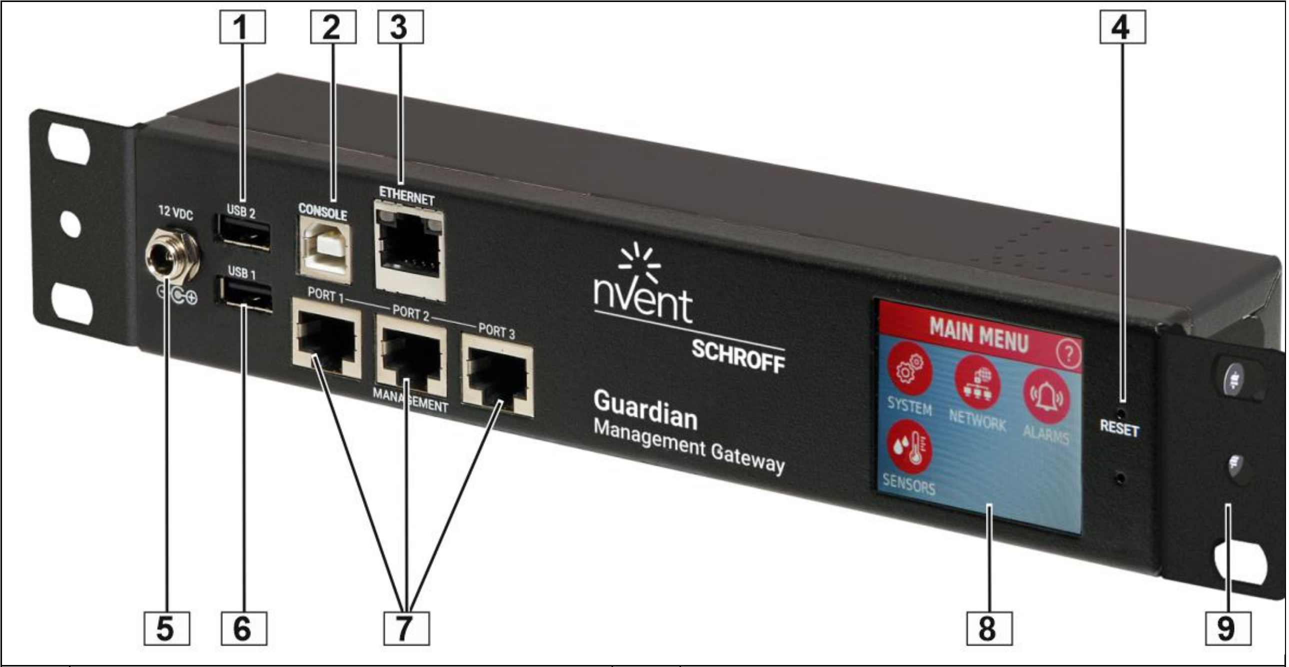
Set-up of the Guardian Management Gateway with security features, sensor configuration, user management, alarm and log management can easily done through a built in Web Interface.

Main access to the Guardian Management Gateway is through the 1 GBE Network interface, supporting industry standard protocols like SNMP, SMTP, HTTPS, BACnet, Modbus/TCP and HPI.

Features:

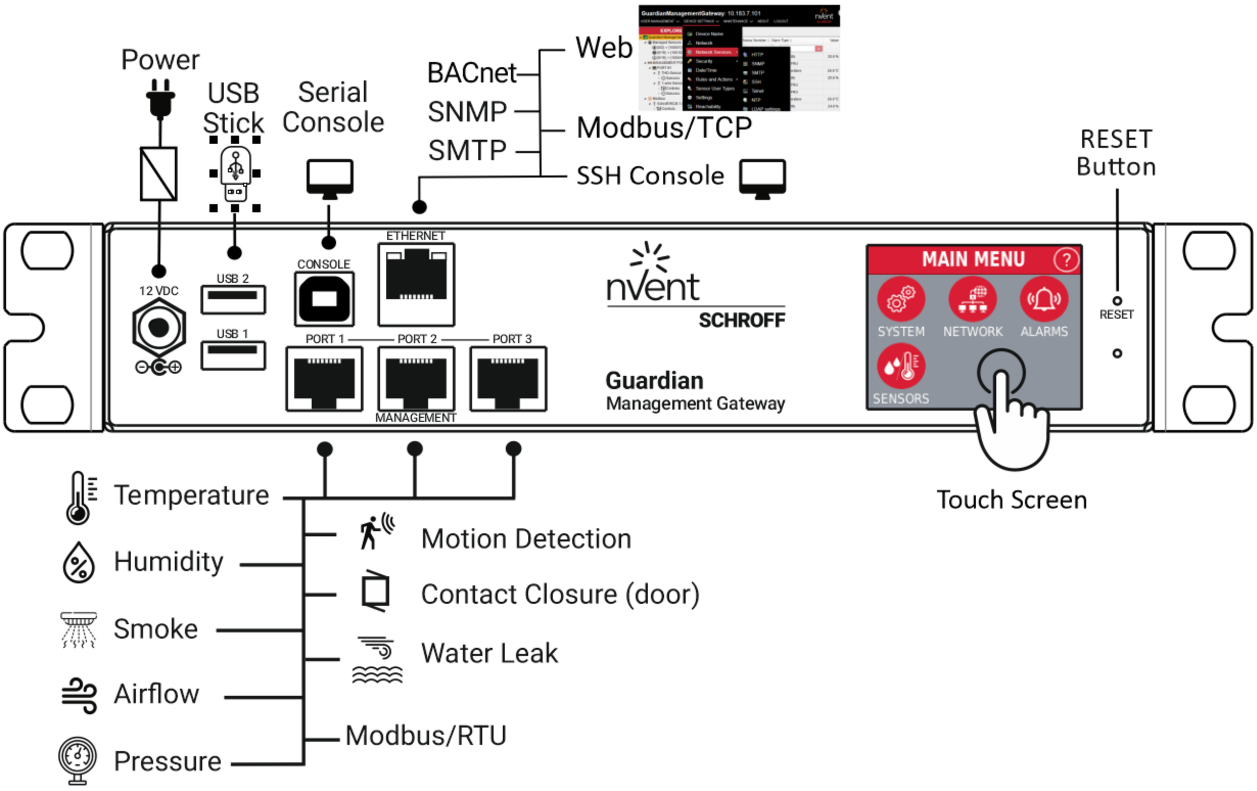
- Data Center environmental monitoring platform
- Compact Design, fits anywhere in a data center rack
- Auto orientation LCD Touch Display
- Web browser GUI or Command Line Interface (CLI) for setup and maintenance.
- Three management ports to connect external sensors and Modbus devices
- Up to 16 sensors/Modbus devices per management port with a cable length of 40 m
- Supports Industry standard network protocols (HTTPS, SNMP, SMTP, Modbus/TCP)
- BACnet support
- AWS support

2 Product Overview Guardian Management Gateway



1	USB Type A Interface 2	6	USB Type A Interface 1
2	USB Type B interface (Console)	7	Sensor Ports
3	Ethernet Interface	8	Touch Display
4	RESET Button	9	Mounting Brackets
5	12 VDC Power Input		

2.1 Guardian Management Gateway Interfaces



The Guardian Management Gateway provides the following interfaces and connectors:

- 1 USB Type-B interface (Console)
- 3 sensor and Modbus device interfaces (RJ45)
- 2 USB Type-A interfaces (USB 1 and USB 2)
- 1 Ethernet interface (RJ45)
- 1 LCD touch display
- 1 Power Barrel Connector 2.1/5.5 mm, female

3 Installing and configuring

ℹ	**This manual describes how to operate and configure the Guardian Management Gateway via the web interface.** **Advanced users can operate and configure the Guardian Management Gateway using a terminal program via the command line interface.** **A User Manual with the Command Line commands is available on request under order number 63972-385.**

ℹ	**For rack mounting and first steps, see the quick start guide, order no.: 63972-380**

3.1 Connect to Network

3.1.1 Wired Connection to LAN

To make a wired connection insert a network cable with RJ45 connector into the socket labelled "Ethernet" and connect the other end to your network device. Once you have a wired connection, you can use the Command Line Interface (CLI) or the Web Interface to access the Guardian Management Gateway

3.1.2 Serial Interface via USB Type B Connector

To use a Command Line Interface (CLI) via the serial interface, connect your computer to the USB Type B connector labelled "CONSOLE".

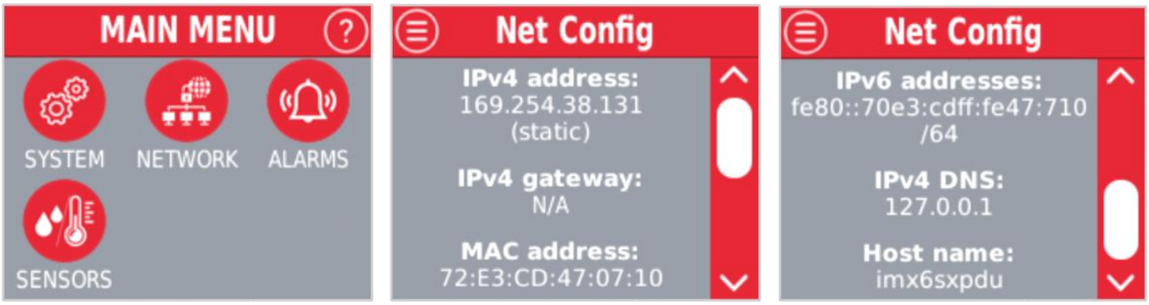
4 Setting up Guardian Management Gateway

After the Guardian Management Gateway is installed and connected to the network, a user can use Command Line Interface (CLI) or Web interface to connect and start communicating with it. For that, the user should know the IP address of the Guardian Management Gateway.

> **By default, the Guardian Management Gateway is configured to obtain its IPv4 address from a DHCP server.**

A DHCP server can be configured either to give the IP address to the Guardian Management Gateway from a dynamic pool (in which case it is not known in advance) or to assign a static IP address based on the MAC address of the Guardian Management Gateway.

In any case, the assigned IP address can be seen by pressing the NETWORK button on the LCD screen of the Guardian Management Gateway.



Net Config

IPv4 address:
169.254.38.131
(static)

IPv4 gateway:
N/A

MAC address:
72:E3:CD:47:07:10

Net Config

IPv6 addresses:
fe80::70e3:cdff:fe47:710
/64

IPv4 DNS:
127.0.0.1

Host name:
imx6sxpdu

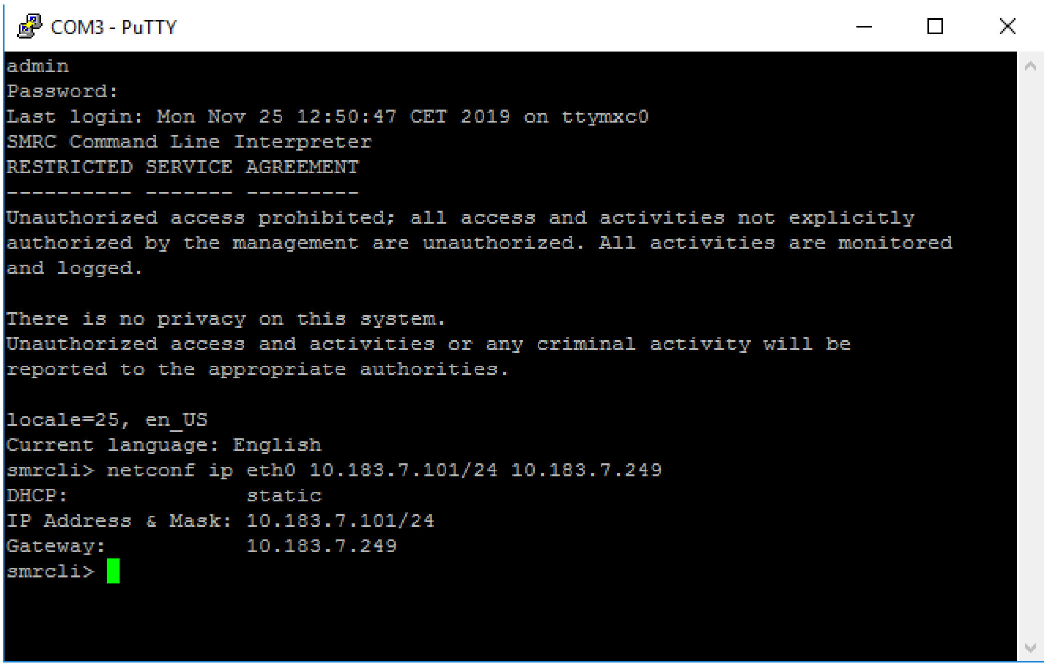
Assign static IP address

To assign a static IP address, complete the following steps:

- Connect your computer to the USB-B port labelled “CONSOLE” with an USB-A/USB-B cable.
- Determine COM port assigned by your computer to the USB Serial connection (Control Panel → System → Hardware → Device Manager → Ports > USB Serial Port).
- Open a terminal program (e.g. PuTTY), set Serial line to the assigned COM port (e.g. COM3), the Speed to 115200, and the Connection type to Serial
- Log in as “admin”
- Password: “admin”
- Assign the IPv4 configuration attributes for the network interface with CLI by entering the following command: `netconf ip <interface> <ip_address>/<mask> [<ipv4_gateway>]`
  - `<interface>` is the name of the adapter (eth0)
  - `<ip_address>` is the IPv4 address assigned to the interface, in the decimal-dot notation
  - `<mask>` is the subnet mask as the number of significant bits; the address with mask may look like `10.183.7.110/24`
  - `<ipv4_gateway>` is the default gateway address in the decimal-dot notation, it is optional here.

  Example:

  ```
  netconf ip eth0 10.183.7.101/24 10.183.7.249
  ```



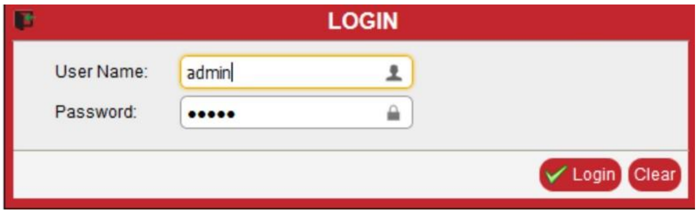
```
admin
Password:
Last login: Mon Nov 25 12:50:47 CET 2019 on ttymxc0
SMRC Command Line Interpreter
RESTRICTED SERVICE AGREEMENT
---------- ------- ---------
Unauthorized access prohibited; all access and activities not explicitly
authorized by the management are unauthorized. All activities are monitored
and logged.

There is no privacy on this system.
Unauthorized access and activities or any criminal activity will be
reported to the appropriate authorities.

locale=25, en_US
Current language: English
smrcli> netconf ip eth0 10.183.7.101/24 10.183.7.249
DHCP:                static
IP Address & Mask:  10.183.7.101/24
Gateway:             10.183.7.249
smrcli>
```

ℹ	To communicate with the Guardian Management Gateway using CLI or Web interface, the user should know a user name and the corresponding password. By default, two users are created: **admin** (password “admin”) with administrative privileges and **user** (password “user”) with normal user privileges. Additional configuration of users can be done after logging in as admin. For security reasons, the password for the admin and user should be changed as early as possible.

5 Getting Started

5.1 Log in using the Web interface

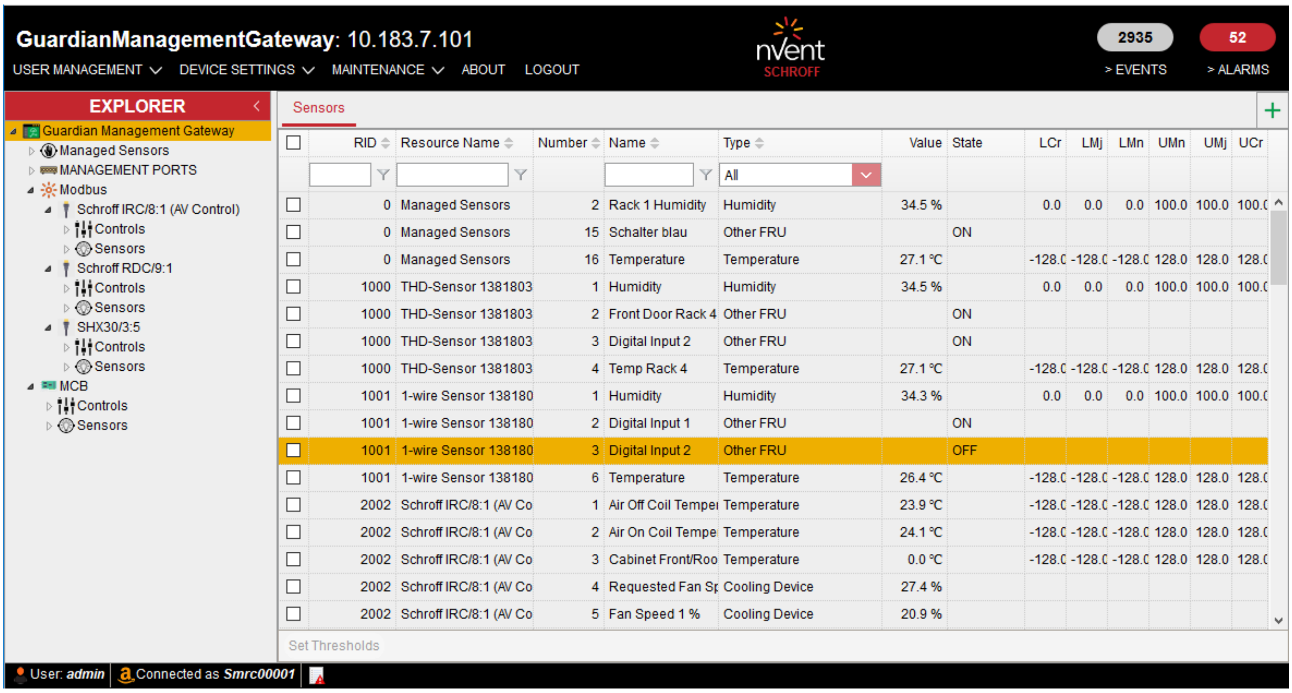
To log in to the Guardian Management Gateway using the Web interface, open the Web browser and point it to the Guardian Management Gateway IP address. The login dialog box appears:



User Name: admin

Password: •••••

After entering the user name and password, and pressing the “Login” button, the main Web interface screen appears:



GuardianManagementGateway: 10.183.7.101

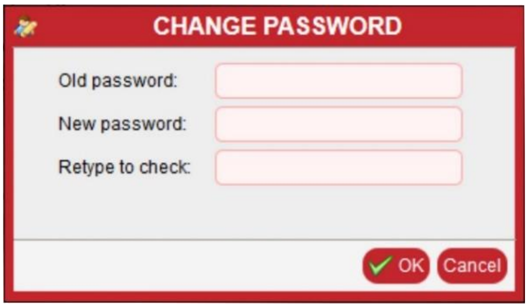
□	RID	Resource Name	Number	Name	Type	Value	State	LCr	LMj	LMn	UMn	UMj	UCr
					All								
□	0	Managed Sensors	2	Rack 1 Humidity	Humidity	34.5 %		0.0	0.0	0.0	100.0	100.0	100.(
□	0	Managed Sensors	15	Schalter blau	Other FRU		ON						
□	0	Managed Sensors	16	Temperature	Temperature	27.1 °C		-128.(	-128.(	-128.(	128.0	128.0	128.(
□	1000	THD-Sensor 1381803	1	Humidity	Humidity	34.5 %		0.0	0.0	0.0	100.0	100.0	100.(
□	1000	THD-Sensor 1381803	2	Front Door Rack 4	Other FRU		ON						
□	1000	THD-Sensor 1381803	3	Digital Input 2	Other FRU		ON						
□	1000	THD-Sensor 1381803	4	Temp Rack 4	Temperature	27.1 °C		-128.(	-128.(	-128.(	128.0	128.0	128.(
□	1001	1-wire Sensor 138180	1	Humidity	Humidity	34.3 %		0.0	0.0	0.0	100.0	100.0	100.(
□	1001	1-wire Sensor 138180	2	Digital Input 1	Other FRU		ON						
□	1001	1-wire Sensor 138180	3	Digital Input 2	Other FRU		OFF						
□	1001	1-wire Sensor 138180	6	Temperature	Temperature	26.4 °C		-128.(	-128.(	-128.(	128.0	128.0	128.(
□	2002	Schroff IRC/8:1 (AV Co	1	Air Off Coil Temper	Temperature	23.9 °C		-128.(	-128.(	-128.(	128.0	128.0	128.(
□	2002	Schroff IRC/8:1 (AV Co	2	Air On Coil Tempe	Temperature	24.1 °C		-128.(	-128.(	-128.(	128.0	128.0	128.(
□	2002	Schroff IRC/8:1 (AV Co	3	Cabinet Front/Roo	Temperature	0.0 °C		-128.(	-128.(	-128.(	128.0	128.0	128.(
□	2002	Schroff IRC/8:1 (AV Co	4	Requested Fan Sp	Cooling Device	27.4 %							
□	2002	Schroff IRC/8:1 (AV Co	5	Fan Speed 1 %	Cooling Device	20.9 %							

User: *admin*

Connected as *Smrc00001*

5.2 Change Password

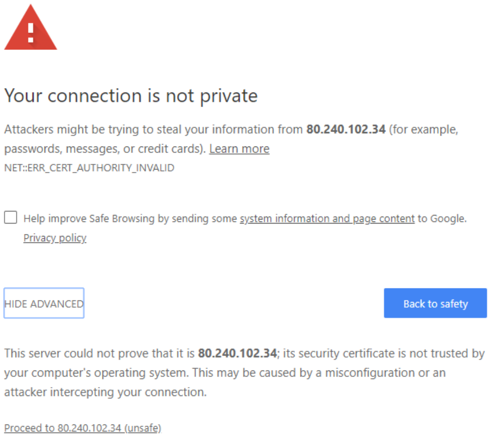
To change the password for the current user, invoke the menu command “USER MANAGEMENT” -> “Change Password”. The “CHANGE PASSWORD” dialog appears, in which the user should type the old password and the new password (two times):



The password will be changed.

5.3 HTTPS Connection

It’s possible to establish a secure HTTP (HTTPS) connection to the Guardian Management Gateway. However the certificate that is originally installed on the Guardian Management Gateway is self-signed, and a warning like this is issued when the connection is established:



Your connection is not private

Attackers might be trying to steal your information from **80.240.102.34** (for example, passwords, messages, or credit cards). Learn more

NET::ERR_CERT_AUTHORITY_INVALID

☐ Help improve Safe Browsing by sending some system information and page content to Google. Privacy policy

This server could not prove that it is **80.240.102.34**; its security certificate is not trusted by your computer's operating system. This may be caused by a misconfiguration or an attacker intercepting your connection.

Proceed to 80.240.102.34 (unsafe)

To get rid of this warning, it is necessary to install a properly signed SSL certificate on the Guardian Management Gateway; it is the user’s responsibility to obtain such a certificate.

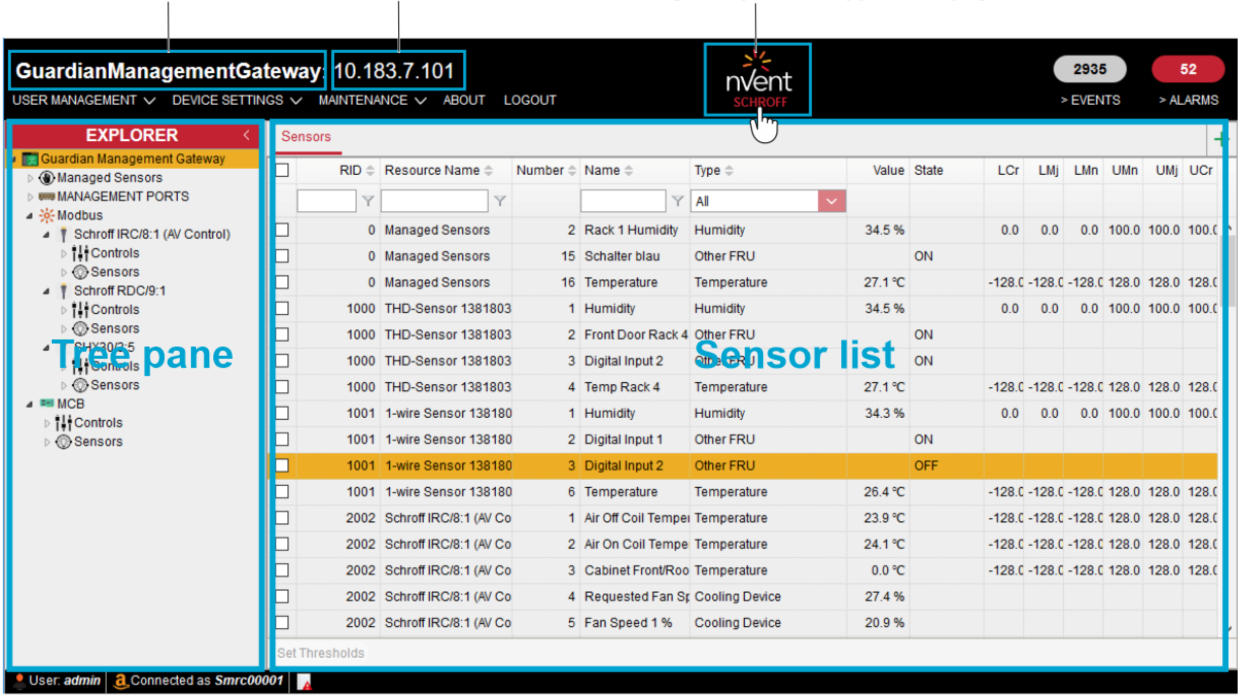
6 Web Interface GUI

6.1 Overview

Device Name

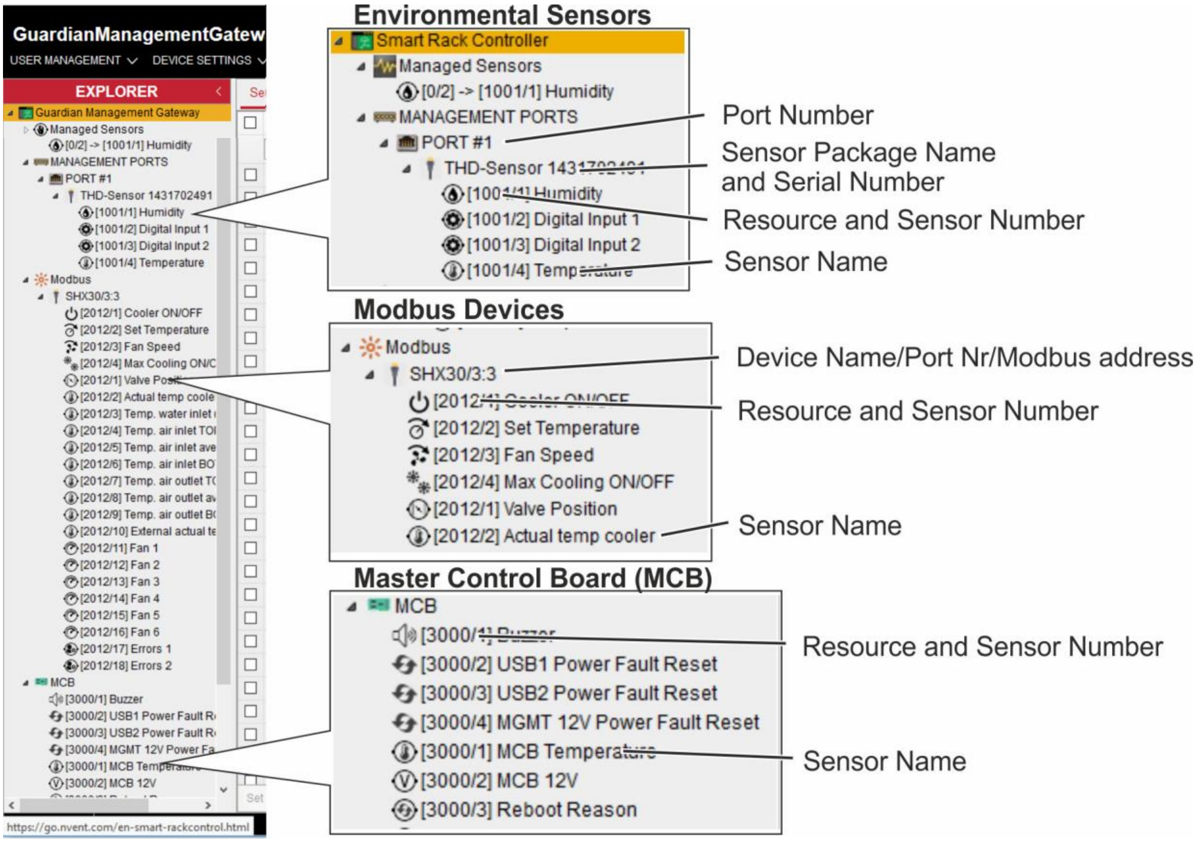
IP Address

Klick on logo to open the support web page



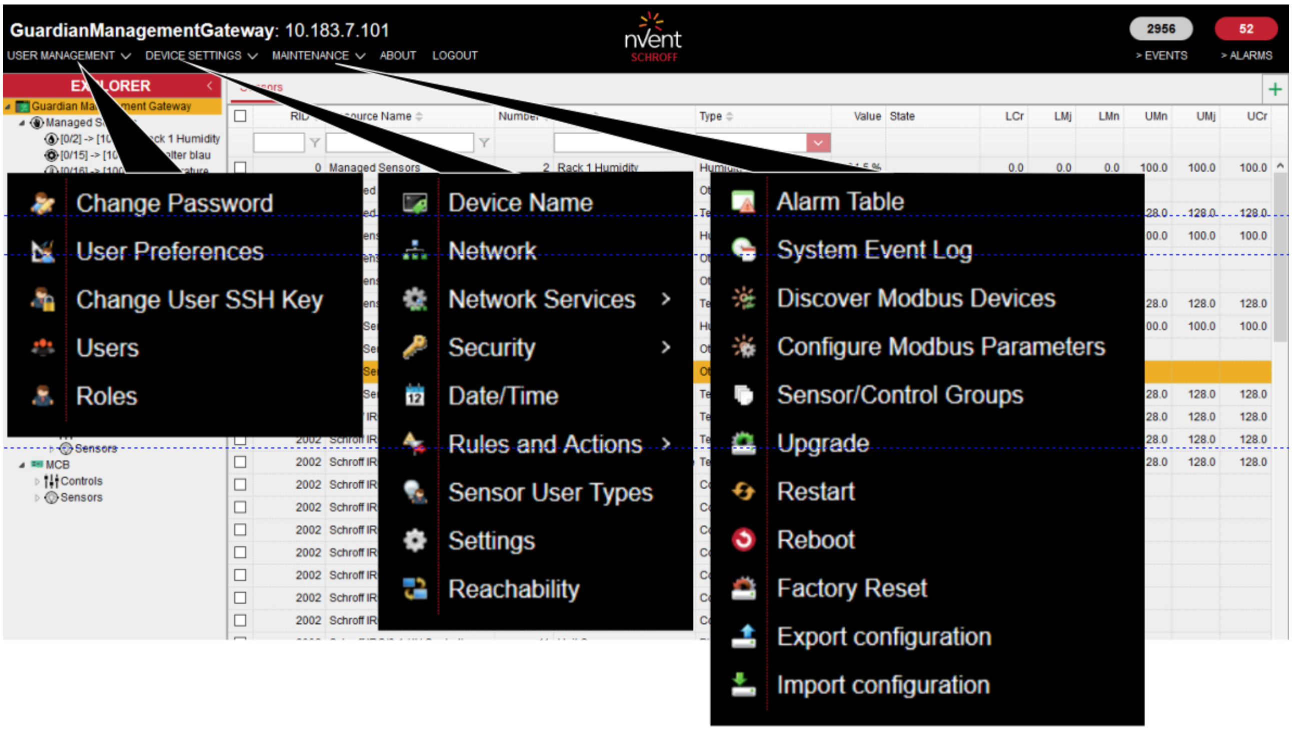
☐	RID	Resource Name	Number	Name	Type	Value	State	LCr	LMj	LMn	UMn	UMj	UCr
					All								
☐	0	Managed Sensors	2	Rack 1 Humidity	Humidity	34.5 %		0.0	0.0	0.0	100.0	100.0	100.(
☐	0	Managed Sensors	15	Schalter blau	Other FRU		ON						
☐	0	Managed Sensors	16	Temperature	Temperature	27.1 °C		-128.(	-128.(	-128.(	128.0	128.0	128.(
☐	1000	THD-Sensor 1381803	1	Humidity	Humidity	34.5 %		0.0	0.0	0.0	100.0	100.0	100.(
☐	1000	THD-Sensor 1381803	2	Front Door Rack 4	Other FRU		ON						
☐	1000	THD-Sensor 1381803	3	Digital Input 2	Other FRU		ON						
☐	1000	THD-Sensor 1381803	4	Temp Rack 4	Temperature	27.1 °C		-128.(	-128.(	-128.(	128.0	128.0	128.(
☐	1001	1-wire Sensor 138180	1	Humidity	Humidity	34.3 %		0.0	0.0	0.0	100.0	100.0	100.(
☐	1001	1-wire Sensor 138180	2	Digital Input 1	Other FRU		ON						
☐	1001	1-wire Sensor 138180	3	Digital Input 2	Other FRU		OFF						
☐	1001	1-wire Sensor 138180	6	Temperature	Temperature	26.4 °C		-128.(	-128.(	-128.(	128.0	128.0	128.(
☐	2002	Schroff IRC/8:1 (AV Co	1	Air Off Coil Tempe|	Temperature	23.9 °C		-128.(	-128.(	-128.(	128.0	128.0	128.(
☐	2002	Schroff IRC/8:1 (AV Co	2	Air On Coil Tempe	Temperature	24.1 °C		-128.(	-128.(	-128.(	128.0	128.0	128.(
☐	2002	Schroff IRC/8:1 (AV Co	3	Cabinet Front/Roo	Temperature	0.0 °C		-128.(	-128.(	-128.(	128.0	128.0	128.(
☐	2002	Schroff IRC/8:1 (AV Co	4	Requested Fan Sp	Cooling Device	27.4 %							
☐	2002	Schroff IRC/8:1 (AV Co	5	Fan Speed 1 %	Cooling Device	20.9 %							

6.1.1 Overview tree pane

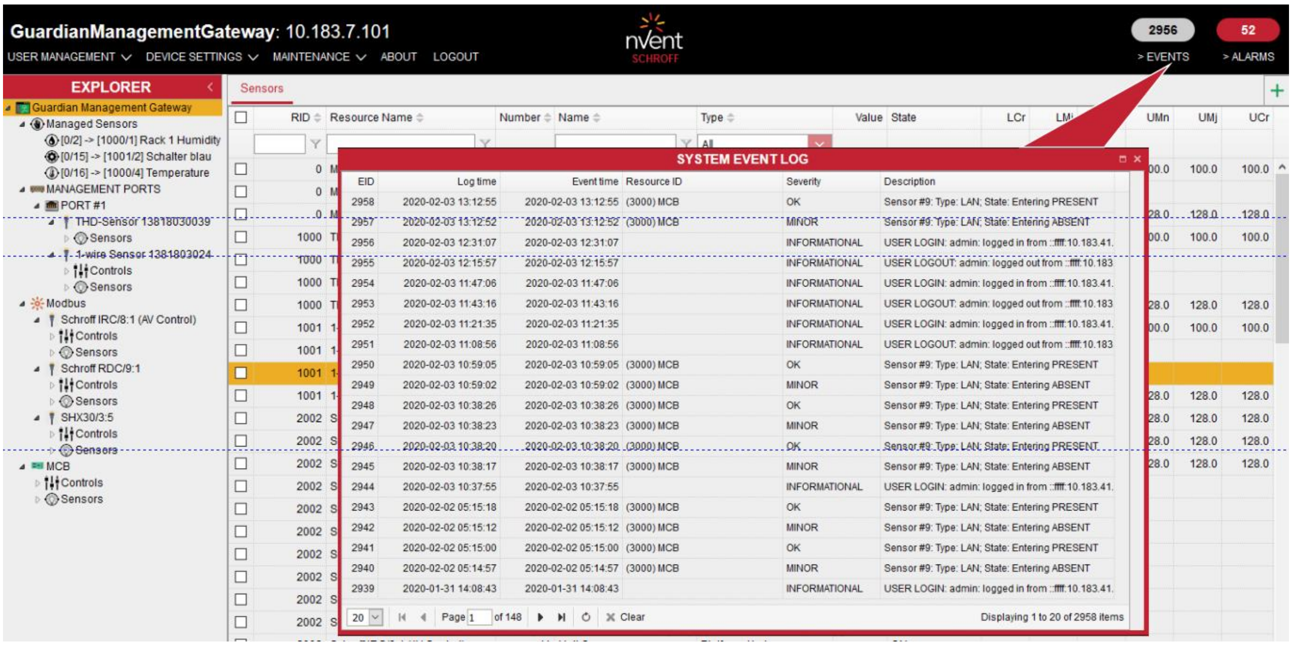


6.1.2 Overview drop down menu

Settings



6.1.3 System Event Log

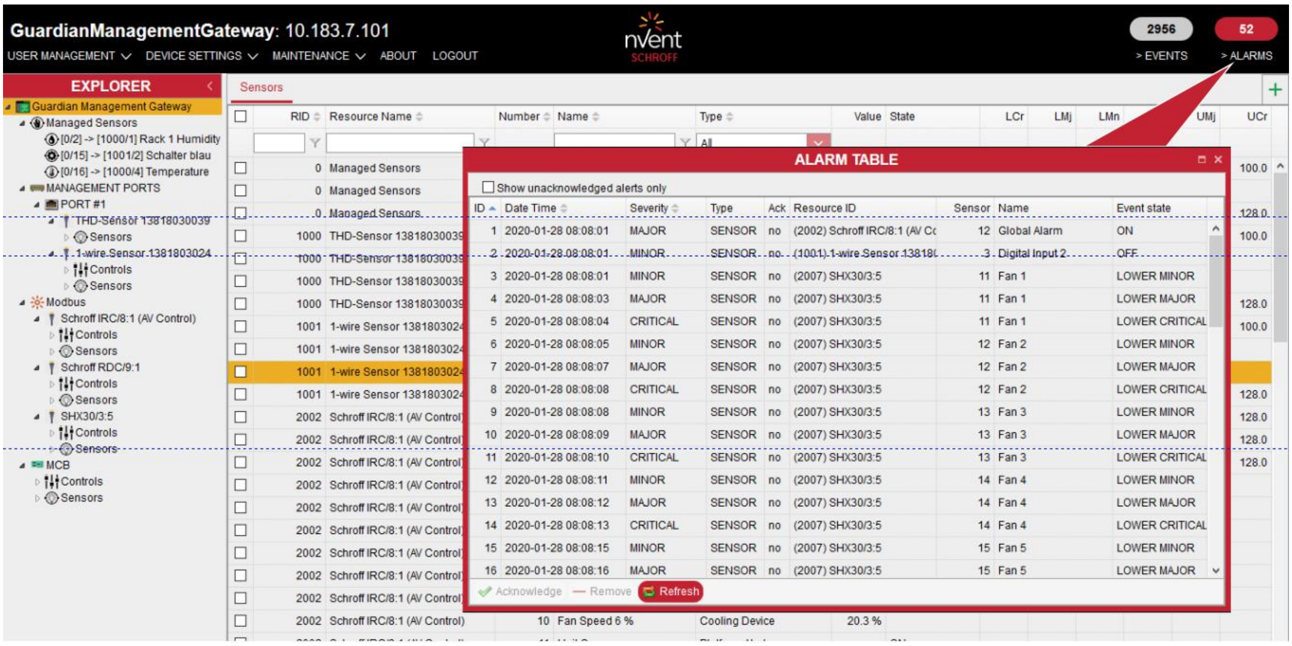


SYSTEM EVENT LOG

EID	Log time	Event time	Resource ID	Severity	Description
2958	2020-02-03 13:12:55	2020-02-03 13:12:55	(3000) MCB	OK	Sensor #9: Type: LAN; State: Entering PRESENT
2957	2020-02-03 13:12:52	2020-02-03 13:12:52	(3000) MCB	MINOR	Sensor #9: Type: LAN; State: Entering ABSENT
2956	2020-02-03 12:31:07	2020-02-03 12:31:07		INFORMATIONAL	USER LOGIN: admin: logged in from ::ffff:10.183.41.
2955	2020-02-03 12:15:57	2020-02-03 12:15:57		INFORMATIONAL	USER LOGOUT: admin: logged out from ::ffff:10.183
2954	2020-02-03 11:47:06	2020-02-03 11:47:06		INFORMATIONAL	USER LOGIN: admin: logged in from ::ffff:10.183.41.
2953	2020-02-03 11:43:16	2020-02-03 11:43:16		INFORMATIONAL	USER LOGOUT: admin: logged out from ::ffff:10.183
2952	2020-02-03 11:21:35	2020-02-03 11:21:35		INFORMATIONAL	USER LOGIN: admin: logged in from ::ffff:10.183.41.
2951	2020-02-03 11:08:56	2020-02-03 11:08:56		INFORMATIONAL	USER LOGOUT: admin: logged out from ::ffff:10.183
2950	2020-02-03 10:59:05	2020-02-03 10:59:05	(3000) MCB	OK	Sensor #9: Type: LAN; State: Entering PRESENT
2949	2020-02-03 10:59:02	2020-02-03 10:59:02	(3000) MCB	MINOR	Sensor #9: Type: LAN; State: Entering ABSENT
2948	2020-02-03 10:38:26	2020-02-03 10:38:26	(3000) MCB	OK	Sensor #9: Type: LAN; State: Entering PRESENT
2947	2020-02-03 10:38:23	2020-02-03 10:38:23	(3000) MCB	MINOR	Sensor #9: Type: LAN; State: Entering ABSENT
2946	2020-02-03 10:38:20	2020-02-03 10:38:20	(3000) MCB	OK	Sensor #9: Type: LAN; State: Entering PRESENT
2945	2020-02-03 10:38:17	2020-02-03 10:38:17	(3000) MCB	MINOR	Sensor #9: Type: LAN; State: Entering ABSENT
2944	2020-02-03 10:37:55	2020-02-03 10:37:55		INFORMATIONAL	USER LOGIN: admin: logged in from ::ffff:10.183.41.
2943	2020-02-02 05:15:18	2020-02-02 05:15:18	(3000) MCB	OK	Sensor #9: Type: LAN; State: Entering PRESENT
2942	2020-02-02 05:15:12	2020-02-02 05:15:12	(3000) MCB	MINOR	Sensor #9: Type: LAN; State: Entering ABSENT
2941	2020-02-02 05:15:00	2020-02-02 05:15:00	(3000) MCB	OK	Sensor #9: Type: LAN; State: Entering PRESENT
2940	2020-02-02 05:14:57	2020-02-02 05:14:57	(3000) MCB	MINOR	Sensor #9: Type: LAN; State: Entering ABSENT
2939	2020-01-31 14:08:43	2020-01-31 14:08:43		INFORMATIONAL	USER LOGIN: admin: logged in from ::ffff:10.183.41.

20 Page 1 of 148 Clear Displaying 1 to 20 of 2958 items

6.1.4 Alarm Table



ALARM TABLE

□ Show unacknowledged alerts only

ID	Date Time	Severity	Type	Ack	Resource ID	Sensor	Name	Event state
1	2020-01-28 08:08:01	MAJOR	SENSOR	no	(2002) Schroff IRC/8:1 (AV Cc	12	Global Alarm	ON
2	2020-01-28 08:08:01	MINOR	SENSOR	no	(1001) 1-wire Sensor 13818(	3	Digital Input 2	OFF
3	2020-01-28 08:08:01	MINOR	SENSOR	no	(2007) SHX30/3:5	11	Fan 1	LOWER MINOR
4	2020-01-28 08:08:03	MAJOR	SENSOR	no	(2007) SHX30/3:5	11	Fan 1	LOWER MAJOR
5	2020-01-28 08:08:04	CRITICAL	SENSOR	no	(2007) SHX30/3:5	11	Fan 1	LOWER CRITICAL
6	2020-01-28 08:08:05	MINOR	SENSOR	no	(2007) SHX30/3:5	12	Fan 2	LOWER MINOR
7	2020-01-28 08:08:07	MAJOR	SENSOR	no	(2007) SHX30/3:5	12	Fan 2	LOWER MAJOR
8	2020-01-28 08:08:08	CRITICAL	SENSOR	no	(2007) SHX30/3:5	12	Fan 2	LOWER CRITICAL
9	2020-01-28 08:08:08	MINOR	SENSOR	no	(2007) SHX30/3:5	13	Fan 3	LOWER MINOR
10	2020-01-28 08:08:09	MAJOR	SENSOR	no	(2007) SHX30/3:5	13	Fan 3	LOWER MAJOR
11	2020-01-28 08:08:10	CRITICAL	SENSOR	no	(2007) SHX30/3:5	13	Fan 3	LOWER CRITICAL
12	2020-01-28 08:08:11	MINOR	SENSOR	no	(2007) SHX30/3:5	14	Fan 4	LOWER MINOR
13	2020-01-28 08:08:12	MAJOR	SENSOR	no	(2007) SHX30/3:5	14	Fan 4	LOWER MAJOR
14	2020-01-28 08:08:13	CRITICAL	SENSOR	no	(2007) SHX30/3:5	14	Fan 4	LOWER CRITICAL
15	2020-01-28 08:08:15	MINOR	SENSOR	no	(2007) SHX30/3:5	15	Fan 5	LOWER MINOR
16	2020-01-28 08:08:16	MAJOR	SENSOR	no	(2007) SHX30/3:5	15	Fan 5	LOWER MAJOR

7 Managing External Devices

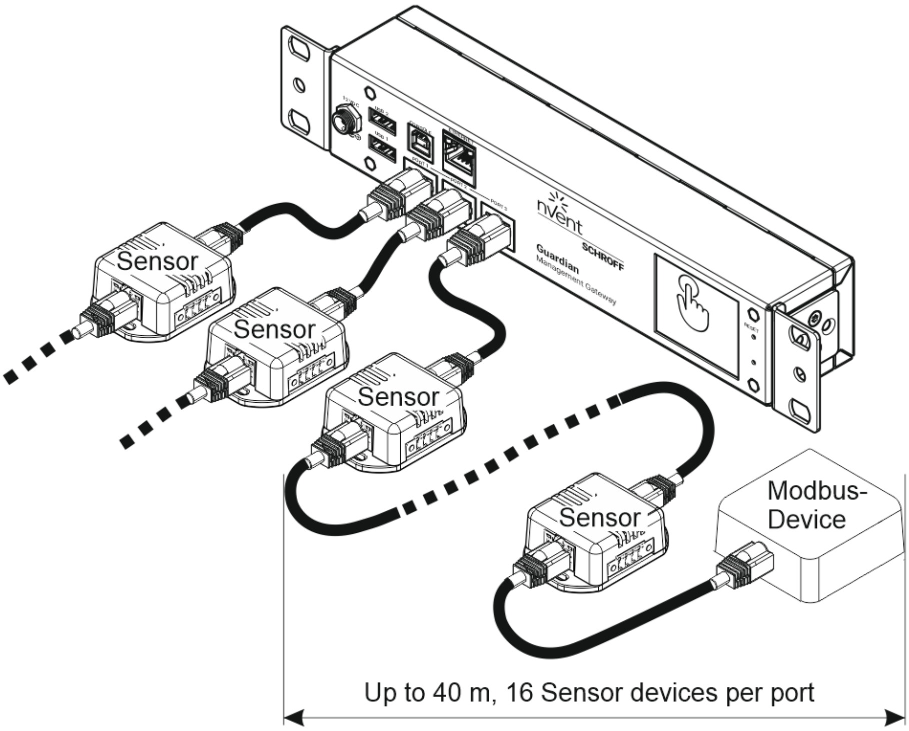
7.1 Managing Schroff environmental sensors

Only nVent SCHROFF Guardian sensor devices are supported by Guardian Management Gateway.

The Guardian Management Gateway offers three sensor management ports with each port is capable of monitoring up to 16 sensor devices with a total cable length of 40 meters per port.

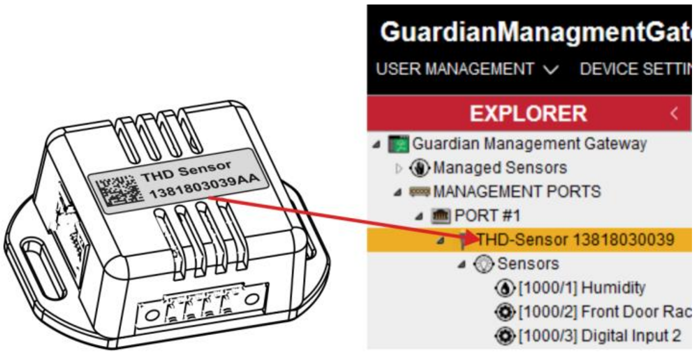
The sensors can be chained together and connected to one of the three interface (RJ45) ports labelled: "MANAGEMENT" on the Guardian Management Gateway.

The sensors are hot-pluggable, that means, they can be connected and disconnected at runtime, without restart or reboot.



Example sensor types:

(T)	Temperature Sensor
(TH)	Multi Sensor (Temperature / Humidity)
(THD)	Multi Sensor (Temperature / Humidity / 2x Digital Input)
(D)	Digital Sensor 1 (2x Digital Input)



GuardianManagmentGat

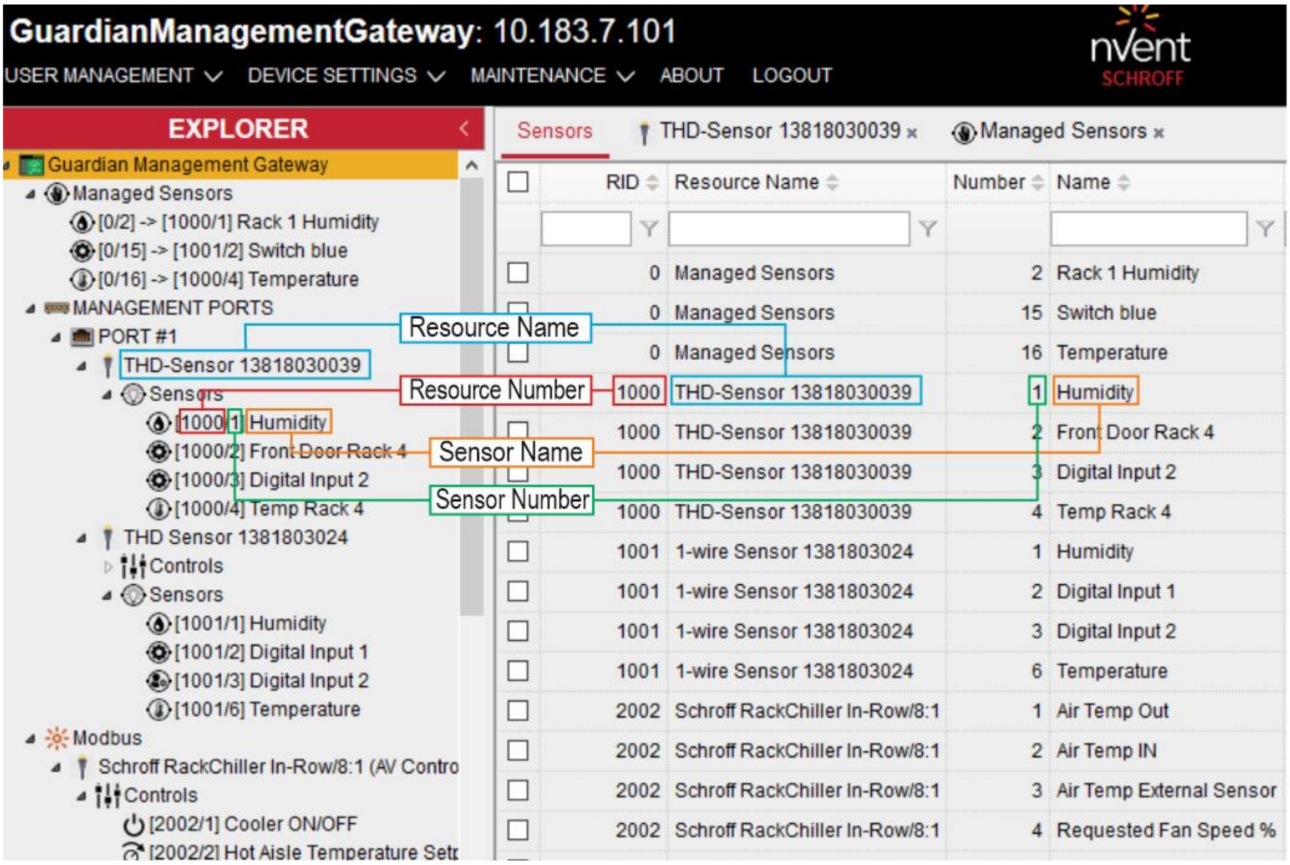
USER MANAGEMENT ⌄ DEVICE SETTI

EXPLORER

- Guardian Management Gateway
  - Managed Sensors
  - MANAGEMENT PORTS
    - PORT #1
      - THD-Sensor 13818030039
        - Sensors
          - [1000/1] Humidity
          - [1000/2] Front Door Rac
          - [1000/3] Digital Input 2

7.1.1 Overview Schroff sensor devices

If a sensor device is connected, it appears in the web interface.



GuardianManagementGateway: 10.183.7.101

USER MANAGEMENT ∨ DEVICE SETTINGS ∨ MAINTENANCE ∨ ABOUT LOGOUT

EXPLORER

- Guardian Management Gateway
  - Managed Sensors
    - [0/2] -> [1000/1] Rack 1 Humidity
    - [0/15] -> [1001/2] Switch blue
    - [0/16] -> [1000/4] Temperature
  - MANAGEMENT PORTS
    - PORT #1
      - THD-Sensor 13818030039
        - Sensors
          - [1000/1] Humidity
          - [1000/2] Front Door Rack 4
          - [1000/3] Digital Input 2
          - [1000/4] Temp Rack 4
      - THD Sensor 1381803024
        - Controls
        - Sensors
          - [1001/1] Humidity
          - [1001/2] Digital Input 1
          - [1001/3] Digital Input 2
          - [1001/6] Temperature
  - Modbus
    - Schroff RackChiller In-Row/8:1 (AV Contro
      - Controls
        - [2002/1] Cooler ON/OFF
        - [2002/2] Hot Aisle Temperature Setr

Sensors | THD-Sensor 13818030039 × | Managed Sensors ×

☐	RID	Resource Name	Number	Name
☐	0	Managed Sensors	2	Rack 1 Humidity
☐	0	Managed Sensors	15	Switch blue
☐	0	Managed Sensors	16	Temperature
☐	1000	THD-Sensor 13818030039	1	Humidity
☐	1000	THD-Sensor 13818030039	2	Front Door Rack 4
☐	1000	THD-Sensor 13818030039	3	Digital Input 2
☐	1000	THD-Sensor 13818030039	4	Temp Rack 4
☐	1001	1-wire Sensor 1381803024	1	Humidity
☐	1001	1-wire Sensor 1381803024	2	Digital Input 1
☐	1001	1-wire Sensor 1381803024	3	Digital Input 2
☐	1001	1-wire Sensor 1381803024	6	Temperature
☐	2002	Schroff RackChiller In-Row/8:1	1	Air Temp Out
☐	2002	Schroff RackChiller In-Row/8:1	2	Air Temp IN
☐	2002	Schroff RackChiller In-Row/8:1	3	Air Temp External Sensor
☐	2002	Schroff RackChiller In-Row/8:1	4	Requested Fan Speed %

Each sensor device is represented as a separate resource with a number in the range 1000-1999, each resource represents one device.

Each resource has the following properties:

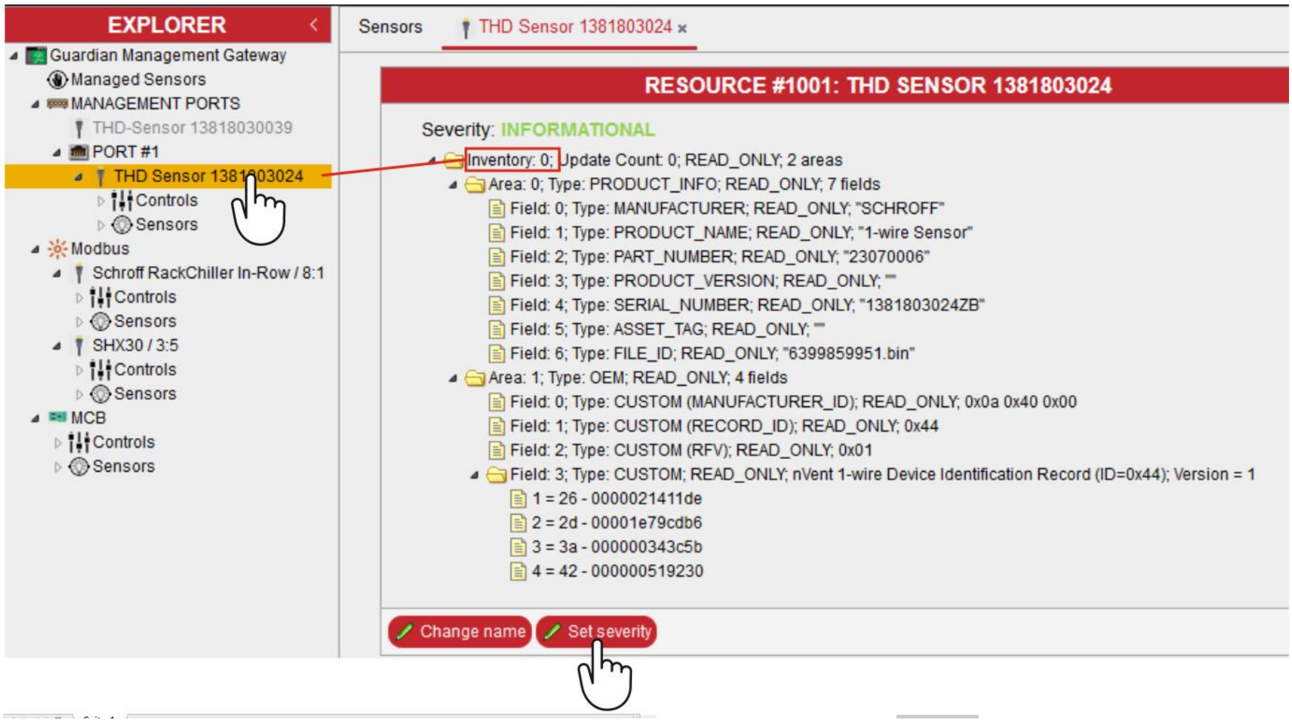
- Resource number, which uniquely identifies the resource in the system
  - The resource number is assigned when the device is first connected to the Guardian Management Gateway. Since each sensor device has a unique serial number, the resource number is associated with the device serial number at this point. When the device is extracted and reinstalled later, Guardian Management Gateway will try to keep the same resource number for it.
- Resource name (also called “resource tag”); this is a human-readable name of the resource that can be changed by the user

Each sensor device exposes the following sensors and controls:

- Sensor “Temperature” report the temperature measured on the device, in degrees C
- Sensor “Humidity” reports the humidity measured on the device, in percentage values
- Sensor “Digital Input 1” this discrete sensor reports the current state of the Digital Input 1 (ON or OFF)
- Sensor “Digital Input 2” this discrete sensor reports the current state of the Digital Input 2 (ON or OFF)
- Controls are a future option

ℹ	The digital inputs are pulled to “High”, so the default state is ON.

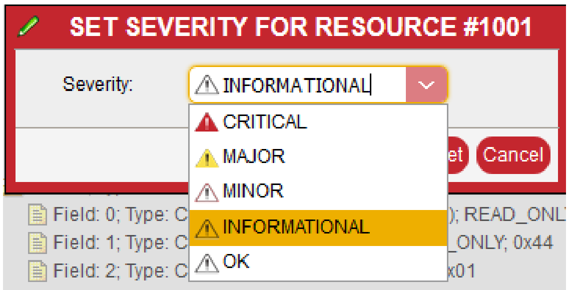
The inventory #0 is present on the resource representing the environmental sensor device. This inventory is in IPMI FRU information format and contains minimal information about the device, including its part number, serial number and manufacturer name, and the Device Identification record in the nVent OEM format. The inventory is read-only and stored in the device EEPROM.



```
Severity: INFORMATIONAL
Inventory: 0; Update Count: 0; READ_ONLY; 2 areas
  Area: 0; Type: PRODUCT_INFO; READ_ONLY; 7 fields
    Field: 0; Type: MANUFACTURER; READ_ONLY; "SCHROFF"
    Field: 1; Type: PRODUCT_NAME; READ_ONLY; "1-wire Sensor"
    Field: 2; Type: PART_NUMBER; READ_ONLY; "23070006"
    Field: 3; Type: PRODUCT_VERSION; READ_ONLY; ""
    Field: 4; Type: SERIAL_NUMBER; READ_ONLY; "1381803024ZB"
    Field: 5; Type: ASSET_TAG; READ_ONLY; ""
    Field: 6; Type: FILE_ID; READ_ONLY; "6399859951.bin"
  Area: 1; Type: OEM; READ_ONLY; 4 fields
    Field: 0; Type: CUSTOM (MANUFACTURER_ID); READ_ONLY; 0x0a 0x40 0x00
    Field: 1; Type: CUSTOM (RECORD_ID); READ_ONLY; 0x44
    Field: 2; Type: CUSTOM (RFV); READ_ONLY; 0x01
    Field: 3; Type: CUSTOM; READ_ONLY; nVent 1-wire Device Identification Record (ID=0x44); Version = 1
      1 = 26 - 0000021411de
      2 = 2d - 00001e79cdb6
      3 = 3a - 000000343c5b
      4 = 42 - 000000519230
```

Severity

When a resource (sensor device) is removed, an event or alarm is generated. The severity can be set by clicking on the Set severity button.



Severity: INFORMATIONAL

```
Field: 0; Type: C          ); READ_ONL
Field: 1; Type: C          _ONLY; 0x44
Field: 2; Type: C          x01
```

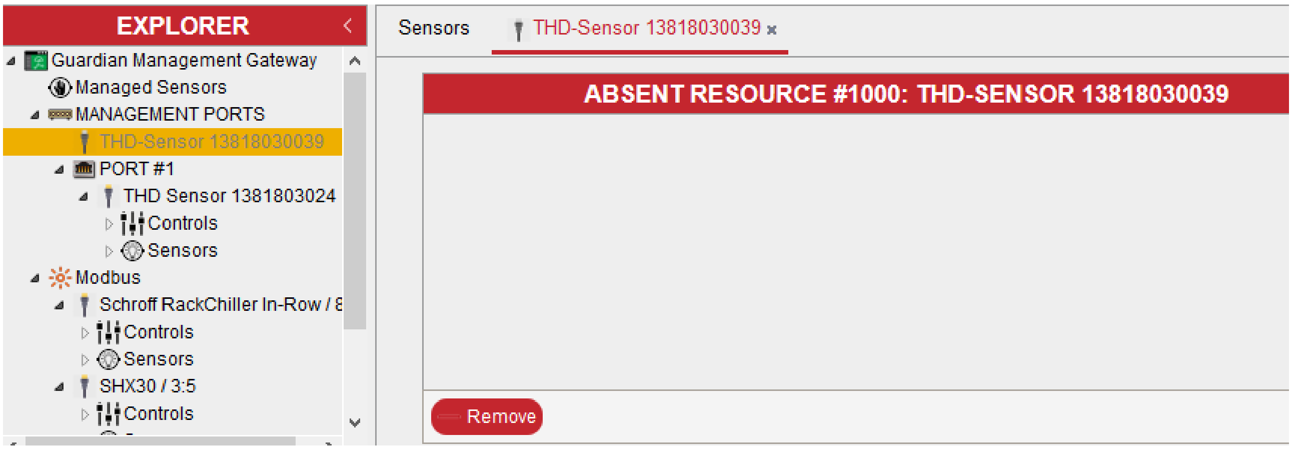
For more information, see HPI model: resources, sensors, controls

7.1.2 Remove sensor device permanently

Sensor devices which are absent (disconnected or broken) are grayed out in the tree pane.

When the sensor is reconnected, it will appear normal again.

If the sensor is to be permanently removed and thus the resource number released again, the sensor must be removed by pressing the "Remove" button.



7.2 Managing Modbus devices

The Guardian Management Gateway supports external Modbus devices communicating over Modbus TCP or the Modbus serial protocol.

Modbus devices communicating over Modbus serial protocol must be connected to the external interface (management) ports, sharing these ports with environmental sensor devices.

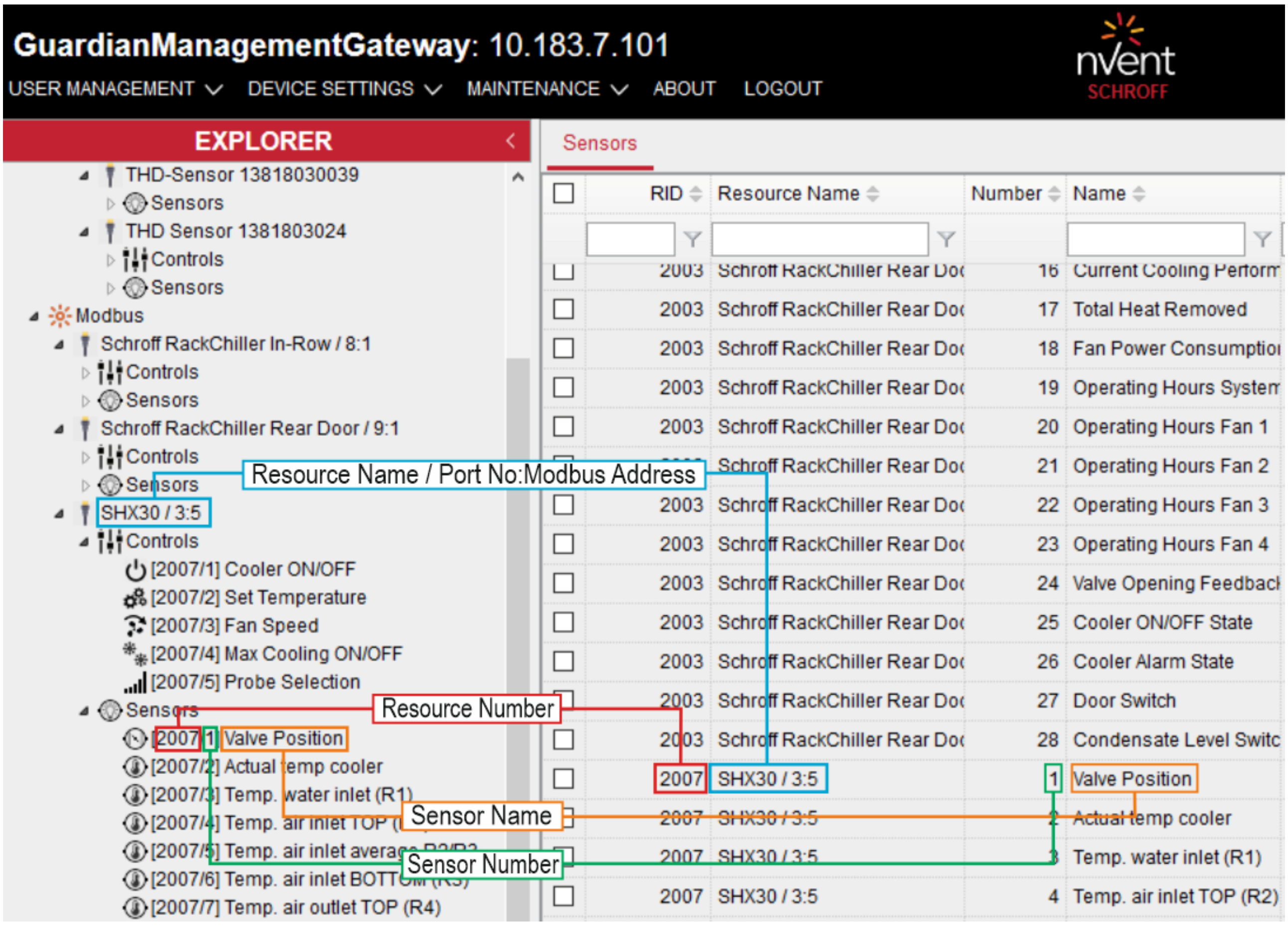
From the software perspective, currently Schroff Side Heat Exchangers (SHX-30 and compatibles), TT_SIM leak detection cable controllers are supported; new versions of firmware may add support for other devices.

Modbus devices connected over TCP are also supported. From the software perspective, currently of these devices only the RackChiller controllers are supported.

Modbus devices are represented by the HPI resources in the range 2000 - 2999, each resource represents one device.

Each device must have a unique combination of a Modbus address and the number of the interface (management port) to which they are connected (1, 2 or 3).

ℹ	**Port number = interface number!** **For TCP-connected devices the interface number is 8 and higher.**



- THD-Sensor 13818030039
  - Sensors
- THD Sensor 1381803024
  - Controls
  - Sensors
- Modbus
  - Schroff RackChiller In-Row / 8:1
    - Controls
    - Sensors
  - Schroff RackChiller Rear Door / 9:1
    - Controls
    - Sensors
  - SHX30 / 3:5
    - Controls
      - [2007/1] Cooler ON/OFF
      - [2007/2] Set Temperature
      - [2007/3] Fan Speed
      - [2007/4] Max Cooling ON/OFF
      - [2007/5] Probe Selection
    - Sensors
      - [2007/1] Valve Position
      - [2007/2] Actual temp cooler
      - [2007/3] Temp. water inlet (R1)
      - [2007/4] Temp. air inlet TOP (
      - [2007/5] Temp. air inlet average R2/R3
      - [2007/6] Temp. air inlet BOTTOM (R3)
      - [2007/7] Temp. air outlet TOP (R4)

☐	RID	Resource Name	Number	Name
☐	2003	Schroff RackChiller Rear Doo	16	Current Cooling Perform
☐	2003	Schroff RackChiller Rear Doo	17	Total Heat Removed
☐	2003	Schroff RackChiller Rear Doo	18	Fan Power Consumption
☐	2003	Schroff RackChiller Rear Doo	19	Operating Hours System
☐	2003	Schroff RackChiller Rear Doo	20	Operating Hours Fan 1
☐	2003	Schroff RackChiller Rear Doo	21	Operating Hours Fan 2
☐	2003	Schroff RackChiller Rear Doo	22	Operating Hours Fan 3
☐	2003	Schroff RackChiller Rear Doo	23	Operating Hours Fan 4
☐	2003	Schroff RackChiller Rear Doo	24	Valve Opening Feedback
☐	2003	Schroff RackChiller Rear Doo	25	Cooler ON/OFF State
☐	2003	Schroff RackChiller Rear Doo	26	Cooler Alarm State
☐	2003	Schroff RackChiller Rear Doo	27	Door Switch
☐	2003	Schroff RackChiller Rear Doo	28	Condensate Level Switc
☐	2007	SHX30 / 3:5	1	Valve Position
☐	2007	SHX30 / 3:5	2	Actual temp cooler
☐	2007	SHX30 / 3:5	3	Temp. water inlet (R1)
☐	2007	SHX30 / 3:5	4	Temp. air inlet TOP (R2)

Resource Name / Port No:Modbus Address

Resource Number

Sensor Name

Sensor Number

7.2.1 Connecting serial Modbus devices

Before connecting a serial Modbus device to one of the three interface (RJ45) ports labelled: "MANAGEMENT", adjust the serial port settings (baud rate, number of data and stop bits, parity, and the type of Modbus protocol: ASCII or binary).

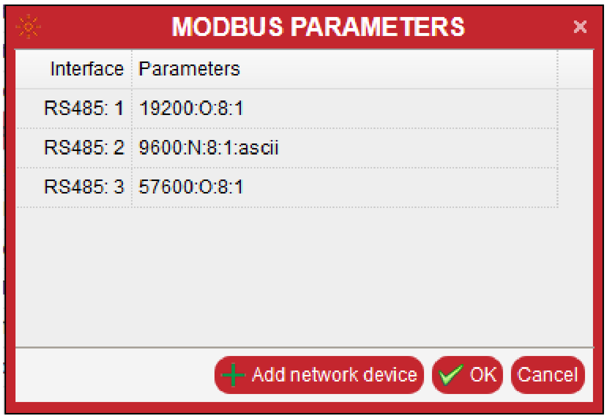
For example, Schroff Side Heat Exchangers (SHX30) use the speed of 19200 – 57600 baud, odd parity, 8 data bits, 1 stop bit and binary Modbus protocol.

TT_SIM leak detection cable controllers use 9600 baud, no parity, 8 data bits, 1 stop bit and ASCII Modbus protocol. To accommodate different Modbus devices, Guardian Management Gateway supports its own set of settings for each external interface ports.

Devices with different requirements to the serial port settings should be connected to different interface ports.

7.2.2 Configure Port Settings

To manage Modbus serial settings in the Web interface, invoke the dialog "**MODBUS PARAMETERS**" via the menu command "**MAINTENANCE**" -> "**Configure Modbus Parameters**". The dialog allows the user to edit serial settings (as strings) for all supported external interfaces. After changing the settings, press the OK button to apply the changes.


MODBUS PARAMETERS

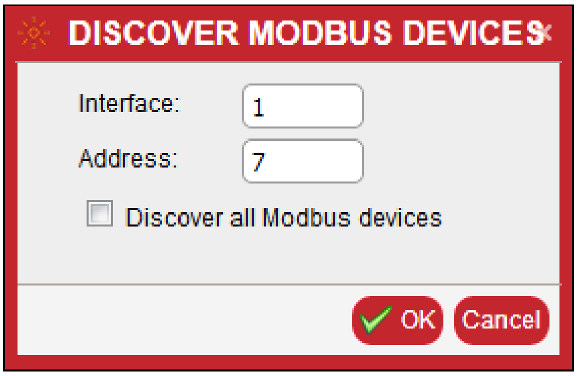
Interface	Parameters
RS485: 1	19200:O:8:1
RS485: 2	9600:N:8:1:ascii
RS485: 3	57600:O:8:1

7.2.3 Discovering serial Modbus devices

Modbus devices are semi hot-pluggable: hot extraction is recognized automatically, but to recognize hot inserted devices, a special discovery process should be run (this is because the discovery of new Modbus devices can be quite slow and resource-consuming).

To discover Modbus devices in the Web interface, invoke the command:
"**MAINTENANCE**" -> "**DISCOVER MODBUS DEVICES**".
A dialog appears where the user can specify the interface number and address for directed discovery of a specific device.
If the target address is not known, press the checkbox "**Discover all Modbus devices**" to discover all devices.
If the address of the new Modbus device is known, it is recommended to perform the "directed discovery" which is much faster.

Press the "OK" button to perform the discovery (directed or generic).



DISCOVER MODBUS DEVICES

Interface: 1

Address: 7

□ Discover all Modbus devices

7.2.4 Connecting TCP-connected Modbus devices

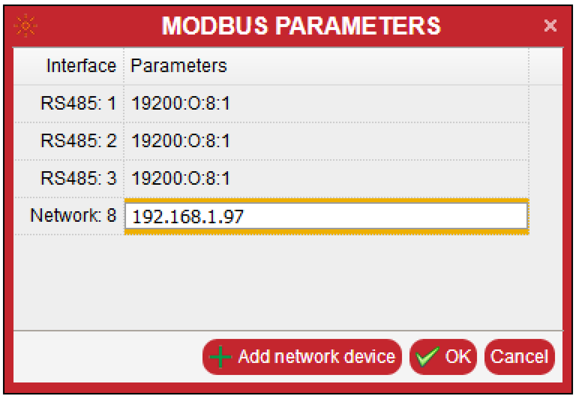
Before You can configure your interface settings for a Modbus TCP device, be sure that the Modbus device is already connected to your network, otherwise the configuration failed!

To establish a TCP connection to the target Modbus device via the Web interface, invoke the dialog:

"**MODBUS PARAMETERS**" via the menu command "**MAINTENANCE**" -> "**Configure Modbus Parameters**".

The dialog allows the user to enter or edit the IP address for a Modbus TCP interface (The virtual interface number for TCP-connected devices is 8 or higher).

After setting or changing the IP address, press the OK button to apply the changes; a failure will be reported if a TCP connection to the target address cannot be established.



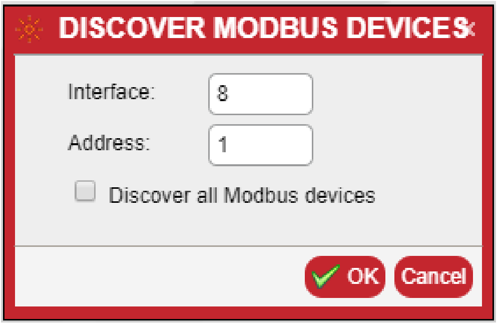
MODBUS PARAMETERS

Interface	Parameters
RS485: 1	19200:O:8:1
RS485: 2	19200:O:8:1
RS485: 3	19200:O:8:1
Network: 8	192.168.1.97

7.2.5 Discovering TCP-connected Modbus devices

To discover TCP-connected Modbus devices in the Web interface, invoke the command:
“**MAINTENANCE**” -> “**DISCOVER MODBUS DEVICES**”.

A dialog appears where the user can specify the interface number and address for directed discovery of a specific device. Specify the previously selected TCP interface number and the device address*, then press the “OK” button to perform the directed discovery of the TCP connected device.



DISCOVER MODBUS DEVICES

Interface: 8

Address: 1

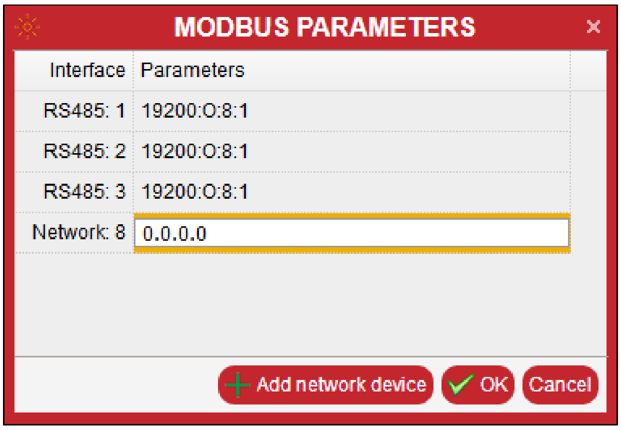
□ Discover all Modbus devices

***The address can be individually assigned by the user in the range 1 to 255.**
This address is not the IP-address!

To reset a TCP connection to the target Modbus device via the Web interface, invoke the dialog:

“**MODBUS PARAMETERS**” via the menu command “**MAINTENANCE**” -> “**Configure Modbus Parameters**”.

To add a new network interface press the “Add network interface” button. To disable the supported network interface, assign “0.0.0.0” as its IP address.

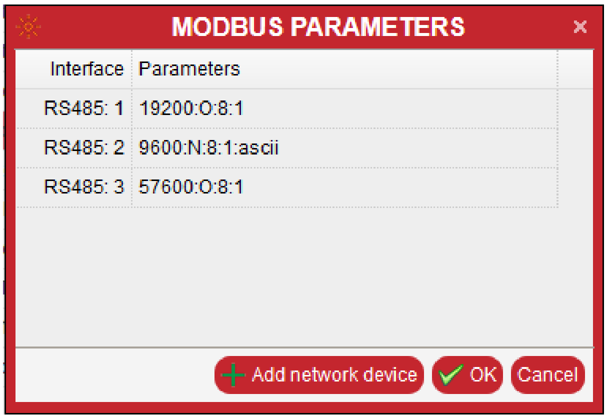


MODBUS PARAMETERS

Interface	Parameters
RS485: 1	19200:O:8:1
RS485: 2	19200:O:8:1
RS485: 3	19200:O:8:1
Network: 8	0.0.0.0

7.2.6 Managing Schroff Side Heat Exchangers SHX-30

Configure/check serial port setting via the menu command "**MAINTENANCE**" -> "**Configure Modbus Parameters**". In this example port (interface) 1 and 3 are configured for the SHX30.

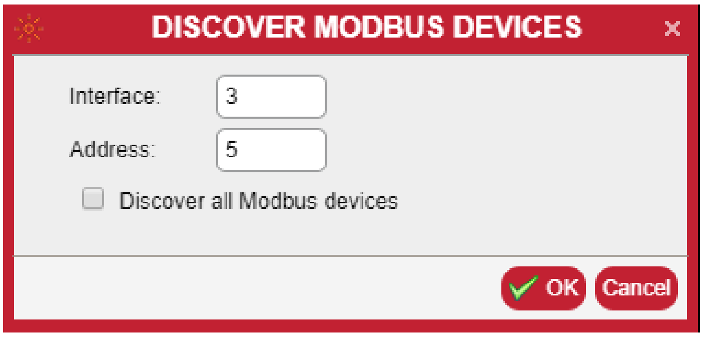


MODBUS PARAMETERS

Interface	Parameters
RS485: 1	19200:O:8:1
RS485: 2	9600:N:8:1:ascii
RS485: 3	57600:O:8:1

Schroff Side Heat Exchangers (SHX30) use the speed of 19200 – 57600 baud, odd parity, 8 data bits, 1 stop bit and binary Modbus protocol.

Connect the SHX30 to a management port (assuming it is connected to port 3 and the Modbus address is 5) and invoke the command: "**MAINTENANCE**" -> "**DISCOVER MODBUS DEVICES**".



DISCOVER MODBUS DEVICES

Interface: 3

Address: 5

□ Discover all Modbus devices

After entering the port and Modbus address, click OK. It discovers the Modbus present at that location.

For Schroff Side Heat Exchangers (SHX-30), the following sensors and controls are available:

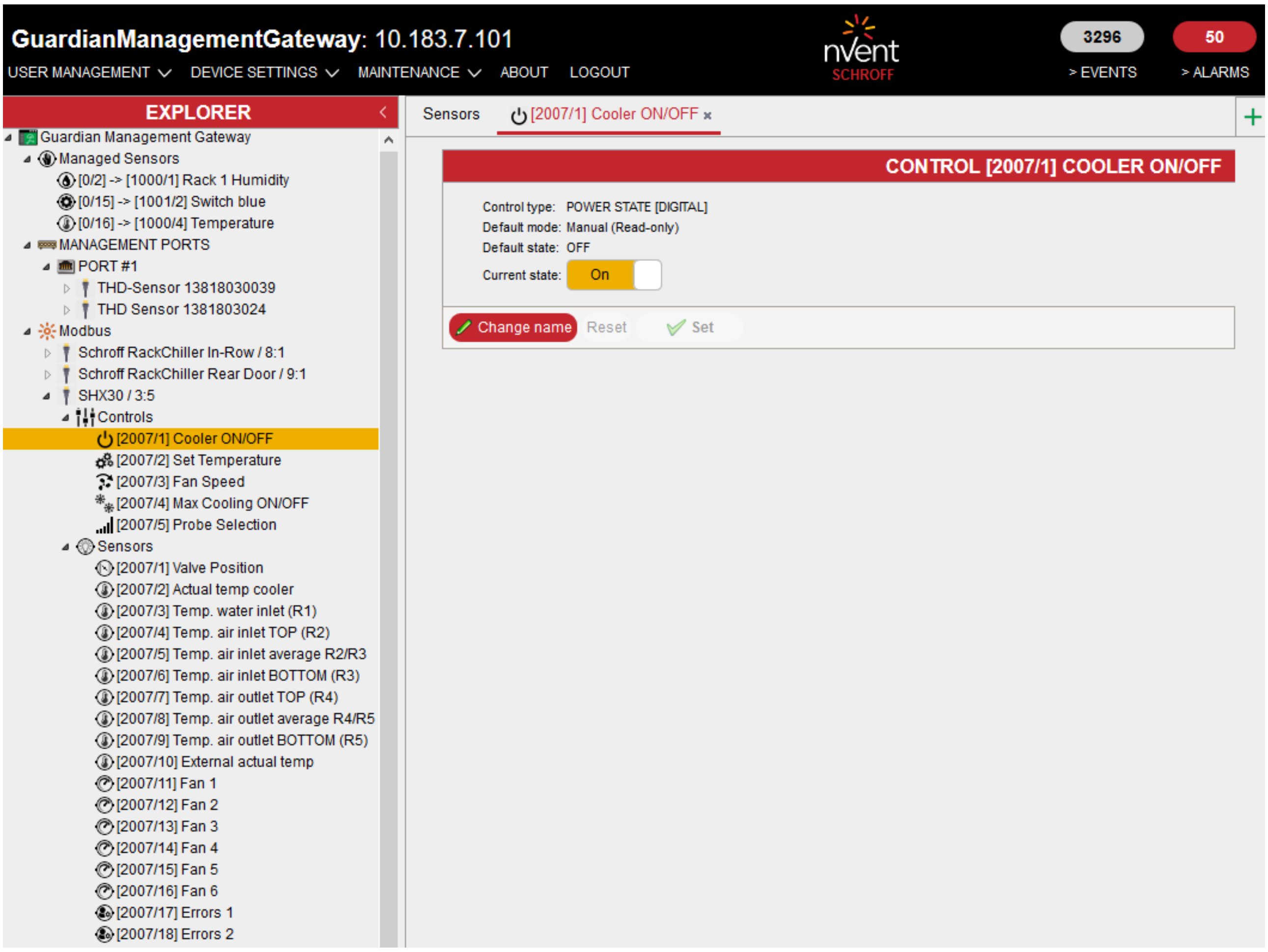
Controls:

1	**Cooler ON/OFF**	This digital control can be used to turn the cooler on or off
2	**Set Temperature**	This control is numeric, it sets the temperature set point for the cooler in the range between 18 and 40 degrees C
3	**Fan Speed**	This control is numeric, it specifies the desired fan speed in percentages between 30% and 100%
4	**Max Cooling ON/OFF**	A digital control that allows the user to turn on or off maximum cooling mode

Sensors:

1	**Valve Position**	Reports the current valve position, in percentages of fully open state (0 to 100)
2	**Actual temp cooler**	Reports the actual average outlet temperature
3	**Temp. water inlet (R1)**	Reports the cooling water inlet temperature
4	**Temp. air inlet TOP (R2)**	Reports the upper air inlet temperature
5	**Temp. air inlet average R2/R3**	Reports the average air inlet temperature
6	**Temp. air inlet BOTTOM (R3)**	Reports the bottom air inlet temperature
7	**Temp. air outlet TOP (R4)**	Reports the upper air outlet temperature
8	**Temp. air outlet average R4/R5**	Reports the average air inlet temperature
9	**Temp. air outlet BOTTOM (R5)**	Reports the bottom air inlet temperature
10	**External actual temp**	Reports the temperature of an external temp. sensor
11	**Fan 1**	Reports the speed of Fan 1 in revs/min
12	**Fan 2**	Reports the speed of Fan 2 in revs/min
13	**Fan 3**	Reports the speed of Fan 3 in revs/min
14	**Fan 4**	Reports the speed of Fan 4 in revs/min
15	**Fan 5**	Reports the speed of Fan 5 in revs/min
16	**Fan 6**	Reports the speed of Fan 6 in revs/min
17	**Errors 1**	Discrete sensors that report various errors detected by the heat exchanger in their state masks; multiple bits may be simultaneously set in their state masks
18	**Errors2**	Discrete sensors that report various errors detected by the heat exchanger in their state masks; multiple bits may be simultaneously set in their state masks

Tree pane in the Web interface:



GuardianManagementGateway: 10.183.7.101

USER MANAGEMENT ∨ DEVICE SETTINGS ∨ MAINTENANCE ∨ ABOUT LOGOUT

3296 > EVENTS 50 > ALARMS

EXPLORER

- Guardian Management Gateway
  - Managed Sensors
    - [0/2] -> [1000/1] Rack 1 Humidity
    - [0/15] -> [1001/2] Switch blue
    - [0/16] -> [1000/4] Temperature
  - MANAGEMENT PORTS
    - PORT #1
      - THD-Sensor 13818030039
      - THD Sensor 1381803024
  - Modbus
    - Schroff RackChiller In-Row / 8:1
    - Schroff RackChiller Rear Door / 9:1
    - SHX30 / 3:5
      - Controls
        - [2007/1] Cooler ON/OFF
        - [2007/2] Set Temperature
        - [2007/3] Fan Speed
        - [2007/4] Max Cooling ON/OFF
        - [2007/5] Probe Selection
      - Sensors
        - [2007/1] Valve Position
        - [2007/2] Actual temp cooler
        - [2007/3] Temp. water inlet (R1)
        - [2007/4] Temp. air inlet TOP (R2)
        - [2007/5] Temp. air inlet average R2/R3
        - [2007/6] Temp. air inlet BOTTOM (R3)
        - [2007/7] Temp. air outlet TOP (R4)
        - [2007/8] Temp. air outlet average R4/R5
        - [2007/9] Temp. air outlet BOTTOM (R5)
        - [2007/10] External actual temp
        - [2007/11] Fan 1
        - [2007/12] Fan 2
        - [2007/13] Fan 3
        - [2007/14] Fan 4
        - [2007/15] Fan 5
        - [2007/16] Fan 6
        - [2007/17] Errors 1
        - [2007/18] Errors 2

Sensors [2007/1] Cooler ON/OFF ×

CONTROL [2007/1] COOLER ON/OFF

Control type: POWER STATE [DIGITAL]
Default mode: Manual (Read-only)
Default state: OFF
Current state: On

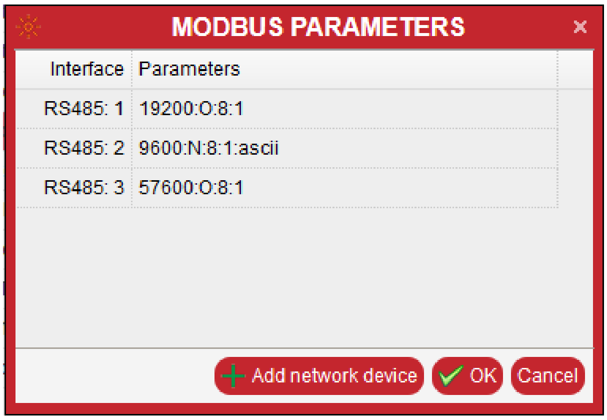
To retrieve the sensor or control data, click on the sensor or control for which you want to know the data.

Inventory is not available for Schroff Side Heat Exchangers (SHX).

7.2.7 Managing TT_SIM Leak detection cable controllers

Connect the device to a management port (assuming it is connected to Interface (Management Port) **2** and the Modbus address is 200).

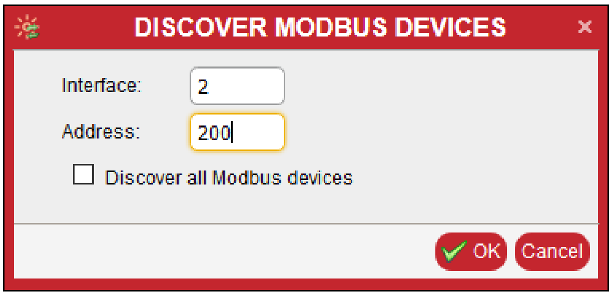
Assign the port (interface) settings to port 2:



MODBUS PARAMETERS

Interface	Parameters
RS485: 1	19200:O:8:1
RS485: 2	9600:N:8:1:ascii
RS485: 3	57600:O:8:1

Discover the Modbus device:



DISCOVER MODBUS DEVICES

Interface: 2

Address: 200

☐ Discover all Modbus devices

For TT_SIM Leak detection cable controllers, the following sensors are available:

1	**Leak**	A discrete sensor that reports whether a leak has been detected
2	**Contamination**	A discrete sensor that reports whether cable contamination has been detected
3	**Leak Location**	Reports the leak location, in meters
4	**Cable Break**	A discrete sensor that reports whether the cable has been physically broken
5	**Fault**	A discrete sensor that reports whether any other fault has been detected
6	**Circuit Length**	Reports the total cable length, in meters
7	**Detection Current**	Reports the current in the cable, in milliamperes
8	**Status**	Reports the current contents of the controller status word, as an opaque numeric value

To retrieve sensor values for the TT_SIM Leak detection cable controllers, click on the resp. sensor in the tree pane of the Web interface

Inventory and controls are not available for TT_SIM Leak detection cable controllers.

ℹ	**The default Modbus address of TT_SIM Leak detection cable controller is either 199 or 200.**

7.2.8 Managing Schroff RackChiller devices

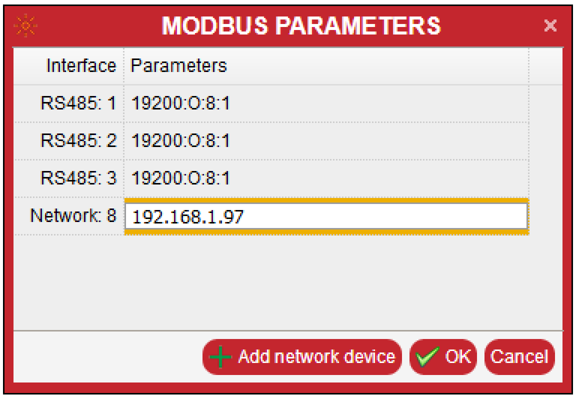
Schroff RackChiller devices can be accessed only via TCP.

Connect the device to your network (assuming the IP address is `192.168.1.97`.

To establish a TCP connection to the target Modbus device via the Web interface, invoke the dialog:

"**MODBUS PARAMETERS**" via the menu command "**MAINTENANCE**" -> "**Configure Modbus Parameters**".

After setting or changing the IP address, press the OK button to apply the changes; a failure will be reported if a TCP connection to the target address cannot be established.



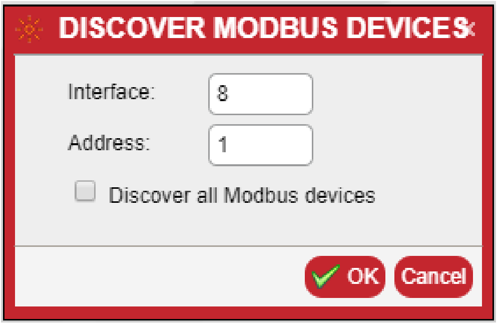
MODBUS PARAMETERS

Interface	Parameters
RS485: 1	19200:O:8:1
RS485: 2	19200:O:8:1
RS485: 3	19200:O:8:1
Network: 8	192.168.1.97

The following command succeeds only if the connection is successfully established, that means that the Modbus device is already connected to your network!

Discover the Modbus device:
To discover TCP-connected Modbus devices in the Web interface, invoke the command:
"**MAINTENANCE**" -> "**DISCOVER MODBUS DEVICES**".

Specify the interface number and address*, then press the "OK" button to perform the directed.



DISCOVER MODBUS DEVICES

Interface: 8

Address: 1

□ Discover all Modbus devices

***The address can be individually assigned by the user in the range 1 to 255.**
This address is not the IP-address!

RackChiller Rear Door:

For Schroff RackChiller Rear Door devices, the following sensors and controls are available (NOTE: this list may be expanded in the future):

Controls:

1	**Cooler ON/OFF**	This digital control can be used to turn the cooler on or off
2	**Max Cooling ON/OFF**	Allows the user to turn on or off maximum cooling mode
3	**Temperature Control Variable**	Set the control variable for the temperature regulation. The following parameters are available: (0) Manual (Opening ration water valve in %) (1) Outlet Temp Air Top (2) Outlet Temp Air Bottom (3) Average Outlet Air Temp (default) (4) Temp Water Outlet (5) dT Water Inlet/Outlet
4	**Temperature Setpoint**	Setpoint for the temperature. If the control is made by a temperature sensor, the temperature can be set in ° C (° F), with manual control, the opening ratio of the water valve can be set manually.
5	**Manual Water Valve Position**	When the Temperature Control Variable is set to “Manual”, the opening ratio of the water valve can be set manually in %.
6	**Fan Speed Control Mode**	Control variable for the fan speed. The following parameters are available: (0) Pressure Difference (1) Manual (%)
7	**Pressure Differential Setpoint**	Setpoint for controlling the fan speed. If the control is via the differential pressure sensor, the pressure can be set in the range of -150 Pa to +150 Pa. Negative differential pressure means that the pressure in the cabinet is higher than the ambient pressure. A setting of approx. +20 Pa is recommended.
8	**Manual Fan Level**	When the Fan Speed Control Variable is set to “Manual”, the fan speed can be set from 20 - 100 %.

Sensors:

1	**Air Temp Out Top**	Reading of the upper temperature sensor at the air outlet (Cold air)
2	**Air Temp Out Bottom**	Reading of the lower temperature sensor at the air outlet (Cold air)
3	**Air Temp In Top**	Reading of the upper temperature sensor located in front of the RackChiller (Warm air)
4	**Air Temp In Bottom**	Reading of the lower temperature sensor located in front of the RackChiller (Warm air)
5	**Air Differential Pressure**	Differential Pressure Inside/Outside cabinet
6	**Water Temp In**	Temperature Water inlet
7	**Water Temp Out**	Temperature Water outlet
8	**Water Flow**	Water flow
9	**Water Pressure**	Water pressure
10	**Requested Valve Opening**	Requested opening ratio of the water valve in %
11	**Requested Fan Speed**	Reports the requested fan speed by the controller
12	**Speed Fan 1**	Reports the actual speed of fan 1 (upper fan)
13	**Speed Fan 2**	Reports the actual speed of fan 2
14	**Speed Fan 3**	Reports the actual speed of fan 3
15	**Speed Fan 4**	Reports the actual speed of fan 4 (lower fan)
16	**Current Cooling Performance**	Reports the Current Cooling Performance
17	**Total Heat Removed**	Reports the Total Heat Removed in kW/h
18	**Fan Power Consumption**	Reports the actual power consumption of the fans
29	**Operating Hours System**	Reports the operating hours of the RackChiller
20	**Operating Hours Fan 1**	Reports the operating hours of fan 1
21	**Operating Hours Fan 2**	Reports the operating hours of fan 2
22	**Operating Hours Fan 3**	Reports the operating hours of fan 3
23	**Operating Hours Fan 4**	Reports the operating hours of fan 4
24	**Valve Opening feedback**	Reports the actual valve opening ratio
25	**Cooler ON/OFF State**	Reports the Cooler ON/OFF State
26	**Cooler Alarm State**	Reports the cooler alarm state
27	**Door Switch**	Reports the state of an optional door switch
28	**Condensate Level Switch**	Reports the state of an optional switch for the water level in the condensate tray

RackChiller In-Row:

For Schroff RackChiller In-Row devices, the following sensors and controls are available (NOTE: this list may be expanded in the future):

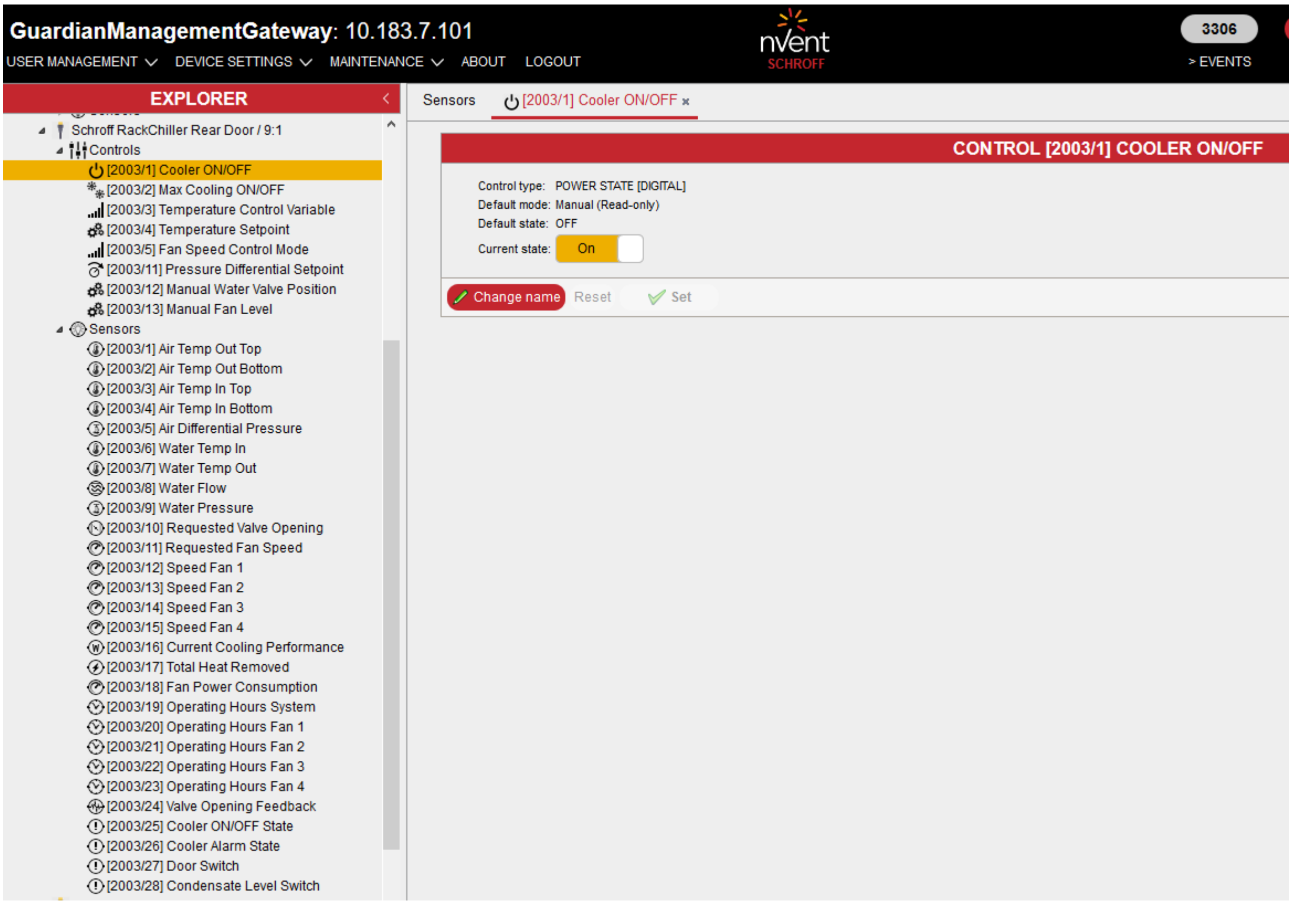
Controls:

1	**Cooler ON/OFF**	This digital control can be used to turn the cooler on or off
2	**Hot Aisle Temperature Setpoint**	
3	**Hot Aisle Temperature Differential**	
4	**Cold Aisle Temperature Differential**	

Sensors:

1	**Air Temp Out**	Reading of the temperature sensor at the air outlet (Cold air)
2	**Air Temp In**	Reading of the temperature sensor at the air inlet (Warm air)
3	**Air Temp ext. Sensor**	Reading of the an external temperature sensor
4	**Requested Fan Speed**	Reports the requested fan speed by the controller
5	**Requested Valve Opening**	Requested opening ratio of the water valve in %
6	**Valve Opening feedback**	Reports the actual valve opening ratio
7	**Fan Speed 1 %**	Reports the actual speed of fan 1
8	**Fan Speed 2 %**	Reports the actual speed of fan 2
9	**Fan Speed 3 %**	Reports the actual speed of fan 3
10	**Fan Speed 4 %**	Reports the actual speed of fan 4
11	**Fan Speed 5 %**	Reports the actual speed of fan 5
12	**Fan Speed 6 %**	Reports the actual speed of fan 6
13	**Cooler ON/OFF State**	Reports the Cooler ON/OFF State
14	**Cooler Alarm State**	Reports the cooler alarm state

To retrieve control or sensor values for the Rack Chiller, click on the resp. control or sensor in the tree pane of the Web interface.



- Schroff RackChiller Rear Door / 9:1
  - Controls
    - [2003/1] Cooler ON/OFF
    - [2003/2] Max Cooling ON/OFF
    - [2003/3] Temperature Control Variable
    - [2003/4] Temperature Setpoint
    - [2003/5] Fan Speed Control Mode
    - [2003/11] Pressure Differential Setpoint
    - [2003/12] Manual Water Valve Position
    - [2003/13] Manual Fan Level
  - Sensors
    - [2003/1] Air Temp Out Top
    - [2003/2] Air Temp Out Bottom
    - [2003/3] Air Temp In Top
    - [2003/4] Air Temp In Bottom
    - [2003/5] Air Differential Pressure
    - [2003/6] Water Temp In
    - [2003/7] Water Temp Out
    - [2003/8] Water Flow
    - [2003/9] Water Pressure
    - [2003/10] Requested Valve Opening
    - [2003/11] Requested Fan Speed
    - [2003/12] Speed Fan 1
    - [2003/13] Speed Fan 2
    - [2003/14] Speed Fan 3
    - [2003/15] Speed Fan 4
    - [2003/16] Current Cooling Performance
    - [2003/17] Total Heat Removed
    - [2003/18] Fan Power Consumption
    - [2003/19] Operating Hours System
    - [2003/20] Operating Hours Fan 1
    - [2003/21] Operating Hours Fan 2
    - [2003/22] Operating Hours Fan 3
    - [2003/23] Operating Hours Fan 4
    - [2003/24] Valve Opening Feedback
    - [2003/25] Cooler ON/OFF State
    - [2003/26] Cooler Alarm State
    - [2003/27] Door Switch
    - [2003/28] Condensate Level Switch

CONTROL [2003/1] COOLER ON/OFF

Control type: POWER STATE [DIGITAL]
Default mode: Manual (Read-only)
Default state: OFF
Current state: On

7.3 Reachability

For user convenience, Guardian Management Gateway provides a facility to detect whether a certain system (server) is reachable over the network. It does this by periodically pinging the given address and storing the results in a special table. When a registered system becomes reachable (ping becomes successful) or becomes unreachable (ping becomes unsuccessful), Guardian Management Gateway changes the state of the target system in the table and generates a corresponding event (“Server reachable” or “Server unreachable”). Another pair of events are generated when systems are added to the reachability verification list (“Server Monitoring Starts”) or deleted from the list (“Server Monitoring Stops”). These events are sent as HPI software events, are subject to event filtering and are placed into the System Event Log.

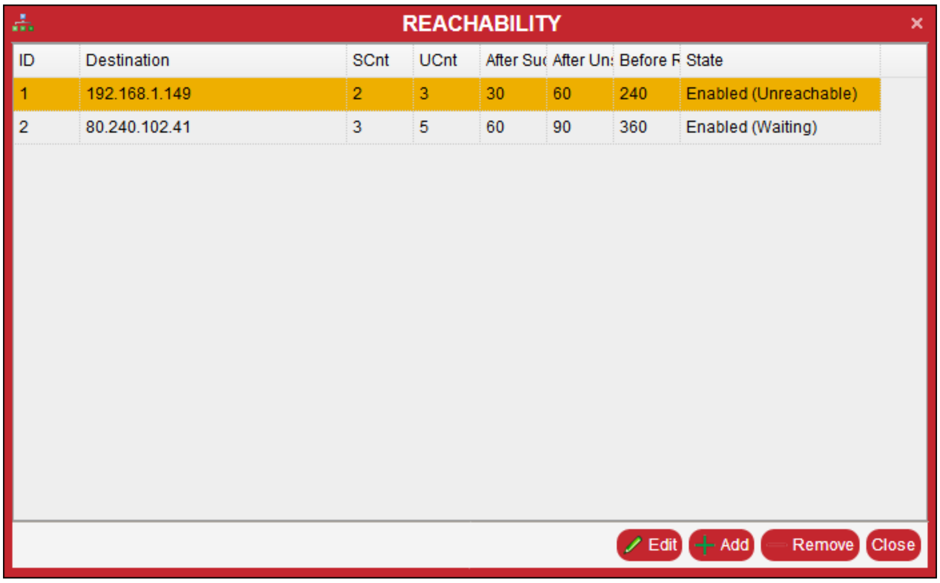
The following actions are available for user in connection with the Reachability feature:

- Add an IP address or the name of the system to the reachability verification list, and specify ping parameters
- Update ping parameters for the specified position in the list
- Enable/disable pinging for the previously specified system, by its position in the list
- Get the current reachability verification list, with system names or IP addresses, their status and ping parameters.

The following ping parameters can be specified for a certain system:

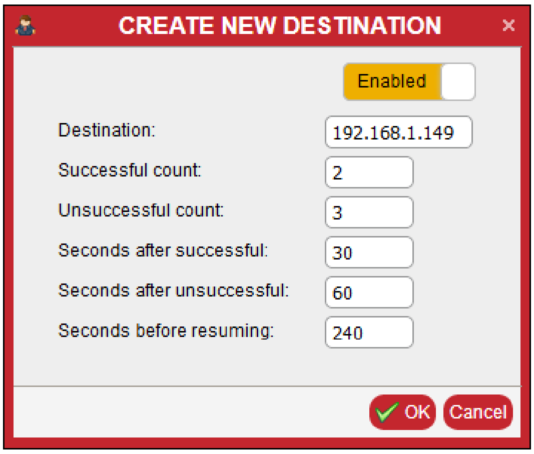
- Success count: after how many successful pings the system should be considered reachable
- Unsuccessful count: after how many unsuccessful pings the system should be considered unreachable
- Seconds after successful: a delay in seconds between a successful ping and the next ping
- Seconds after unsuccessful: a delay in seconds between an unsuccessful ping and the next ping (unless the target has been considered unreachable after this unsuccessful ping)
- Seconds before resuming: a delay in seconds to resume pinging after that target has been considered unreachable.
- Whether to enable reachability test for this system (true/false).
-

To manage the reachability verification list from the Web interface, use the Reachability dialog, invoked via the menu commands “**DEVICE SETTINGS**” -> “**Reachability**”.



ID	Destination	SCnt	UCnt	After Su(	After Un:	Before F	State
1	192.168.1.149	2	3	30	60	240	Enabled (Unreachable)
2	80.240.102.41	3	5	60	90	360	Enabled (Waiting)

To add a new entry in the reachability verification list press the button “Add”.



CREATE NEW DESTINATION

Enabled

Destination: 192.168.1.149

Successful count: 2

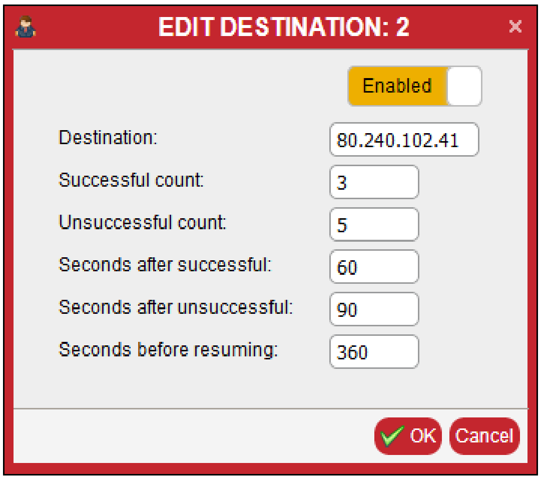
Unsuccessful count: 3

Seconds after successful: 30

Seconds after unsuccessful: 60

Seconds before resuming: 240

To edit an existing entry move the cursor over the entry and press the button “Edit”.



EDIT DESTINATION: 2

Enabled

Destination: 80.240.102.41

Successful count: 3

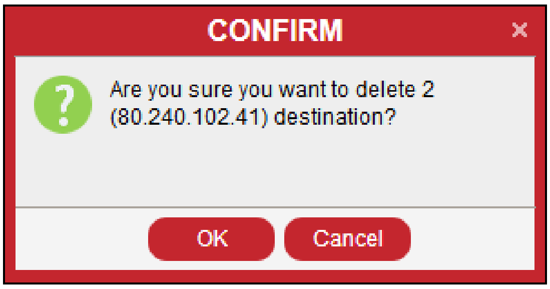
Unsuccessful count: 5

Seconds after successful: 60

Seconds after unsuccessful: 90

Seconds before resuming: 360

To delete an existing entry from the reachability verification list move the cursor over the entry and press the button “Remove”. The “Confirm” dialog window is generated.



CONFIRM

Are you sure you want to delete 2 (80.240.102.41) destination?

8 HPI model: resources, sensors, controls

The software architecture of the Guardian Management Gateway conforms to the Hardware Platform Interface (HPI) model by Service Availability Forum. This model is defined in the Hardware Platform Interface Specification.

Hardware Platform Interface provides an abstract model of underlying hardware, using abstract concepts of resources, sensors, controls and inventory.

A system comprises multiple resources, and the resource population is dynamic, that is, it may change over time.

Existing resources may be removed from the system and new resources may appear.

Each resource abstracts a hardware field-replaceable unit (FRU) and includes multiple sensors, controls and an optional inventory.

- Sensors abstract physical sensor devices
- Controls abstract physical control mechanisms (e.g. GPIO pins in the output mode)
- Inventory is a data storage that contains information about the resource in a standardized format.

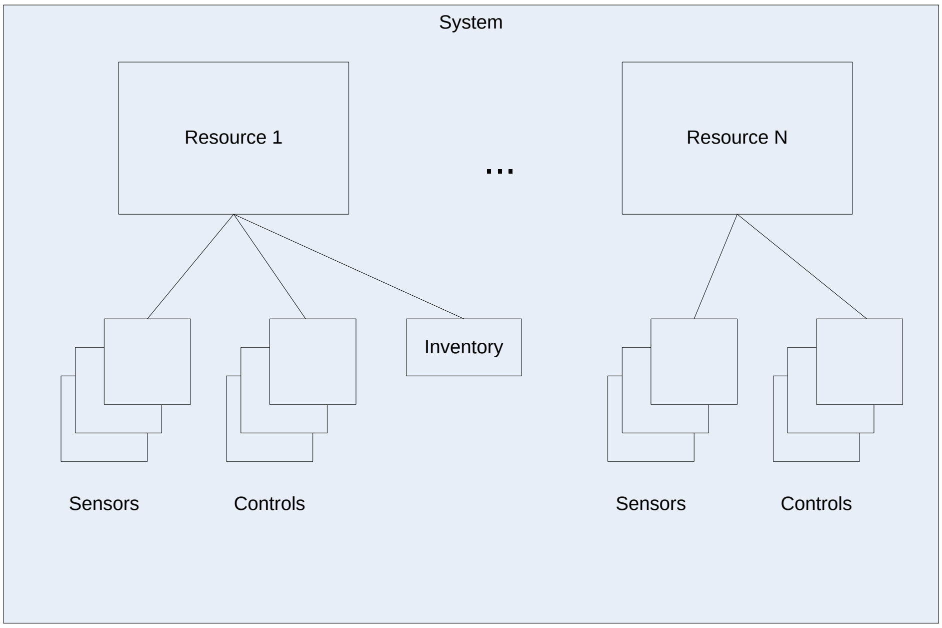
When something important happens in the system (e.g. a configuration change or an alert condition on a sensor) an event is generated.

Events are data packets in a standardized format, they are processed by the event filters (and can generated subsidiary actions, like sending an e-mail message or an SMS, or executing some predefined actions, or generating an SNMP trap).

All events are stored in the system event log where they can be examined in a later time.

8.1 Resources

In the Guardian Management Gateway architecture, resources normally represent hardware FRUs (though some of them may be hot-inserted and removed while some remain static).



Each resource has the following properties:

- Resource number, which uniquely identifies the resource in the system
- Resource name (also called "resource tag"); this is a human-readable name of the resource that can be changed by the user
- Capabilities; this is the mask of binary flags that identifies what capabilities the resource has. The most commonly used capabilities are:
  - Resource contains sensors
  - Resource contains controls
  - Resource hosts an inventory
- Resource entity path, which identifies the position of the resource in the hierarchy of entities in the system (in the machine-readable form).
- Resource severity, that identifies the severity of an event generated when this resource is removed.

The resource numbers are fixed and assigned as follows:

Resource 0 ("Managed Sensors"): this resource is virtual. It holds managed sensors and the inventory for the whole Guardian Management Gateway.

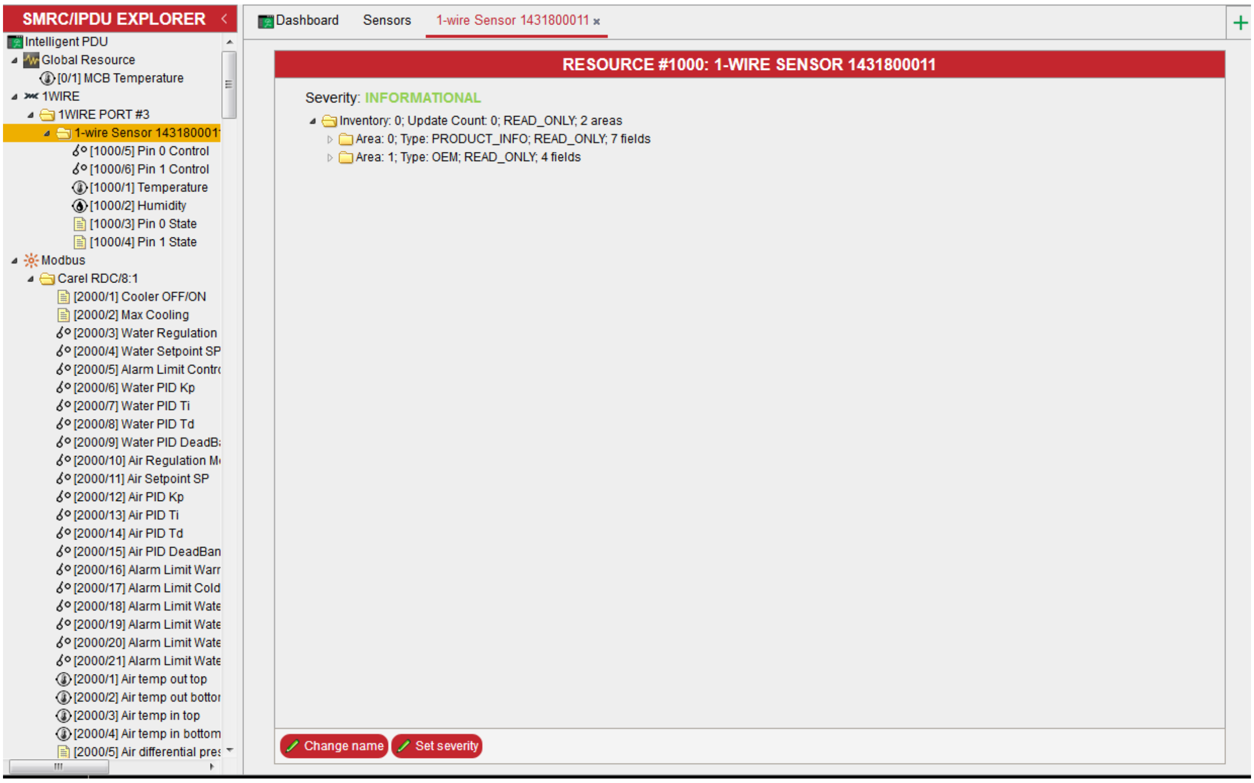
Resources 1000 - 1999: these resources represent Schroff environmental sensor devices. These devices are hot-swappable and carry several sensors and controls on them. Each sensor device holds an inventory that contains the serial number of the device.

Resources 2000 - 2999: these resources represent Modbus devices.

Currently, the Schroff SHX-30 cooling units, the Schroff RackChillers and the TraceTek TTSIM-1A leak detection alarm unit are supported as Modbus devices. These devices are hot-swappable, but in the case of hot insertion, a manual discovery of a new device should be initiated by the user.

Resource 3000: the master control board (MCB). This is the board which hosts the single-board computer on which the management software is run.

In the Web interfaces, resources are visible in the left (tree) pane of the screen as tree nodes.



Intelligent PDU
Global Resource
[0/1] MCB Temperature
1WIRE
1WIRE PORT #3
1-wire Sensor 143180001
[1000/5] Pin 0 Control
[1000/6] Pin 1 Control
[1000/1] Temperature
[1000/2] Humidity
[1000/3] Pin 0 State
[1000/4] Pin 1 State
Modbus
Carel RDC/8:1
[2000/1] Cooler OFF/ON
[2000/2] Max Cooling
[2000/3] Water Regulation
[2000/4] Water Setpoint SP
[2000/5] Alarm Limit Contr
[2000/6] Water PID Kp
[2000/7] Water PID Ti
[2000/8] Water PID Td
[2000/9] Water PID DeadB
[2000/10] Air Regulation M
[2000/11] Air Setpoint SP
[2000/12] Air PID Kp
[2000/13] Air PID Ti
[2000/14] Air PID Td
[2000/15] Air PID DeadBan
[2000/16] Alarm Limit Warr
[2000/17] Alarm Limit Cold
[2000/18] Alarm Limit Wate
[2000/19] Alarm Limit Wate
[2000/20] Alarm Limit Wate
[2000/21] Alarm Limit Wate
[2000/1] Air temp out top
[2000/2] Air temp out bottor
[2000/3] Air temp in top
[2000/4] Air temp in bottom
[2000/5] Air differential pres

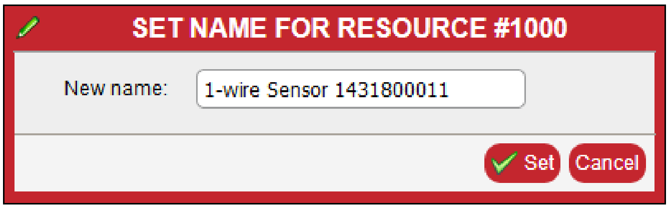
RESOURCE #1000: 1-WIRE SENSOR 1431800011

Severity: INFORMATIONAL

Inventory: 0; Update Count: 0; READ_ONLY; 2 areas
Area: 0; Type: PRODUCT_INFO; READ_ONLY; 7 fields
Area: 1; Type: OEM; READ_ONLY; 4 fields

8.1.1 Change a resource name

To change the resource name, select the resource in the tree pane. In the middle pane, the resource inventory will be shown. Press the button “Change name” at the bottom of the screen.
The dialog box “Set Name for Resource #NNN” appears:



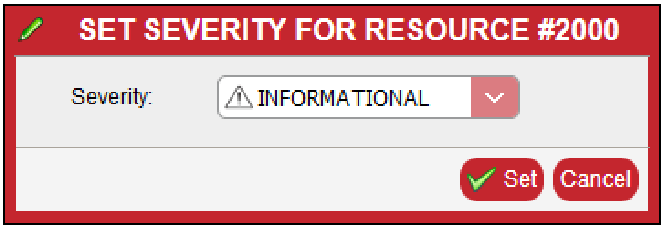
SET NAME FOR RESOURCE #1000

New name: 1-wire Sensor 1431800011

Change the current resource name and press the “Set” button.

8.1.2 Change the resource severity

Press the button “Set severity” at the bottom of the screen. The dialog box “Set Severity for Resource #NNN” appears:



SET SEVERITY FOR RESOURCE #2000

Severity: INFORMATIONAL

Change the current resource severity and press the “Set” button.

8.2 Sensors

Sensors in the HPI model represent devices that collect and report information about the environment and the state of the system itself. These devices can be physical or logical, and are considered components of FRUs (which are represented as resources).

ℹ	**Each sensor belongs to some resource.**

8.2.1 Numeric and discrete sensors

There are two classes of sensors: numeric and discrete.

Numeric sensors

Numeric sensors report a numeric reading and the state mask.

The reading can be signed integer, unsigned integer or a floating-point number.

In all cases the values are 64 bits in size.

Thresholds

Thresholds can be specified for a numeric sensor.

There are three upper thresholds:

- Upper Critical Threshold
- Upper Major Threshold
- Upper Minor Threshold

and three lower thresholds:

- Lower Critical Threshold
- Lower Major Threshold
- Lower Minor Threshold

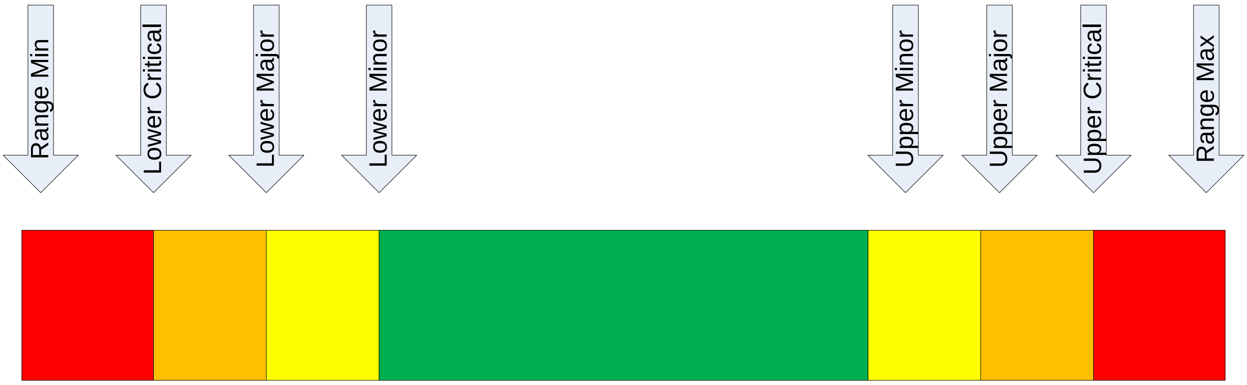
Not all thresholds need to be defined. Normally the sensor value should be between lower and upper thresholds. The state mask for numeric sensors reports which thresholds are crossed at a given moment of time. When a sensor value crosses a threshold, this is considered an abnormal situation, and an event may be generated (if the sensor configuration and event enable mask allows it).

Hysteresis

Hysteresis values can be specified to prevent generation of numerous events when the sensor reading oscillates in a vicinity of a certain threshold.

- Positive Hysteresis
- Negative Hysteresis

Usually the position of the thresholds corresponds to the figure below, and the sensor value is normally located between lower minor and upper minor thresholds.



A hysteresis is taken into account when the sensor value goes back into the normal range, crossing an upper threshold in the downside direction or lower threshold in the upside direction. For a de-assertion event, the sensor value should become less than Threshold – Positive Hysteresis in the first case, and greater than Threshold + Negative Hysteresis in the second case.

Discrete sensors

Discrete sensors do not report a numeric reading, they report only the state mask.

The state mask can comprise up to 16 states.

A sensor may be in several states simultaneously, but most often it is only in one single state at a given moment of time.

Discrete sensors may generate events when changing states.

An example of a discrete sensor can be:

- A presence sensor (some entity is present or absent)
- A failure sensor for a component (component operational / component failed) or
- A reboot reason sensor (with the set of states corresponding to different reasons of last reboot, e.g. “power up”, “hardware reset”, “software reboot”, “reset after upgrade”, etc.).

For a discrete sensor, event severity is assigned to each state and can be changed by a user.

This severity is propagated to the event that is generated when the sensor gets into this state, and allows to distinguish between “normal” and “abnormal” states for the sensor.

- A “normal” state should be assigned the severity “OK” or “Informational”
- An “abnormal” state should be assigned the severity “Minor”, “Major” or “Critical” depending on the severity of this abnormality.

For sensors with thresholds, event severity corresponds to the severity of the corresponding threshold and should not be changed by a user.

8.2.2 Sensor attributes and configuration parameters

Each sensor has static attributes and dynamic configuration parameters.

- Static attributes are defined when the sensor is created by the system and are read-only for a user.
- Dynamic configuration parameters can be changed by a user.

Static sensor attributes:

- Sensor number
- Sensor type (e.g. temperature, voltage, humidity, presence)
- Event category (threshold-based, state asserted/deasserted, reboot reason)
- Can sensor be dynamically enabled or disabled?
- Event control: can events be globally enabled/disabled, has the sensor per-state event control?
- Bit mask of supported states
- Data format: is sensor numeric or discrete?
- For a numeric sensor:
    - Type of the numeric reading (integer, unsigned, float)
    - Base units (meters, volts, amperes, etc.)
    - Modifier units (e,g, seconds)
    - Modifier use (multiply or divide)
    - Base units factor (e.g. kilo=10^3, mini=10^{-3}, etc)
    - Modifier units factor
    - Is sensor reading a percentage?
    - Range of valid sensor readings
    - Accuracy, resolution, tolerance for the sensor reading
    - Supported thresholds and hysteresis, as a bit mask
    - Writable thresholds and hysteresis, as a bit mask (those that can be changed by a user)
- For a discrete sensor:
    - Default assignment of severities to event states (identifies normal and abnormal states for the sensor)

Dynamic sensor attributes:

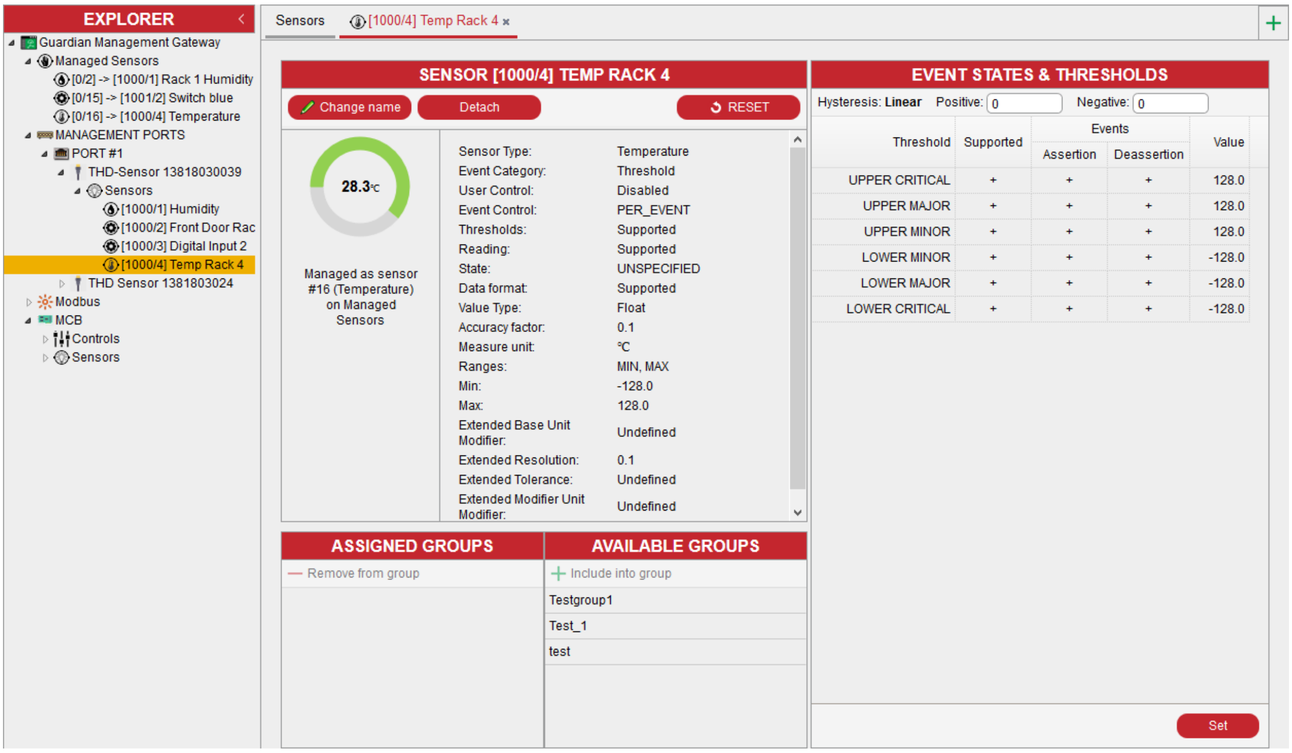
The user can change the following configuration parameters for a sensor:

- Enable or disable the sensor (if static attributes allow that)
- Enable or disable events for specific states and/or globally (if static attributes allow that)
- Change sensor human-readable name
- Change values of supported thresholds and hysteresis
- Change polling period for the sensor (in milliseconds)
- Change assertion delay count (for how long a threshold should be crossed to generate an event); this setting prevents spontaneous events in the case of random errors in sensor readings
- Change severities assigned to specific event states.

8.2.3 Managing sensors with the Web interface

Managing specific sensors

To manage a specific sensor, choose it in the left (tree) pane; instruments are shown under the resources that own them:



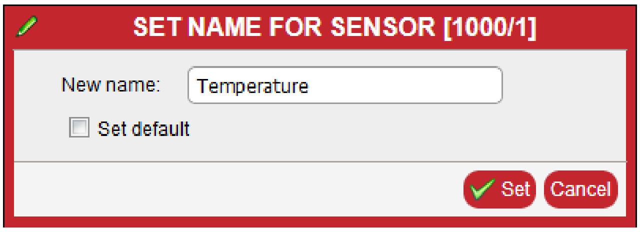
Sensor Type:	Temperature
Event Category:	Threshold
User Control:	Disabled
Event Control:	PER_EVENT
Thresholds:	Supported
Reading:	Supported
State:	UNSPECIFIED
Data format:	Supported
Value Type:	Float
Accuracy factor:	0.1
Measure unit:	°C
Ranges:	MIN, MAX
Min:	-128.0
Max:	128.0
Extended Base Unit Modifier:	Undefined
Extended Resolution:	0.1
Extended Tolerance:	Undefined
Extended Modifier Unit Modifier:	Undefined

Hysteresis: **Linear** Positive: 0 Negative: 0

Threshold	Supported	Events		Value
		Assertion	Deassertion	
UPPER CRITICAL	+	+	+	128.0
UPPER MAJOR	+	+	+	128.0
UPPER MINOR	+	+	+	128.0
LOWER MINOR	+	+	+	-128.0
LOWER MAJOR	+	+	+	-128.0
LOWER CRITICAL	+	+	+	-128.0

When a numeric sensor is selected in the left pane, the middle pane shows the sensor value and properties, and the right pane shows threshold and hysteresis values.

To change the name of the sensor, press the button “Change name”. The dialog for choosing the new name will open:



New name: Temperature

□ Set default

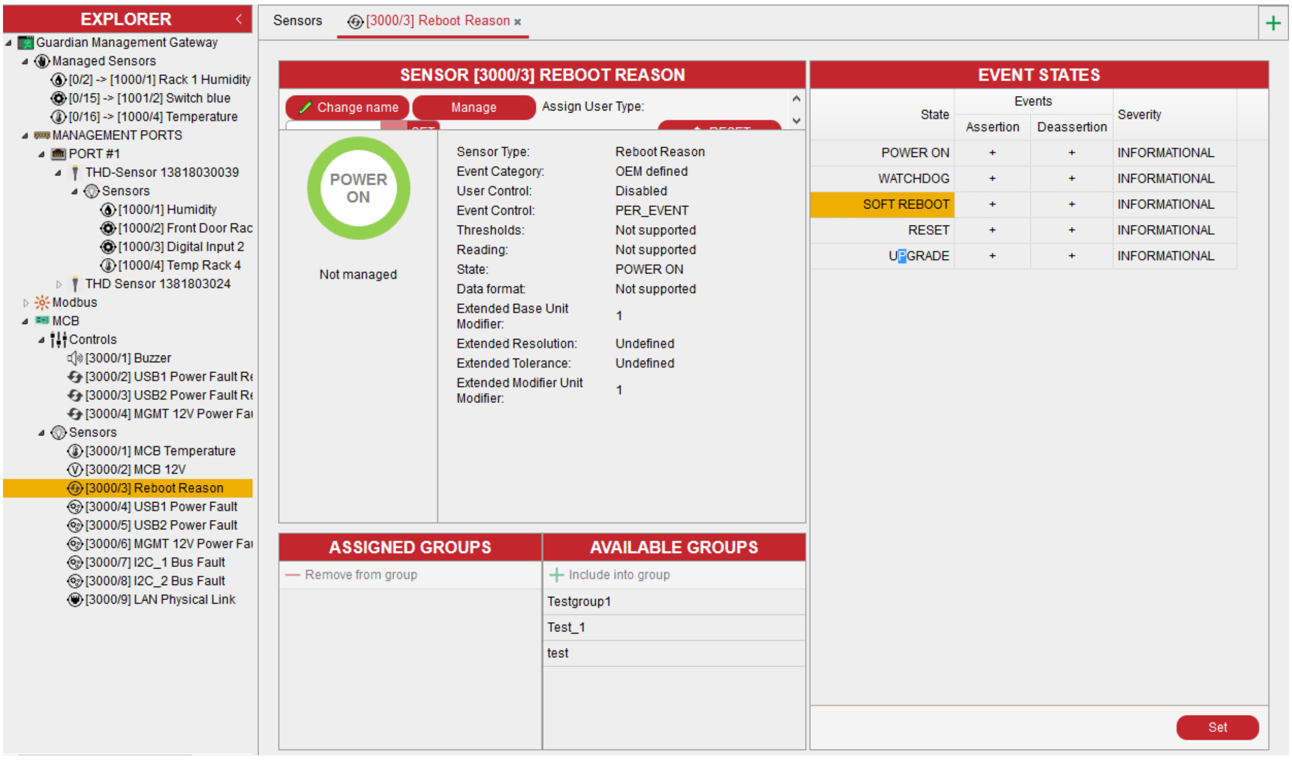
Type the new name and press the “Set” button; the sensor name will be changed.

To restore the default name of the sensor, check the “Set default” checkbox and press the “Set” button.

To change thresholds and hysteresis, change the corresponding values in the right pane and press the “Set” button in the right pane.

Managing discrete sensors:

When a discrete sensor is selected in the left pane, the middle upper pane shows the sensor state and properties, the middle lower pane shows the groups to which the sensor belongs, and the right pane shows supported event states, their event severities and event enables.



Guardian Management Gateway
- Managed Sensors
  - [0/2] -> [1000/1] Rack 1 Humidity
  - [0/15] -> [1001/2] Switch blue
  - [0/16] -> [1000/4] Temperature
- MANAGEMENT PORTS
  - PORT #1
    - THD-Sensor 13818030039
      - Sensors
        - [1000/1] Humidity
        - [1000/2] Front Door Rac
        - [1000/3] Digital Input 2
        - [1000/4] Temp Rack 4
    - THD Sensor 1381803024
- Modbus
- MCB
  - Controls
    - [3000/1] Buzzer
    - [3000/2] USB1 Power Fault Re
    - [3000/3] USB2 Power Fault Re
    - [3000/4] MGMT 12V Power Fai
  - Sensors
    - [3000/1] MCB Temperature
    - [3000/2] MCB 12V
    - [3000/3] Reboot Reason
    - [3000/4] USB1 Power Fault
    - [3000/5] USB2 Power Fault
    - [3000/6] MGMT 12V Power Fai
    - [3000/7] I2C_1 Bus Fault
    - [3000/8] I2C_2 Bus Fault
    - [3000/9] LAN Physical Link

SENSOR [3000/3] REBOOT REASON

Sensor Type:	Reboot Reason
Event Category:	OEM defined
User Control:	Disabled
Event Control:	PER_EVENT
Thresholds:	Not supported
Reading:	Not supported
State:	POWER ON
Data format:	Not supported
Extended Base Unit Modifier:	1
Extended Resolution:	Undefined
Extended Tolerance:	Undefined
Extended Modifier Unit Modifier:	1

EVENT STATES

State	Events		Severity
	Assertion	Deassertion	
POWER ON	+	+	INFORMATIONAL
WATCHDOG	+	+	INFORMATIONAL
SOFT REBOOT	+	+	INFORMATIONAL
RESET	+	+	INFORMATIONAL
UPGRADE	+	+	INFORMATIONAL

ASSIGNED GROUPS

AVAILABLE GROUPS

Testgroup1

Test_1

test

To change the name of the sensor, press the button “Change name”, as in the case of a numeric sensor.

To change event enables and event severities, change the corresponding values in the right pane and press the “Set” button in the right pane.

To include the sensor to a group, check the correspondent checkbox in the “Groups” pane.

8.2.4 User-Defined Sensor Types

There is a number of built-in sensor types which are identified by small integer numbers and listed in Table 5. However a user can define his/her own type for discrete sensors, in order to specify meaningful names to the states of the corresponding sensor and define severities appropriately. Then these user-defined sensor types can be assigned to sensors.

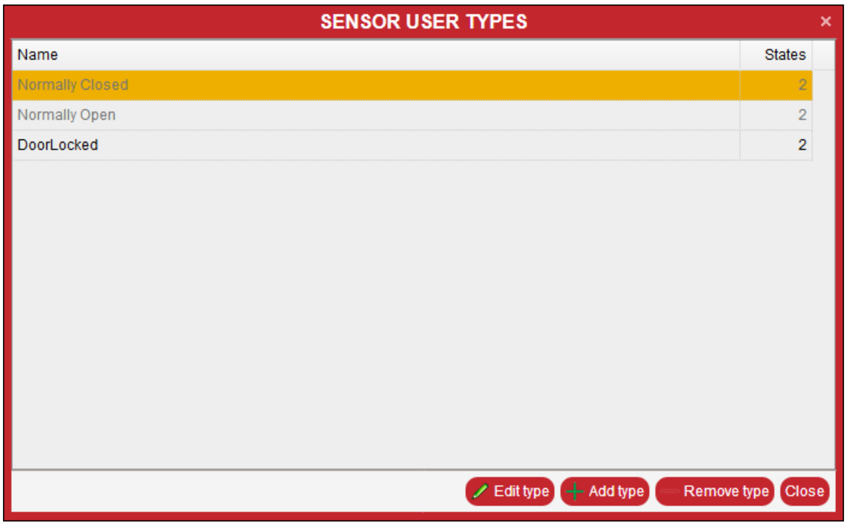
User-defined sensor types must have unique names that identify them.

Also they are assigned numeric identifiers from a specially designated range, so that these types can be used in sensor events and other data structures where numeric sensor types are required.

There are two pre-defined sensor types: “Normally Closed” and “Normally Open”.

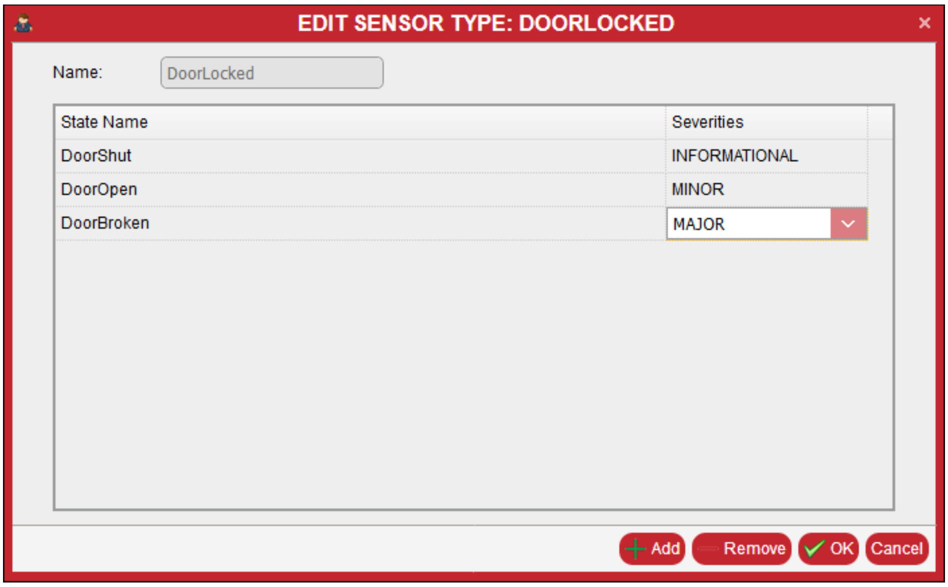
These type are included in the user-defined sensor types, but they can’t be edited or deleted. These sensor types may be assigned to a discrete sensor with two states.

From the Web interface, to get the list of the user-defined sensor types, invoke the dialog box with the menu items “Device Settings” -> “Sensor User Types”. The dialog box “Sensor User Types” allows the user to create a new user-defined sensor type (the button “Add type”), to delete an existing user-defined sensor type (the button “Remove type”), to edit an existing user-defined sensor type (the button “Edit type”).


SENSOR USER TYPES

Name	States
Normally Closed	2
Normally Open	2
DoorLocked	2

To edit an existing user-defined sensor type, press the “Edit type” button. The “Add” and “Remove” buttons in the “Edit Sensor Type” window allow to add and to remove named sensor states (and their severities), respectively. Severities are chosen from a drop-down box with a predefined list of values.

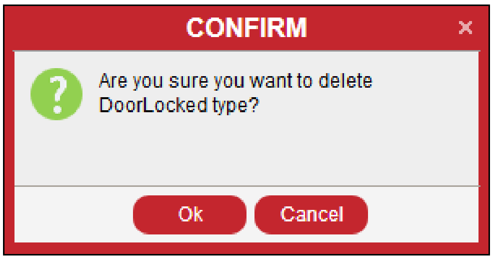


EDIT SENSOR TYPE: DOORLOCKED

Name: DoorLocked

State Name	Severities
DoorShut	INFORMATIONAL
DoorOpen	MINOR
DoorBroken	MAJOR

To delete an existing user-defined sensor type, press the “Remove type” button; a confirmation dialog appears that asks the user to confirm the deletion of the specific sensor type:



CONFIRM

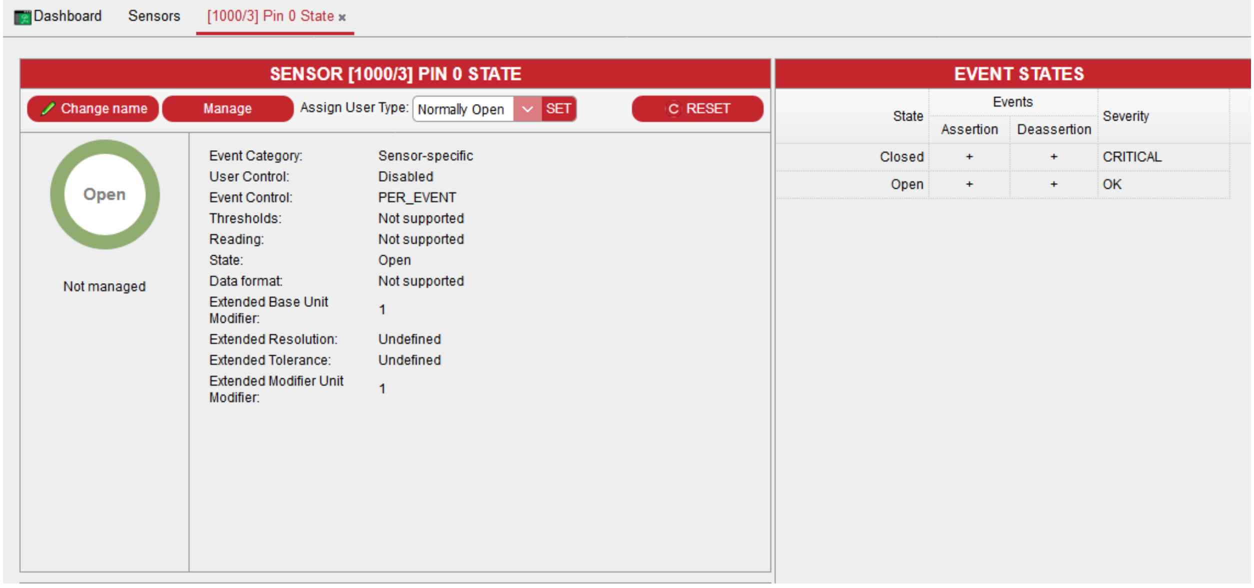
Are you sure you want to delete DoorLocked type?

8.2.5 Assigning sensor types to sensors

It is possible to assign either a built-in type or a user-defined type to a sensor. In the case of a built-in type, the type is designated by the numeric identifier. In the case of a user-defined type, the type should be defined by its name. If the type is specified by its numeric identifier, the event category number is also specified, because the meaning of sensor states depends not only on the sensor type but on the event category as well.

Event category numbers are described in Table 7. For user-defined sensor types specified by name, the event category is set to the value $0x7E$ (Sensor-specific events).

To assign a user-defined sensor type to a discrete sensor, select this discrete sensor in the left pane, then choose a user-defined sensor type from a drop-down box with the list of all the user-defined sensor types in the middle upper pane. Press the “SET” button.



SENSOR [1000/3] PIN 0 STATE

Assign User Type: Normally Open

Event Category:	Sensor-specific
User Control:	Disabled
Event Control:	PER_EVENT
Thresholds:	Not supported
Reading:	Not supported
State:	Open
Data format:	Not supported
Extended Base Unit Modifier:	1
Extended Resolution:	Undefined
Extended Tolerance:	Undefined
Extended Modifier Unit Modifier:	1

EVENT STATES

State	Events		Severity
	Assertion	Deassertion	
Closed	+	+	CRITICAL
Open	+	+	OK

8.3 Controls

Controls represent the means to change state of some physical or logical objects programmatically. For example, a GPIO that controls the state of a door lock (open/closed) can be represented as a control in the HPI model. Another example of a control can be a PWM register that determines the speed of a fan.

A control has the following attributes:

- Control number, that identifies the control within the resource that owns it
- Human-readable control name
- Control type
- Output type: the type of physical control output, e.g. dry contact-closure, fan speed or LED
- Current mode (automatic or manual)
- Default mode and state
- For analog controls, the allowed range of values.

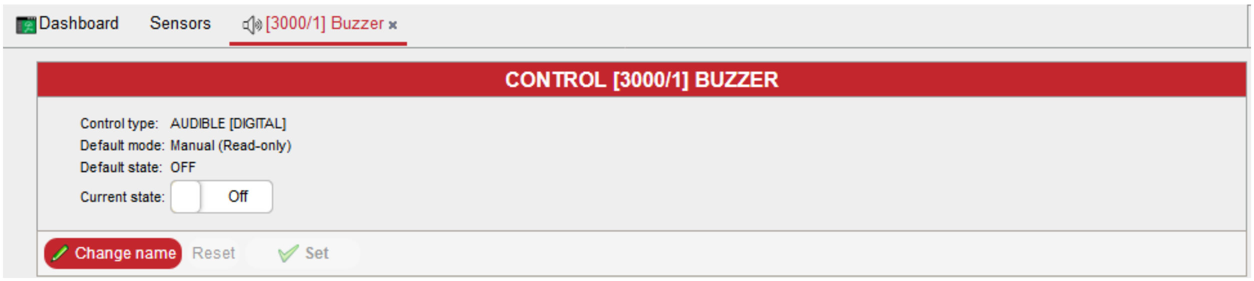
The following types exist for controls:

- Digital: these controls can be in one of the two states, On and Off. In addition, pulse operations (Pulse On and Pulse Off), may be supported for digital controls; these operations set the specific state for a control for a small period of time, and then return back to the previous state
- Discrete: these controls can be in one of several predefined states, which are specified by an integer enumeration
- Analog: these controls have a numeric (integer) value which can be set by the user
- Float analog: same as analog, but the value can be a floating-point number.

A control can be in one of the two modes: automatic and manual. In automatic mode, the control state or value is chosen automatically and the user can only read it. In manual mode, the user directly specifies the state or value for the control. Not all controls support automatic mode; a fan PWM control can be one example of a control supporting automatic mode.

8.3.1 Examples

With the Web interface, controls and sensors are shown together in the list of instruments in the tree (left) pane below the corresponding resource. To manipulate a control, select it in the tree pane; the control management pane will be shown on the right:



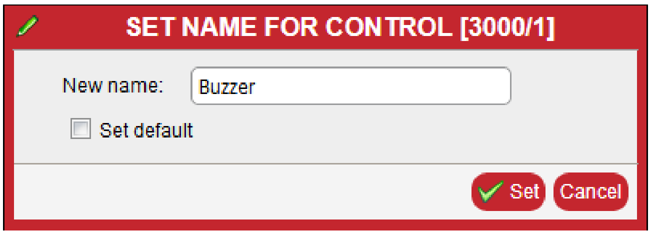
CONTROL [3000/1] BUZZER

Control type: AUDIBLE [DIGITAL]
Default mode: Manual (Read-only)
Default state: OFF
Current state: Off

The information shown on the screen about the control includes its number, name, type, output type, default mode, default state, current mode and current state or value.

To change the current state/value of the control, enter the new value or toggle the state in the field “Current State” and press the “Set” button. The new value or state will be set.

To change the name of the control, press the button “Change name”. The dialog asking for the new control name will appear:



SET NAME FOR CONTROL [3000/1]

New name: Buzzer

□ Set default

Enter the new name for the control in the text box and press the “Set” button. The control name will be changed.

To restore the default name of the control, check the “Set default” checkbox and press the “Set” button.

8.4 Events

Events represent the method for an HPI system to notify the environment about state and configuration changes in it. In Guardian Management Gateway, the subset of the whole HPI set of events is supported.

8.4.1 Event categories

Events, that a Guardian Management Gateway can generate, can be split in several categories:

Resource events

- Resource events are sent when:
    - Resource added: Event is sent when a new resource is added to the system (hot-inserted)
    - Resource removed: Event is sent when a resource is removed from the system (hot-extracted)
    - Resource updated: Event is sent when the population of instruments (sensors, controls, inventory) changes for a given resource

Sensor Events

- Sensor events are sent when:
    - For numeric sensors, when a sensor value crosses one of its thresholds; depending on the event enable mask, events can be sent when the sensor value goes outside a threshold, returns back or both
    - For discrete sensor, when the sensor changes its state; also the event enable mask determines for which state changes the event is generated

Software Events

- Software events are sent by software when certain actions are initiated by the user or other software-related conditions occur; for example:
    - When a user logs in or logs out
    - When the Guardian Management Gateway connects to a wireless LAN or disconnects from a wireless LAN
    - When a specific server, which is being monitored, becomes reachable or unreachable

Upgrade Events

- Upgrade events are sent when a firmware upgrade takes place and indicate different phases of the upgrade process.

8.4.2 Event parameters

Events are data packets that have standardized format. Besides the type of the event, they carry parameters which vary depending on the event type.

All events include the following parameters:

- Event type
- Timestamp (when the event happened)
- Severity (can be one of Critical, Major, Minor, Informational or OK).

Resource Events

Resource events carry the resource ID as the only additional parameter; this resource ID identifies the resource that has been added, removed or changed its instrument population.

Sensor events

For sensor events, the following additional parameters are provided:

- Resource ID and sensor number, for the sensor that originated the event
- Sensor type (e.g. temperature, voltage, humidity, etc.)
- Event category (one specific event category is threshold crossing)
- Is the event condition asserted or deasserted?
- For threshold-crossing events on numeric sensors:
    - Which threshold has been crossed?
    - The sensor value that triggered the event
    - The value of the threshold that has been crossed
- For sensor state change events on discrete sensors:
    - A single state being asserted or deasserted that triggered the event
    - The current state mask of the sensor

Software events

Software events contain the following additional parameters: the specific event type (e.g. “user logs in”) and a text string that describes the event, in a human-readable form.

Upgrade events

Upgrade events carry the enumeration that identifies the current stage of the upgrade process, as the only significant additional parameter.

8.4.3 Event processing

After being generated, the event passes through event filters that may initiate certain actions based on the event type and values of event parameters.
Configuration of event filters and actions is discussed in the chapter 18 Events and Actions.

Finally, the event is stored in the System Event Log on the Guardian Management Gateway where it can be examined later.

8.5 Inventory

An inventory contains information about a resource in a special structured format. The format used for Guardian Management Gateway is the IPMI FRU Information format, described in [2]. Information is represented in several standard sections, followed by a number of OEM-specific records of variable length.

The following standard sections, defined in the IPMI FRU Information format, are used with Guardian Management Gateway resources:

- Board information area. This section contains information about the hardware aspects of the resource, including date and time of manufacturing, manufacturer name, board product name, part number and serial number
- Product information area. This section contains information about the general aspects of the resource, or about the resource as a separate product; it includes manufacturer name, product name, product version, part number, serial number and optional asset tag.

The chassis information area and the internal use area, also defined in the IPMI FRU Information format, are not used in the Guardian Management Gateway.

The following OEM-specific variable-length records are used with Guardian Management Gateway resources (the OEM is nVent/Schroff for all records):

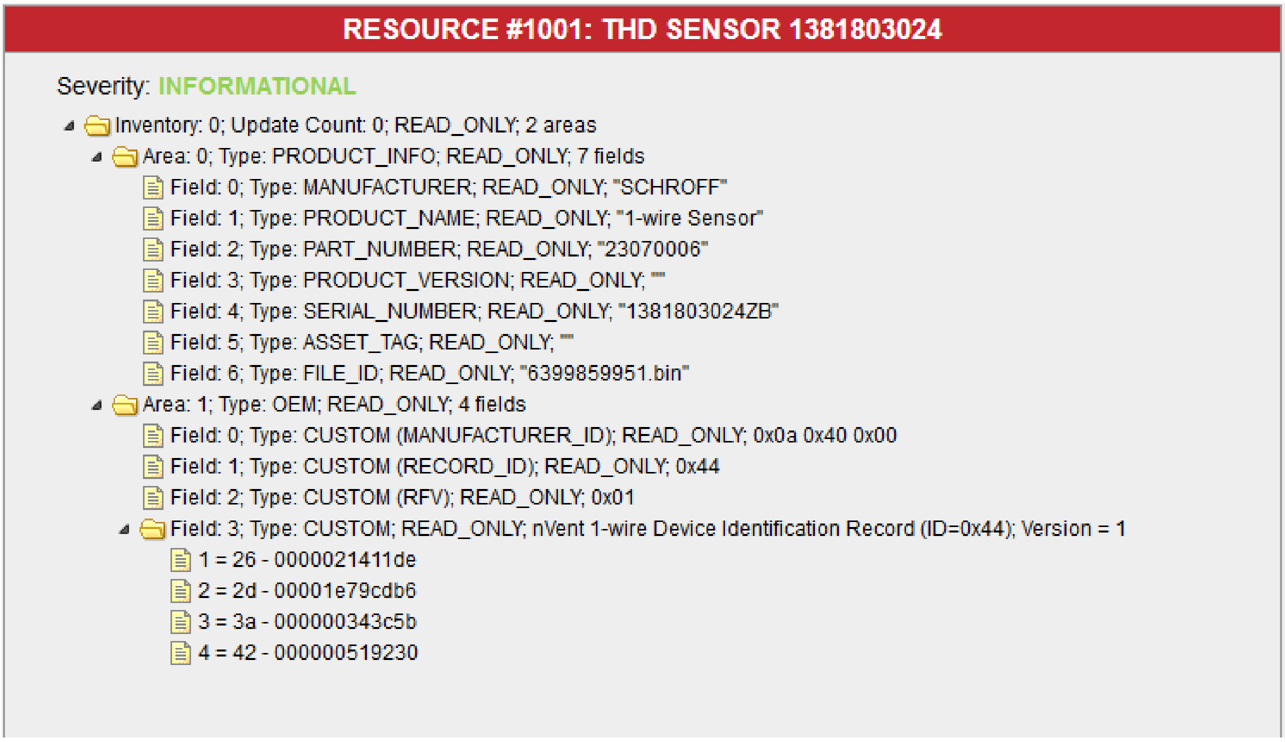
- Guardian Management Gateway configuration record; this record describes general configuration of power-related aspects of the Guardian Management Gateway
- LCD calibration parameters record; this record contains calibration parameters of the LCD screen
- Sensor device identification record; this record identifies the components of a specific Schroff sensor device.

For the specific types of Guardian Management Gateway resources, inventory contains the following areas and records:

- Managed sensors resource (`0`): the corresponding inventory describes the Guardian Management Gateway as a whole. It contains the board information area, the product information area and the Guardian Management Gateway configuration record
- Schroff environmental sensor resources (`1000 – 1999`) : the inventory contains the product information area and the sensor device identification record (no board information area is included because of the limited inventory size on these devices)
- MCB (`3000`) : the inventory contains the board information area, the product information area and the LCD calibration parameters record

From the user perspective, inventory is always read-only; only read access is provided to it.

With the Web interface, a user can view the inventory on the global resource, sensor devices and on the MCB. To do that, point to the corresponding item in the tree (left) pane; the inventory will be shown in the hierarchical format in the pane in the middle of the screen. Use arrows to the left of the hierarchical items to open and close the corresponding branches of the hierarchy.



RESOURCE #1001: THD SENSOR 1381803024

```
Severity: INFORMATIONAL
Inventory: 0; Update Count: 0; READ_ONLY; 2 areas
  Area: 0; Type: PRODUCT_INFO; READ_ONLY; 7 fields
    Field: 0; Type: MANUFACTURER; READ_ONLY; "SCHROFF"
    Field: 1; Type: PRODUCT_NAME; READ_ONLY; "1-wire Sensor"
    Field: 2; Type: PART_NUMBER; READ_ONLY; "23070006"
    Field: 3; Type: PRODUCT_VERSION; READ_ONLY; ""
    Field: 4; Type: SERIAL_NUMBER; READ_ONLY; "1381803024ZB"
    Field: 5; Type: ASSET_TAG; READ_ONLY; ""
    Field: 6; Type: FILE_ID; READ_ONLY; "6399859951.bin"
  Area: 1; Type: OEM; READ_ONLY; 4 fields
    Field: 0; Type: CUSTOM (MANUFACTURER_ID); READ_ONLY; 0x0a 0x40 0x00
    Field: 1; Type: CUSTOM (RECORD_ID); READ_ONLY; 0x44
    Field: 2; Type: CUSTOM (RFV); READ_ONLY; 0x01
    Field: 3; Type: CUSTOM; READ_ONLY; nVent 1-wire Device Identification Record (ID=0x44); Version = 1
      1 = 26 - 0000021411de
      2 = 2d - 00001e79cdb6
      3 = 3a - 000000343c5b
      4 = 42 - 000000519230
```

9 Managed Sensors

Managed sensors allow a user to designate a subset of the whole set of sensors in the Guardian Management Gateway (which may be quite large), work with this subset and apply additional management actions to it.

The sensors in this subset are mapped to resource 0 (“Managed sensors”) and are allocated available sensor numbers on that resource. They still continue to be available at their original resource number and sensor number; sensor numbers on resource 0 are just aliases.

One possible use of managed sensors is to collect most important sensors in one place to be able to manage them efficiently (since an Guardian Management Gateway, depending on configuration, may have hundreds and even thousands of different sensors).

9.1 Features of managed sensors

Sensors on resource 0 can have the following additional attributes compared to regular sensors:

- User-defined sensor name, as an arbitrary text string
- Description, as an arbitrary text string
- User-defined sensor type and subtype, as arbitrary text strings (in addition to normal HPI sensor type that is defined for all sensors)
- Location attributes in the form of X, Y and Z coordinates, in the form of arbitrary text strings (for the Z coordinate, a numeric representation can be chosen).

The attributes listed above are opaque for the Guardian Management Gateway but are kept persistent across reboots. These attributes are associated with the sensor representation on resource 0 (that is, they are not visible if the sensor is accessed by its original resource number and sensor number).

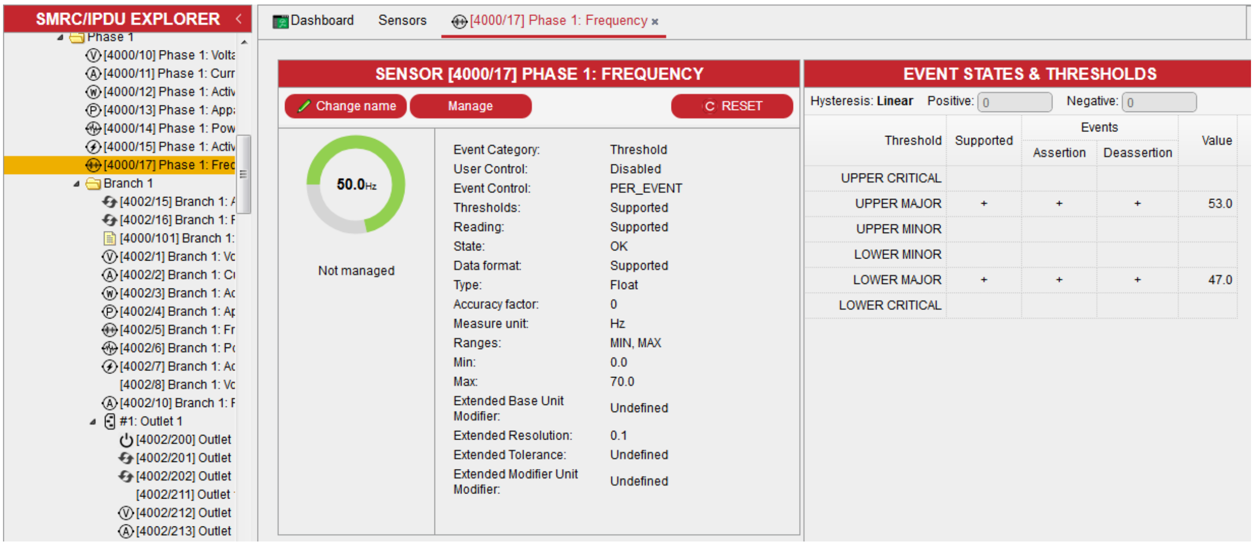
In addition, sensor logging applies to managed sensors. Sensor logging involves periodic polling of managed sensors and calculating some statistics for the sensor values over time.

9.2 Attaching and detaching managed sensors

To make sensor a managed sensor, a user should first choose the actual sensor, by resource and sensor number, and then map (attach) it to resource *0*. Sensor mapping is persistent across reboots. If a sensor belongs to a hot-swappable resource, the mapping is also automatically restored when the resource is hot-inserted (if the resource was previously hot-extracted or was not present during the initial start).

With the Web interface, attaching and detaching can be performed in the following way:

To create a managed sensor, find the actual sensor in the tree pane on the left side on the screen and click on it. The sensor information pane contains information whether the sensor is managed and, if the sensor is not attached yet, the "Manage" button.



SENSOR [4000/17] PHASE 1: FREQUENCY

Not managed

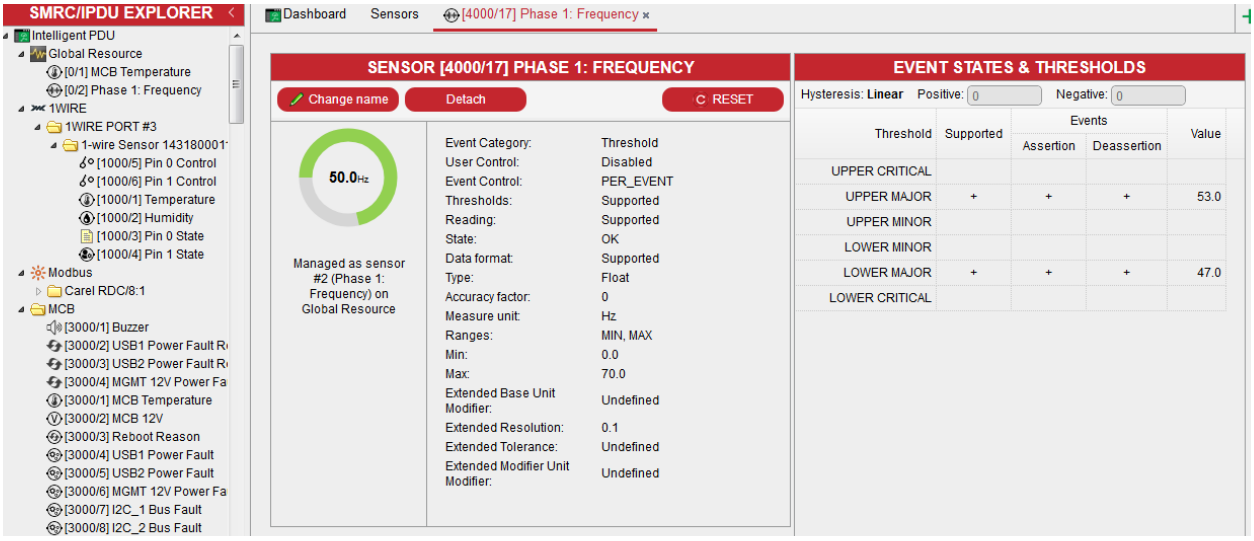
Event Category:	Threshold
User Control:	Disabled
Event Control:	PER_EVENT
Thresholds:	Supported
Reading:	Supported
State:	OK
Data format:	Supported
Type:	Float
Accuracy factor:	0
Measure unit:	Hz
Ranges:	MIN, MAX
Min:	0.0
Max:	70.0
Extended Base Unit Modifier:	Undefined
Extended Resolution:	0.1
Extended Tolerance:	Undefined
Extended Modifier Unit Modifier:	Undefined

EVENT STATES & THRESHOLDS

Hysteresis: **Linear** Positive: 0 Negative: 0

Threshold	Supported	Events		Value
		Assertion	Deassertion	
UPPER CRITICAL				
UPPER MAJOR	+	+	+	53.0
UPPER MINOR				
LOWER MINOR				
LOWER MAJOR	+	+	+	47.0
LOWER CRITICAL				

Press the "Manage" button to attach the sensor. The sensor information pane changes to reflect that the sensor is now managed and shows the managed sensor number; the "Manage" button becomes unavailable:



SENSOR [4000/17] PHASE 1: FREQUENCY

Managed as sensor #2 (Phase 1: Frequency) on Global Resource

Event Category:	Threshold
User Control:	Disabled
Event Control:	PER_EVENT
Thresholds:	Supported
Reading:	Supported
State:	OK
Data format:	Supported
Type:	Float
Accuracy factor:	0
Measure unit:	Hz
Ranges:	MIN, MAX
Min:	0.0
Max:	70.0
Extended Base Unit Modifier:	Undefined
Extended Resolution:	0.1
Extended Tolerance:	Undefined
Extended Modifier Unit Modifier:	Undefined

EVENT STATES & THRESHOLDS

Hysteresis: **Linear** Positive: 0 Negative: 0

Threshold	Supported	Events		Value
		Assertion	Deassertion	
UPPER CRITICAL				
UPPER MAJOR	+	+	+	53.0
UPPER MINOR				
LOWER MINOR				
LOWER MAJOR	+	+	+	47.0
LOWER CRITICAL				

The list of managed sensors is shown in the tree pane on the left side of the screen, under "Managed Sensors":

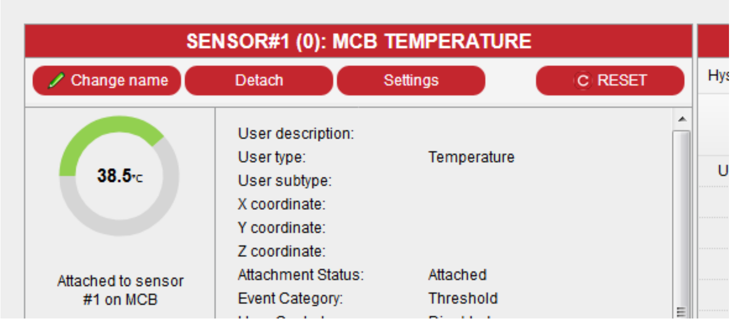
In the sensor information pane for a managed sensor, the title line shows information about the original sensor.

To delete a managed sensor, press the “Detach” button in the title line of the sensor information pane, either for the managed sensor on resource *0*, or for the actual sensor. The corresponding managed sensor on resource *0* will disappear.

Unlike CLI, in the Web interface there is no way to attach a sensor to a specific managed sensor number on resource *0*.

9.3 Managing attributes of managed sensors

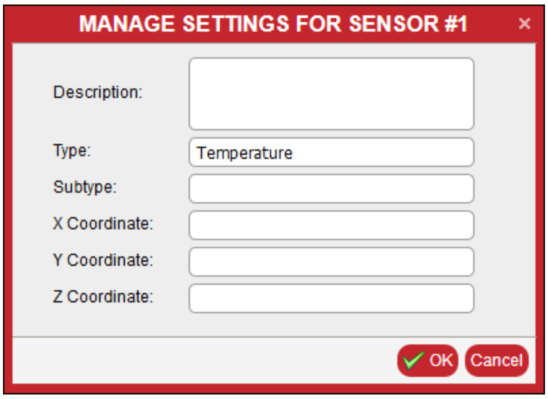
To view and modify managed sensor attributes (except the user-defined sensor name) with the Web interface, press the “Settings” button in the title bar of the sensor information pane.



Attached to sensor #1 on MCB

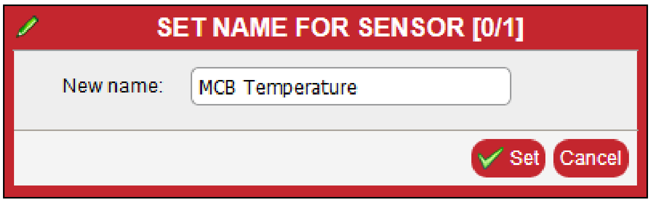
User description:
User type: Temperature
User subtype:
X coordinate:
Y coordinate:
Z coordinate:
Attachment Status: Attached
Event Category: Threshold

The “Manage settings” dialog appears. The user can view and edit the text of sensor attributes and press the “OK” button to save the values. Web interface automatically removes the numeric restriction for the Z coordinate value, if the value entered in the corresponding field is not numeric.



Description:
Type: Temperature
Subtype:
X Coordinate:
Y Coordinate:
Z Coordinate:

To change the managed sensor name with the Web interface, use the same mechanism as with a regular sensor: choose the target sensor in the left tree pane and press the “Change name” button in the title bar of the sensor information pane. The dialog box appears that allow the user to edit the sensor name and save changes.



New name: MCB Temperature

The “Set default” operation is not supported for managed sensors.

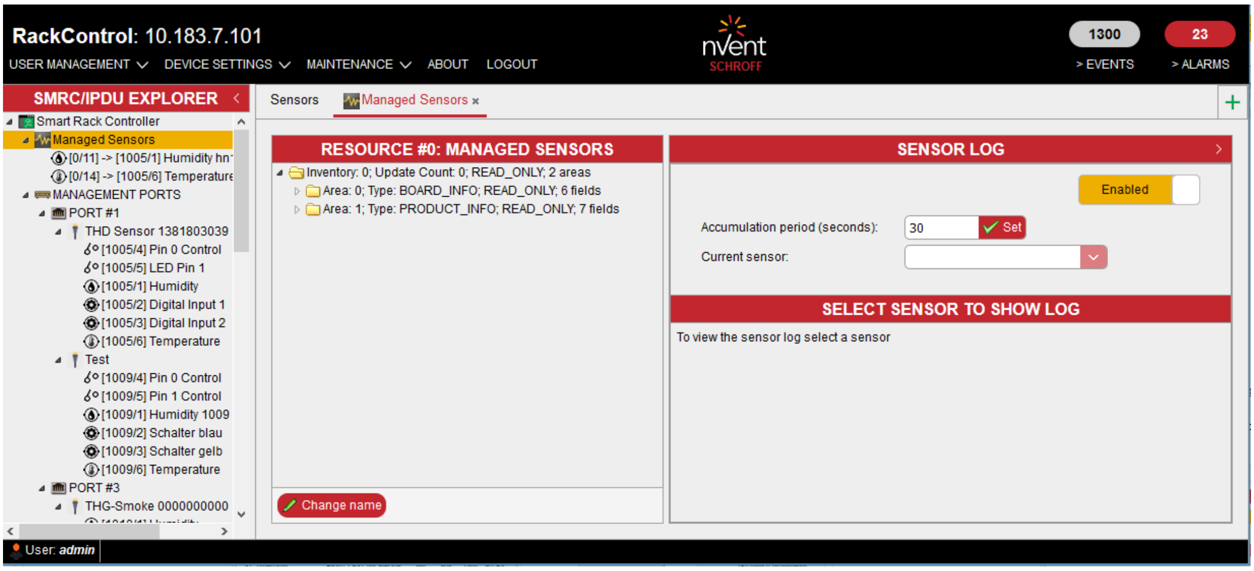
9.4 Logging for managed sensors

The logging facility for managed sensors implements periodic polling and accumulation of sensor values. It works as follows:

- The logging period is defined, during which managed sensor values are accumulated; the typical duration of this period is about 30 seconds and this value is configurable for all sensors
- During this period, the sensors are periodically polled; the typical period is about 3 seconds but can be configured separately for each sensor (this is the configuration parameter “Polling period” which is defined for all sensors, not just managed sensors)
- After the period is finished, the following values are calculated for each managed sensor:
    - number of polls during the period
    - number of polls in which the sensor reported a value (the sensor could return the condition “sensor reading unavailable” during some polls)
    - average sensor value for the period
    - minimal sensor value for the period
    - maximal sensor value for the period
    - dispersion of the sensor value during the period
    - accumulated event state mask during the period (it includes all sensor states that were detected during the period)
- These values are stored in an entry of a ring buffer; there is a separate ring buffer for each managed sensor and the number of entries in each buffer is fixed to 16 entries. When all entries are filled in, the buffer wraps around.

The logging facility can be enabled and disabled by the user.

To get access to the managed sensor log facility with the Web interface, choose the “Managed Sensors” in the left tree pane; the “Sensor Log” pane will be on the right side of the screen. Choose the sensor from “Current Sensor” combo-box to see the sensor log for a specific managed sensor. Also, the controls for changing the accumulation period and enabling/disabling sensor log exist on this page.



Inventory: 0; Update Count: 0; READ_ONLY; 2 areas
Area: 0; Type: BOARD_INFO; READ_ONLY; 6 fields
Area: 1; Type: PRODUCT_INFO; READ_ONLY; 7 fields

Accumulation period (seconds): 30
Current sensor:

To view the sensor log select a sensor

10 Group Operations on Controls and Sensors

It is possible to group controls and sensors into larger entities and perform group operations on them.

For controls, the group operation is setting all controls in the group to the same state. For example, for digital controls this can involve setting all controls to the state ON or to the state OFF. For analog or discrete controls, all controls are assigned the same value.

For group control operations, it is possible to specify the order of processing of controls and, for each control, a delay after processing that control.

A group control operation can be executed synchronously or asynchronously. For synchronous execution, the caller waits until the operation is complete. However, since a group control operation may take a long time, asynchronous execution is also supported. In that case, the caller does not wait until the execution is complete, but gets control back immediately after execution is started and can then poll for the progress of the execution.

For sensors, the group operation can be setting thresholds to all sensors in the group to the same values, or calculating some aggregate value over the values of sensors comprising the group. The supported aggregates include sum, average, dispersion, minimum and maximum values, and others. For example, if homogeneous sensors from several resources are grouped together, it is possible to calculate an average value from the values of these sensors.

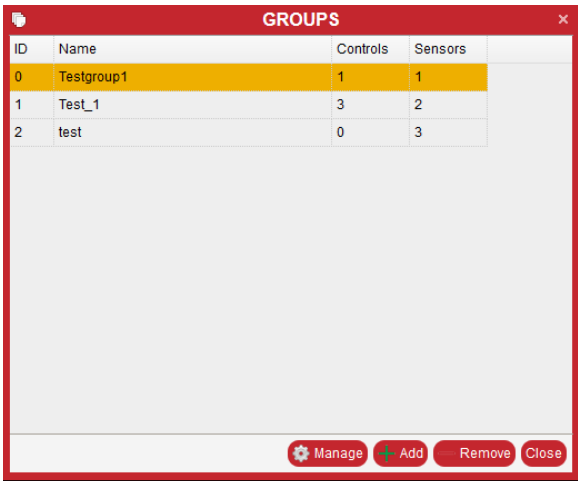
To facilitate group operations, a special named entity (a group) is created by a user, and then, sensors and/or controls are added to that entity. The order of adding controls to the group corresponds to the order in which controls are processed for a group operation. The order of adding sensors to the group is not significant.

It is possible to have separate groups for working with sensors and controls or use one group to work with several sensors and several controls. Operations with sensors and controls in one group are handled independently.

The following operations with groups are supported:

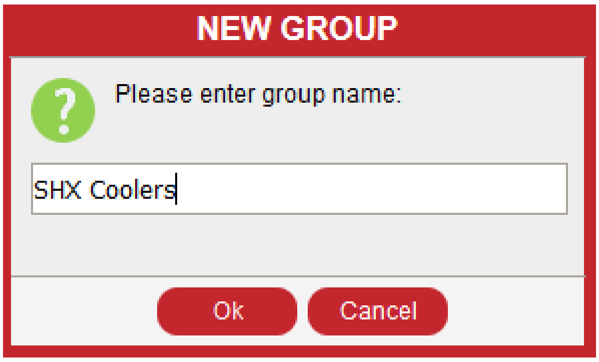
- Create a new group and assign a name to it
- Delete an existing group by name
- Get the list of existing groups
- Add a control to the group, specify the delay after processing this control
- Delete a control from the group, by control number or by position
- List all controls and sensors in the group
- Assign a state to all controls in the group, synchronously
- Assign a state to all controls in the group, asynchronously
- Show progress of an asynchronous group control operation
- Cancel an asynchronous group control operation
- Add a sensor to the group
- Delete a sensor from the group, by sensor number or by position
- Set thresholds to certain values to all sensors in the group
- Calculate an aggregate over all sensors in the group.

From the Web interface, to get the list of the groups , invoke the dialog box with the menu items “Maintenance” -> “Sensor/Control Groups”. The dialog box “Groups” allows the user to create a new group (the button “Add”), delete an existing group (the button “Remove”), edit a group or perform a group operation on it (the button “Manage”).



ID	Name	Controls	Sensors
0	Testgroup1	1	1
1	Test_1	3	2
2	test	0	3

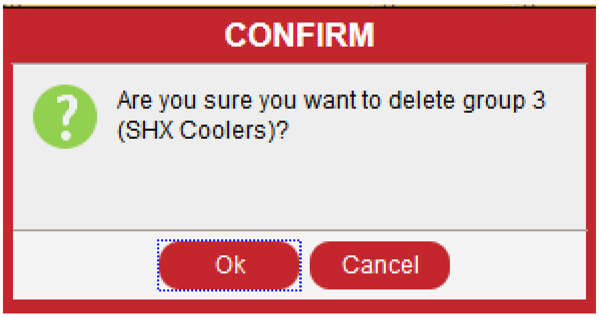
To create a new group, press the “Add” button; a “New group” dialog appears that asks the user for the name of the new group that should be unique:



Please enter group name:

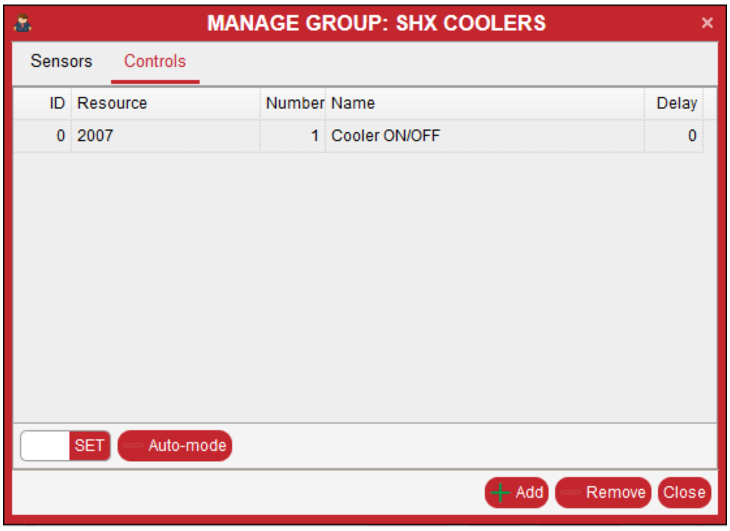
SHX Coolers

To delete a group, press the “Remove” button; a confirmation dialog appears that asks the user to confirm the deletion of the specific group:



Are you sure you want to delete group 3 (SHX Coolers)?

To edit a group or perform a group operation on it, press the “Manage” button. The “Manage group” window appears that contains two tabs: “Sensors”. “Controls”.

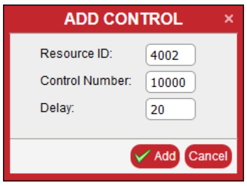


MANAGE GROUP: SHX COOLERS

ID	Resource	Number	Name	Delay
0	2007	1	Cooler ON/OFF	0

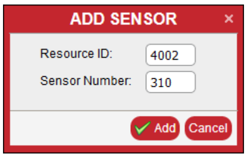
On each of the tabs, there is an “Add” button that allows a user to add the corresponding instrument to the group, and the “Remove” button that allows the user to remove the selected instrument from the group.

When an “Add” button is pressed, a “new instrument” dialog shows up that asks the user for the identification and parameters of the new instrument. For a control, these are the resource ID, the control number and the mandatory delay after setting the control state (in milliseconds). For a sensor, these are the resource ID and the sensor number.



ADD CONTROL

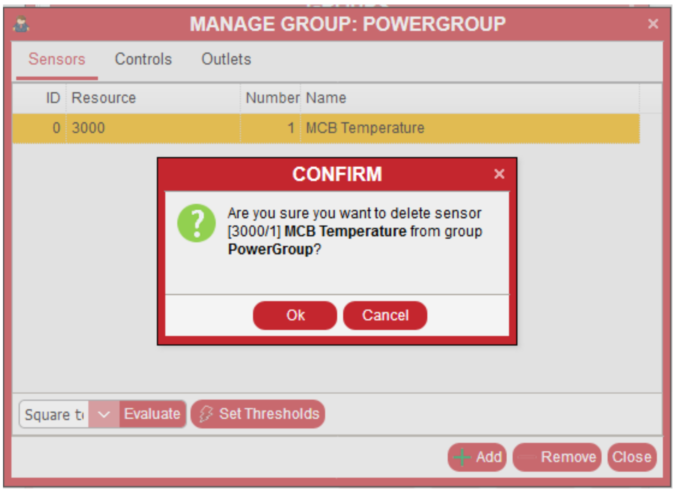
Resource ID: 4002
Control Number: 10000
Delay: 20



ADD SENSOR

Resource ID: 4002
Sensor Number: 310

When a “Remove” button is pressed, and a group item is selected, a confirmation dialog shows up, and if the user presses the “OK” button, the corresponding item is deleted.



MANAGE GROUP: POWERGROUP

ID	Resource	Number	Name
0	3000	1	MCB Temperature

CONFIRM

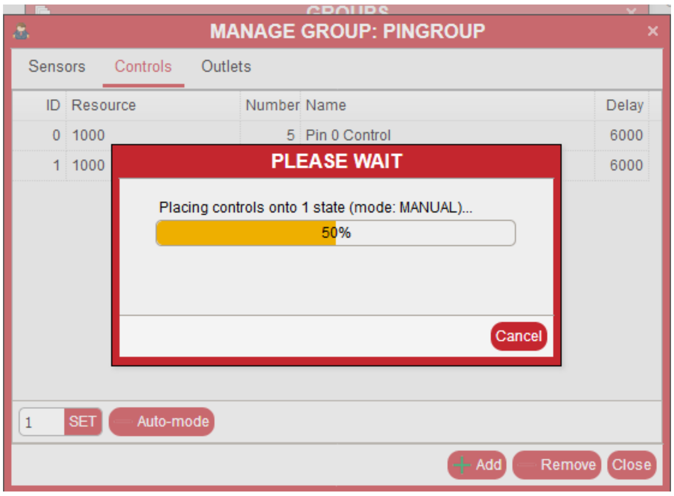
Are you sure you want to delete sensor [3000/1] **MCB Temperature** from group **PowerGroup**?

To set all controls in the group to the same state, select the “Controls” tab, enter the numeric state in the edit field in the left-bottom corner, and press the “SET” button.

For digital sensors, use the number *0* for the “Off” state, *1* for the “On” state, *2* for the “Pulse Off” state, and *3* for the “Pulse On” state.

To set all controls in the group to Automatic mode, press the “Auto-mode” button.

In both cases, an asynchronous operation is started and a progress bar indicating the progress of the operation is shown. While the operation is in progress, it can be cancelled by pressing the “Cancel” button.



MANAGE GROUP: PINGROUP

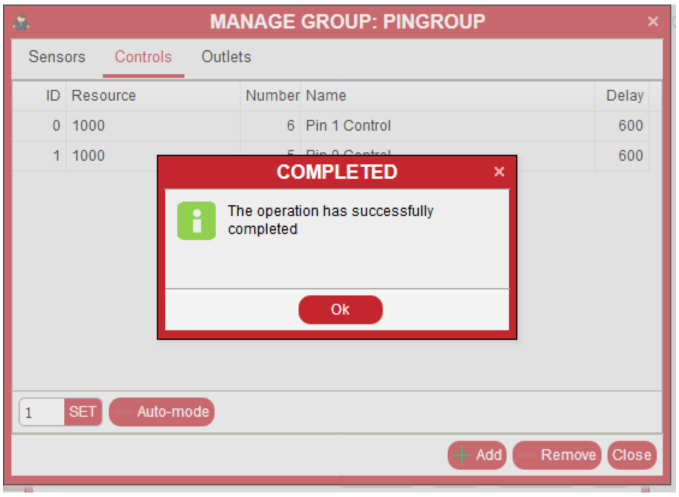
ID	Resource	Number	Name	Delay
0	1000	5	Pin 0 Control	6000
1	1000			6000

PLEASE WAIT

Placing controls onto 1 state (mode: MANUAL)...

50%

After the operation is completed, a message box indicating the successful completion is shown. Press the “OK” button to dismiss it:



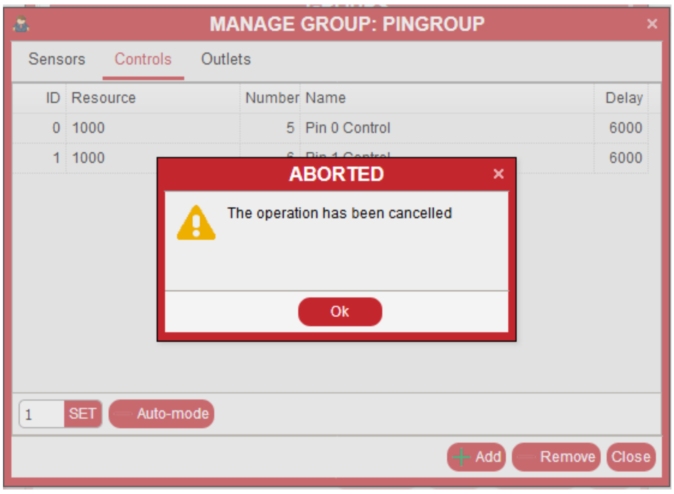
MANAGE GROUP: PINGROUP

ID	Resource	Number	Name	Delay
0	1000	6	Pin 1 Control	600
1	1000	5	Pin 0 Control	600

COMPLETED

The operation has successfully completed

If the user presses the “Cancel” button, the operation is cancelled and the corresponding message box is shown; also pressing the “OK” button will dismiss it.



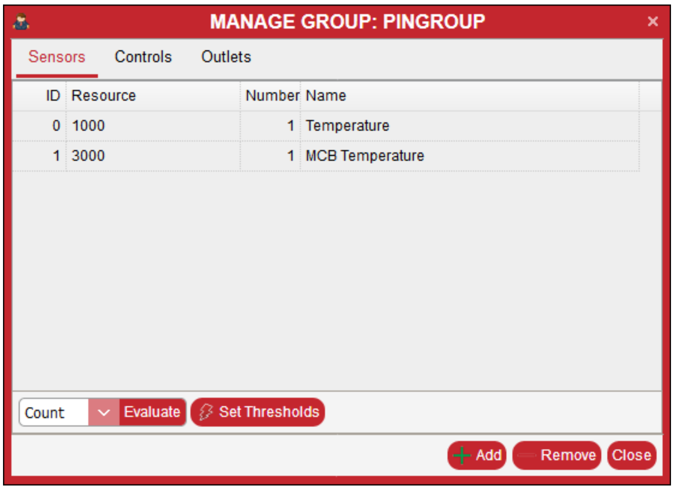
MANAGE GROUP: PINGROUP

ID	Resource	Number	Name	Delay
0	1000	5	Pin 0 Control	6000
1	1000	6	Pin 1 Control	6000

ABORTED

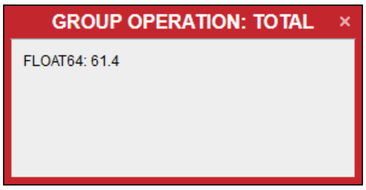
The operation has been cancelled

To initiate an aggregate operation on all sensors in a group, select the “Sensors” tab, choose the operation in the drop-down box in the left-bottom corner and press the button “Evaluate”.



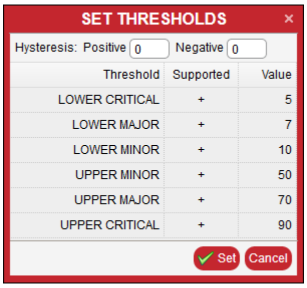
ID	Resource	Number	Name
0	1000	1	Temperature
1	3000	1	MCB Temperature

The “Group operation” window is generated that shows the result:



```
FLOAT64: 61.4
```

To set a threshold (or hysteresis) value for all sensors in a group press the button “Set threshold”, fill in the fields in the “Set Thresholds” dialog, and press the “Set” button.



Hysteresis: Positive 0 Negative 0

Threshold	Supported	Value
LOWER CRITICAL	+	5
LOWER MAJOR	+	7
LOWER MINOR	+	10
UPPER MINOR	+	50
UPPER MAJOR	+	70
UPPER CRITICAL	+	90

11 Users, Roles and Privileges

A list of valid users exists for a Guardian Management Gateway. A valid user can log in to some upper Guardian Management Gateway interface: Web, CLI (with SSH) or SNMP. The list of Guardian Management Gateway users is integrated with the list of users in the underlying Linux.

With respect to user permissions, the role-based model is used.

Each user is associated with one or more roles (e.g. “administrator”, “normal user” or “power user”). Each role has a set of privileges associated with it. A user possesses a certain privilege if at least one role this user is associated with, has this privilege.

For each user, the following attributes are defined:

- User name (used as a logon name)
- Password (not stored explicitly, but only as a hash)
- Full user name
- User phone number
- User e-mail address
- “User enabled” flag
- The list of roles associated with the user
- Preferred measurement units
- Web session preferences
- SSH public key
- SNMPv3 settings
- Preferred interface language (English, German, French)

External users are the users not listed in the list of valid users but that can be authenticated using external means (e.g. via LDAP). For such users, the entry in the user list is created during first logon. The role “ExternalUserRole” is used and other attributes are defined as empty strings. These attributes can later be redefined by an administrator.

For each role, the following attributes are defined:

- Role name (used in the user attributes)
- Role description (arbitrary text)
- Privilege mask (a bit mask where bit 1 means the corresponding privilege is included).

The following privileges are currently defined (this list may be extended in future versions of the firmware):

- Administrator – allows all possible actions
- Change Authentication – allows changing authentication settings
- Change Date/Time – allows changing date and time
- Change Event Setup – allows changing event setup and event filters
- Change External Sensor Configuration – allows changing attributes of hardware sensors (1-Wire,MCB, but excluding Modbus devices)
- Change LHX Configuration – allows changing attributes of Modbus devices
- Change Network Configuration – allows changing network configuration
- Change Own Password – allows a user to change own password
- Change Security Settings – allows changing security-related settings
- Change SNMP Settings – allows changing SNMP settings
- Change User Settings – allows changing user settings (for other users)
- Change Guardian Management Gateway Configuration – allows changing Guardian Management Gateway-related settings
- Clear Event Log – allows clearing the System event log
- Firmware Update – allows to perform Guardian Management Gateway firmware updates
- Perform Reset – allows the user to perform reset (Restart, Reboot and Factory Reset operations)
- View Event Setup – allows the user to view event setup and event filters
- View Event Log – allows the user to view the System event log
- View Security Settings – allows the user to view security-related settings
- View SNMP Settings – allows to view SNMP settings

- View User Settings – allows to view user settings (for other users)
- View Guardian Management Gateway Configuration – allows to view Guardian Management Gateway-related settings

On the first start of an Guardian Management Gateway, or after a factory reset, the following default list of users and roles exists on the Guardian Management Gateway:

- User “admin”, password “admin”, enabled, list of roles consists of one role “AdministratorRole”.
- User “user”, password “user”, enabled, list of roles consists of one role “UserRole”.
- User “guest”, password “guest”, disabled, list of roles consists of one role “ReadOnlyUserRole”.
- Role “AdministratorRole”: includes the following privileges: Administrator.
- Role “UserRole”: includes the following privileges: Change Date/Time, Change Event Setup, Change External Sensor Configuration, Change LHX Configuration, Change Network Configuration, Change Own Password, Change Guardian Management Gateway Configuration, Clear Event Log, Firmware Update, Perform Reset, View Event Setup, View Event Log, View Security Settings, View SNMP Settings, View User Settings, View Guardian Management Gateway Configuration, Use Groups, Change Group Configuration.
- Role “ReadOnlyUserRole”: includes the following privileges: View Event Setup, View Event Log, View Security Settings, View SNMP Settings, View User Settings, View Guardian Management Gateway Configuration.
- Role “ExternalUserRole”: includes the following privileges: Perform Reset, View Event Setup, View Event Log, View Security Settings, View SNMP Settings, View User Settings, View Guardian Management Gateway Configuration. This role is automatically used for external users.

This configuration of roles and users can be subsequently changed by a user having the Change User Settings privilege.

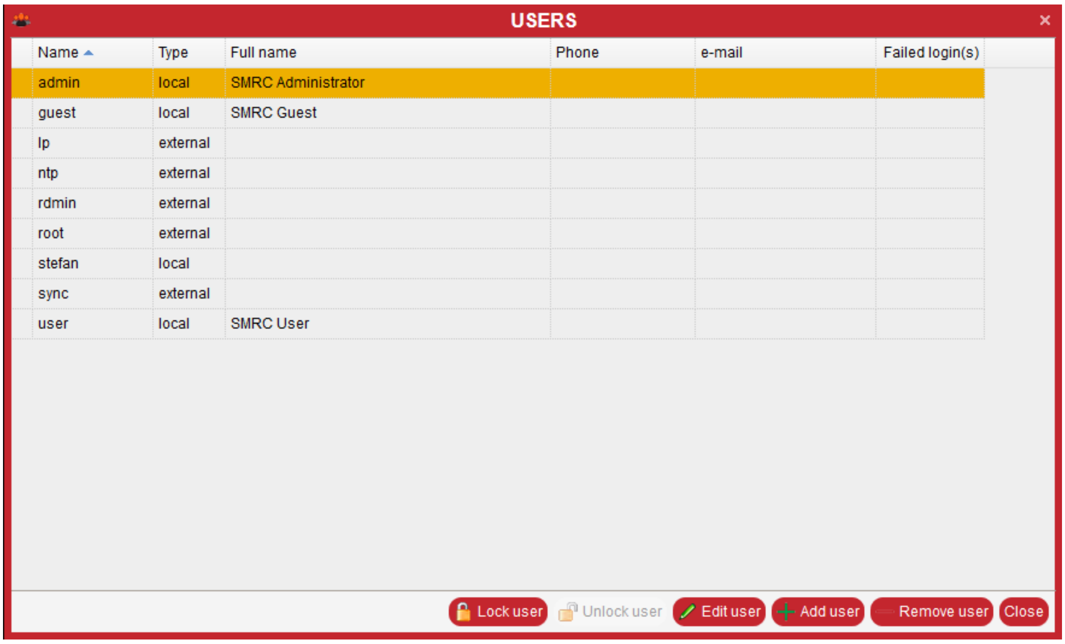
The following management operations are defined for the users:

- Get user information by index. The returned information includes all user attributes except the password and the list of roles. This operation allows enumerate existing users.
- Get user information by user name
- Create a user, specifying user name, password, attributes and the list of roles
- Delete a user by user name
- Change user attributes by user name
- Change user password (by user name or for the current user)
- Set user SSH public key by user name
- Get SNMPv3 attributes for a user by user name
- Set SNMPv3 attributes for a user by user name
- Get list of roles for a user by user name
- Set list of roles for a user by user name
- Get the mask of privileges that user has, by user name
- Get role information by index. The returned information includes role name, description and the privilege mask. This operation allows enumerate existing roles.
- Get role information by role name
- Create a role
- Delete a role by role name
- Change role information by role name
- Add privileges to the existing role, by role name
- Remove privileges from the existing role, by role name.

11.1 Create a new user

Before you create a new user, check the privileges of the existing roles. If the privileges of the existing roles are not suitable for the new user, first create a new role.

From the Web interface, to create a new user, invoke the dialog box with the menu items “User Management” -> “Users” and press the button “Add user”.



USERS

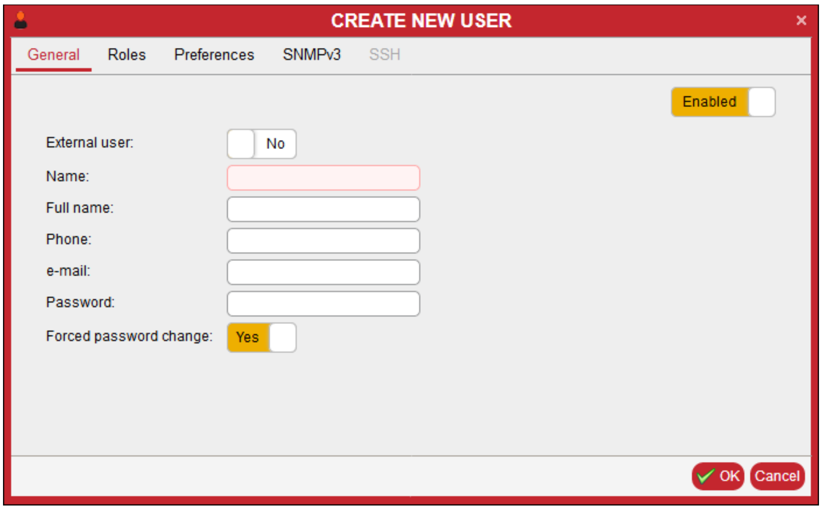
Name	Type	Full name	Phone	e-mail	Failed login(s)
admin	local	SMRC Administrator			
guest	local	SMRC Guest			
lp	external				
ntp	external				
rdmin	external				
root	external				
stefan	local				
sync	external				
user	local	SMRC User			

There are four tabs in the “Create new user” window: “General”, “Roles”, “Preferences”, “SNMPv3”.

The fields “Name” and “Password” are mandatory, other fields are optional.

The **Name** shall only consist of lowercase letters without spaces.

The **Password** must have sufficient complexity, namely, it should be 8 characters or longer, include at least one lowercase letter, one uppercase letter, one digit and one special character.



CREATE NEW USER

Enabled

External user: No

Name:

Full name:

Phone:

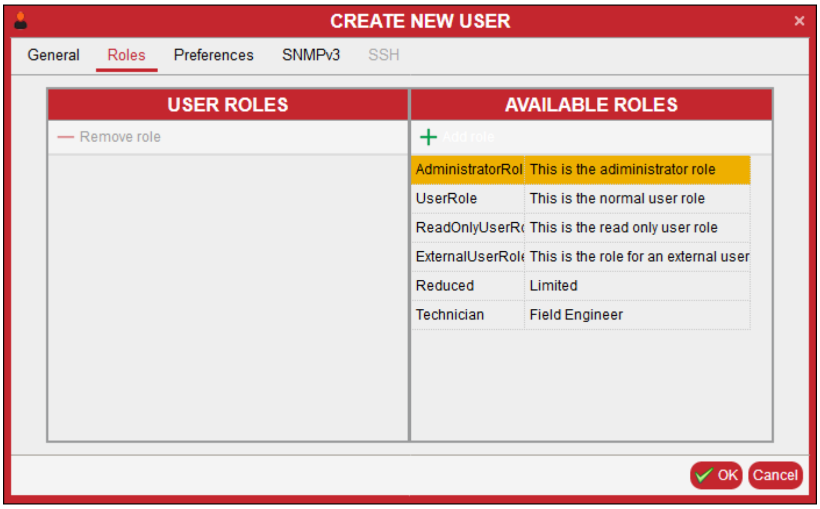
e-mail:

Password:

Forced password change: Yes

11.2 Set roles for a new user

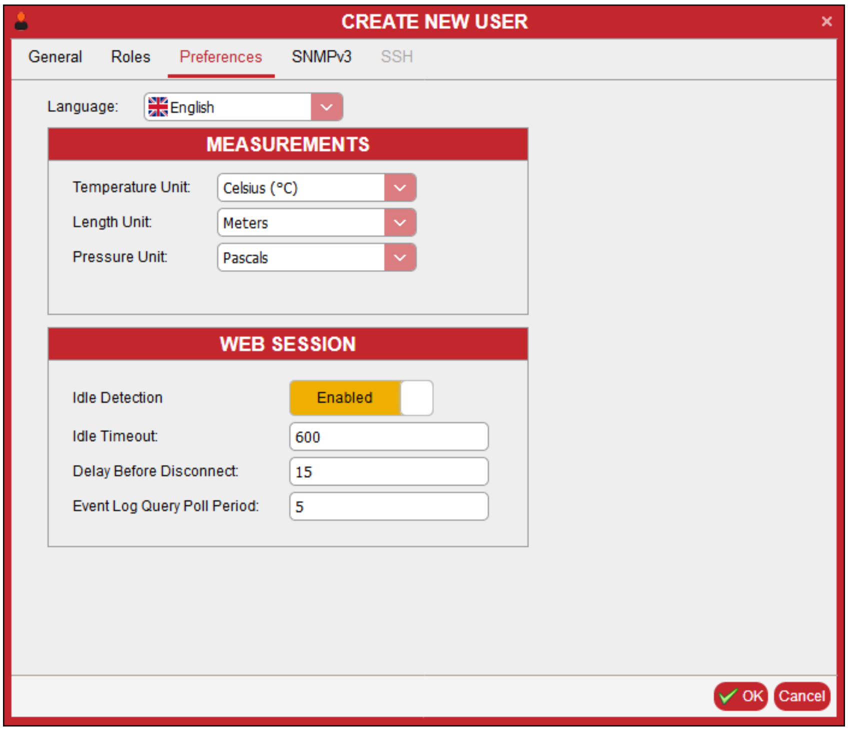
To set roles for a new user use the “Roles” tab. There is no default role.



AdministratorRol	This is the adiministrator role
UserRole	This is the normal user role
ReadOnlyUserR	This is the read only user role
ExternalUserRol	This is the role for an external user
Reduced	Limited
Technician	Field Engineer

11.3 Set preferred measurement units

To set preferred measurement units for a new user use the “Preferences” tab. The default preferred measurement units are suggested.



Language: English

Temperature Unit: Celsius (°C)

Length Unit: Meters

Pressure Unit: Pascals

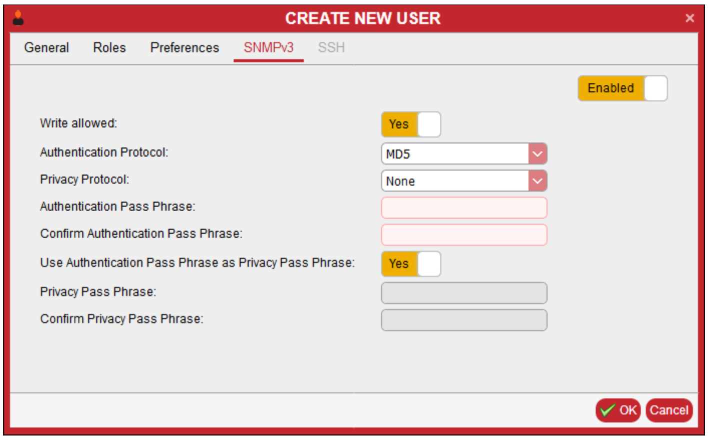
Idle Detection: Enabled

Idle Timeout: 600

Delay Before Disconnect: 15

Event Log Query Poll Period: 5

Optionally, SNMPv3 attributes may be set for a new user in the “SNMPv3” tab.



Enabled

Write allowed: Yes

Authentication Protocol: MD5

Privacy Protocol: None

Authentication Pass Phrase:

Confirm Authentication Pass Phrase:

Use Authentication Pass Phrase as Privacy Pass Phrase: Yes

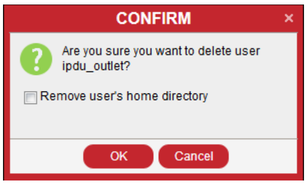
Privacy Pass Phrase:

Confirm Privacy Pass Phrase:

When all the attributes of a new user are set, press the “OK” button and a new user will be created.

11.4 Delete an existing user

From the Web interface, to delete an existing user, invoke the dialog box with the menu items “User Management” -> “Users”, move the cursor to the correspondent line and press the button “Remove user”. The “Confirm” window with two buttons, “OK” and “Cancel”, is generated. If the “Remove user’s home directory” checkbox is checked, the user’s home directory is deleted. In the other (default) case, the directory remains intact.

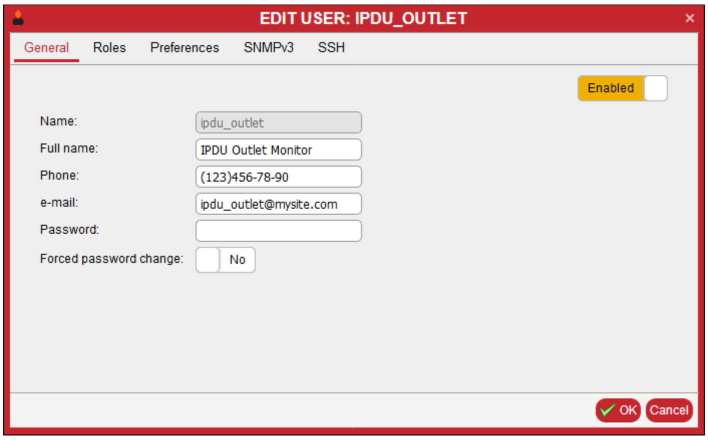

CONFIRM

Are you sure you want to delete user ipdu_outlet?

□ Remove user's home directory

11.5 Edit an existing user

From the Web interface, to edit an existing user, invoke the dialog box with the menu items “User Management” -> “Users”, move the cursor to the correspondent line and press the button “Edit user”. There are 5 tabs in the “Edit user” window: “General”, “Roles”, “Preferences”, “SNMPv3”, “SSH”. The fields “Name” and “Password” are mandatory, other fields are optional.


EDIT USER: IPDU_OUTLET

Enabled

Name: ipdu_outlet

Full name: IPDU Outlet Monitor

Phone: (123)456-78-90

e-mail: ipdu_outlet@mysite.com

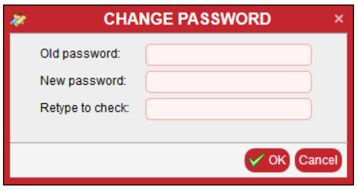
Password:

Forced password change: No

11.6 Lock an user

To lock an user, invoke the dialog box with the menu items “User Management” -> “Users”, move the cursor to the correspondent line and press the button “Lock user.” To unlock a user, invoke the dialog box with the menu items “User Management” -> “Users”, move the cursor to the correspondent line and press the button “Unlock user.”

To change own password, invoke the dialog box with the menu items “User Management” -> “Change Password.”


CHANGE PASSWORD

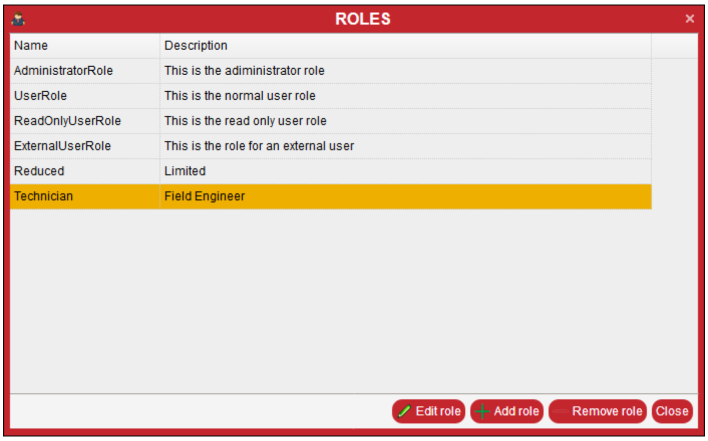
Old password:

New password:

Retype to check:

11.7 Get the list of the roles

From the Web interface, to get the list of the roles, invoke the dialog box with the menu items “User Management” -> “Roles”.

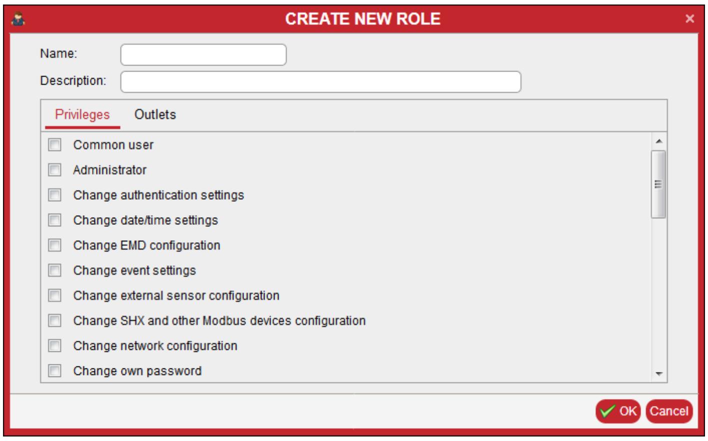


ROLES

Name	Description
AdministratorRole	This is the adiministrator role
UserRole	This is the normal user role
ReadOnlyUserRole	This is the read only user role
ExternalUserRole	This is the role for an external user
Reduced	Limited
Technician	Field Engineer

11.8 Create a new role

To create a new role, press the button “Add role”.



CREATE NEW ROLE

Name:

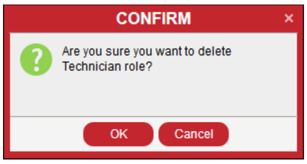
Description:

- □ Common user
- □ Administrator
- □ Change authentication settings
- □ Change date/time settings
- □ Change EMD configuration
- □ Change event settings
- □ Change external sensor configuration
- □ Change SHX and other Modbus devices configuration
- □ Change network configuration
- □ Change own password

Then, fill the fields “Name” and “Description”. For every privilege there is a corresponding checkbox in the window. Check boxes that correspond to the set of privileges of the role and press the button “OK”.

11.9 Delete an existing role

To delete an existing role, move the cursor over the corresponding line and press the button “Remove role”. The “Confirm” window with two buttons, “OK” and “Cancel”, is generated.

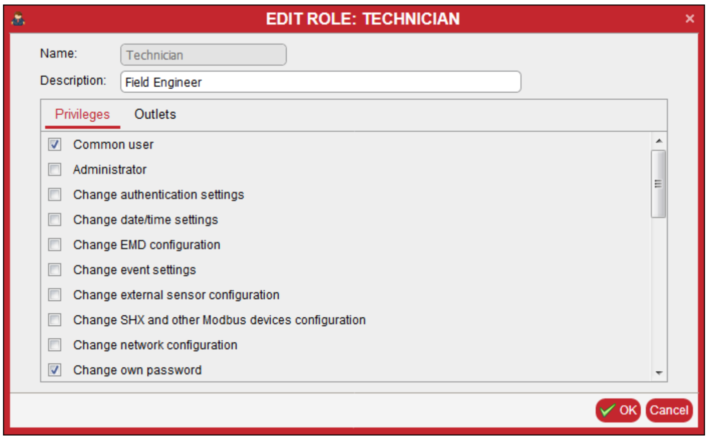


CONFIRM

Are you sure you want to delete Technician role?

11.10 Edit an existing role

To edit an existing role, move the cursor over the corresponding line and press the button “Edit role”.



EDIT ROLE: TECHNICIAN

Name: Technician

Description: Field Engineer

☑ Common user
□ Administrator
□ Change authentication settings
□ Change date/time settings
□ Change EMD configuration
□ Change event settings
□ Change external sensor configuration
□ Change SHX and other Modbus devices configuration
□ Change network configuration
☑ Change own password

Change the description and the set of privileges of the role and press the button “OK”.

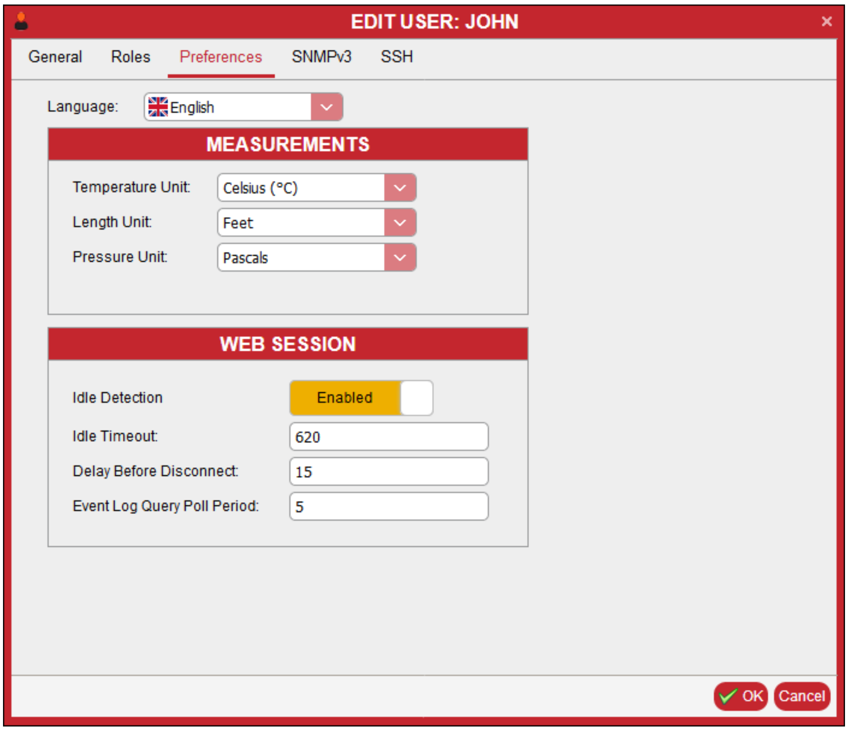
11.11 Preferred Measurement Units

Information about preferred measurement units includes the following:

- Temperature units: Celsius or Fahrenheit degrees
- Length units: Metric (meters) or English (feet)
- Pressure units: Pascal or PSI.

For a new user, measurement units are inherited from global parameters.

From the Web interface, to view and set measurement units for the current user, invoke the dialog box with the menu items “User Management” -> “User Preferences”. For an arbitrary user, choose the user from the list of users (invoked with menu items “User Management” -> “Users”), press the button “Edit user” and choose the tab “Preferences”. In both cases, the same dialog box appears, it gives access to both measurement units and Web session preferences.



Language: English

Temperature Unit: Celsius (°C)

Length Unit: Feet

Pressure Unit: Pascals

Idle Detection: Enabled

Idle Timeout: 620

Delay Before Disconnect: 15

Event Log Query Poll Period: 5

11.12 Web Session Preferences

For the Web session preferences, the following information is included:

- Whether idle detection is enabled: *TRUE* or *FALSE*
- Idle detection timeout, in seconds
- Delay before disconnecting the session after idle state is detected (and the corresponding warning is shown), in seconds
- The period to poll the System Event Log, in seconds.

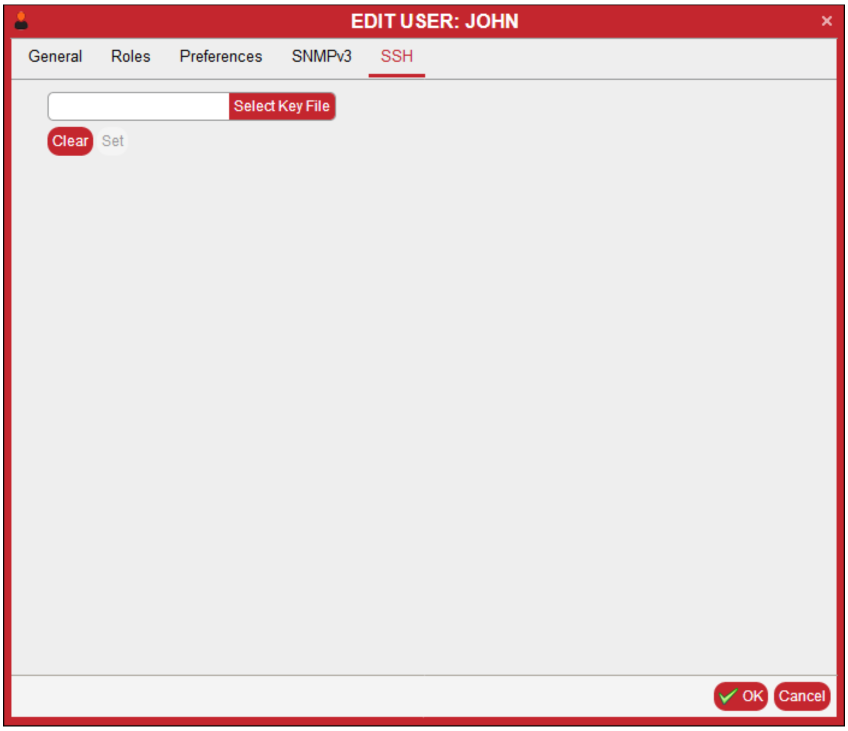
For a new user, Web session parameters are inherited from global parameters.

In the Web interface, Web session preferences appear in the same dialog box as the preferred measurement units (see the previous section).

11.13 SSH public key

SSH supports user authentication by the private/public key pair, without need for entering a password. In that case, a pair of keys is generated for the user on some system, and the public key is stored in a user-specific location `~/.ssh/authorized_keys` on the Guardian Management Gateway file system. During authentication, the SSH client verifies the match of the user's private key (stored on the client) with the user's public key, stored on the Guardian Management Gateway. If the match is successful, the user is authenticated.

To store an SSH public key for the current user on the Guardian Management Gateway with the Web interface, invoke the corresponding dialog via menu: "User Management" -> "Change User SSH Key". To do the same for an arbitrary user, choose the user via "User Management" -> "Users", press the "Edit User" button and choose the "SSH" tab. In both cases, the dialog looks the same:



The key should be present in a local file; press the "Choose Key File" button to choose the file, and then press the "Set" button to store the key on the Guardian Management Gateway. Press the "Clear" button to delete all public keys stored on the Guardian Management Gateway for the given user.

11.14 SNMPv3 User Settings

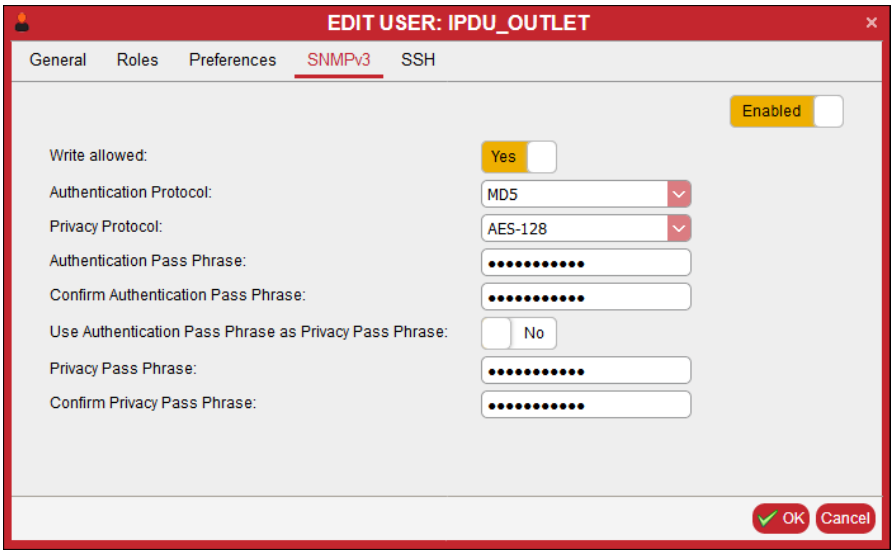
SNMP protocol version 3 implements mandatory user authentication. Therefore, if an external SNMP client communicates with the Guardian Management Gateway using version 3 of the protocol, it should first authenticate itself on the Guardian Management Gateway with some user identity. SNMPv3 user identities on the Guardian Management Gateway match regular user identities; for each user, SNMPv3 identity attributes can be specified. Also, it is possible to configure the SNMP server on the Guardian Management Gateway so that only version 3 of the SNMP protocol is supported; in that case, user authentication is mandatory for any SNMP client.

The following SNMPv3 related settings exist for a user:

- Whether SNMPv3 access is enabled for the user
- Whether the user has read/write (or read/only) access in SNMP terms
- The protocol used for authentication (None, MD5 or SHA1)
- The protocol user for encryption (None, DES, or AES128)
- The passphrase used for authentication
- The passphrase used for encryption

By default, a new or built-in user is not SNMPv3 enabled; an administrator needs to explicitly set SNMPv3-related parameters for each new user, and for built-in users if necessary.

In the Web interface, SNMPv3 user settings can be viewed and changed on the “SNMPv3” tab of the “Edit User” tabbed dialog. This dialog can be invoked by choosing menu items “User Management” -> “Users”, choosing the user and pressing the button “Edit user”.



Write allowed: Yes

Authentication Protocol: MD5

Privacy Protocol: AES-128

Authentication Pass Phrase: ●●●●●●●●●●●

Confirm Authentication Pass Phrase: ●●●●●●●●●●●

Use Authentication Pass Phrase as Privacy Pass Phrase: No

Privacy Pass Phrase: ●●●●●●●●●●●

Confirm Privacy Pass Phrase: ●●●●●●●●●●●

After editing is complete, press the “OK” button to save the SNMPv3 settings for the user.

12 Device Management

This section describes managing global user interface preferences (measurement units, Web interface properties, language), managing other global attributes, viewing device make, model and version and managing time attributes (date, time and time zone).

12.1 Global User Interface Preferences and Other Global Attributes

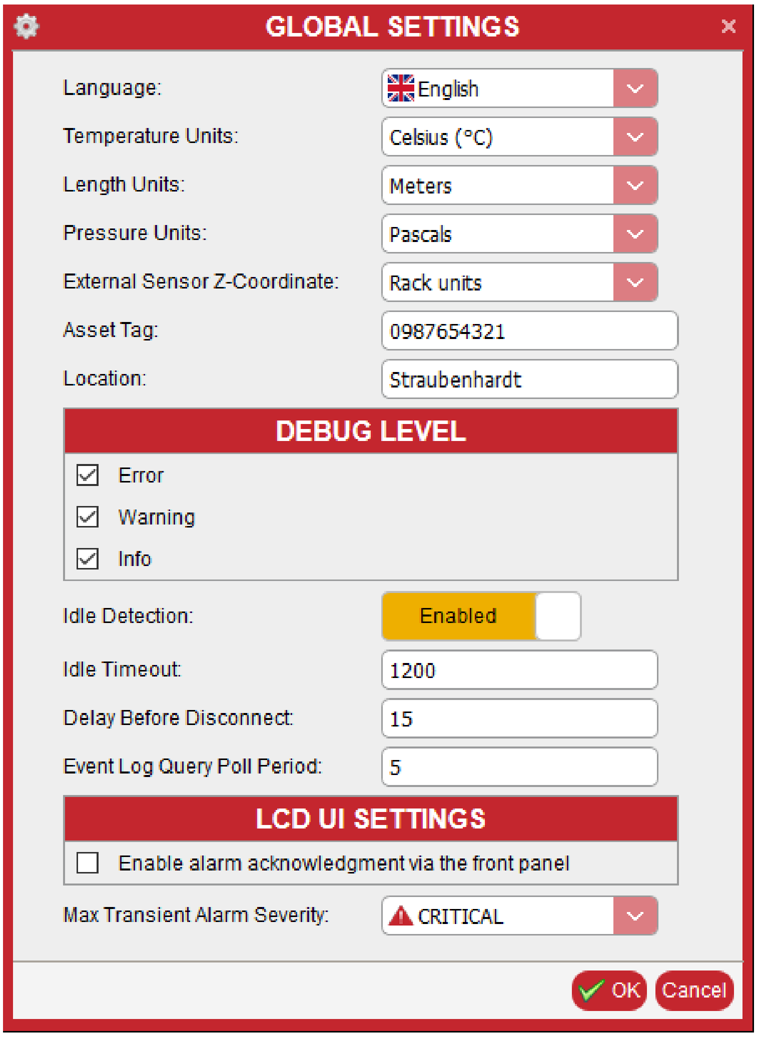
Global user interface preferences are used as defaults assigned to a new user when a new user is created. Also they are used in the contexts where no logged in user exists (e.g. on the LCD interface to the Guardian Management Gateway). They the following items:

- Measurement units:
  - Temperature units: Celsius or Fahrenheit degrees
  - Length units: meters or feet
  - Pressure units: Pascal or PSI.
- Web interface preferences:
  - Whether idle detection is enabled: *TRUE* or *FALSE*
  - Idle detection timeout, in seconds
  - Delay before disconnecting the session after idle state is detected (and the corresponding warning is shown), in seconds
  - The period to poll the System Event Log, in seconds.
- Interface language (English, German, French)
- LCD User Interface flags (an opaque integer number).

Other global attributes include the following items:

- Debug level: the bit mask that indicates the verbosity of log messages that are posted in the system log file */var/log/messages*, the bits have the following meaning:
  - Bit 0 (mask 1): error level
  - Bit 1 (mask 2): warning level
  - Bit 2 (mask 4): informational level
  - Bit 3 (mask 8): verbose level.
- Z-coordinate type for managed sensors: a two-state flag, can have values “Rack Units” or “Arbitrary text”
- Maximum Transient Alarm Severity: the severity level can be one of “Critical”, “Major”, “Minor”, “Informational” or “OK”. If this value is set to a level less than “Critical”, then more severe alarms are not automatically deleted from the Alarm table when the alarm condition goes away. They stay in the Alarm table until manually deleted by the user or until a restart of the iPDU.

To manage global user interface preferences and other global attributes with the Web interface, use the dialog box "Global Settings" that can be invoked with the menu command "Device Settings" -> "Settings". This dialog box allows the user to view and change attributes of both kinds described above:



GLOBAL SETTINGS

Language: English

Temperature Units: Celsius (°C)

Length Units: Meters

Pressure Units: Pascals

External Sensor Z-Coordinate: Rack units

Asset Tag: 0987654321

Location: Straubenhardt

DEBUG LEVEL

☑ Error

☑ Warning

☑ Info

Idle Detection: Enabled

Idle Timeout: 1200

Delay Before Disconnect: 15

Event Log Query Poll Period: 5

LCD UI SETTINGS

☐ Enable alarm acknowledgment via the front panel

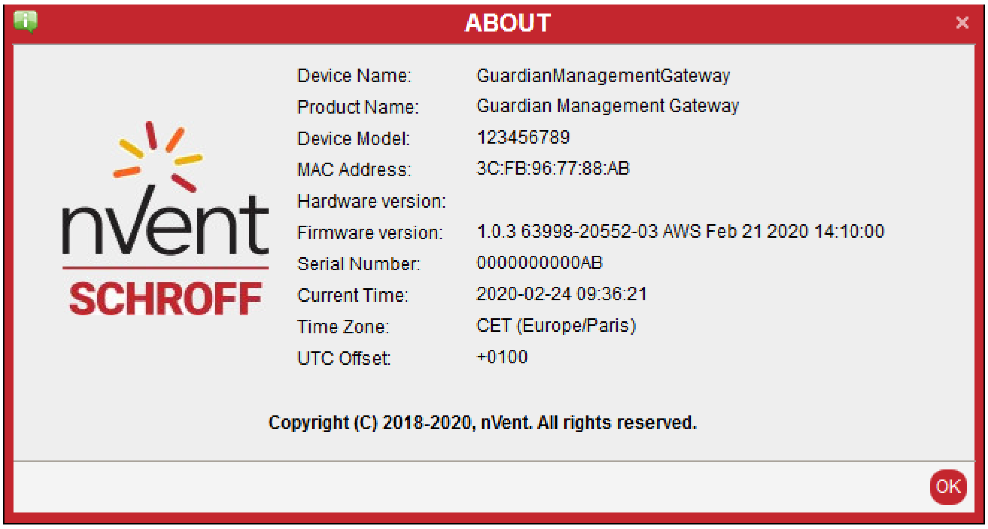
Max Transient Alarm Severity: CRITICAL

12.2 Device attributes, date and time

The device attributes are read-only; they include:

- Device name
- Device model
- Serial number of the device
- Hardware version
- Software (firmware) version: it includes the application version, Schroff version of the image, versions of separate image components (U-Boot, Linux kernel and the root file system).

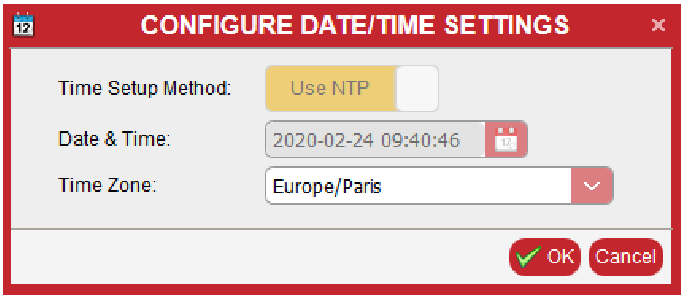
In the Web interface, device attributes and the current time are shown in the "**ABOUT**" dialog, which is invoked with the menu command "About":


ABOUT

Device Name: GuardianManagementGateway
Product Name: Guardian Management Gateway
Device Model: 123456789
MAC Address: 3C:FB:96:77:88:AB
Hardware version:
Firmware version: 1.0.3 63998-20552-03 AWS Feb 21 2020 14:10:00
Serial Number: 0000000000AB
Current Time: 2020-02-24 09:36:21
Time Zone: CET (Europe/Paris)
UTC Offset: +0100

Copyright (C) 2018-2020, nVent. All rights reserved.

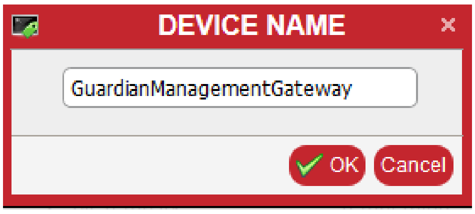
To change the current time and time zone with the Web interface, use the "Configure Date/Time Settings" dialog, which is invoked with the menu command "Device Settings" -> "Date/Time".


CONFIGURE DATE/TIME SETTINGS

Time Setup Method: Use NTP
Date & Time: 2020-02-24 09:40:46
Time Zone: Europe/Paris

In this dialog, the user can set the date, time (if it is set manually), and choose the time zone. In the picture above, date and time are set via NTP.

To change the device name, use the "Device Name" dialog, which is invoked with the menu command "Device Settings" -> "Device Name". Spaces are not allowed in the device name.


DEVICE NAME

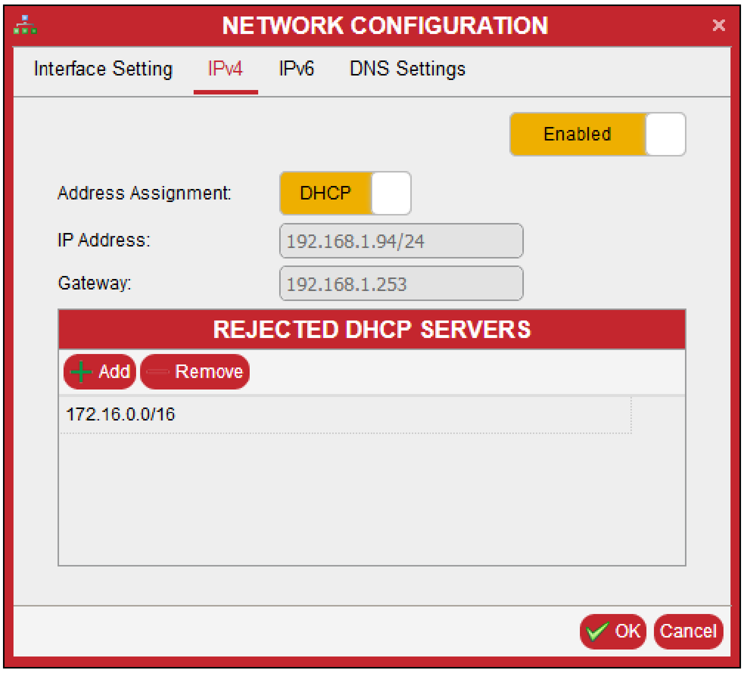
GuardianManagementGateway

13 Network Configuration

Network configuration consists of the following parts:

- Network adapter configuration – applicable to wired Ethernet adapters (except for getting MAC address which applies to all network adapters)
- IPv4 address, subnet mask and default IPv4 gateway assignment – applies to each supported network adapter separately
- List of IPv6 addresses, subnet mask and default IPv6 gateway assignment – applies to each supported network adapter separately
- IPv4 and IPv6 DNS server configuration – applies to the whole system
- Additional DNS attributes: host name and DNS domain search path – apply to the whole system
- List of DHCPv4 and DHCPv6 rejected servers - applies to the whole system

To view and edit network configuration in general with the Web interface, use the tabbed dialog box “Network Configuration” invoked via the menu command “Device Settings” -> “Network”. Different tabs give access to different parts of the network configuration.



Address Assignment: DHCP

IP Address: 192.168.1.94/24

Gateway: 192.168.1.253

172.16.0.0/16

13.1 Network adapter configuration

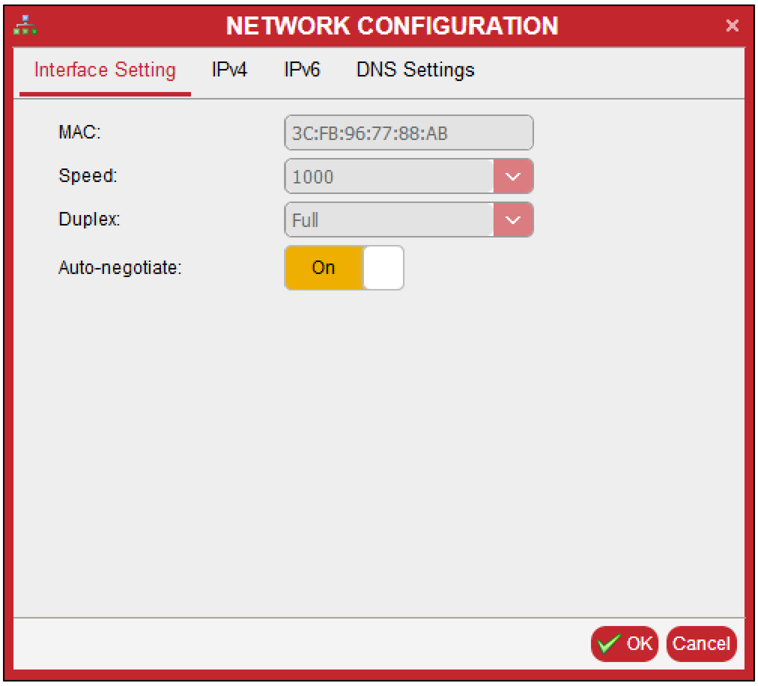
For each network adapter, the following low-level attributes can be configured:

- MAC address (read-only)
- Interface mode and speed (applicable on `eth0`):
  - Auto-negotiate flag: true or false
  - Duplex mode: half-duplex or full-duplex
  - Speed: 10 Mbit/s or 100 Mbit/s (higher speeds are set as 100 Mbit/s).

If the auto-negotiate flag is set to true, interface speed and duplex mode are set up automatically from the transmission media; manual settings are ignored.

By default, auto-negotiation on `eth0` is turned on, so duplex mode and speed are automatically assigned.

With the Web interface, low-level attributes for `eth0` can be edited on the "Interface Setting" tab of the "Network Configuration" dialog box.



MAC: 3C:FB:96:77:88:AB

Speed: 1000

Duplex: Full

Auto-negotiate: On

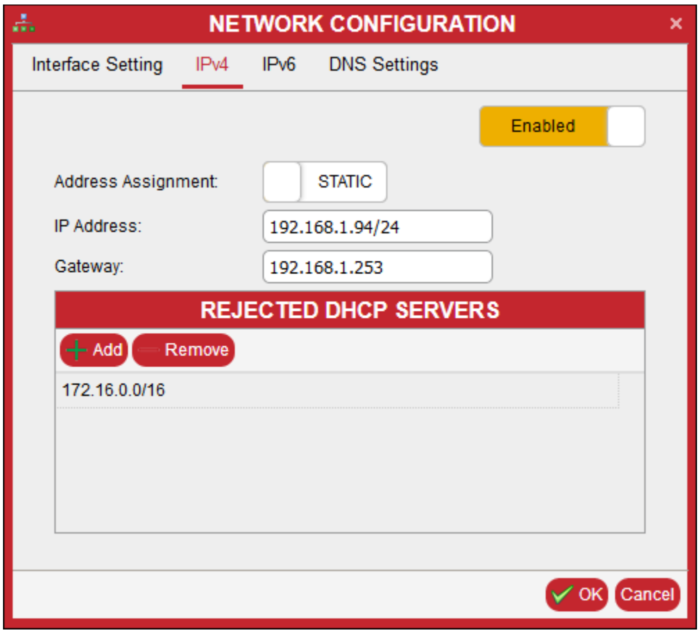
13.2 IPv4 configuration

For each network adapter, the following IPv4 configuration attributes can be configured:

- IPv4 address of the Guardian Management Gateway
- Subnet mask
- Default IPv4 gateway address
- Address assignment type (static or DHCP).

By default, the IPv4 address, subnet mask and the default gateway address are assigned automatically from DHCP. The user should set the assignment type to static to assign these attributes manually.

With the Web interface, IPv4 configuration attributes for an interface can be edited on the "IPv4 Settings" tab of the "Network Configuration" dialog box.



Address Assignment: STATIC

IP Address: 192.168.1.94/24

Gateway: 192.168.1.253

172.16.0.0/16

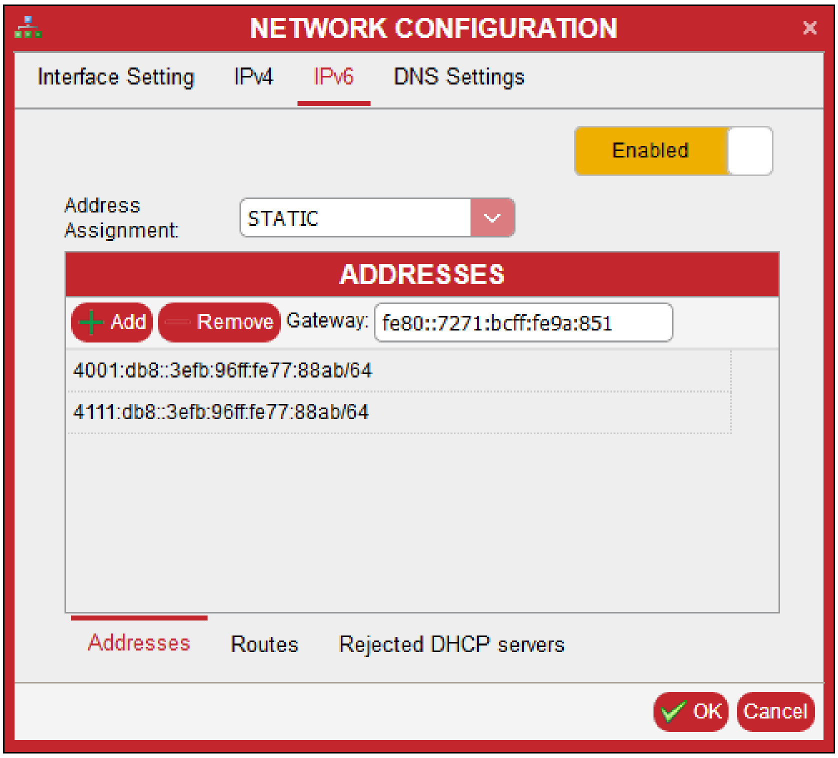
13.3 IPv6 configuration

For each network adapter, the following IPv6 configuration attributes can be configured:

- List of IPv6 addresses with subnet prefixes
- Default IPv6 gateway address.

By default, no IPv6 addresses are configured for a network adapter; an IPv6 address with the link scope is usually auto-configured for each network adapter by the system.

With the Web interface, IPv6 configuration attributes for an interface can be edited on the "IPv6 Settings" tab of the "Network Configuration" dialog box. Besides the IPv6 default gateway, this dialog box also shows the list of currently assigned IPv6 addresses for the given iPDU network interface.



Enabled

Address Assignment: STATIC

Gateway: fe80::7271:bcff:fe9a:851

4001:db8::3efb:96ff:fe77:88ab/64

4111:db8::3efb:96ff:fe77:88ab/64

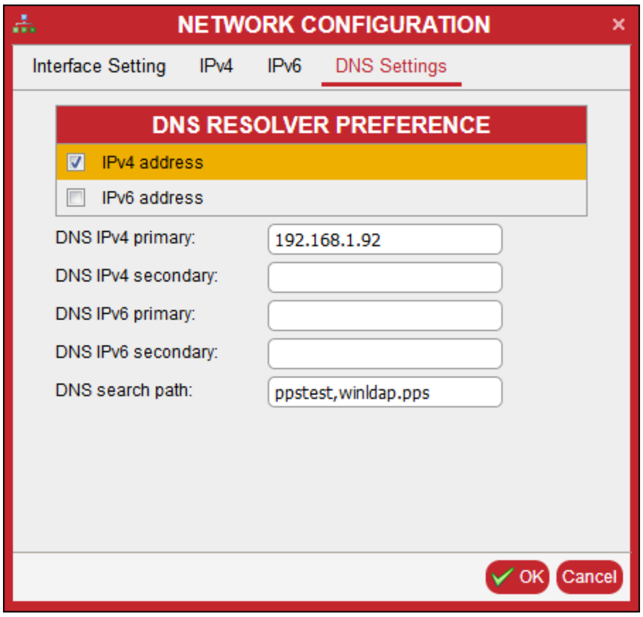
13.4 DNS server configuration

These configuration attributes specify the location of the DNS server; they are system-wide, but server addresses are defined separately for IPv4 and IPv6 protocols and a choice can be made between them:

- DNS resolver preference flag: which DNS settings to prefer, IPv4 or IPv6?
- IPv4 address of the primary DNS server
- IPv4 address of the secondary DNS server
- IPv6 address of the primary DNS server
- IPv6 address of the secondary DNS server.

By default, in the case of automatic IPv4 address assignment, IPv4 information takes preference and DNS server addresses are provided by the corresponding DHCPv4 server.

With the Web interface, DNS server addresses can be edited on the "DNS Settings" tab of the "Network Configuration" dialog box.



☑ IPv4 address
□ IPv6 address

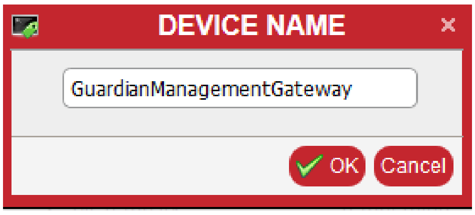
DNS IPv4 primary: 192.168.1.92
DNS IPv4 secondary:
DNS IPv6 primary:
DNS IPv6 secondary:
DNS search path: ppstest,winldap.pps

13.5 Additional configurable DNS attributes

These attributes are system-wide and include the following:

- Guardian Management Gateway host name
- DNS domain search path.

By default, these attributes are not set, but the DHCP server can provide the host name.



GuardianManagementGateway

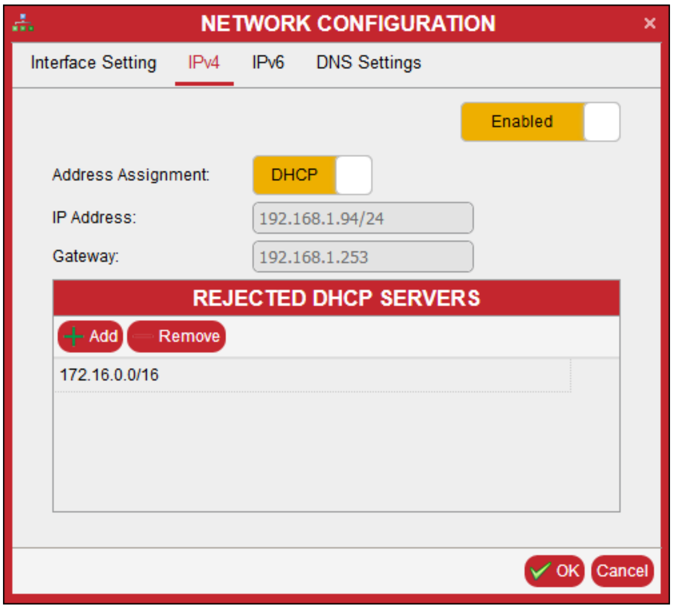
Device Name can be changed under Device Settings with option Device Name.

13.6 List of rejected DHCP servers

In some cases, when DHCP is used, it may be necessary to avoid accepting configuration from certain DHCP servers, which are available in the local network. It is possible to configure the list of rejected DHCP server addresses (both for DHCPv4 and for DHCPv6); the Guardian Management Gateway will not accept configuration parameters from these servers.

Two address lists can be configured: the list of IPv4 addresses for DHCPv4 and the list of IPv6 addresses for DHCPv6.

With the Web interface, the list of rejected DHCPv4 servers can be edited on the “IPv4” tab of the “Network Configuration” dialog box.



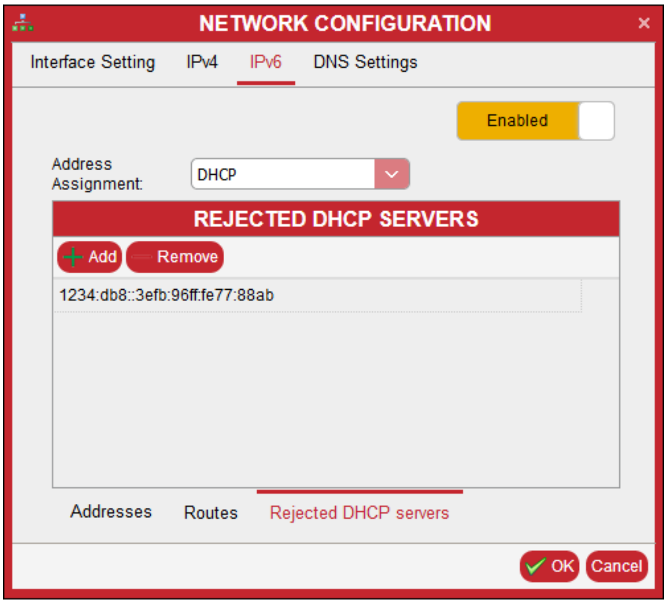
Address Assignment: DHCP

IP Address: 192.168.1.94/24

Gateway: 192.168.1.253

172.16.0.0/16

The list of rejected DHCPv6 servers can be edited on the “IPv6” tab of the “Network Configuration” dialog box.

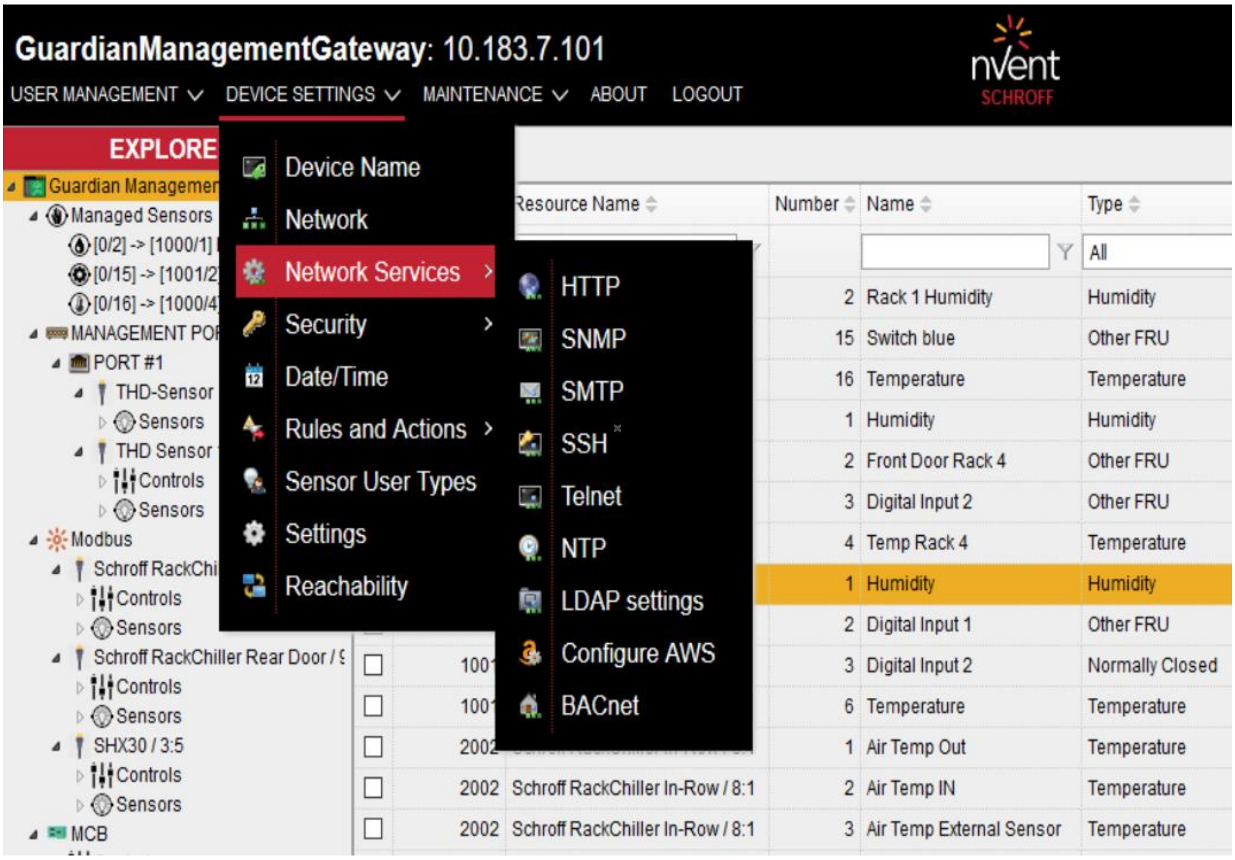


Address Assignment: DHCP

1234:db8::3efb:96ff:fe77:88ab

14 Network Service Configuration

The user via the web interface can configure several network services provided by Guardian Management Gateway or CLI commands. These services include HTTP/HTTPS, Telnet, SSH, SMTP, SNMP and NTP.



		Resource Name	Number	Name	Type
					All
			2	Rack 1 Humidity	Humidity
			15	Switch blue	Other FRU
			16	Temperature	Temperature
			1	Humidity	Humidity
			2	Front Door Rack 4	Other FRU
			3	Digital Input 2	Other FRU
			4	Temp Rack 4	Temperature
			1	Humidity	Humidity
			2	Digital Input 1	Other FRU
☐	100		3	Digital Input 2	Normally Closed
☐	100		6	Temperature	Temperature
☐	2002		1	Air Temp Out	Temperature
☐	2002	Schroff RackChiller In-Row / 8:1	2	Air Temp IN	Temperature
☐	2002	Schroff RackChiller In-Row / 8:1	3	Air Temp External Sensor	Temperature

14.1 HTTP/HTTPS configuration

HTTP and HTTPS network services provide Web interface to the Guardian Management Gateway. On the Guardian Management Gateway, the web server program `lighttpd` provides these services. HTTPS, unlike HTTP, provides secure access to the Guardian Management Gateway over the Web interface, the corresponding traffic is encrypted.

For HTTP and HTTPS, the user can configure the following parameters:

HTTP port

HTTPS port

Enforce HTTPS (a logical flag, if *TRUE*, only secure access is allowed to the Guardian Management Gateway).

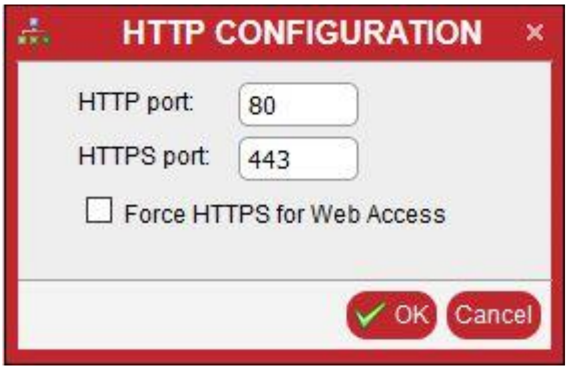
Default settings:

HTTP port = 80

HTTPS port = 443

Enforce HTTPS = False

To change the HTTP/HTTPS configuration, invoke the menu command “Device Settings” -> “Network Services” -> “HTTP”. The “HTTP Configuration” dialog appears.



HTTP port: 80

HTTPS port: 443

□ Force HTTPS for Web Access

14.2 SNMP Configuration

SNMP service provides SNMP interface to the Guardian Management Gateway. On the Guardian Management Gateway, the SNMP server program `snmpd` provides this service.

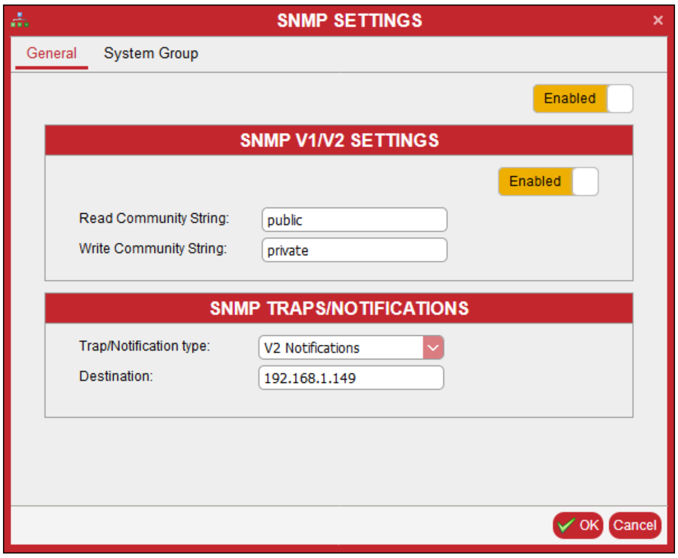
For SNMP, the user can configure the following parameters:

- Whether SNMP service is enabled (`TRUE`/`FALSE`)
- Whether SNMP v1/v2 legacy protocols are enabled; if false, only secure SNMPv3 protocol can be used to communicate to the Guardian Management Gateway over the SNMP interface
- Read community string
- Write community string
- The “Sys Name” string
- The “Sys Contact” string
- The “Sys Location” string
- IP address of the SNMP trap destination system
- Whether to use SNMPv2 format for SNMP traps (if `FALSE`, SNMPv1 format is used).

Default settings:

- SNMP service is enabled = `TRUE`
- SNMP v1/v2 legacy protocols are enabled = `TRUE`
- Read community string = `public`
- Write community string = `private`
- The “Sys Name” string = “” (empty string)
- The “Sys Contact” string = `root`
- The “Sys Location” string = `unknown`
- IP address of the SNMP trap destination system = not specified
- Whether to use SNMPv2 format for SNMP traps = `TRUE`.

To change the SNMP parameters, invoke the menu command “Device Settings” -> “Network Services” -> “SNMP”. The “SNMP Settings” dialog appears. There are two tabs in this dialog box: “General” and “System Group”.



SNMP V1/V2 SETTINGS

Enabled

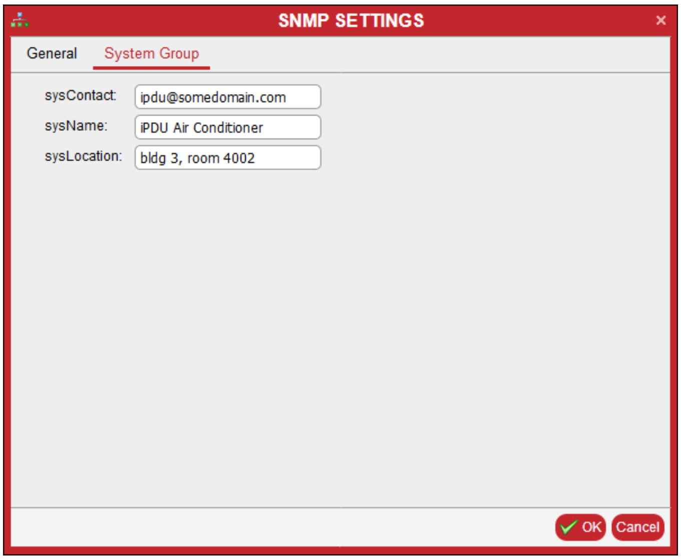
Read Community String: public

Write Community String: private

SNMP TRAPS/NOTIFICATIONS

Trap/Notification type: V2 Notifications

Destination: 192.168.1.149



sysContact: ipdu@somedomain.com

sysName: iPDU Air Conditioner

sysLocation: bldg 3, room 4002

14.3 SMTP Configuration

SMTP service allows sending e-mail. On the Guardian Management Gateway, the SMTP client exists that is able to connect to an external SMTP server and send e-mail messages. Sending e-mail is one of the actions that can be specified in event filtering. In that case, the message body is constructed on the base of the event just received.

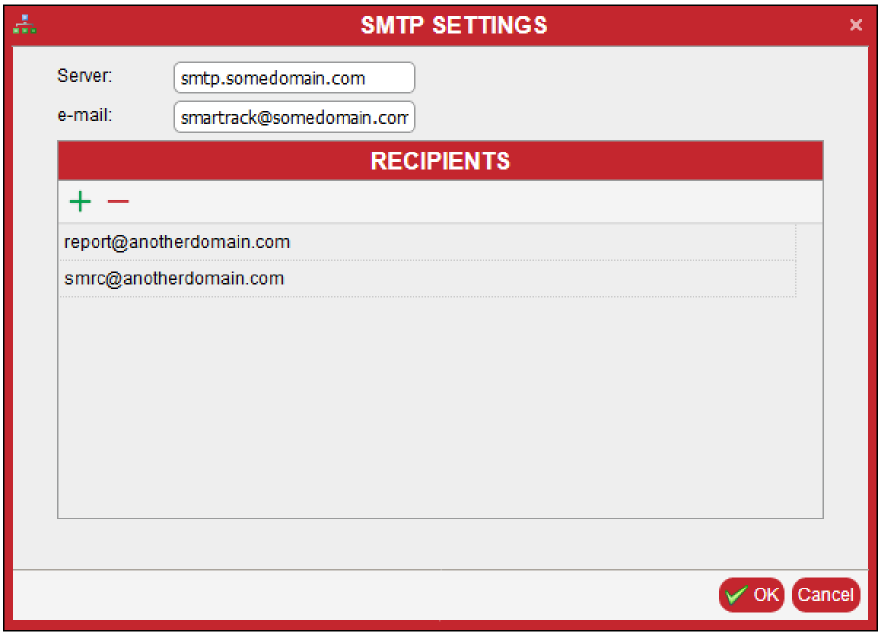
The user can configure the following SMTP parameters:

- Guardian Management Gateway own e-mail address
- Name or IP address of the SMTP server
- The default list of recipient e-mail addresses (comma-separated).

These parameters are specified once for all event filters; other parameters, like the e-mail subject line and the actual list of recipients, are specified as parameters for a specific action in a specific event filter.

By default, these parameters are not specified (empty strings).

To change the SMTP parameters, invoke the menu command "Device Settings" -> "Network Services" -> "SMTP". The "SMTP Settings" dialog appears.



SMTP SETTINGS

Server: smtp.somedomain.com

e-mail: smartrack@somedomain.com

RECIPIENTS

report@anotherdomain.com

smrc@anotherdomain.com

14.4 SSH Configuration

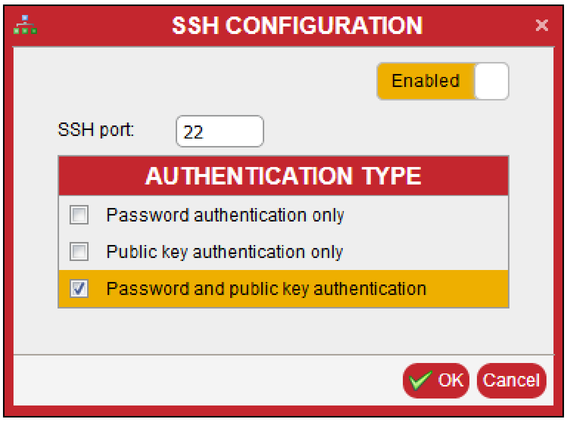
SSH service allows secure terminal access to an Guardian Management Gateway; SSH traffic is encrypted in transit. SSH protocol is the preferred instrument for terminal access to an Guardian Management Gateway. On the Guardian Management Gateway, SSH service is provided by the `sshd` daemon.

For SSH, the user can configure the following parameters:

- Whether SSH service is enabled
- SSH port
- Supported SSH authorization methods: by password, by public key or both.

By default, SSH service is enabled on port 22, with both authorization methods (password and public key) supported.

To change the SSH configuration, invoke the menu command "Device Settings" -> "Network Services" -> "SSH". The "SSH Configuration" dialog appears.



SSH CONFIGURATION

Enabled

SSH port: 22

AUTHENTICATION TYPE

□ Password authentication only

□ Public key authentication only

☑ Password and public key authentication

14.5 Telnet Configuration

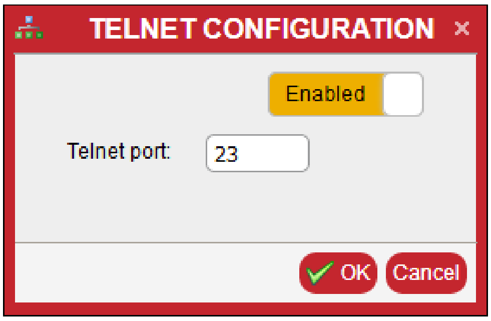
Telnet service allows terminal access to an Guardian Management Gateway. Telnet protocol is not secure, so SSH protocol is the preferred instrument for terminal access to an Guardian Management Gateway. On the Guardian Management Gateway, Telnet service is provided by the `telnetd` daemon.

For Telnet, the user can configure the following parameters:

- Whether Telnet service is enabled
- Telnet port.

By default, Telnet service is enabled on port 23.

To change the Telnet configuration, invoke the menu command "Device Settings" -> "Network Services" -> "Telnet". The "Telnet Configuration" dialog appears.



TELNET CONFIGURATION

Enabled

Telnet port: 23

14.6 NTP Configuration

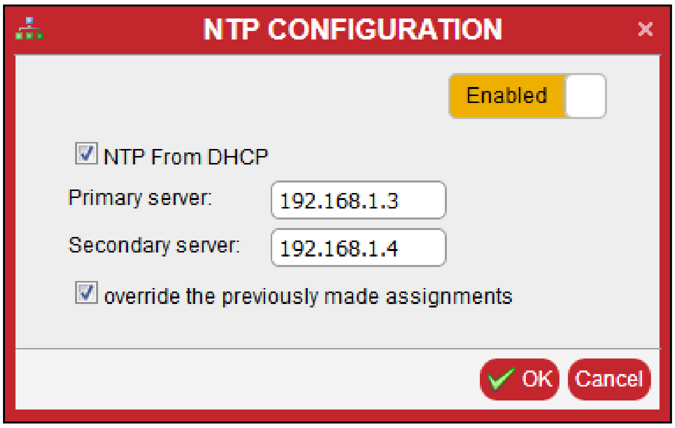
NTP service allows time synchronization with external servers. On the Guardian Management Gateway, the NTP client exists that is able to connect to an external NTP server and obtain current time from it.

The user can configure the following NTP parameters:

- Whether NTP client functionality is enabled (if *FALSE*, system time must be manually set by the administrator)
- Whether NTP client should obtain NTP server addresses via DHCP (if *FALSE*, NTP server addresses must be set manually)
- Name or IP address of the primary NTP server (if obtaining via DHCP is disabled)
- Name or IP address of the secondary NTP server (if obtaining via DHCP is disabled).

By default, NTP client is enabled and obtaining NTP server addresses via DHCP is enabled; primary and secondary NTP server addresses are not set.

To change the NTP configuration, invoke the menu command “Device Settings” -> “Network Services” -> “NTP”. The “NTP Configuration” dialog appears.



NTP CONFIGURATION

Enabled

☑ NTP From DHCP

Primary server: 192.168.1.3

Secondary server: 192.168.1.4

☑ override the previously made assignments

15 LDAP Configuration

Guardian Management Gateway supports authenticating users via LDAP. If this method is used, user records are stored on a remote server and user authentication on the Guardian Management Gateway involves communication with that server. The Guardian Management Gateway itself may not have information about the user that intends to log in; if this is the case and remote authentication is successful, this user is considered an "external user" and user information record with default attributes is created on the Guardian Management Gateway for that user; in particular, that user is assigned the predefined role "ExternalUserRole", that defines its privileges with respect to the Guardian Management Gateway.

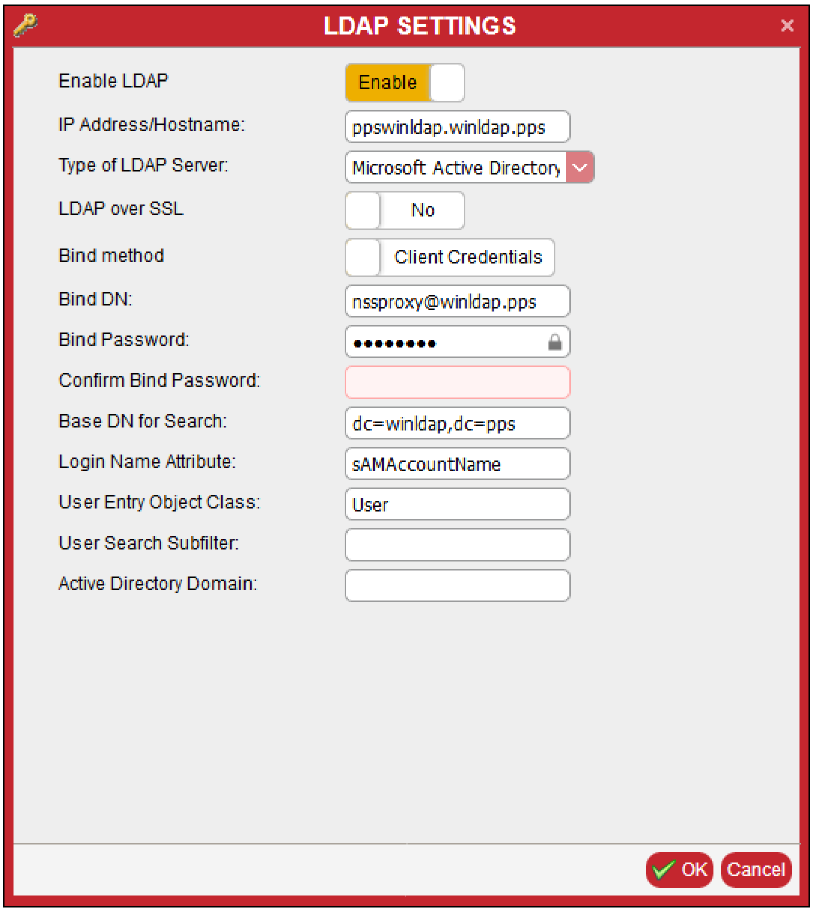
LDAP configuration parameters, that a user can view and set, include the following:

- Whether logging in via LDAP is enabled (*TRUE*/*FALSE*)
- LDAP server name (URI)
- Server type (OpenLDAP or ActiveDirectory, other parameters may need to be set differently depending on the server type)
- Whether to use SSL for connection to the LDAP server
- SSL port number (if SSL is used)
- SSL certificate for the server
- Whether the client uses an anonymous bind to the LDAP server to authenticate a user
- Distinguished name used for binding (only if anonymous bind is not used)
- Password used for binding (only if anonymous bind is not used)
- Distinguished name used as search base
- Login name attribute (normally "sAMAccountName" for ActiveDirectory servers or empty for OpenLDAP servers)
- User entry object class (normally "User")
- User search subfilter
- Extra configuration options (as a sequence of strings in the format `<option> <value>`, separated by the newline characters)

By default, LDAP-based logins are disabled and all other parameters are undefined.

When LDAP is enabled by a user, all LDAP configuration parameters should also be supplied by the user in the same command or dialog box. When LDAP is disabled by a user, all other configuration parameters become undefined and need not be specified.

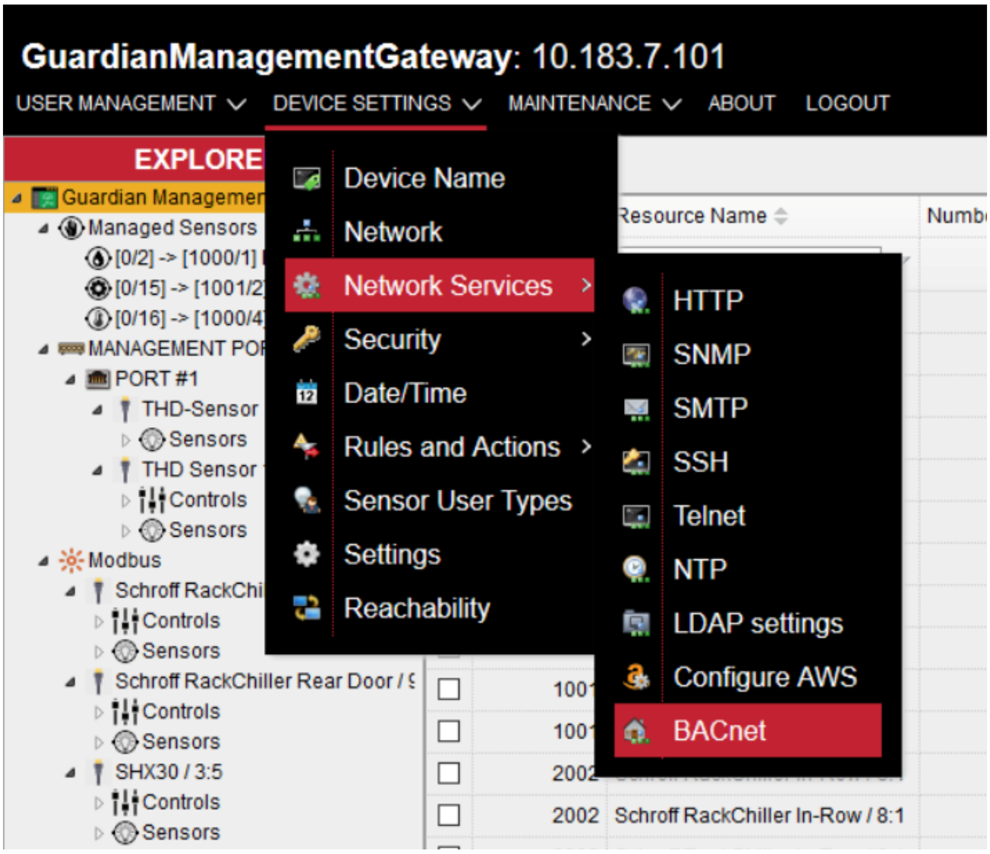
In the Web interface, LDAP is configured via a dialog box that is accessible via menu items “Device Settings” -> “Network Services” -> “LDAP Settings”. This dialog box allows the user to specify all LDAP configuration parameters. For the SSL certificate, local path to the certificate should be specified; the certificate will be downloaded to the fixed place in the Guardian Management Gateway file system.



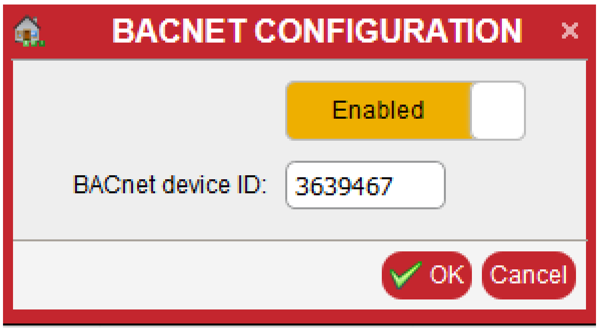
Enable LDAP	Enable
IP Address/Hostname:	ppswinldap.winldap.pps
Type of LDAP Server:	Microsoft Active Directory
LDAP over SSL	No
Bind method	Client Credentials
Bind DN:	nssproxy@winldap.pps
Bind Password:	••••••••
Confirm Bind Password:	
Base DN for Search:	dc=winldap,dc=pps
Login Name Attribute:	sAMAccountName
User Entry Object Class:	User
User Search Subfilter:	
Active Directory Domain:	

16 BACnet

In the Web interface, BACnet is configured via a dialog box that is accessible via menu items “Device Settings” -> “Network Services” -> “BACnet”.



This dialog box allows the user to enable BACnet and change the BACnet device ID.



BACNET CONFIGURATION

Enabled

BACnet device ID: 3639467

16.1 BACnet Overview

HPI objects are mapped to BACnet objects as follows:

HPI	Mapping	BACnet
Guardian Management Gateway	← →	Device Object
Resource	← →	Structured View objects
Sensor	← →	Analog Input, Binary Input and Multi-State Input objects
Control	← →	Analog Output and Binary Output objects
HPI Event	← →	BACnet Event
HPI Alarm Table	← →	Get Alarm Summary service

- A single Guardian Management Gateway device is mapped to the Device object. The Device object instance ID is by default based on the MAC address of the device but can be changed by the user.
- HPI resources are mapped to Structured View objects.
- HPI sensors are mapped to Analog Input, Binary Input and Multi-State Input objects; the corresponding object belongs to the Structured View object which corresponds to the resource – owner of the sensor.
- HPI controls are mapped to Analog Output and Binary Output objects; the corresponding object belongs to the Structured View object which corresponds to the resource – owner of the control.
- HPI events are mapped to BACnet events and are forwarded to BACnet clients via the subscriptions in the Notification Class object.
- HPI Alarm Table is exposed to BACnet via the services Get Alarm Summary, Acknowledge Alarm.
- HPI Event Log is mapped to the BACnet Event Log object (in progress).

16.2 Device Object

A single device object exists for the Guardian Management. This object describes global properties of the device. Object instance for the device object should be unique among all BACnet device objects in the network. By default it is based on the MAC address of the Guardian Management, but can be changed by the user via CLI or web interface.

Properties:

Property Name	Access	Property Source
Object Type	RO	8 = “Device”
Object Instance	RO	Unique number, based on MAC address by default (but can be redefined by the user)
Object name	RO	Device name
Description	RO	Device name
Apdu Timeout	RO	3000
Application Software Version	RO	“1.0”
Database Revision	RO	1
Daylight Saving Status	RO	Based on the current timezone
Device Address Binding	RO	Empty list
Firmware Revision	RO	SMRC firmware version
Local Date	RO	Current system date
Local Time	RO	Current system time

Location	RO	Based on the current timezone
Max Apdu Length Accepted	RO	1476
Model Name	RO	Model name from the inventory
Number Of Apdu Retries	RO	3
Protocol Object Types Supported	RO	The bit string – the mask of supported object types
Protocol Version	RO	1
Protocol Revision	RO	12
Protocol Services Supported	RO	The bit string – the mask of supported protocol services
Segmentation Supported	RO	False (should become True after segmentation is implemented)
System Status	RO	"Operational"
Utc Offset	RO	Based on the current timezone
Vendor Identifier	RO	1094 ("Nvent Thermal Management")
Vendor Name	RO	Manufacturer name from the inventory

16.3 Structured View

HPI Resources are mapped to Structured View objects. All sensors and controls which belong to the resource are mapped to the Subordinate List for this Structured View.

Properties:

PROPERTY NAME	ACCESS	PROPERTY SOURCE
Object Type	RO	29 = "Structured View"
Object Instance	RO	Small integer number, assigned sequentially
Object name	RO	Resource name
Description	RO	Resource name
Subordinate List	RO	List of object identifiers for the mapped sensors/controls which belong to this resource

16.4 Analog Input

HPI analog sensors (sensors that have numeric values) are mapped to Analog Input objects.

Properties:

PROPERTY NAME	ACCESS	PROPERTY SOURCE
Object Type	RO	0 = “Analog Input”
Object Instance	RO	Small integer number, assigned sequentially
Object name	RO	Sensor name
Description	RO	Sensor name
Present Value	RO	Current numeric value of the sensor
Event State	RO	“Normal” if the sensor value is within thresholds “Offnormal” if the sensor value is beyond thresholds
Out of service	RO	False
Units	RO	Sensor units (in BACnet encoding)
Reliability	RO	“No Fault Detected”
Status Flags	RO	Flag “In Alarm” is set if the sensor value is beyond thresholds; otherwise no flags are set

16.5 Binary Input

HPI discrete sensors with two states are mapped to Binary Input objects.

Properties:

PROPERTY NAME	ACCESS	PROPERTY SOURCE
Object Type	RO	3 = "Binary Input"
Object Instance	RO	Small integer number, assigned sequentially
Object name	RO	Sensor name
Description	RO	Sensor name
Polarity	RO	"Normal"
Present Value	RO	0 if the sensor is in the first state; 1 if the sensor is in the second state
Event State	RO	"Offnormal" if the severity of the current sensor state is MINOR, MAJOR or CRITICAL "Normal" otherwise
Out of service	RO	False
Units	RO	Empty (discrete sensors do not have units)
Reliability	RO	"No Fault Detected"
Status Flags	RO	Flag "In Alarm" is set if the severity of the current sensor state is MINOR, MAJOR or CRITICAL; otherwise no flags are set

16.6 Multi-State Input

HPI discrete sensors with more than two states are mapped to Multi-State Input objects.

Properties:

PROPERTY NAME	ACCESS	PROPERTY SOURCE
Object Type	RO	13 = “Multi-State Input”
Object Instance	RO	Small integer number, assigned sequentially
Object name	RO	Sensor name
Description	RO	Sensor name
Present Value	RO	Bit string of sensor states; each bit = 1 if the state is asserted and 0 if the state is not asserted
Event State	RO	“Offnormal” if the severity of any of the asserted states is MINOR, MAJOR or CRITICAL “Normal” otherwise
Out of service	RO	False
Units	RO	Empty (discrete sensors do not have units)
Reliability	RO	“No Fault Detected”
Status Flags	RO	Flag “In Alarm” is set if the severity of any of the asserted states is MINOR, MAJOR or CRITICAL; otherwise no flags are set

16.7 Analog Output

HPI analog and discrete controls are mapped to Analog Output objects.

Properties:

PROPERTY NAME	ACCESS	PROPERTY SOURCE
Object Type	RO	1 = “Analog Output”
Object Instance	RO	Small integer number, assigned sequentially
Object name	RO	Control name
Description	RO	Control name
Present Value	RW	Current numeric value of the control, can be set
Priority Array	RO	The array of priorities and last values assigned to the control at that priority (according to the regular BACnet semantics)
Relinquish Default	RO	The default value for the control (specified in the static attributes of the control)
Event State	RO	“Normal”
Out of service	RO	False
Units	RO	Control units if information about units is available (in BACnet encoding)
Status Flags	RO	No flags are set

16.8 Binary Output

HPI digital controls are mapped to Binary Output objects.

Properties:

PROPERTY NAME	ACCESS	PROPERTY SOURCE
Object Type	RO	4 = “Binary Output”
Object Instance	RO	Small integer number, assigned sequentially
Object name	RO	Control name
Description	RO	Control name
Polarity	RO	“Normal”
Present Value	RW	Control value: 0 for the “Off” state, 1 for the “On” state, can be set
Priority Array	RO	The array of priorities and last values assigned to the control at that priority (according to the regular BACnet semantics)
Relinquish Default	RO	The default value for the control (specified in the static attributes of the control)
Event State	RO	“Normal”
Out of service	RO	False
Units	RO	None (no units are defined for digital controls)
Status Flags	RO	No flags are set

16.9 Mapping HPI events to BACnet events

The following HPI events are mapped to BACnet events:

- Sensor events (both from analog and discrete sensors)
- Software events (auditing, logins, logouts)

Events are dispatched to BACnet clients via subscriptions in the Notification Class object. There is a single Notification Class object for each Guardian Management Gateway instance, with Object Instance = 0. Its implementation is fully provided by the bacnet-stack library. To start receiving events, the client should create a subscription in the Notification Class object.

Event structure fields are specified below separately for each event type

Sensor events from analog sensors (note that both assertion and deassertion events are sent for threshold crossing):

FIELD NAME	FIELD VALUE
Process Identifier	PID of the BACnet server process
Notification Class	0
Object Identifier	Object identifier for the corresponding Analog Input object
Timestamp	Current date and time
Event Type	“Out of Range”
Notify Type	“Alarm” for threshold-crossing assertion events “Event” otherwise
Ack Required	False
From State	“Off Normal” if some thresholds were crossed before the event, “Normal” otherwise
To State	“Off Normal” if some thresholds are crossed after the event, “Normal” otherwise
Exceeding Value	The current value of the sensor
Exceeded Limit	The value of the threshold which has been crossed
Status Flags	Flag “In Alarm” is set if and only if “To State” = “Off Normal”; other flags are cleared
Text	A text string indicating in human-readable form, which threshold is exceeded, and including the sensor value and the threshold value

Sensor events from discrete sensors with two states:

FIELD NAME	FIELD VALUE
Process Identifier	PID of the BACnet server process
Notification Class	0
Object Identifier	Object identifier for the corresponding Binary Input object
Timestamp	Current date and time
Event Type	“Change of State”
Notify Type	“Alarm” for assertion events with event severity = MINOR, MAJOR or CRITICAL “Event” otherwise
Ack Required	False

From State	“Off Normal” if sensor severity state before the event was MINOR, MAJOR or CRITICAL, “Normal” otherwise
To State	“Off Normal” if sensor severity state after the event is MINOR, MAJOR or CRITICAL, “Normal” otherwise
New State	The index of the new sensor state (BINARY_INACTIVE for the first state, BINARY_ACTIVE for the second state)
Status Flags	Flag “In Alarm” is set if and only if “To State” = “Off Normal”; other flags are cleared
Text	Description of the state transition in human-readable form

Sensor events from discrete sensors with more than two states:

FIELD NAME	FIELD VALUE
Process Identifier	PID of the BACnet server process
Notification Class	0
Object Identifier	Object identifier for the corresponding Multi-State Input object
Timestamp	Current date and time
Event Type	“Change of Bit String”
Notify Type	“Alarm” for assertion events with event severity = MINOR, MAJOR or CRITICAL “Event” otherwise
Ack Required	False
From State	“Off Normal” if sensor severity state before the event was MINOR, MAJOR or CRITICAL, “Normal” otherwise
To State	“Off Normal” if sensor severity state after the event is MINOR, MAJOR or CRITICAL, “Normal” otherwise
Referenced Bit String	The current sensor state mask as a bit string
Status Flags	Flag “In Alarm” is set if and only if “To State” = “Off Normal”; other flags are cleared
Text	Description of the state transition in human-readable form

Software events:

FIELD NAME	FIELD VALUE
Process Identifier	PID of the BACnet server process
Notification Class	0
Object Identifier	Object identifier for the Device object
Timestamp	Current date and time
Event Type	“Change of State”
Notify Type	“Event”
Ack Required	False

From State	“Normal”
To State	“Normal”
New State	“Normal”
Status Flags	All flags are cleared
Text	The text portion the original event

16.10 Mapping alarms

The BACnet services Get Alarm Summary and Alarm Acknowledge are implemented by the server and provide direct access to the Guardian Management Gateway alarm table.

In the Get Alarm Summary output, all alarms from one sensor are consolidated into a single BACnet alarm with the alarm state “Transition to Off Normal”, originating from the corresponding BACnet object. The list of consolidated alarms is then provided to the client.

A BACnet alarm is considered unacknowledged, if at least one alarm of those which were consolidated, was unacknowledged.

The fields of the alarm data structure given to the client is shown in the following table.

Alarm fields.

FIELD NAME	FIELD VALUE
Object Identifier	Object identifier for the corresponding Analog Input, Binary Input or Multi-State Input object
Alarm State	“Transition to Off Normal”
Acknowledged Transitions	For the “Transition to Off Normal”: True if all Guardian Management Gateway alarms consolidated into this BACnet alarm are acknowledged, False otherwise For all other transitions: True

The service Alarm Acknowledge is given the following parameters:

- object identifier for the object-originator of the alarm
- the alarm state -must be “Transition to Off Normal”
- the timestamp of the last Get Alarm Summary call.

The server acknowledges all alarms in the Guardian Management Gateway alarm table, associated with the sensor that is mapped to the specified object. However before doing that, it verifies the timestamp to make sure than no alarm originated after the specified timestamp. If this is not the case, then the client does not have the latest information about alarms, and the request is rejected.

16.11 Mapping the system event log

The Guardian Management Gateway system event log maps straightforwardly to the BACnet event log. The following properties of the BACnet event log object are supported:

PROPERTY NAME	ACCESS	PROPERTY SOURCE
Object Type	RO	25 = “Event Log”
Object Instance	RO	0
Object name	RO	“Event Log”
Description	RO	“Event Log”
Event State	RO	“Normal”
Reliability	RO	“No Fault Detected”
Status Flags	RO	No flags are set
Enable	RW	The “enabled” flag from Guardian Management SEL Info. Can be turned on and off from BACnet
Stop When Full	RO	True if OverflowAction == DROP in Guardian Management SEL Info, False otherwise (normally it is False).
Buffer Size	RO	System event log capacity, in records
Record Count	RW	The number of records in the system event log. Can be set to 0 to clear the event log.
Total Record Count	RO	The total number of records added to the system event log, since its creation
Log Buffer	RO	The array of log entries; use the Read Range service to access them

The event log object exposes the event log entries in the property Log Buffer in the form of an array. To access specific event log entries, the client should use the Read Range service. The only supported Read Range request type is “By Position”.

Each event log entry reported to the client consists of the following fields:

FIELD NAME	FIELD VALUE
Timestamp	Timestamp from the corresponding Guardian Management Gateway event log entry
Log Datum	Has type “notification”; contains the event that comprises the corresponding Guardian Management Gateway event log entry, translated to the ConfirmedEventNotification object.

16.12 Mapping the Reinitialize Device service

This service allows a BACnet client to reinitialize the server. The Guardian Management Gateway BACnet server partially implements this service, mapping it to Guardian Management Gateway reboot and restart operations.

This service requires the client to supply a password for the operation. This password is verified by the server and the operation is rejected if the password does not match. For the Guardian Management Gateway BACnet server, this password should be the Guardian Management Gateway password of the user “admin”.

Service operations are mapped according to the following table:

OPERATION	SMRC ACTION
Cold Restart	Reboot operation – the Guardian Management Gateway is rebooted
Warm Restart	Restart operation – the Guardian Management Gateway software is restarted
Start Backup	Not implemented, an error is returned
End Backup	Not implemented, an error is returned
Start Restore	Not implemented, an error is returned
End Restore	Not implemented, an error is returned
Abort Restore	Not implemented, an error is returned

13 Supported BACnet services (protocol commands)

The following table lists the BACnet services (protocol commands) which are supported by the SMRC BACnet server (as a responder).

BACNET SERVICE	CONFIRMED/UNCONFIRMED	SUPPORT STATUS
Who Is	Unconfirmed	Supported
Who Has	Unconfirmed	Supported
I Am	Unconfirmed	Supported
Read Property	Confirmed	Supported
Read Property Multiple	Confirmed	Supported
Read Range	Confirmed	Supported
Write Property	Confirmed	Supported
Write Property Multiple	Confirmed	Supported
Reinitialize Device	Confirmed	Partially supported
UTC Time Synchronization	Unconfirmed	Accepted but no action
Time Synchronization	Unconfirmed	Accepted but no action
Device Communication Control	Confirmed	Partially supported
Acknowledge Alarm	Confirmed	Supported
Get Alarm Summary	Confirmed	Supported

17 Security

In this section, the following facilities are described:

- Firewall
- Role-based firewall
- Login restrictions and password policy
- SSL certificate management
- Restricted Service Agreement

17.1 Firewall

This group of settings specifies Linux firewall rules. Firewall settings are separate for the IPv4 and IPv6 protocols, but have the same structure.

The global firewall settings include:

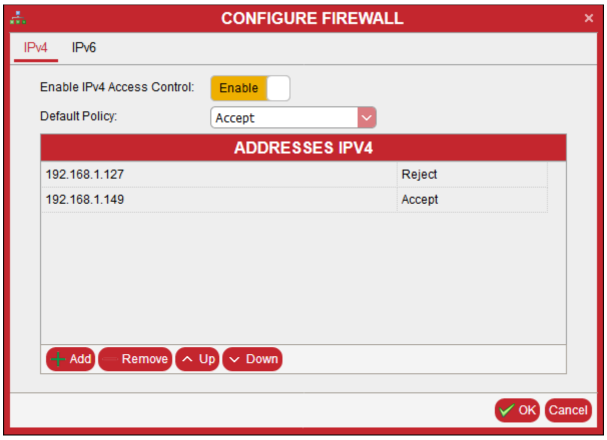
- Enable firewall: true if the firewall is enabled for the given protocol, false otherwise
- Default firewall policy for incoming packets: *ACCEPT*, *REJECT* or *DROP* packets

Each rule defines a network address (a host or subnet address) and the policy that applies to the packets originating from this address. The policy can be *ACCEPT*, *REJECT* or *DROP*. The order of rules is significant: for each incoming packet, the rules are examined in that order and the first matching rule determines the policy to be applied. If no rules match, the default policy is applied.

The following operations are defined for the firewall (separately for IPv4 and IPv6):

- Get “firewall enabled” flag
- Set “firewall enabled” flag
- Get default firewall policy (*ACCEPT*, *REJECT* or *DROP*)
- Set default firewall policy (*ACCEPT*, *REJECT* or *DROP*)
- Get firewall rule by index (index starts from *0*); the information returned includes the network address and policy
- Add firewall rule to the end of the list; the network address and the policy are specified
- Insert firewall rule by index (the new rule is inserted before the rule with index `index`); the network address and the policy are specified
- Modify firewall rule by index (the new rule replaces the rule with index `index`); the network address and the policy are specified
- Delete firewall rule by index

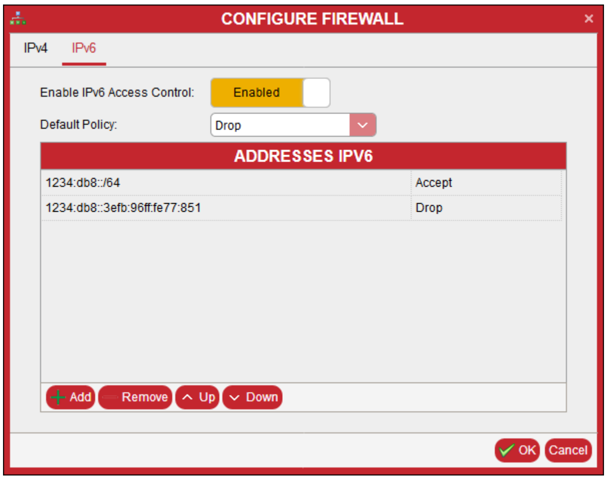
In the Web interface, the global firewall is configured via a dialog box that is accessible via menu items “Device Settings”->”Security”->”Firewall”. There are two tabs in this dialog box: “IPv4” and “IPv6”.



Enable IPv4 Access Control: Enable

Default Policy: Accept

ADDRESSES IPV4	
192.168.1.127	Reject
192.168.1.149	Accept



Enable IPv6 Access Control: Enabled

Default Policy: Drop

ADDRESSES IPV6	
1234:db8::/64	Accept
1234:db8::3efb:96ff:fe77:851	Drop

17.2 Login restrictions and password policy

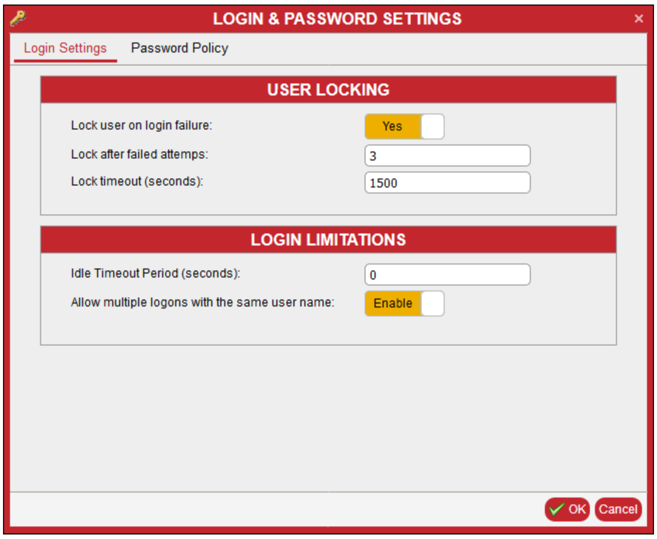
This set of options specifies requirements to password complexity, password aging and login security. The following options exist:

- AllowMultipleLogons – $TRUE$ if multiple logons with the same user name are allowed, $FALSE$ otherwise
- LockAfterFailedAttempts – lock a user (prevent from login) after this number of failed logon attempts, for a certain time
- LockTime – the number of seconds for which the user is locked
- IdleTimeout – the number of seconds; if a user is inactive for this number of seconds, he/she is logged off automatically. Value 0 turns off this feature.
- PasswordAging – $TRUE$ if password aging is enabled (logon passwords expire after some time and need to be changed after that)
- PasswordAgingInterval – the password aging interval, in days
- PasswordHistoryDepth – the system refuses to assign a new password that matches one of the most recent passwords for the user; this parameter specifies how many most recent passwords the system remembers. Value 0 turns off this feature.
- StrongPasswords – $TRUE$ if strong passwords are enforced (the properties of strong passwords are given by subsequent options), $FALSE$ otherwise
- MinStrongPasswordLength – minimum length of a strong password, in characters
- AtLeastOneLcCharacter – $TRUE$ if a strong password must contain at least one lowercase character, $FALSE$ otherwise
- AtLeastOneLcCharacter – $TRUE$ if a strong password must contain at least one lowercase character, $FALSE$ otherwise
- AtLeastOneNumCharacter – $TRUE$ if a strong password must contain at least one numeric character, $FALSE$ otherwise
- AtLeastOneSpecCharacter – $TRUE$ if a strong password must contain at least one special (punctuation) character, $FALSE$ otherwise.

The following operations are defined for the login restrictions and password policy:

- Get login restrictions and password policy (all options)
- Set login restrictions and password policy (all options)
- Check if the specified user is currently locked out
- Unlock the specified user

In the Web interface, the login restrictions and password policy are configured via a dialog box that is accessible via menu items “Device Settings” -> “Security” -> “Login Settings & Password Policy”. There are two tabs in this dialog box: “Login Settings” and “Password Policy”.



USER LOCKING

Lock user on login failure: Yes

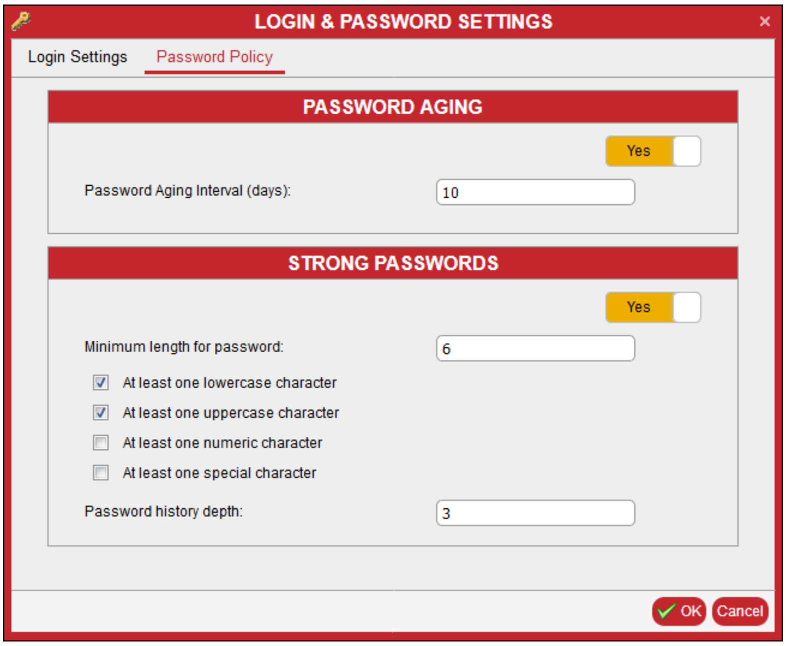
Lock after failed attemps: 3

Lock timeout (seconds): 1500

LOGIN LIMITATIONS

Idle Timeout Period (seconds): 0

Allow multiple logons with the same user name: Enable



PASSWORD AGING

Yes

Password Aging Interval (days): 10

STRONG PASSWORDS

Yes

Minimum length for password: 6

☑ At least one lowercase character

☑ At least one uppercase character

☐ At least one numeric character

☐ At least one special character

Password history depth: 3

17.3 Role-based firewall

This group of settings specifies rules for the role-based firewall. This firewall allows or denies logins for specific users from specific IP address ranges. Firewall settings are separate for the IPv4 and IPv6 protocols, but have the same structure.

The global role-based firewall settings include:

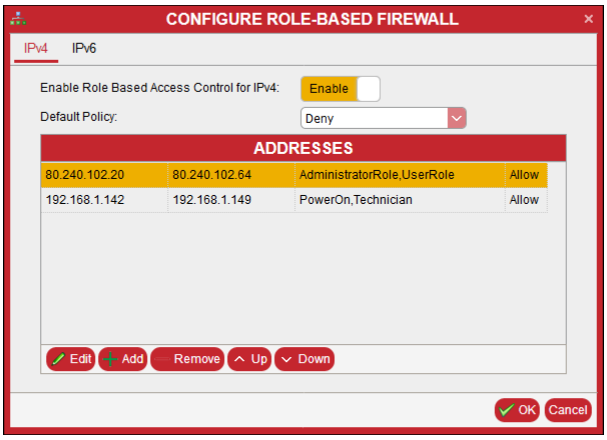
- Enable role-based firewall: true if the role-based firewall is enabled for the given protocol, false otherwise
- Default role-based firewall policy: *ALLOW* or *DENY* login

Each rule defines a range of network addresses (IPv4 or IPv6 addresses), the list of roles and the policy that applies to the login attempt of a user belonging to one of the specified roles, from an IP address belonging to the specified range. The policy can be *ALLOW* or *DENY*. The order of rules is significant: for each login attempt, the rules are examined in that order and the first matching rule determines the policy to be applied. If no rules match, the default policy is applied.

The following operations are defined for the role-based firewall (separately for IPv4 and IPv6):

- Get “role-based firewall enabled” flag
- Set “role-based firewall enabled” flag
- Get default role-based firewall policy (*ALLOW* or *DENY*)
- Set default role-based firewall policy (*ALLOW* or *DENY*)
- Get role-based firewall rule by index (index starts from *0*); the information returned includes the starting and ending IP address, the list of roles and the policy
- Add role-based firewall rule to the end of the list; the starting and ending IP address, the list of roles and the policy are specified
- Insert role-based firewall rule by index (the new rule is inserted before the rule with index `index`); the starting and ending IP address, the list of roles and the policy are specified
- Modify role-based firewall rule by index (the new rule replaces the rule with index `index`); the starting and ending IP address, the list of roles and the policy are specified
- Delete role-based firewall rule by index

In the Web interface, the global role-based firewall is configured via a dialog box that is accessible via menu items “Device Settings”->”Security”->”Role-Based Firewall”.



Enable Role Based Access Control for IPv4: Enable

Default Policy: Deny

ADDRESSES

80.240.102.20	80.240.102.64	AdministratorRole,UserRole	Allow
192.168.1.142	192.168.1.149	PowerOn,Technician	Allow

17.4 SSL Certificate Management

An SSL certificate is a file that is needed for secure HTTP (HTTPS) access to the Guardian Management Gateway; this file is issued by some certificate authority and confirms the identity of a specific HTTPS server, in our case, this is the identity of the Guardian Management Gateway. This file should be installed in a specific location on the Guardian Management Gateway, then it becomes an active certificate and can participate in the secure communication.

By default, no certificate is installed on the Guardian Management Gateway and it is the user’s responsibility to install one (if secure HTTP communication with the Guardian Management Gateway is needed).

There are three kinds of certificates and certificate-related objects that Guardian Management Gateway software can deal with:

- A certificate signed by a certificate authority (CA). This certificate must come from outside, and can be downloaded on the Guardian Management Gateway, stored there and installed as the active certificate
- A self-signed certificate. This certificate is generated on the Guardian Management Gateway and can be stored there and installed as the active certificate. When the active certificate is a self-signed one, Web browsers normally issue a warning when establishing an HTTPS session with the target server; the Web user must acknowledge the security risks to continue the communication
- A certificate sign request (CSR). This is the file that is generated on the Guardian Management Gateway and must be sent to a certificate authority to obtain a valid certificate. This file contains necessary information about the Guardian Management Gateway, its location and ownership.

The set of operations that deal with certificates is different between CLI and Web interface. This is because CLI is executed locally on the Guardian Management Gateway, while the Web client is remote to the Guardian Management Gateway.

With the CLI interface, the following certificate-related operations are supported:

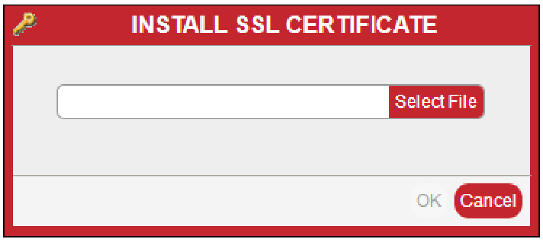
- Generate a self-signed certificate or a certificate sign request. The resulting file is stored in the specified location on the Guardian Management Gateway (by default, this is the user’s home directory) or can be copied to a remote SCP location
- Show the list of existing certificates and CSRs in the specified directory (by default, in the user’s home directory)
- Show the details of the specified certificate or CSR file
- Show the details of the active certificate
- Delete the specified local certificate or CSR file
- Copy a certificate or a CSR file to a remote SCP location or from a remote SCP location
- Install the specified certificate (a local file or a remote file accessible via SCP) as the active certificate.

With the Web interface, the following certificate-related operations are supported:

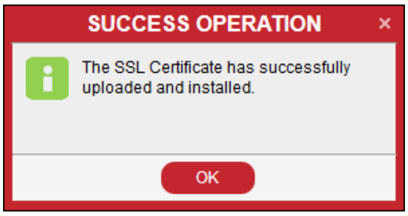
- Copy a certificate from the client system to the Guardian Management Gateway and install it as the active certificate
- Generate a certificate sign request on the Guardian Management Gateway and copy it to the client system
- Generate a self-signed certificate on the Guardian Management Gateway and install it as the active certificate
- Show the details of the active certificate.

In the Web interface, certificate-related operations are implemented as follows:

To copy a certificate from the client system to the Guardian Management Gateway and install it, use the dialog “Install SSL Certificate”, which is invoked with the menu command “Device Settings” -> “Security” -> “SSL Certificate”.

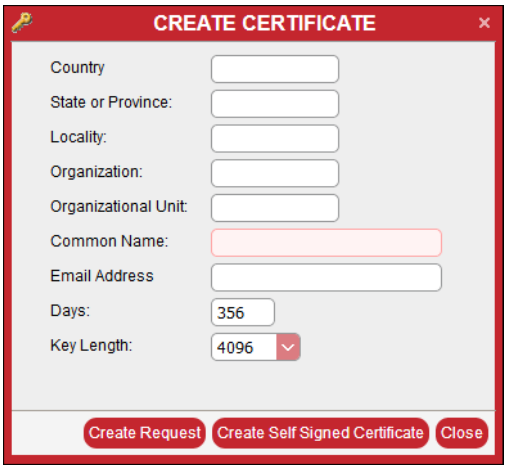


The user should select the local file with the certificate and start the upload by pressing the “OK” button. When the certificate is successfully uploaded, the following window is generated.



The SSL Certificate has successfully uploaded and installed.

To create a new SSL certificate or a CSR use the menu command “Device Settings” -> “Security” -> “Create SSL Certificate”. There are two buttons in the window “Create certificate”: “Create Request and “Create Self Signed Certificate”. A two-letter country code (ISO 3166-1 alpha-2 standard) or three-letter country code (ISO 3166-1 alpha-3 standard) should be written in the “Country” field.



Country
State or Province:
Locality:
Organization:
Organizational Unit:
Common Name:
Email Address
Days: 356
Key Length: 4096

17.5 Restricted Service Agreement

A restricted service agreement (a special security banner) can be shown to a user during the logon, both in CLI and Web interface. In addition, the restricted service agreement can be enforced, which means that the user should explicitly acknowledge it in order to be able to log in.

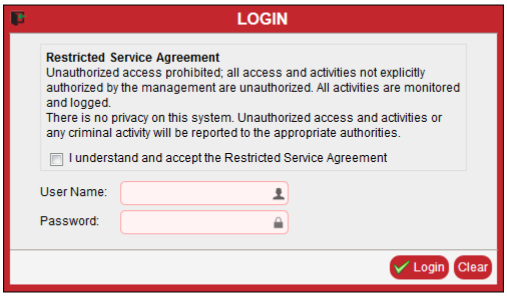
The following attributes are specified for the restricted service agreement:

- The text of restricted service agreement
- Enforce flag (*TRUE*/*FALSE*)

If the restricted service agreement text is configured and it is enforced, the following dialog will be shown during CLI logon:

```
SMRC Command Line Interpreter
RESTRICTED SERVICE AGREEMENT
---------- ------- ---------
Unauthorized access to this system is prohibited; all access and activities not
explicitly authorized by management are unauthorized. All activities are monitored
and logged.


Do you accept the restricted service agreement (y/n)? y
Connection from 80.240.102.63 as testuser
Current language: English
smrcli>
```

In the Web interface, the following dialog is shown:

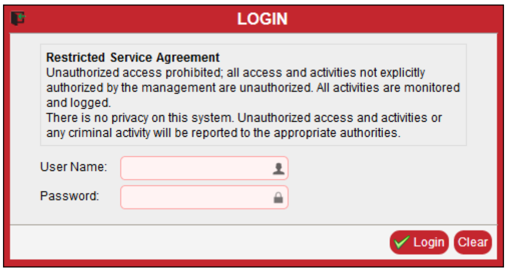


Restricted Service Agreement
Unauthorized access prohibited; all access and activities not explicitly authorized by the management are unauthorized. All activities are monitored and logged.
There is no privacy on this system. Unauthorized access and activities or any criminal activity will be reported to the appropriate authorities.

□ I understand and accept the Restricted Service Agreement

User Name:
Password:

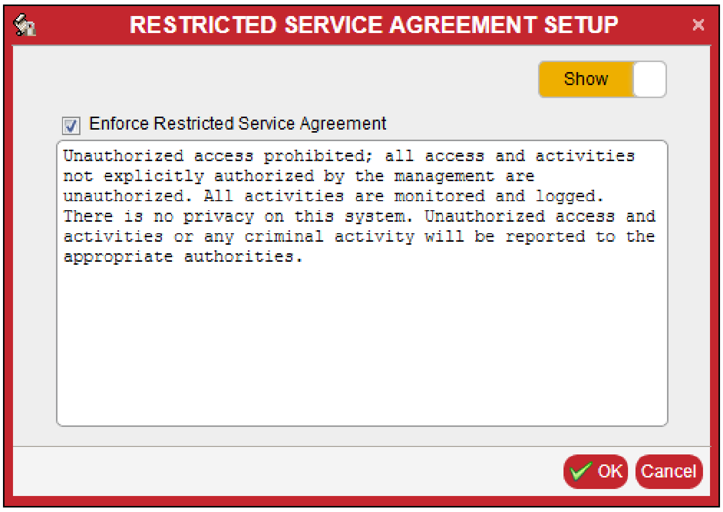
If the restricted service agreement text is configured and it is not enforced, the following dialog is shown:



Restricted Service Agreement
Unauthorized access prohibited; all access and activities not explicitly authorized by the management are unauthorized. All activities are monitored and logged.
There is no privacy on this system. Unauthorized access and activities or any criminal activity will be reported to the appropriate authorities.

User Name:
Password:

To configure the restricted service agreement with the Web interface, use the menu command “Device Settings” -> “Security” -> “Restricted Service Agreement Banner”. If the switch button “Show/Not used” is set to “Show”, the restricted service agreement is shown at every logon. The “Restricted Service Agreement Setup” window contains the checkbox “Enforce Restricted Service Agreement”. It corresponds to the enforce flag. The text area in the window contains the full text of the restricted service agreement. A user with sufficient privileges can edit the text of the restricted service agreement.



RESTRICTED SERVICE AGREEMENT SETUP

Show

☑ Enforce Restricted Service Agreement

```
Unauthorized access prohibited; all access and activities
not explicitly authorized by the management are
unauthorized. All activities are monitored and logged.
There is no privacy on this system. Unauthorized access and
activities or any criminal activity will be reported to the
appropriate authorities.
```

18 Events and Actions

Events are used to notify about state changes in various Guardian Management Gateway subsystems. All generated events are stored in the event log, most events are generated by sensors.

- Threshold-based sensors generate events when thresholds are crossed
- Discrete sensors generate events when sensor state changes

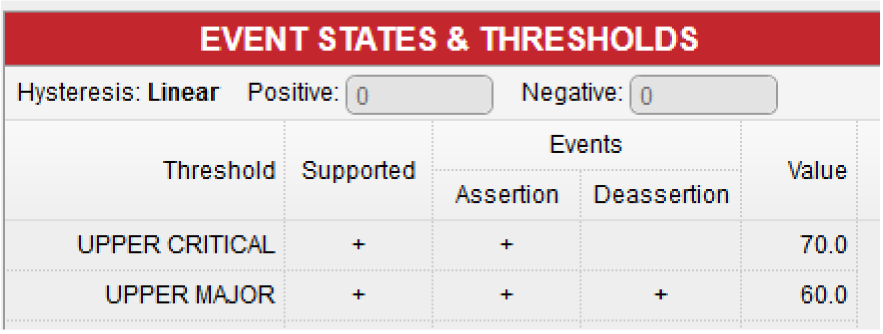
An important feature is that it's possible to define event actions and periodic actions.

Event actions allow to send messages, SNMP notifications and perform device control functions in response to certain events.

Periodic actions allow for automatic device control based on sensor values.

To generate an event, the assertion/deassertion event must be enabled!

Example:



EVENT STATES & THRESHOLDS

Hysteresis: **Linear** Positive: 0 Negative: 0

Threshold	Supported	Events		Value
		Assertion	Deassertion	
UPPER CRITICAL	+	+		70.0
UPPER MAJOR	+	+	+	60.0

When assertion and deassertion is enabled, an event is generated for the threshold crossings in both directions. In this example for a temperature sensor, an event is generated when the temperature exceeds 60 degrees, and the next event is generated when the temperature exceeds 70 degrees. When the temperature falls below 70 degrees no event is generated because deassertion for the Upper Critical threshold is not enabled.

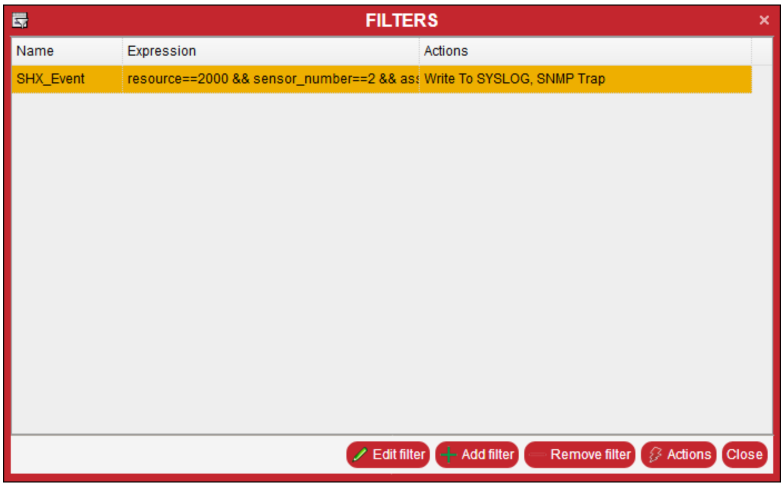
18.1 Event Filters

Event filters allow the user to trigger specified actions such as: send messages, SNMP notifications and perform device control functions to events. Each filter consists of a rule defined by an expression and one or more actions. When an event is generated, the filter expression is evaluated and, if the result is non-zero, the actions belonging to this filter are executed.

If the filter list consists of several filters, at an event the entire list is walked through and all filter expression are evaluated and the resp. actions are executed.

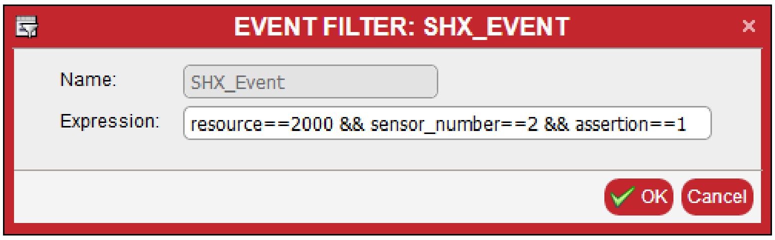
Expressions are evaluated in units defined by the global settings.

The window below can be accessed by selecting the menu item “Device Settings” -> “Event Rules”.



Name	Expression	Actions
SHX_Event	resource==2000 && sensor_number==2 && as	Write To SYSLOG, SNMP Trap

To edit the filter expression, press the button “Edit filter”.



Name: SHX_Event

Expression: resource==2000 && sensor_number==2 && assertion==1

18.2 Actions

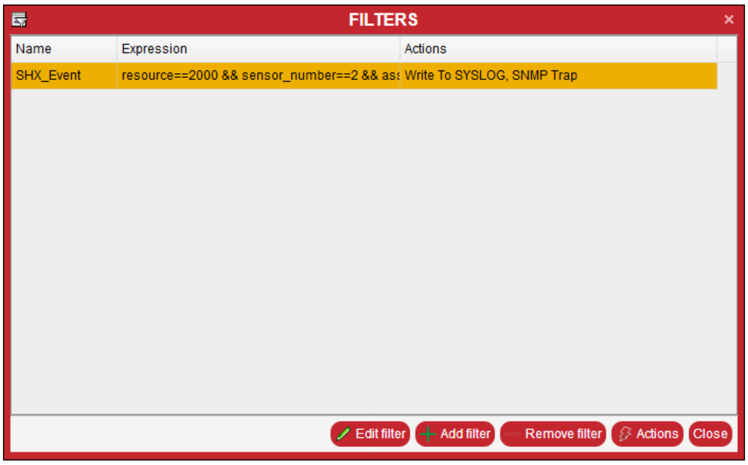
Each action in a filter has a “disposition” parameter. The following dispositions are defined:

- “Always” – the action is always executed.
- “If successful” – the action is executed only if execution of the previous action in a rule was successful.
- “If unsuccessful” – the action is executed only if execution of the previous action in a rule was unsuccessful.

There are several types of actions, they include:

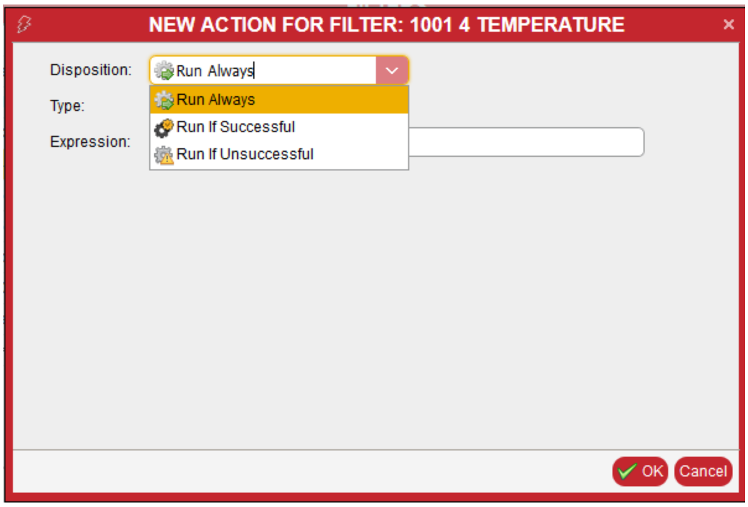
- “Expression” – evaluate an expression (likely with a side effect, e.g. assign a value to a control).
- “Command” – run a CLI command on the SMRC/Guardian Management Gateway.
- “Syslog” – log information about the event into the Linux system log on the SMRC/Guardian Management Gateway.
- “Send mail” – send an e-mail with the information about the event, via the preconfigured SMTP server. The list of recipients and the subject are the parameters of the action.
- “SNMP Trap” – send an SNMP trap (notification) to the previously specified target IP address.
- “Turn Cooling On” – turn on environment cooling, using a previously specified SHX cooling device.
- “Turn Cooling Off” – turn off environment cooling, using a previously specified SHX cooling device.
- “Max Cooling” – set maximum environment cooling, using a previously specified SHX cooling device.
- “Publish MQTT” - If the SMRC/Guardian Management Gateway is AWS enabled, for each event, an MQTT message with the information about the event is sent to AWS.

To edit or add an action press the button “Actions”.



Name	Expression	Actions
SHX_Event	resource==2000 && sensor_number==2 && as	Write To SYSLOG, SNMP Trap

Select the “disposition” parameter:

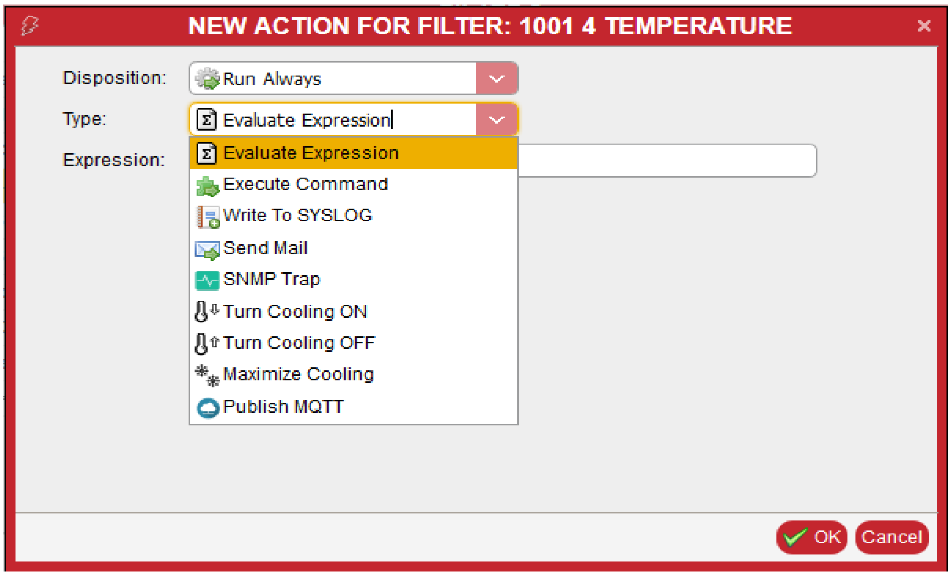


Disposition: Run Always

Type:

Expression:

Select the action type:



Disposition: Run Always

Type: Evaluate Expression

Expression:

The following operations exist for event filters and actions in event filters:

- Create an event filter, specifying its name and filter expression
- Delete an event filter by name
- Enumerate existing event filters
- Get event filter expression by event filter name
- Add an action to the event filter
- Enumerate actions for the given event filter
- Update a specific action for the given event filter
- Remove a specific action from the given event filter.

18.3 Expressions

Guardian Management Gateway allows expressions to be specified as event filtering criteria and as event actions. Also an expression can be directly evaluated by the CLI command `expression`. These expressions conform to certain syntax, similar to the syntax of arithmetic and logical expressions in the C and Java programming languages.

For event filtering, the expression is evaluated and the result determines whether the event passes the filter (the event passes if the result is not *0*).

For actions, the expression is evaluated and the result is ignored. The expression normally has some side effects (e.g. assignment to some variable or control).

For the CLI command `expression`, the expression is evaluated and the result is printed on the CLI console.

18.3.1 Value Types

The result of an expression evaluation is a value, which has a type. A value type can be “integer number”, “real number” or “string”. Boolean values are represented as integer numbers, *1* represents *TRUE*, *0* represents *FALSE*. Both integer and real numbers have 64 bits in size.

18.3.2 Expression Structure

The expression consists of terms connected with operators. Terms include special names, variables, integer, real and string constants, sensor and control designators.

18.3.3 Special Names

Special names designate certain fields in the event that is currently subject to filtering. In the action expressions, these special names designate the fields of the event on which the action is invoked.

In alarm subexpressions (parameters to the functions `alarm_exists` and `alarm_count`) the special names designate the fields of the alarm table entries.

Special names are defined in the following tables, they are case insensitive (i.e. `sensor_number`, `Sensor_Number`, or `SENSOR_NUMBER` variants can be used).

Table 1: Event-related special names

NAME	TYPE	DESCRIPTION
`assertion`	Integer	*0* for deassertion events, *1* for assertion events
`event_category`	Integer	The event category, according to HPI definition (e.g. *1* for threshold events, *2* for usage state events, etc.)
`resource`	Integer	The resource ID which sourced the event. This value is *−1* if not applicable.
`is_fumi`	Integer	*1* if the event is an HPI FUMI (upgrade-related) event, *0* otherwise
`is_sensor`	Integer	*1* if the event is originated by a sensor, *0* otherwise

NAME	TYPE	DESCRIPTION
sensor_number	Integer	The number of the sensor that originated the event. This value is −1 if not applicable
managed_sensor	Integer	The number of the managed sensor that corresponds to the physical sensor that originated the event. This value is -1 if not applicable
sensor_state	Integer	The single sensor state that, asserted or deasserted, caused the event; it is represented as a bit mask with a single bit set. This value is 0 if not applicable
sensor_type	Integer	The type of the sensor that originated the event, according to HPI definition (e.g. 1 for Temperature sensors, 2 for Voltage sensors, etc). A string value is returned in the contexts that allow string values (e.g "Temperature", "Voltage")
severity	Integer	Event severity according to HPI definition (0 for Critical, 1 for Major, 2 for Minor, 3 for Informational, 4 for OK). A string value is returned in the contexts that allow string values: "Critical", "Major, "Minor", "Informational" and "OK".
upper_critical	boolean	True if the event indicates crossing of the corresponding threshold (including deassertion events), false otherwise.
upper_major	boolean	True if the event indicates crossing of the corresponding threshold (including deassertion events), false otherwise.
upper_minor	boolean	True if the event indicates crossing of the corresponding threshold (including deassertion events), false otherwise.
lower_critical	boolean	True if the event indicates crossing of the corresponding threshold (including deassertion events), false otherwise.
lower_major	boolean	True if the event indicates crossing of the corresponding threshold (including deassertion events), false otherwise.
lower_minor	boolean	True if the event indicates crossing of the corresponding threshold (including deassertion events), false otherwise.

Table 2: Alarm-related special names

NAME	TYPE	DESCRIPTION
acknowledged	Integer	1 if the alarm is acknowledged, 0 otherwise
event_category	Integer	The event category of the alarm, according to HPI definition (e.g. 1 for threshold events, 2 for usage state events, etc.)
resource	Integer	The resource ID which sourced the event that caused the alarm. This value is −1 if not applicable.
is_sensor	Integer	1 if the alarm is caused by a sensor event, 0 otherwise
sensor_number	Integer	The number of the sensor that originated the event. This value is −1 if not applicable
sensor_state	Integer	The single sensor state that, asserted or deasserted, caused the alarm; it is represented as a bit mask with a single bit set. This value is 0 if not applicable
sensor_type	Integer	The type of the sensor that originated the event, according to HPI definition (e.g. 1 for Temperature sensors, 2 for Voltage sensors, etc.). A string value is returned in the contexts that allow string values (e.g. "Temperature", "Voltage")
severity	Integer	Alarm severity according to HPI definition (0 for Critical, 1 for Major, 2 for Minor, 3 for Informational, 4 for OK). A string value is returned in the contexts that allow string values: "Critical", "Major, "Minor", "Informational" and "OK".

18.3.4 **Variables**

Variable names start with *$* and further consist of alphanumeric characters. They are case insensitive (e.g. *$var1*, *$Var1* and *$VAR1* designate the same variable). The variables are global variables that are created when they are first referenced; the variable value is integer *0* at this point. Values of variables can be integer numbers, real numbers or strings. The type is associated with the value, not with the variable.

Variable values set in one expression are preserved after the evaluation of this expression is complete and can later be used in other expressions.

18.3.5 **Sensor items**

A sensor item has the following syntax:

```
sensor-item ::= "SENSOR" "(" resource-id "," sensor-number ")" [ "." sensor-item-tail ]
```

```
sensor-item-tail ::= "UNR" | "UC" | "UNC" | "LNR" | "LC" | "LNC" | "FAILED" | "INITIAL_UPDATE" | state-number
```

The value of a sensor item is calculated as follows:

If `sensor-item-tail` is omitted, the value is the numeric sensor value.

If a `sensor-item-tail` is present, the value is of the item is *1* or *0*, depending on whether the sensor is in the specified state. The states `UNR`, `UC`, `UNC`, `LNR`, `LC`, `LNC` indicate whether the sensor is beyond the corresponding threshold. The state `FAILED` indicates whether the sensor reading has failed. The state `INITIAL_UPDATE` indicates whether the numeric value of the sensor is not available. A state number indicates whether the corresponding state is set in the sensor event state mask.

18.3.6 **Control items**

A control item has the following syntax:

```
control-item ::= "CONTROL" "(" resource-id "," control-number ")"
```

The value of a control item is numeric; it's the result of the “get” operation applied to the corresponding control. For digital controls, the result is *1* if the control is in the *ON* state and *0* if it is in the *OFF* state.

Control items can be targets of an assignment operation. Assigning a value to a control means setting the control to this value (and to “manual” mode in the HPI sense). For digital controls, assigning *1* sets the control to the *ON* state, assigning *0* sets the control to the *OFF* state.

18.3.7 **Constants**

Integer and real constants have usual representation (e.g. *25*, *2.5*). String constants are enclosed in double quotes (e.g. “string”). The value of a constant is this constant.

18.3.8 **Operators**

The following table lists all operators, with their arity and priority for binary operators:

Table 3: Operators

OPERATOR	ARITY	PRIORITY	DEFINITION
!	1		*NOT operator*. The operand must be numeric. The result is 1 if applied to 0 and 0 if applied to any non-zero value.
~	1		*Complement operator*. The operand must be numeric. The result is a bit-wise complement of the operand.
-	1		*Negation operator*. The operand must be numeric. The result is the operand subtracted from 0.
*	2	1	*Multiplication*. The operands must be numeric. The result is the product of the operands. If one of the operands is a real number, the result is a real number.
/	2	1	*Division*. The operands must be numeric. If one of the operands is a real number, the result is a real number, otherwise the operation is integer division.
%	2	1	*Remainder*. The operands must be integer. The result is the remainder of division of the first operand by the second operand.
+	2	2	*Addition*. For numeric operands, the result is the sum of the operands. If one of the operands is a real number, the result is a real number. This operation is also applicable to string values and yields their concatenation.
-	2	2	*Subtraction*. The operands must be numeric. The result is the difference of the operands. If one of the operands is a real number, the result is a real number.
<<	2	3	*Left shift*. The operands must be numeric. The result is the result of the left shift of the first operand by the number of bits specified by the second operand.
>>	2	3	*Right shift*. The operands must be numeric. The result is the result of the right shift of the first operand by the number of bits specified by the second operand.
==	2	4	*Equal*. Compares the two operands for equality and yields 1 if they are equal and 0 if they are not equal. String values can also be compared; if one of the operands is a string value, and the other is not, the other value is converted to a string.
!=	2	4	*Not Equal*. Compares the two operands for inequality and yields 1 if they are not equal and 0 if they are equal. String values can also be compared; if one of the operands is a string value, and the other is not, the other value is converted to a string.
<	2	4	*Less*. Compares the two operands and yields 1 if the first operand is less than the second and 0 otherwise. String values can also be compared; if one of the operands is a string value, and the other is not, the other value is converted to a string.
>	2	4	*Greater*. Compares the two operands and yields 1 if the first operand is greater than the second and 0 otherwise. String values can also be compared; if one of the operands is a string value, and the other is not, the other value is converted to a string.

OPERATOR	ARITY	PRIORITY	DEFINITION
<=	2	4	*Less or Equal*. Compares the two operands and yields 1 if the first operand is less or equal than the second and 0 otherwise. String values can also be compared; if one of the operands is a string value, and the other is not, the other value is converted to a string.
>=	2	4	*Greater or Equal*. Compares the two operands and yields 1 if the first operand is greater or equal than the second and 0 otherwise l. String values can also be compared; if one of the operands is a string value, and the other is not, the other value is converted to a string.
&	2	5	*Bitwise AND*. The operands must be numeric. The result is the result of the bitwise AND of the two operands. The type of the result is integer.
|	2	6	*Bitwise OR*. The operands must be numeric. The result is the result of the bitwise OR of the two operands. The type of the result is integer.
^	2	6	*Bitwise XOR*. The operands must be numeric. The result is the result of the bitwise XOR of the two operands. The type of the result is integer.
&&	2	7	*Logical short-circuit AND*. The operator evaluates the first operand, and if the result is 0, it yields 0 and does not evaluate the second operand. Otherwise, it evaluates the second operand and returns the resulting value as the result of the whole operation.
||	2	8	*Logical short-circuit OR*. The operator evaluates the first operand, and if the result is not 0, it yields this result as the result of the whole operation and does not evaluate the second operand. Otherwise (if the first operand evaluates to 0), it evaluates the second operand and returns the resulting value as the result of the whole operation.
=	2	9	*Assignment*. This operator is right-associative. The first operand (assignment target) must be a variable or a control item. The operator evaluates the second operand and assigns the resulting value to the assignment target and yields it as the result of the operation (allowing chained assignments)
?: IF...THEN...ELSE	3	10	This is the *conditional operator*, can be represented in the form a ? b : c or IF a THEN b ELSE c. First a is evaluated, then depending on the value of a (non-zero or 0), subexpression b or c respectively is evaluated and the corresponding value is returned.
,	2	11	*Comma operator*. The first operand is evaluated, the value is thrown away, and then the second operand is evaluated and its value is the value of the whole operation.

18.3.9 **Alarm-related functions**

Alarm-related functions alarm_exists and alarm_count return information about the presence of certain entries in the alarm table. Both functions take one argument, which is a predicate expression evaluated over all alarm table entries. Special names in this expression refer to the fields in the alarm table entry.

The function alarm_exists returns TRUE if the predicate returns TRUE for at least one entry, FALSE otherwise.

The function alarm_count returns the number of alarm table entries for which the predicate returns TRUE. For example, alarm_count(1) returns the total number of entries in the alarm table.

18.3.10 Aggregate functions

These functions implement aggregate operations on groups. Values of all sensors in the group are evaluated and aggregated according to the specific function. All these functions have a single parameter that should be a group name. The functions, their return types and their semantics are listed in the table below:

Table 4. Aggregate functions

FUNCTION NAME	TYPE	DESCRIPTION
`count`	Integer	The number of sensors that return valid readings.
`total`	Real	Sum of readings of all sensors in the group.
`minimum`	Real	The minimal reading among all sensors in the group.
`maximum`	Real	The maximal reading among all sensors in the group.
`average`	Real	The average reading among all sensors in the group (`total()` divided by `count()`)
`square_total`	Real	Sum of squares of readings of all sensors in the group.
`dispersion`	Real	The dispersion of readings of all sensors in the group.
`state_count`	Integer	The number of sensors in the group that return state mask.
`state_and`	Integer	The aggregate AND of state masks for all sensors in the group.
`state_or`	Integer	The aggregate OR of state masks for all sensors in the group.
`state_xor`	Integer	The aggregate XOR of state masks for all sensors in the group.

18.4 Examples for event filtering rules and expressions

The following expression can be used for an event filtering.

```
resource==1000 && sensor_number==1 && assertion==1
```

If an assertion event is generated by sensor #1 at resource 1000, the event passes the filter, → an action is triggered.

- The special names `resource`, `sensor_number`, `assertion` are defined in Table 1.
- The operators `==`, `&&` are defined in Table 3.

This expression can be used for an event filtering.

```
is_fumi==1 && severity!=1
```

The event passes the filter if it is a HPI FUMI (upgrade-related) event with severity other than *MAJOR*.

- The special names `is_fumi`, `severity` are defined in Table 1.
- The operators `==`, `!=` are defined in Table 3.

The following expression can be used for an event filtering.

```
SENSOR(3000,1).UNR && sensor_type==2
```

An event passes the filter if the type of the sensor that originated the event is “Voltage” and the sensor #1 at resource 3000 is beyond the Upper Non-Recoverable threshold. The special name `sensor_type` is defined in Table 1. The operators `==`, `&&` are defined in Table 3.

The following expression makes use of an alarm-related function.

```
SENSOR(3000,1)> 25. && alarm_exists(is_sensor && resource == 4002 && acknowledged
== 0)
```

The expression evaluates to *TRUE* if the sensor reading of sensor #1 at resource 3000 exceeds 25.0 and there is an unacknowledged alarm caused by a sensor event, and the resource ID that sourced the sensor event is 4002. The special names *is_sensor*, *resource*, *acknowledged* are defined in Table 2. The operators *==*, *&&*, *>* are defined in Table 3.

The following expression makes use of an aggregate function.

```
IF total("Even Outlets") > 70 THEN CONTROL(3000,1)=1 ELSE CONTROL(3000,1)=0
```

The expression evaluates the sum of sensor readings of all the sensors in the group ‘Even Outlets’ and compares it to *70*. It is supposed that the group contains at least one sensor. If the comparison holds *TRUE*, the control #1 at resource 3000 is set the *ON* state. If the comparison holds *FALSE*, the control #1 at resource 3000 is set the *OFF* state. The aggregate function *total()* is defined in Table 4. Its only argument is a group name. In this specific case the group name should be put into the quotes ("") since it contains a whitespace character. The operators *>*, *IF..THEN..ELSE* are defined in Table 3.

18.5 Periodic actions

Periodic actions are objects, similar to event filters but intended to run certain actions periodically (instead of as a reaction to an event). Periodic actions can be used to implement environment management algorithms (e.g. cooling management) on an iPDU. Similar to event filters, each periodic action consists of a predicate expression and one or more actions. Periodically (the value of the period is specified when the periodic action is created), the periodic action is invoked: that is, the predicate expression is evaluated and, if the result is non-zero, the corresponding actions are executed. Expressions are evaluated in units defined by the global settings.

Periodic actions share their name space with event filters (there can be no event filter and periodic action with the same name), and action management for them uses the same commands as for event filters.

The following operations exist for periodic actions:

- Create a periodic action, specifying its name, predicate expression and invocation period (in seconds)
- Delete a periodic action by name
- Enumerate existing periodic actions
- Get predicate expression by periodic action name
- Add an action to the periodic action
- Enumerate actions for the given periodic action
- Update a specific action for the given periodic action
- Remove a specific action from the given periodic action.

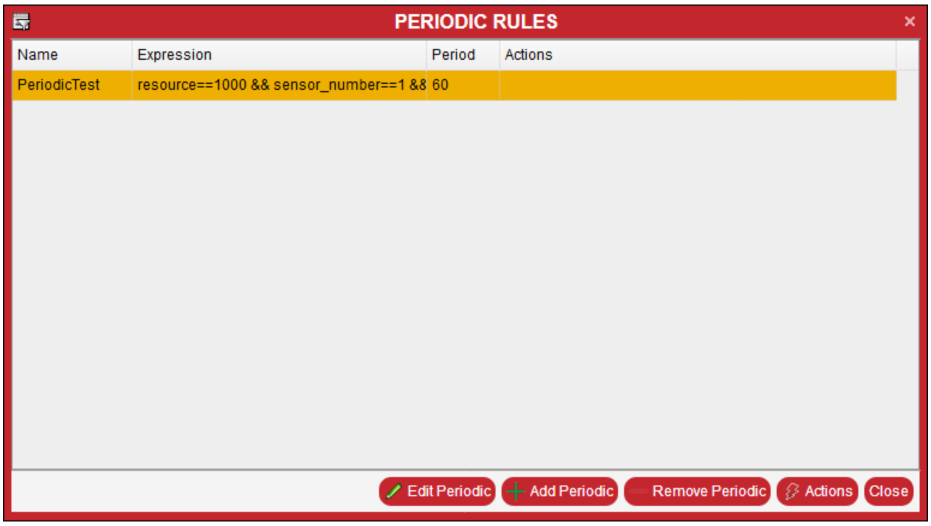
To manage periodic actions in CLI, use commands `periodic` and `action`.

For the command `periodic,` use its subcommands as follows:

- Use the command `periodic add` to create a new periodic action, specify the periodic action name, the predicate expression and the period in seconds
- Use the command `periodic delete` to delete a periodic action by name
- Use the command `periodic list` to see the list of defined periodic actions
- Use the command `periodic show` to see information about a specific periodic action by its name.

Use the command `action` to manage the action list for a specific periodic action; the usage scenarios are the same as for the event filters.

This window below can be accessed by selecting the menu item “Device Settings” -> “Periodic Rules”.



Name	Expression	Period	Actions
PeriodicTest	resource==1000 && sensor_number==1 &&	60	

To edit the predicate expression, press the button “Edit Periodic”.

To edit the action list of the periodic expression press the button “Actions”.

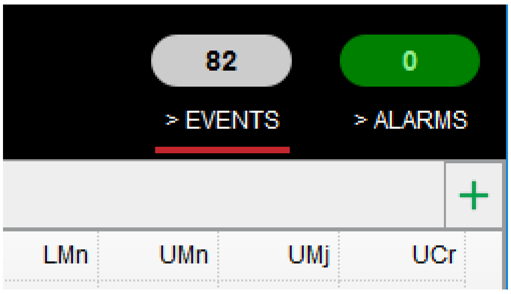
18.6 Event log (SEL)

The event log is maintained on the Guardian Management Gateway in the format of HPI System Event Log (SEL). All generated events are stored in the event log. Event log storage capacity is 10000 events by default, but can be reduced by the user. When the event log reaches its capacity, the oldest events become deleted. Also, a user can clear the event log at any moment. Other than that, the event log is read-only for the user.

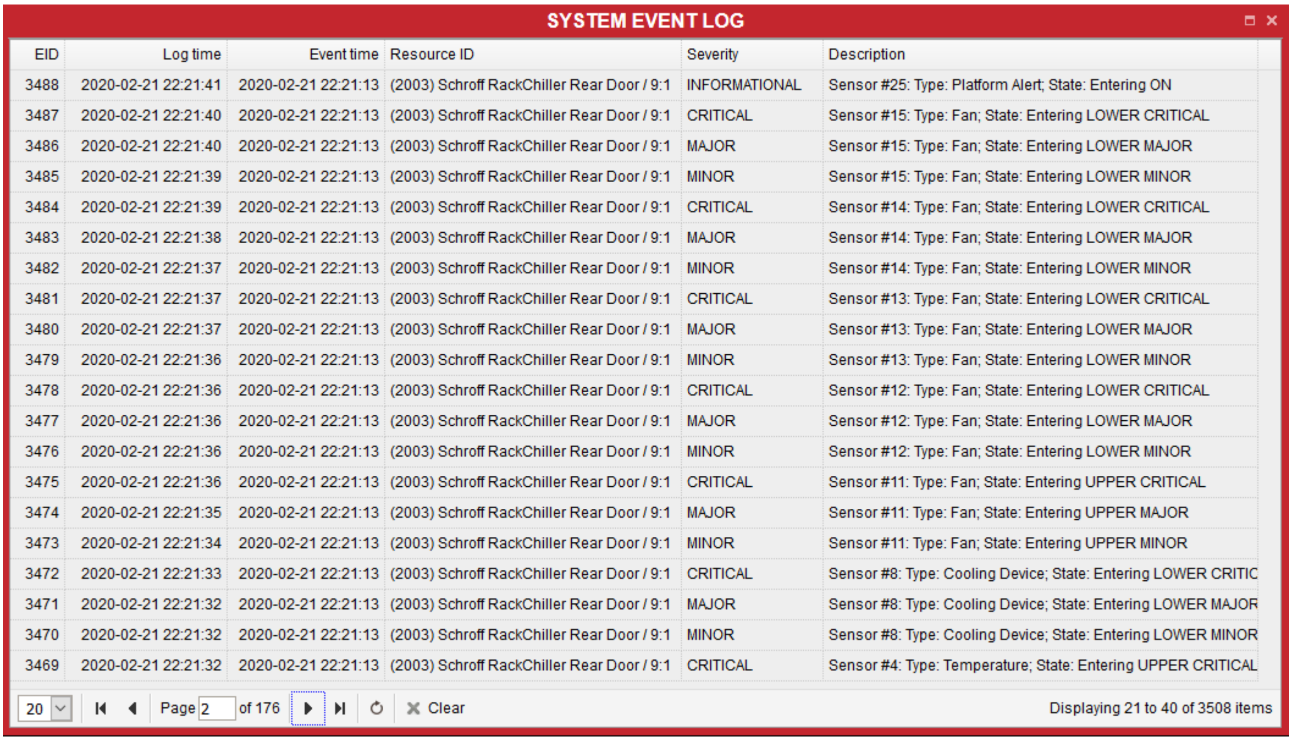
The following operations are defined for the system event log:

- Get information about the event log as a whole
- Enumerate entries in the event log
- Clear the event log.

In the Web interface, the event log is represented by the “System Event Log” window. This window can be accessed by selecting the menu item “Maintenance” -> “System Event Log” or by pressing the SEL indicator on the status.



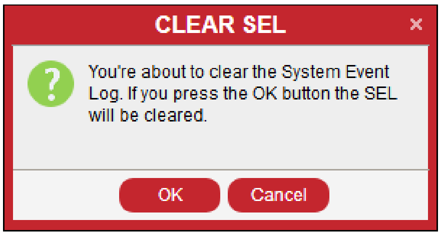
The “System Event Log” window shows event log entries page by page. Control items at the bottom of the window allow the user to navigate to the previous page and to the next page, navigate to the beginning or to the end of the event log, change the number of items per page, refresh the view or clear the event log.



EID	Log time	Event time	Resource ID	Severity	Description
3488	2020-02-21 22:21:41	2020-02-21 22:21:13	(2003) Schroff RackChiller Rear Door / 9:1	INFORMATIONAL	Sensor #25: Type: Platform Alert; State: Entering ON
3487	2020-02-21 22:21:40	2020-02-21 22:21:13	(2003) Schroff RackChiller Rear Door / 9:1	CRITICAL	Sensor #15: Type: Fan; State: Entering LOWER CRITICAL
3486	2020-02-21 22:21:40	2020-02-21 22:21:13	(2003) Schroff RackChiller Rear Door / 9:1	MAJOR	Sensor #15: Type: Fan; State: Entering LOWER MAJOR
3485	2020-02-21 22:21:39	2020-02-21 22:21:13	(2003) Schroff RackChiller Rear Door / 9:1	MINOR	Sensor #15: Type: Fan; State: Entering LOWER MINOR
3484	2020-02-21 22:21:39	2020-02-21 22:21:13	(2003) Schroff RackChiller Rear Door / 9:1	CRITICAL	Sensor #14: Type: Fan; State: Entering LOWER CRITICAL
3483	2020-02-21 22:21:38	2020-02-21 22:21:13	(2003) Schroff RackChiller Rear Door / 9:1	MAJOR	Sensor #14: Type: Fan; State: Entering LOWER MAJOR
3482	2020-02-21 22:21:37	2020-02-21 22:21:13	(2003) Schroff RackChiller Rear Door / 9:1	MINOR	Sensor #14: Type: Fan; State: Entering LOWER MINOR
3481	2020-02-21 22:21:37	2020-02-21 22:21:13	(2003) Schroff RackChiller Rear Door / 9:1	CRITICAL	Sensor #13: Type: Fan; State: Entering LOWER CRITICAL
3480	2020-02-21 22:21:37	2020-02-21 22:21:13	(2003) Schroff RackChiller Rear Door / 9:1	MAJOR	Sensor #13: Type: Fan; State: Entering LOWER MAJOR
3479	2020-02-21 22:21:36	2020-02-21 22:21:13	(2003) Schroff RackChiller Rear Door / 9:1	MINOR	Sensor #13: Type: Fan; State: Entering LOWER MINOR
3478	2020-02-21 22:21:36	2020-02-21 22:21:13	(2003) Schroff RackChiller Rear Door / 9:1	CRITICAL	Sensor #12: Type: Fan; State: Entering LOWER CRITICAL
3477	2020-02-21 22:21:36	2020-02-21 22:21:13	(2003) Schroff RackChiller Rear Door / 9:1	MAJOR	Sensor #12: Type: Fan; State: Entering LOWER MAJOR
3476	2020-02-21 22:21:36	2020-02-21 22:21:13	(2003) Schroff RackChiller Rear Door / 9:1	MINOR	Sensor #12: Type: Fan; State: Entering LOWER MINOR
3475	2020-02-21 22:21:36	2020-02-21 22:21:13	(2003) Schroff RackChiller Rear Door / 9:1	CRITICAL	Sensor #11: Type: Fan; State: Entering UPPER CRITICAL
3474	2020-02-21 22:21:35	2020-02-21 22:21:13	(2003) Schroff RackChiller Rear Door / 9:1	MAJOR	Sensor #11: Type: Fan; State: Entering UPPER MAJOR
3473	2020-02-21 22:21:34	2020-02-21 22:21:13	(2003) Schroff RackChiller Rear Door / 9:1	MINOR	Sensor #11: Type: Fan; State: Entering UPPER MINOR
3472	2020-02-21 22:21:33	2020-02-21 22:21:13	(2003) Schroff RackChiller Rear Door / 9:1	CRITICAL	Sensor #8: Type: Cooling Device; State: Entering LOWER CRITIC
3471	2020-02-21 22:21:32	2020-02-21 22:21:13	(2003) Schroff RackChiller Rear Door / 9:1	MAJOR	Sensor #8: Type: Cooling Device; State: Entering LOWER MAJOR
3470	2020-02-21 22:21:32	2020-02-21 22:21:13	(2003) Schroff RackChiller Rear Door / 9:1	MINOR	Sensor #8: Type: Cooling Device; State: Entering LOWER MINOR
3469	2020-02-21 22:21:32	2020-02-21 22:21:13	(2003) Schroff RackChiller Rear Door / 9:1	CRITICAL	Sensor #4: Type: Temperature; State: Entering UPPER CRITICAL

Displaying 21 to 40 of 3508 items

To clear the event log (erase all entries), press the “Clear” button at the bottom of the window (this button is visible only if the current user is privileged enough to clear the event log). The confirmation dialog will appear:



You're about to clear the System Event Log. If you press the OK button the SEL will be cleared.

Press the OK button to confirm the intention to clear the event log.

19 Alarm Table

Guardian Management Gateway maintains the HPI Alarm table, which is the table of active alarms and represents an aggregated view on any anomalies in the current state of the Guardian Management Gateway. There is only one alarm table on the Guardian Management Gateway. Each alarm is caused by the corresponding alarm condition; alarm conditions are typically associated with sensors. Guardian Management Gateway currently supports only sensor-based alarm conditions.

For a threshold-based sensor, an alarm condition occurs when the sensor reading goes beyond a threshold, and ceases to exist when the sensor reading goes back. If multiple thresholds are crossed at once, multiple alarm conditions are generated.

For a discrete sensor, an alarm condition occurs when the sensor goes into a state with the severity Minor, Major or Critical, and disappears when the sensor leaves this state.

Alarms are associated with events; an alarm is added to the alarm table in response to the event that indicates that the corresponding alarm condition has appeared.

An alarm can be automatically removed from the alarm table is response to the event that indicates that the corresponding alarm condition has disappeared. Or, an alarm can be “sticky” and stay in the alarm table until it is deleted manually by a user. This behavior depends on the global parameter “maximum transient alarm severity”. Alarms with the severity greater than the value of this parameter stay permanently in the alarm table; alarms with the severity less or equal than value of this parameter are automatically removed when the corresponding alarm condition goes away. By default, the value of this parameter is set to “Critical”, which means that all alarms atr transient, but it can be changed by a user. For example, if this parameter is set to “Minor”, then alarms with severity “Critical” and “Major” will stay in the alarm table permanently, while alarms with the severity “Minor” and below will be transient.

An alarm can be acknowledged by a user, meaning that the user has recognized the presence of this alarm. Initially an alarm is unacknowledged. Acknowledged and unacknowledged alarms are shown differently in CLI and Web interfaces.

A user can manually delete an alarm from the alarm table. Transient alarms can be deleted by the user even while the alarm condition is active.

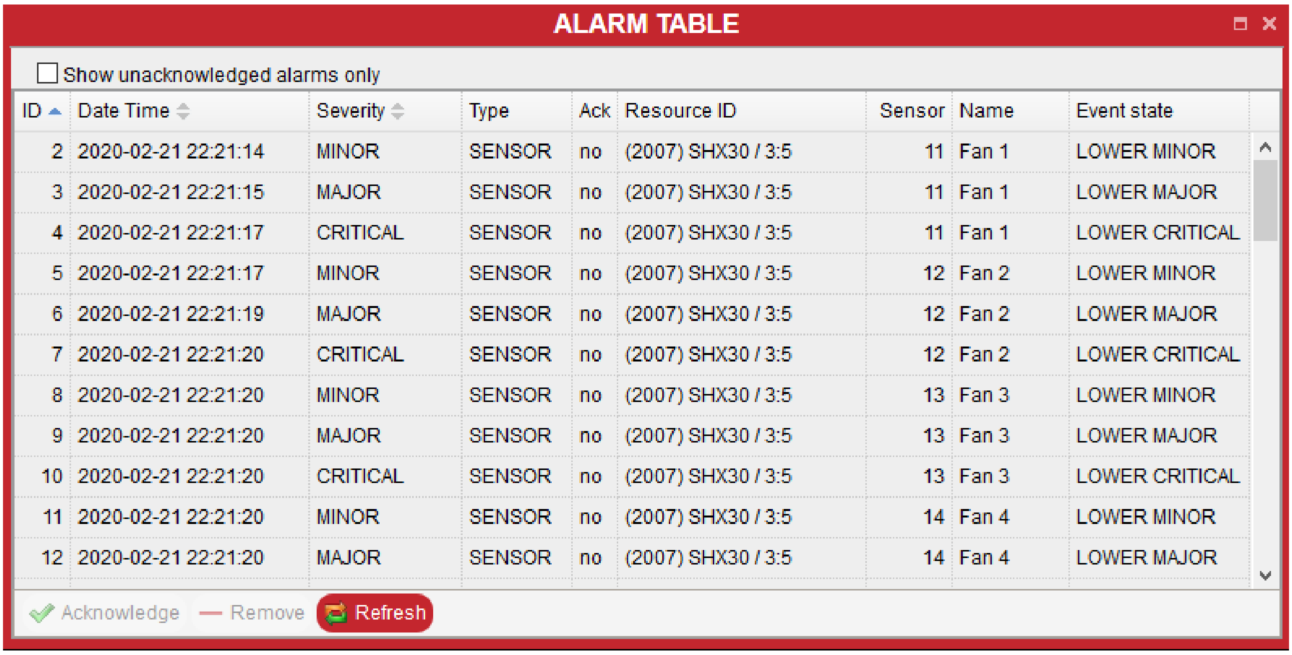
For each alarm table entry (active alarm), the following information is available:

- Alarm ID – the index of the alarm in the table
- Timestamp - when the alarm was created
- Alarm severity – can be Minor, Major or Critical; corresponds to the severity of the event that caused the alarm
- Acknowledged state (`yes` or `no`)
- Alarm condition; for sensor-based alarms, the alarm condition contains the following fields:
    - Entity path of the entity related to the alarm condition
    - Resource number
    - Sensor number
    - Event state (sensor state) that caused the alarm condition

A user can perform the following operations with the alarm table and specific alarms in it:

- View the alarm table as a whole
- View information about a specific alarm
- Acknowledge a specific alarm
- Delete a specific alarm from the alarm table

In the Web interface, the alarm table is represented by a separate table-like window which is invoked by the menu command “Maintenance” -> “Alarm Table”. Each line in the table represents one alarm.



ID	Date Time	Severity	Type	Ack	Resource ID	Sensor	Name	Event state
2	2020-02-21 22:21:14	MINOR	SENSOR	no	(2007) SHX30 / 3:5	11	Fan 1	LOWER MINOR
3	2020-02-21 22:21:15	MAJOR	SENSOR	no	(2007) SHX30 / 3:5	11	Fan 1	LOWER MAJOR
4	2020-02-21 22:21:17	CRITICAL	SENSOR	no	(2007) SHX30 / 3:5	11	Fan 1	LOWER CRITICAL
5	2020-02-21 22:21:17	MINOR	SENSOR	no	(2007) SHX30 / 3:5	12	Fan 2	LOWER MINOR
6	2020-02-21 22:21:19	MAJOR	SENSOR	no	(2007) SHX30 / 3:5	12	Fan 2	LOWER MAJOR
7	2020-02-21 22:21:20	CRITICAL	SENSOR	no	(2007) SHX30 / 3:5	12	Fan 2	LOWER CRITICAL
8	2020-02-21 22:21:20	MINOR	SENSOR	no	(2007) SHX30 / 3:5	13	Fan 3	LOWER MINOR
9	2020-02-21 22:21:20	MAJOR	SENSOR	no	(2007) SHX30 / 3:5	13	Fan 3	LOWER MAJOR
10	2020-02-21 22:21:20	CRITICAL	SENSOR	no	(2007) SHX30 / 3:5	13	Fan 3	LOWER CRITICAL
11	2020-02-21 22:21:20	MINOR	SENSOR	no	(2007) SHX30 / 3:5	14	Fan 4	LOWER MINOR
12	2020-02-21 22:21:20	MAJOR	SENSOR	no	(2007) SHX30 / 3:5	14	Fan 4	LOWER MAJOR

To acknowledge an alarm, select it in the table window and press the “Acknowledge” button at the bottom of the window.

To delete an alarm, select it in the table window and press the “Remove” button at the bottom of the window.

To refresh current view, press the “Refresh” button at the bottom of the window.

20 MCB Instruments

The Master Control Board (MCB) of the Guardian Management Gateway hosts the single-board computer that runs Guardian Management Gateway firmware. Also it hosts several hardware entities that are exposed to the user as sensor and controls. The MCB itself is exposed as the resource 3000.

There are following sensors and controls on the MCB resource:

- Sensor "MCB Temperature" (#1): reports the temperature measured on the MCB, in degrees C
- Sensor "MCB 12V" (#2): reports the 12V voltage on the MCB, in volts
- Sensor "Reboot Reason" (#3): the discrete sensor reports the reason of the last reboot of the Guardian Management Gateway (see details below)
- Sensor "USB1 Power Fault" (#4): the discrete sensor that reports the power fault state of USB1 interface (see below for the state meaning for this and subsequent sensors)
- Sensor "USB2 Power Fault" (#5): the discrete sensor that reports the power fault state of USB2 interface
- Sensor "MGMT 12V Power Fault" (#6): the discrete sensor that reports the fault state of the external +12V power line
- Sensor "I2C_1 Bus Fault" (#7): the discrete sensor that reports the fault state of the I2C bus #1
- Sensor "I2C_2 Bus Fault" (#8): the discrete sensor that reports the fault state of the I2C bus #2
- Control "Buzzer" (#1): a digital control that controls a buzzer located on the MCB, set to *ON* to turn the buzzer on, set to *OFF* to turn it off
- Control "USB1 Power Fault Reset" (#2), a digital control, set to `Pulse ON` to reset the power fault state of the USB1 interface
- Control "USB2 Power Fault Reset" (#3), a digital control, set to `Pulse ON` to reset the power fault state of the USB2 interface
- Control "MGMT 12V Power Fault Reset" (#4), a digital control, set to `Pulse ON`" to reset the power fault state of the external +12V interface

For the "Reboot Reason" sensor, the states have the following meaning:

- State 0 (State Mask 1): the last boot was a power-on
- State 1 (State Mask 2): the last reboot was caused by a watchdog timer
- State 2 (State Mask 4): the last reboot was caused by software (e.g. a CLI command or Web interface action)
- State 3 (State Mask 8): the last reboot was caused by hardware reset
- State 4 (State Mask 0x10): the last reboot was caused by a firmware upgrade.

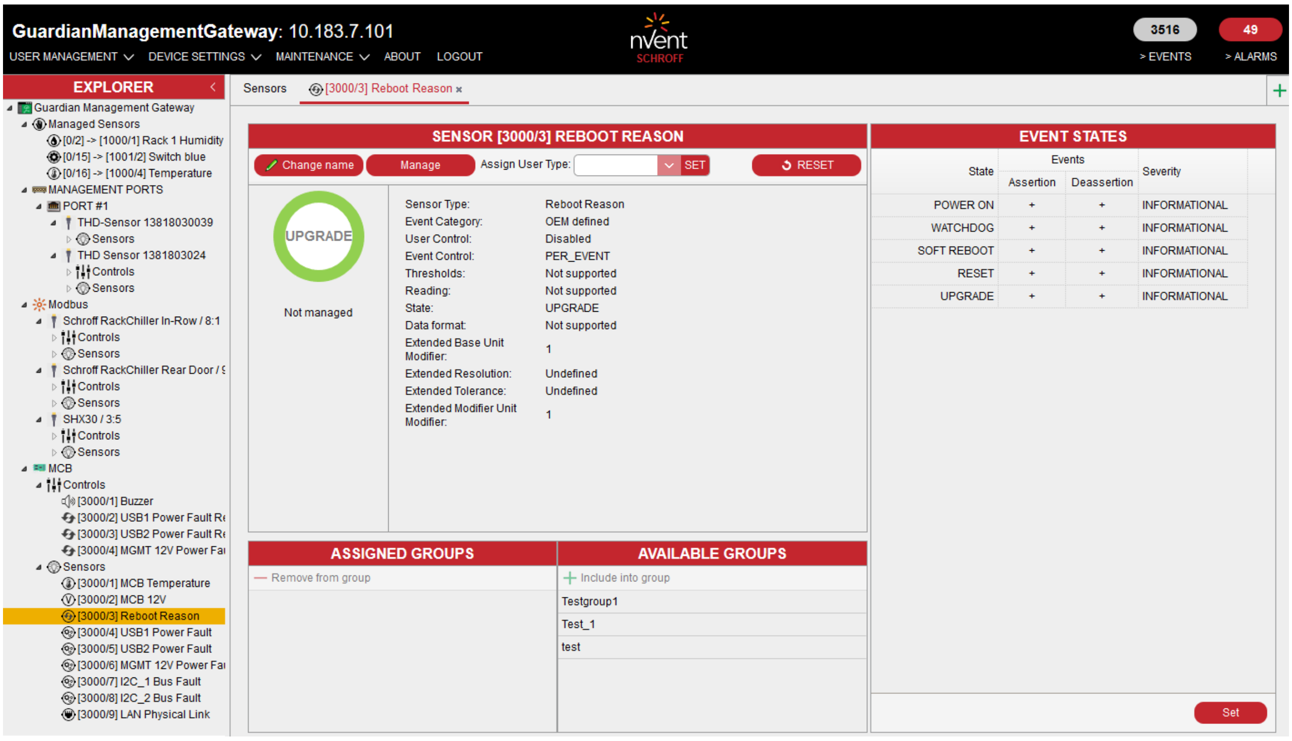
For the fault sensors, the states have the following meaning:

- State 0 (State Mask 1): no fault
- State 1 (State Mask 2): a fault is present
- State 2 (State Mask 4): the corresponding subsystem is turned off

For the LAN state sensors, the states have the following meaning:

1. State 0 (State Mask 1): no LAN physical link
2. State 1 (State Mask 2): the LAN physical link is present

The inventory #0 is present on the MCB resource. This inventory contains standard FRU information about the MCB (manufacturer, product name, serial number, etc.) and the LCD Calibration Parameters record in the nVent OEM format. It is read-only and stored in the EEPROM physically located on the MCB.



SENSOR [3000/3] REBOOT REASON

Sensor Type: Reboot Reason
Event Category: OEM defined
User Control: Disabled
Event Control: PER_EVENT
Thresholds: Not supported
Reading: Not supported
State: UPGRADE
Data format: Not supported
Extended Base Unit Modifier: 1
Extended Resolution: Undefined
Extended Tolerance: Undefined
Extended Modifier Unit Modifier: 1

EVENT STATES

State	Events		Severity
	Assertion	Deassertion	
POWER ON	+	+	INFORMATIONAL
WATCHDOG	+	+	INFORMATIONAL
SOFT REBOOT	+	+	INFORMATIONAL
RESET	+	+	INFORMATIONAL
UPGRADE	+	+	INFORMATIONAL

ASSIGNED GROUPS

— Remove from group

AVAILABLE GROUPS

+ Include into group

Testgroup1

Test_1

test

21 Restart, Reboot and Factory Reset

There are three types of restart applicable to a Schroff Guardian Management Gateway:

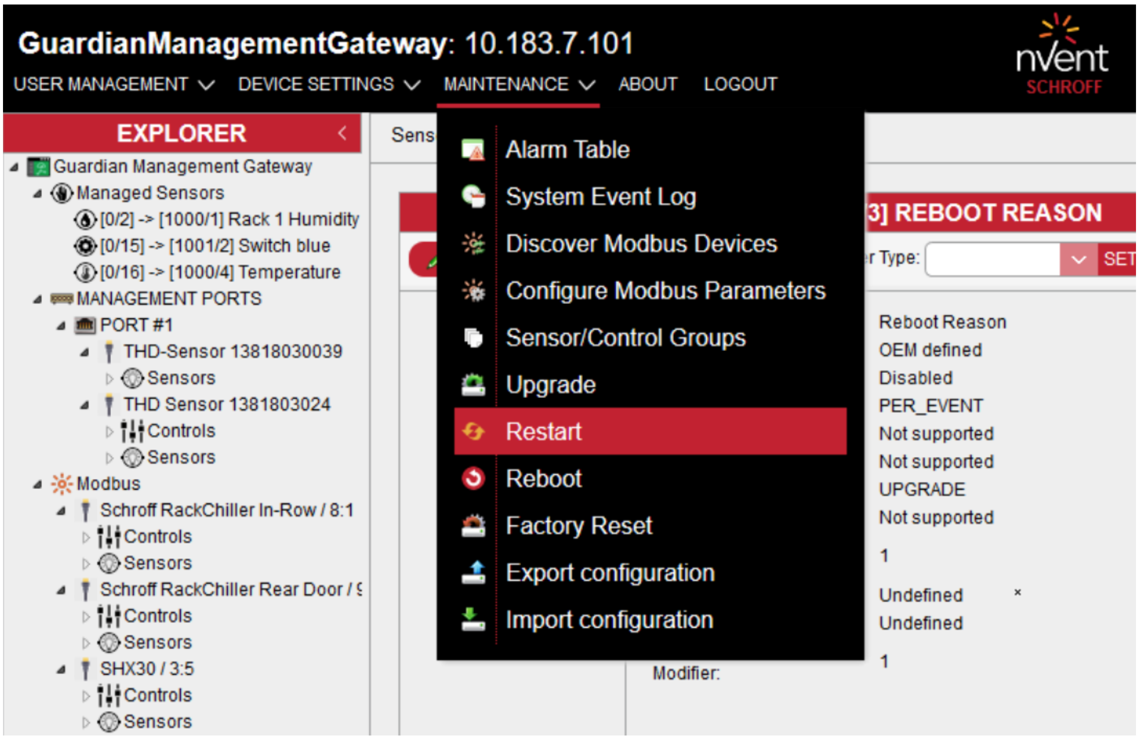
1. Restart is a termination and relaunch of the application (`smrc`) that runs on the MCB CPU and manages the Guardian Management Gateway functionality. The operating system (Linux) running on that CPU is not affected. This is the fastest type of restart
2. Reboot means reboot of the operating system running on the MCB CPU. After the restart of the operating system, the managing application is started automatically. A reboot takes longer than a restart, because the operating system gets involved. A reboot can be caused by a hardware reset or by a software command
3. Factory reset involves clearing of all configuration data on the Guardian Management Gateway and return to factory default settings. A factory reset involves a reboot.

Reboot (hardware reset) and factory reset can be initiated from the front panel, by pressing hardware buttons; there are two buttons, “Reset” and “Recovery” which are recessed to prevent them from being accidentally pressed. A sharp object like a tip of a pen is needed to press them.

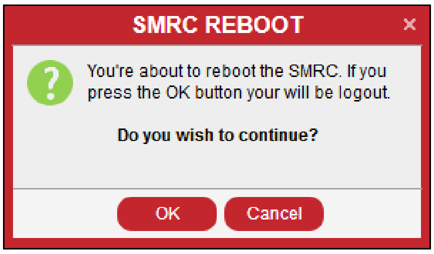
To initiate a hardware reset from the front panel, press the “Reset” button.

To initiate a factory reset from the front panel, press and hold the “Recovery” button, then press the “Reset” button and then release the “Recovery” button.

To initiate a restart, reboot or factory reset with the Web interface, use the menu commands “Maintenance” -> “Restart”, “Maintenance” -> “Reboot” and “Maintenance” -> “Factory Reset”, respectively. In all three cases, a confirmation dialog is shown to prevent accidental invocation of the command.



For example, in the case of reboot, the following dialog will be shown:



You're about to reboot the SMRC. If you press the OK button your will be logout.

Do you wish to continue?

22 Firmware Upgrade

Updated firmware images are periodically released by nVent and made accessible to customers.

Upgrading Guardian Management Gateway firmware is done in the following way:

1. A new firmware image is downloaded on the Guardian Management Gateway.
2. The firmware is installed to the flash partition.
3. The Guardian Management Gateway is rebooted to activate the new firmware.

Each firmware image contains everything needed for Guardian Management Gateway operation: the U-Boot, the Linux operating system kernel and the root file system that hosts system utilities and applications.

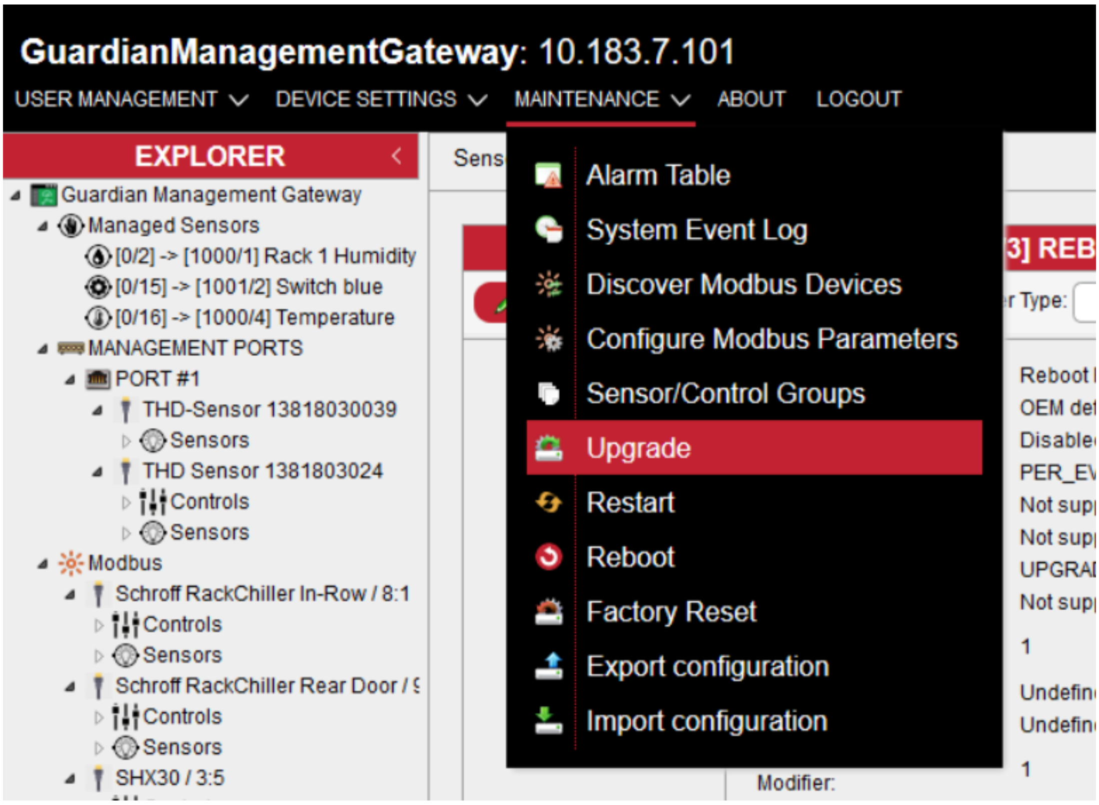
The image is protected by a digital signature (SHA256 digest) to ensure its integrity and authenticity. The signature is added to the image when it is created. When the installation of a firmware upgrade image is requested, the image signature is verified against a public key stored on the iPDU file system. If the signature is absent or is not valid, the image is rejected.

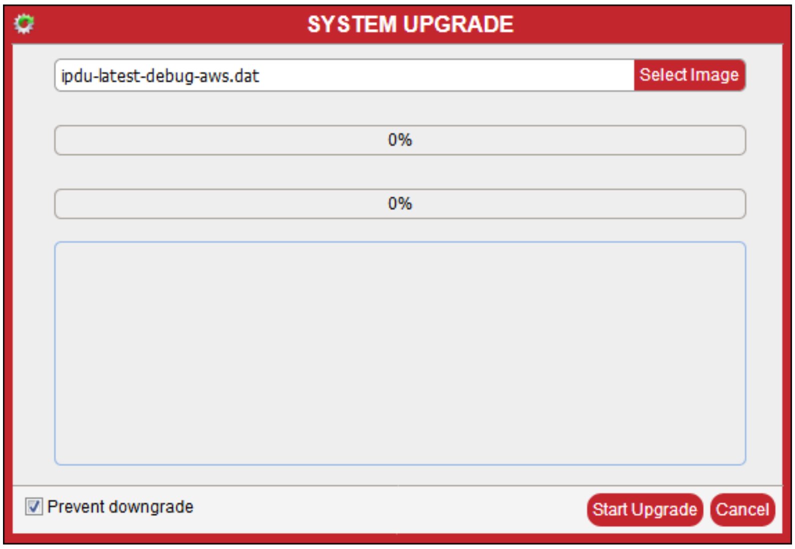
If the signature is valid, the following conditions are guaranteed to be met:

4. The image is not corrupted (since otherwise the signature would no longer match the calculated digest).
5. The image comes from nVent (since no one else has our private key, which is necessary to generate a valid signature).

Firmware upgrade does not affect the Guardian Management Gateway configuration (i.e. does not reset settings previously made).

To perform firmware upgrade via the Web interface, use the “System Upgrade” dialog invoked with the menu command “Maintenance” -> “Upgrade”.



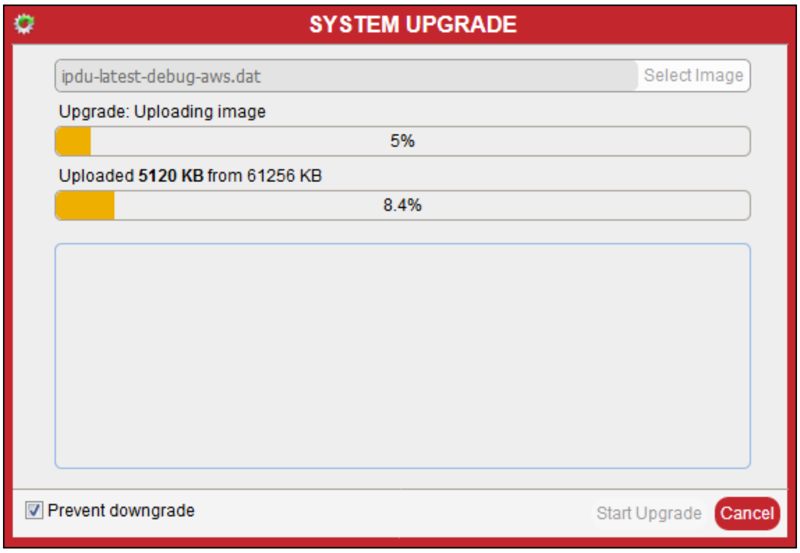

SYSTEM UPGRADE

ipdu-latest-debug-aws.dat

In this dialog, the user chooses the upgrade image located on the local (client) file system. This file is downloaded to the Guardian Management Gateway (into the temporary directory), then installed in the flash partition and then the Guardian Management Gateway is rebooted to activate the new firmware.

Check the check box "Prevent downgrade" (it is checked by default) to disallow downgrade (installation of images with the version less or equal to the current version); the meaning of this check box is opposite to the meaning of the option $-f$ for the CLI.

After the image file is chosen, press the button "Start Upgrade" to start image download and installation. The installation progress is reflected in the progress bars (the first progress bar corresponds to the whole firmware upgrade procedure, the second progress bar reflects the progress of a specific firmware upgrade stage). Press "Cancel" during this phase to cancel the upgrade.

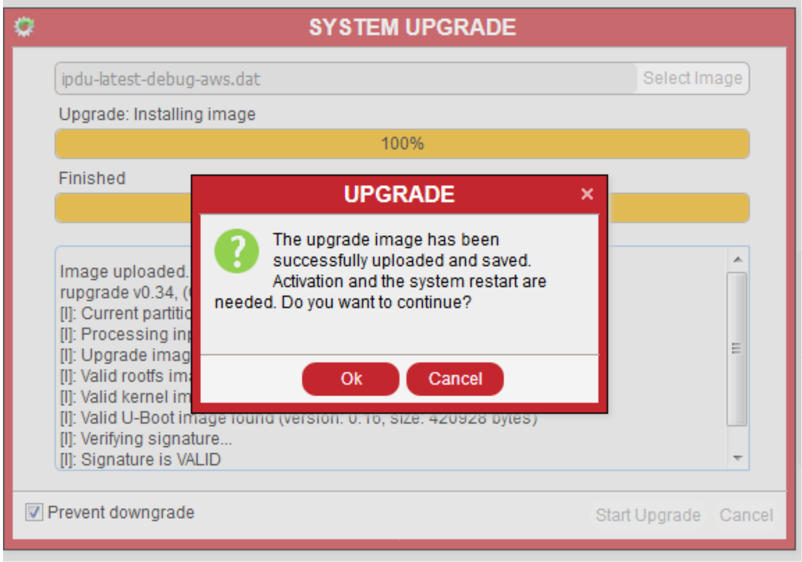

SYSTEM UPGRADE

ipdu-latest-debug-aws.dat

Upgrade: Uploading image

Uploaded **5120 KB** from 61256 KB

After the image is downloaded and installed in the flash partition, the user is presented with the dialog, asking to confirm activation of the new image:



SYSTEM UPGRADE

ipdu-latest-debug-aws.dat

Upgrade: Installing image

100%

Finished

```
Image uploaded.
rupgrade v0.34, (
[I]: Current partitio
[I]: Processing in
[I]: Upgrade imag
[I]: Valid rootfs im
[I]: Valid kernel im
[I]: Valid U-Boot image found (version: 0.16, size: 420928 bytes)
[I]: Verifying signature...
[I]: Signature is VALID
```

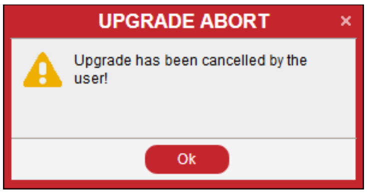
UPGRADE

The upgrade image has been successfully uploaded and saved. Activation and the system restart are needed. Do you want to continue?

☑ Prevent downgrade

If the user agrees to the request, then the 01Guardian Management Gateway is rebooted and the new image is activated.

If the user declines the activation request, the upgrade is canceled and the image installation is rolled back. The window below informs the user:



UPGRADE ABORT

Upgrade has been cancelled by the user!

The currently running firmware will continue to run, even after future reboots.

23 Saving and Loading Configuration

Guardian Management Gateway configuration includes data items and values that are persistent across system reboots and SMRC application restarts. It is stored in the flash file system as a collection of JSON files, each file storing data for a specific component of the configuration.

In addition, configuration is periodically archived and stored in the Guardian Management Gateway EEPROM. This is to facilitate hot swapping of management controller boards (MCBs) between Guardian Management Gateways. In the case of a hot swap, the configuration stays with the Guardian Management Gateway and can be obtained and applied on the newly inserted MCB.

Configuration consists of following components:

- Global settings
- Network configuration settings
- Host name
- Network service configuration settings
- List of users and their properties
- List of roles and their properties
- SNMPv3 user settings
- Security settings (firewalls and login restrictions)
- SSL certificate for the HTTP server
- LDAP settings
- Rules for events handling with corresponding actions
- User-defined resource names
- Configuration of physical sensors (with user-defined sensor names)
- Configuration of controls (user names assigned to controls)
- List of sensor/control groups, their contents and properties
- List of managed sensors and their properties
- Server reachability settings
- Resource map for 1-wire devices
- Resource map for Modbus devices

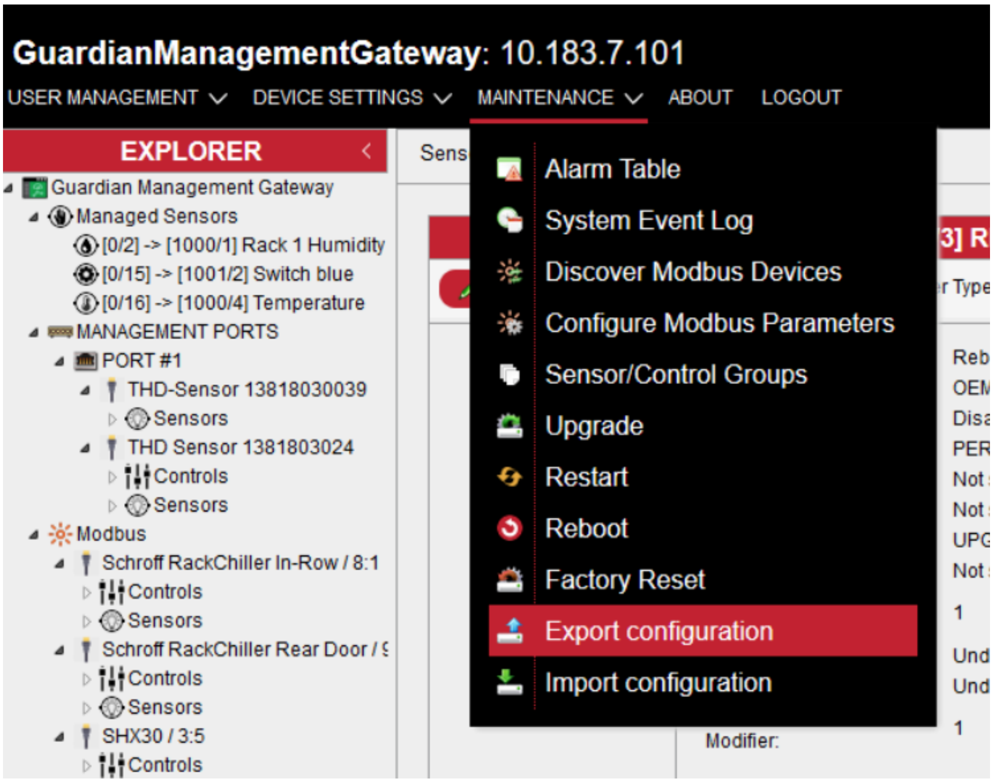
For a user, the following actions are available:

- Save configuration in an archive and download it to an external server or copy to the USB stick
- Load and apply configuration from an archive from an external server or from the USB stick
- View the list of available configuration archives on the USB stick.

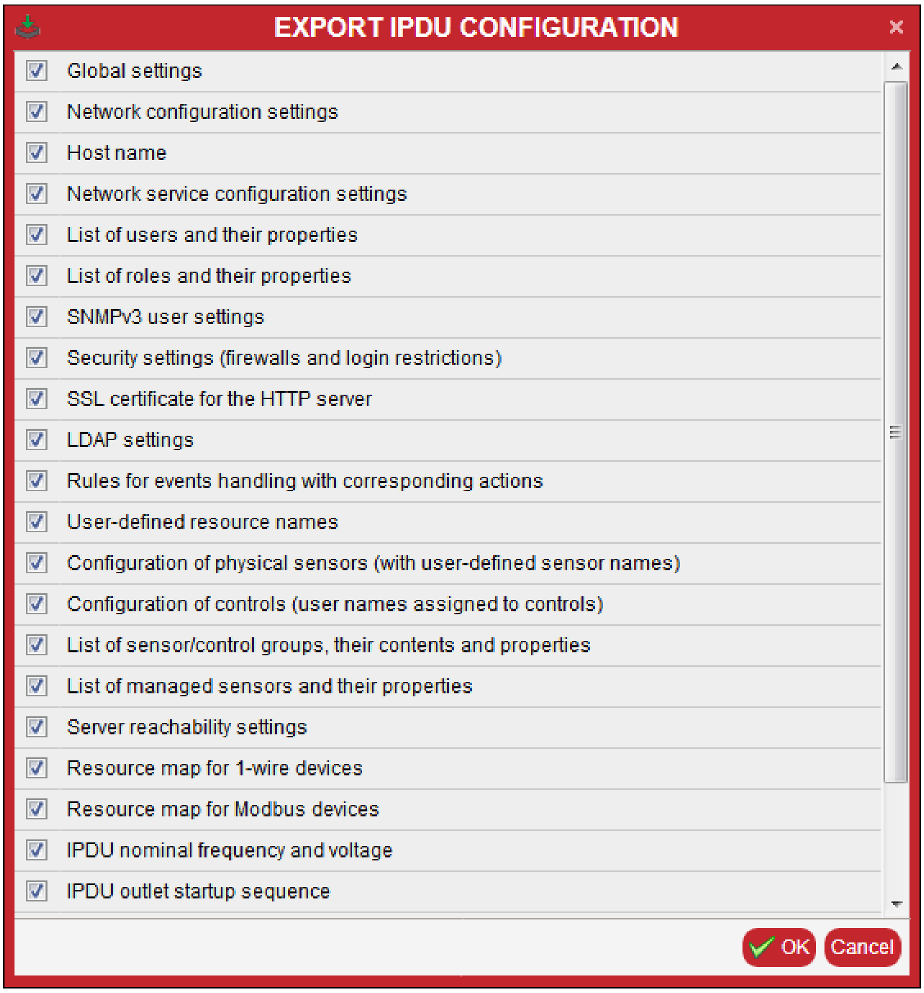
A configuration archive saved on one Guardian Management Gateway can then be loaded on another Guardian Management Gateway in order to duplicate configuration from one Guardian Management Gateway to another. It is also possible to apply this operation to multiple Guardian Management Gateways in turn if their configuration should be similar. The components that are different between these Guardian Management Gateways should not be included into the configuration archive during the save operation. This configuration transfer could be done with the use of the USB stick, or of a remote location via the Web interface.

In addition, the scenario of MCB hot replacement should be considered. In this case, configuration is loaded from the EEPROM belonging to the Guardian Management Gateway and is applied on the newly inserted MCB.

23.1 Saving current configuration



With the Web interface, the configuration archive can be downloaded from the Guardian Management Gateway to the client system. To do that, invoke the menu command “Maintenance” -> “Export configuration”. The dialog appears, in which the user can choose the components to be saved:



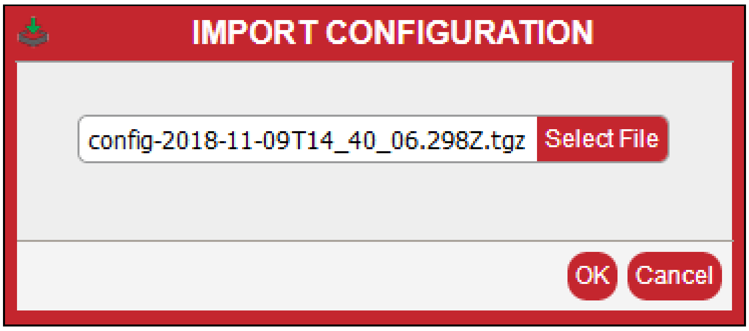
EXPORT IPDU CONFIGURATION

- ☑ Global settings
- ☑ Network configuration settings
- ☑ Host name
- ☑ Network service configuration settings
- ☑ List of users and their properties
- ☑ List of roles and their properties
- ☑ SNMPv3 user settings
- ☑ Security settings (firewalls and login restrictions)
- ☑ SSL certificate for the HTTP server
- ☑ LDAP settings
- ☑ Rules for events handling with corresponding actions
- ☑ User-defined resource names
- ☑ Configuration of physical sensors (with user-defined sensor names)
- ☑ Configuration of controls (user names assigned to controls)
- ☑ List of sensor/control groups, their contents and properties
- ☑ List of managed sensors and their properties
- ☑ Server reachability settings
- ☑ Resource map for 1-wire devices
- ☑ Resource map for Modbus devices
- ☑ IPDU nominal frequency and voltage
- ☑ IPDU outlet startup sequence

After the desired set of components is chosen, the configuration archive is created on the Guardian Management Gateway and then downloaded to the client system (normally to the system “Download” directory).

23.2 Loading configuration

In the Web interface, a configuration archive can be uploaded from the client system to the Guardian Management Gateway and the configuration will be applied. To do that, invoke the menu command “Maintenance” -> “Import configuration”. The dialog “Import Configuration” appears, in which the user can choose the configuration archive to upload:



IMPORT CONFIGURATION

config-2018-11-09T14_40_06.298Z.tgz

After the user chooses the target file and presses the OK button, the configuration archive is transferred to the Guardian Management Gateway, the SMRC application is restarted and the new configuration is applied.

24 Using SNMP

The Guardian Management Gateway supports a Simple Network Management Protocol (SNMP) interface to that allows accessing configuration, control variables and sensor readings. The following groups of variables are supported by this interface:

- System Configuration
- Power management: plugs, phases, branches and outlets
- Physical Sensors
- Managed Sensors, including sensor log
- Controls
- Schroff SHX Devices
- Server reachability table
- System Event Log

According to SNMP rules, the variables from these groups are represented via a hierarchical data model, each variable identified via an object identifier (OID). These object identifiers have a common root OID:

```
iso(1).org(3).dod(6).internet(1).private(4).enterprises(1).nvent(16394).products
(2). smartGuardian Management Gatewayler(3).smrc(1)
```

16394 is a unique private enterprise number for nVent, Schroff GmbH (formerly Pentair Technical Products, referenced as nVent in this document), obtained from IANA. In the remainder of this section, the root OID is denoted as *<ROOTOID>*.

The structure of the branches of the SNMP variables tree is described in the following subsections.

The definition of SNMP variables provided by the Smart Rack Controller is contained in a Management Information Base (MIB) file *SMRC-MIB.txt*. This file should be installed on the management system (the client system, that interacts with the Smart Rack Controller over the network). It depends on the SNMP client software how the MIB file should be installed on the management system; usually this file must be placed in a special location on the management system or compiled with a MIB compiler. If the MIB is not installed on the client system, SNMP communication with the Smart Rack Controller is still possible; however symbolic names for the OIDs are not available and OIDs should be used in numeric form.

It should be mentioned that access to some SNMP variables may require communication with physical devices or EEPROM data read operations to be invoked. In some cases such operations (e.g. accessing Controls) may take a rather long time. It is recommended to set the SNMP client timeout to 15 seconds. For example to retrieve the entire Smart Rack Controller tree (i.e. everything starting with *<ROOTOID>*) from server with IP address 192.168.0.1 via SNMPv1, run

```
$ snmpwalk -v1 -c public -t 15 192.168.0.1 1.3.6.1.4.1.16394.2.3.1
```

or, assuming SMRC-MIB is installed at SNMP client system, run

```
$ snmpwalk -v1 -c public -t 15 192.168.0.1 SMRC-MIB::smrc
```

Examples in this chapter refer to OIDs in numeric form e.g. *<ROOTOID>.1.1.4.0* which means the variable can be accessed via the following command:

```
$ snmpwalk -v1 -c public -t 15 192.168.0.1 SMRC-MIB::smrc.1.1.4.0
```

24.1 Guardian Management Gateway specific data types

The Management Information Base (MIB) file defines several additional data types like SensorType, SensorUnit, SensorCategory, EventType, SeverityType, ControlType and ControlOutput that are used in variable description below. To avoid text duplication, this chapter describes such data types in the tables below.

Table 5: SensorType values

VALUE	DESCRIPTION		VALUE	DESCRIPTION
1	Temperature		26	Other FRU
2	Voltage		27	Cable Interconnect
3	Current		28	Terminator
4	Fan (Tachometer)		29	System Boot Initiated
5	Physical Security		30	Boot Error
6	Platform Violation		31	OS Boot
7	Processor		32	OS Critical Stop
8	Power Supply		33	Slot Connector
9	Power Unit		34	System ACPI Power State
10	Cooling Device		35	reserved
11	Other Units Based Sensor		36	Platform Alert
12	Memory		37	Entity Presence
13	Drive Slot		38	Monitor ASIC IC
14	Post Memory Resize		39	LAN
15	System FW Progress		40	Management Subsystem Health
16	Event Logging Disabled		41	Battery
17	reserved		42	System Audit
18	System Event		43	Version Change
19	Critical Interrupt		160	Operational
20	Button		192	OEM Sensor
21	Module Board		65537	Comm Channel Link State
22	Microcontroller Coprocessor		65538	Management Bus State
23	Add In Card		65539	Comm Channel Bus State
24	Chassis		65540	Config Data
25	Chipset		65541	Power Budget

Table 6: SensorUnit values

VALUE	DESCRIPTION		VALUE	DESCRIPTION
-1	Unspecified		46	Ft-Lb
1	Degrees C		47	Oz-In
2	Degrees F		48	Gauss
3	Degrees K		49	Gilberts
4	Volts		50	Henry

Value	Description		Value	Description
5	Amps		51	Millihenry
6	Watts		52	Farad
7	Joules		53	Microfarad
8	Coulombs		54	Ohms
9	VA		55	Siemens
10	Nits		56	Mole
11	Lumen		57	Becquerel
12	Lux		58	Ppm
13	Candela		59	reserved
14	Kpa		60	Decibels
15	Psi		61	Dba
16	Newton		62	Dbc
17	Cfm		63	Gray
18	Rpm		64	Sievert
19	Hz		65	Color Temp Degrees K
20	Microseconds		66	Bits
21	Milliseconds		67	Kilobits
22	Seconds		68	Megabits
23	Minutes		69	Gigabits
24	Hours		70	Bytes
25	Days		71	Kilobytes
26	Weeks		72	Megabytes
27	Mil		73	Gigabytes
28	Inches		74	Words
29	Feet		75	DWords
30	Cubic Inches		76	QWords
31	Cubic Feet		77	Lines
32	mm		78	Hits
33	cm		79	Misses
34	m		80	Retries
35	Cubic cm		81	Resets
36	Cubic m		82	Overruns
37	Liters		83	Underruns
38	Fluid Ounce		84	Collisions
39	Radians		85	Packets
40	Steradians		86	Messages
41	Revolutions		87	Characters
42	Cycles		88	errors
43	Gravities		89	Correctable Errors

Value	Description		Value	Description
44	Ounces		90	Uncorrectable Errors
45	Pounds			

Table 7: `SensorCategory` values

Value	Description		Value	Description
-1	Unspecified		7	Severity
1	Threshold		8	Presence
2	Usage		9	Enable
3	State		10	Availability
4	Predicted Fail		11	Redundancy
5	Limit		126	Sensor Specific
6	Performance		127	Generic

Table 8: `EventType` values

Value	Description
1	Resource
2	Domain
3	Sensor
4	Sensor Enable Change
5	Hot Swap
6	Watchdog
7	HPI SW
8	OEM
9	User
10	DIMI
11	DIMI Update
12	FUMI

Table 9: `SeverityType` values

Value	Description
1	Critical
2	Major
3	Minor
4	Informational
5	OK
241	Debug
255	All

Table 10: `ControlType` values

VALUE	DESCRIPTION
1	Digital
2	Discrete
3	Analog
4	Stream
5	Text
193	OEM

Table 11: `ControlOutput` values

VALUE	DESCRIPTION
1	Generic
2	LED
3	Fan Speed
4	Dry Contact Closure
5	Power Supply Inhibit
6	Audible
7	Front Panel Lockout
8	Power Interlock
9	Power State
10	LCD Display
11	OEM
12	Generic Address
13	IP Address
14	Resource ID
15	Power Budget
16	Activate
17	Reset

24.2 Configuration MIB variables

The variables defined in this section contain information about the Guardian Management Gateway configuration, including configuration of system, controls, sensors, managed sensors, managed sensor log. Currently, most of the configuration variables are read-only but in future, the number of read-write variables may be increased, to improve management capabilities via the SNMP interface.

Basic system configuration variables have the following OID, where <var> is the variable index:

<ROOTOID>.1.1.<var>.0

Table 12: Basic system configuration indices

VARIABLE	INDEX	TYPE	ACCESS MODE	DESCRIPTION
shxCount	1	INTEGER	Read-only	The number of SHX units (Carel RDC included) supported.
unitName	2	STRING	Read-write	System host name.
hardwareVersion	3	STRING	Read-only	Hardware version of the main board.
firmwareVersion	4	STRING	Read-only	System firmware version.
utcOffset	5	STRING	Read-only	UTC offset of the system time.
resourceCount	6	INTEGER	Read-only	The number of resources in the system.
totalSensorCount	7	INTEGER	Read-only	The number of external (physical) sensors.
managedSensorCount	8	INTEGER	Read-only	The number of managed sensors.

VARIABLE	INDEX	TYPE	ACCESS MODE	DESCRIPTION
externalSensorsZCoordinateUnits	9	INTEGER	Read-only	External Sensor Z Coordinate units: Freeform Text (T) or Rack Units (U).
serverCount	10	INTEGER	Read-only	The number of entries in serverReachabilityTable.
model	11	STRING	Read-only	The device model name.
cascadedDeviceConnected	12	INTEGER (TruthValue)	Read-only	Reserved for future use
unitsTemperature	14	STRING	Read-only	The global temperature measurement units: Celsius or Fahrenheit.
unitsLength	15	STRING	Read-only	The global length measurement units: Meters or Feet.
unitsPressure	16	STRING	Read-only	The global pressure measurement units: PSI or Pascals.

For example, to retrieve the system firmware version, use the following OID:

```
<ROOTOID>.1.1.4.0
snmpwalk -v1 -c private 80.240.102.34 SMRC-MIB::unitConfiguration
SMRC-MIB::shxCount.0 = INTEGER: 0
SMRC-MIB::unitName.0 = STRING: Ipdu00001
SMRC-MIB::hardwareVersion.0 = STRING: 0.1
SMRC-MIB::firmwareVersion.0 = STRING: 0.95.1 63998-20551-17 AWS
Nov 21 2018
18:28:15
SMRC-MIB::utcOffset.0 = STRING: +0000
SMRC-MIB::resourceCount.0 = INTEGER: 6
…..
```

Also, there is the networkConfigurationTable table in this section that contains parameters of the system network interfaces that have the following OIDs, where <var> is the variable index from the table below and <entry> is the entry number:

<ROOTOID>.1.1.13.1.<var>.<entry>

Table 13: Network interface table variables

VARIABLE	INDEX	TYPE	ACCESS MODE	DESCRIPTION
networkInterfaceId	1	INTEGER	Read-only	Index of the network interface, equal to <entry>
networkInterfaceName	2	STRING	Read-only	Network interface name, e.g “eth0” or “wlan0”.
networkInterfaceMacAddress	3	STRING	Read-only	MAC Address.
networkInterfaceIPv4UseDHCP	4	INTEGER (TruthValue)	Read-only	Indicates whether IPv4 DHCP used: true (1) or false (2).

VARIABLE	INDEX	TYPE	ACCESS MODE	DESCRIPTION
networkInterfaceIpV4Address	5	STRING	Read-only	IPv4 address with the number of significant bits in the network mask, e.g. “192.16.1.35/24".
networkInterfaceIpV4Gateway	6	STRING	Read-only	IPv4 gateway address.
networkInterfaceIPv6UseDHCP	7	INTEGER (TruthValue)	Read-only	Indicates whether IPv6 DHCP used: true (1) or false (2).
networkInterfaceIpV6Addresses	8	STRING	Read-only	IPv6 address with scope.

The following command retrieves information on network interfaces at the Guardian Management Gateway.

```
snmpwalk -v1 -c private 192.168.0.1 SMRC-MIB::networkConfigurationTable
SMRC-MIB::networkInterfaceName.1 = STRING: lo
SMRC-MIB::networkInterfaceName.2 = STRING: eth0
SMRC-MIB::networkInterfaceName.3 = STRING: eth1
SMRC-MIB::networkInterfaceName.4 = STRING: sit0
SMRC-MIB::networkInterfaceMacAddress.1 = STRING: 00:00:00:00:00:00
….
```

The shxConfiguration sub-branch contains details on Side Heat eXchangers (SHX) in the system in three tables: shxConfigurationTable, shxSensorCountTable and shxSensorConfigurationTable.

The shxConfigurationTable exposes SHX device parameters that have the following OIDs, where <var> is the variable index described below and <resource> is the resource ID of the SHX device.

<ROOTOID>.1.2.1.1.<var>.<resource>

Table 14: SHX Configuration table variables

VARIABLE	INDEX	TYPE	ACCESS MODE	DESCRIPTION
shxResourceId	1	INTEGER	Read-only	Resource ID of the SHX device, equal to <resource>. Resource IDs for SHX devices are in the range 2000 to 2999.
shxOperationalState	2	INTEGER	Read-write	The operational state of the SHX controller: disconnected(-1), offline(2) or online(1). To switch SHX power state while controller is connected, set shxOperationalState to 2 (offline) or 1 (online).
shxValvePosition	3	INTEGER	Read-only	The current opening state of the water valve (in percentages from 0 to 100).
shxCoolerTempSetpoint	4	INTEGER	Read-write	The setpoint for the desired temperature.
shxFanPerformanceSetpoint	5	INTEGER	Read-write	The fan performance setpoint, in percents
shxMaximumCooling	6	INTEGER (TruthValue)	Read-write	Indicates whether maximum cooling is requested (1) or not (2). To request maximum cooling, set shxMaximumCoolingState to 1 (true).
shxAlertState	7	INTEGER (TruthValue)	Read-write	Indicates whether SHX controller is in alert state (1) or not (2). To acknowledge alert status, set shxAlertState to 2 (false).

VARIABLE	INDEX	TYPE	ACCESS MODE	DESCRIPTION
shxModel	8	STRING	Read-only	The model identifier of an SHX controller
shxFirmwareVersion	9	STRING	Read-only	The firmware version of an SHX controller

For example, to retrieve firmware versions of SHX devices, use the following OID:

<ROOTOID>.1.2.1.1.9

The shxSensorCountTable exposes the number of sensors of SHX devices that have the following OIDs, where <resource> is the resource ID of SHX device.

<ROOTOID>.1.2.2.1.2.<resource>

The shxSensorConfigurationTable exposes sensor parameters of SHX devices that have the following OIDs, where <var> is the variable index described below, <resource> is the resource ID of SHX device and <sensor> is the sensor number.

<ROOTOID>.1.2.3.1.<var>.<resource>.<sensor>

Table 15: SHX Sensor Configuration table variables

VARIABLE	INDEX	TYPE	ACCESS MODE	DESCRIPTION
shxSensorId	1	INTEGER	Read-only	Sensor ID of the SHX device, equal to <sensor>.
shxInterface	2	INTEGER	Read-only	SHX sensor interface number.
shxAddress	3	INTEGER	Read-only	SHX sensor device address.
shxSensorName	4	STRING	Read-only	User-defined name of the sensor (e.g. Fan Speed 1).
shxSensorType	5	SensorType	Read-only	The sensor type. This data type is described in Table 5.
shxSensorCategory	6	SensorCategory	Read-only	The sensor category. This data type is described in Table 7.
shxSensorEnableControl	7	INTEGER (TruthValue)	Read- only	Indicates whether sensor control is enabled(1) or disabled(2).
shxSensorEventControl	8	INTEGER	Read-only	The sensor event control: Per-Event(1), Read-Only Masks(2) or Read-Only(3).
shxSensorAssertEventMask	9	STRING	Read-only	Bitmask of allowed Assertion events from the sensor, e.g. “0x003F”.
shxSensorDeassertEventMask	10	STRING	Read-only	Bitmask of allowed Deassertion events from the sensor, e.g. “0x003F”.
shxSensorIsReadingSupported	11	INTEGER (TruthValue)	Read-only	Indicates whether sensor reading is supported(1) or not supported(2).
shxSensorBaseUnit	12	SensorUnit	Read-only	The base units (this data type is described in Table 6). This parameter does not apply to discrete sensors.
shxSensorModifierUnit	13	SensorUnit	Read-only	The sensor modifier unit (this data type is described in Table 6 in the section **Fehler! Verweisquelle konnte nicht gefunden werden.**).
shxSensorModifierUse	14	INTEGER	Read-only	A sensor modifier unit use: Basic Over Modifier(1), Basic Times Modifier(2) or None(-1).

VARIABLE	INDEX	TYPE	ACCESS MODE	DESCRIPTION
shxSensorPercentage	15	INTEGER (TruthValue)	Read-only	Indicated whether the sensor reading is returned in percents (1) or not (2).
shxSensorAccuracy	16	FLOAT64	Read-only	The sensor accuracy.
shxSensorResolution	17	FLOAT64	Read-only	The sensor resolution.
shxSensorTolerance	18	FLOAT64	Read-only	The sensor tolerance.
shxSensorMaximum	19	FLOAT64	Read-only	The largest possible value. This parameter does not apply to discrete sensors.
shxSensorMinimum	20	FLOAT64	Read-only	The smallest possible value. This parameter does not apply to discrete sensors.
shxSensorThresholdsIsAccessible	21	INTEGER (TruthValue)	Read-only	Indicates whether sensor thresholds are accessible(1) or not (2).
shxSensorLowerCriticalThreshold	22	FLOAT64	Read- write	The lower critical threshold. This parameter does not apply to discrete sensors.
shxSensorLowerMajorThreshold	23	FLOAT64	Read- write	The lower major threshold. This parameter does not apply to discrete sensors.
shxSensorLowerMinorThreshold	24	FLOAT64	Read- write	The lower minor threshold. This parameter does not apply to discrete sensors.
shxSensorUpperCriticalThreshold	25	FLOAT64	Read- write	The upper critical threshold. This parameter does not apply to discrete sensors.
shxSensorUpperMajorThreshold	26	FLOAT64	Read- write	The upper major threshold. This parameter does not apply to discrete sensors.
shxSensorUpperMinorThreshold	27	FLOAT64	Read- write	The upper minor threshold. This parameter does not apply to discrete sensors.
shxSensorPositiveHysteresis	28	FLOAT64	Read- write	The positive hysteresis used for deassertions. This parameter does not apply to discrete sensors.
shxSensorNegativeHysteresis	29	FLOAT64	Read- write	The negative hysteresis used for deassertions. This parameter does not apply to discrete sensors.
shxSensorPollInterval	30	INTEGER	Read- write	The sensor polling interval in milliseconds.
shxSensorAssertionDelayCount	31	INTEGER	Read- write	The delay measured in samples before a state is asserted. If the value is zero, then the state is asserted as soon as it is detected; if it is non-zero, say n, then the assertion condition must exist for n+1 consecutive samples before the corresponding assertion event is reported.

For example, to retrieve all SHX sensor names, use the following OID:

<ROOTOID>.1.2.3.1.4

The managedSensorConfigurationTable exposes sensor parameters of managed sensors (i.e. virtual replicas of physical sensors located at resource 0) that have the following OIDs, where <var> is the variable index described below and <msensor> is the managed sensor number.

<ROOTOID>.1.3.1.<var>.<msensor>

Table 16: Managed sensor configuration table variables

VARIABLE	INDEX	TYPE	ACCESS MODE	DESCRIPTION
managedSensorId	1	INTEGER	Read-only	Managed sensor ID, equal to <msensor>.
managedSensorType	2	SensorType	Read-only	The sensor type. This data type is described in the previous chapter.
managedSensorName	3	STRING	Read- write	The user-defined name of the sensor (e.g. Fan Speed 1); defaults to the original physical sensor name if not changed by the user.
managedSensorDescription	4	STRING	Read- write	The user-defined description of the sensor.
managedSensorXCoordinate	5	STRING	Read- write	The X coordinate of the sensor location.
managedSensorYCoordinate	6	STRING	Read- write	The Y coordinate of the sensor location.
managedSensorZCoordinate	7	STRING	Read- write	The Z coordinate of the sensor location.
managedSensorSubtype	8	STRING	Read- write	Type of measurement in case the sensor type is discrete.
managedSensorCategory	9	SensorCategory	Read-only	The sensor category. This data type is described in Table 7.
managedSensorEnableControl	10	INTEGER (TruthValue)	Read- only	Indicates whether sensor control is enabled(1) or disabled(2).
managedSensorEventControl	11	INTEGER	Read-only	The sensor event control: Per-Event(1), Read-Only Masks(2) or Read-Only(3).
managedSensorAssertEventMask	12	STRING	Read-only	Bitmask of allowed Assertion events from the sensor, e.g. “0x003F”.
managedSensorDeassertEventMask	13	STRING	Read-only	Bitmask of allowed Deassertion events from the sensor, e.g. “0x003F”.
managedSensorIsReadingSupported	14	INTEGER (TruthValue)	Read-only	Indicates whether sensor reading is supported(1) or not supported(2).
managedSensorBaseUnit	15	SensorUnit	Read-only	The base units (this data type is described in Table 6). This parameter does not apply to discrete sensors.
managedSensorModifierUnit	16	SensorUnit	Read-only	The sensor modifier unit (data type is described in Table 6).

VARIABLE	INDEX	TYPE	ACCESS MODE	DESCRIPTION
managedSensorModifierUse	17	INTEGER	Read-only	A sensor modifier unit use: Basic Over Modifier(1), Basic Times Modifier(2) or None(-1).
managedSensorPercentage	18	INTEGER (TruthValue)	Read-only	Indicated whether the sensor reading is returned in percents (1) or not (2).
managedSensorAccuracy	19	FLOAT64	Read-only	The sensor accuracy: how close (in percents) the measurement is to the actual value. This parameter does not apply to discrete sensors.
managedSensorResolution	20	FLOAT64	Read-only	The sensor resolution: the minimum difference between any two measured values. This parameter does not apply to discrete sensors.
managedSensorTolerance	21	FLOAT64	Read-only	The sensor tolerance: the difference between a sensor value and the actual value. This parameter does not apply to discrete sensors.
managedSensorMaximum	22	FLOAT64	Read-only	The biggest possible value. This parameter does not apply to discrete sensors.
managedSensorMinimum	23	FLOAT64	Read-only	The smallest possible value. This parameter does not apply to discrete sensors.
managedSensorThresholdsIsAccessible	24	INTEGER (TruthValue)	Read-only	Indicates whether sensor thresholds are accessible (1) or not (2).
managedSensorLowerCriticalThreshold	25	FLOAT64	Read- write	The lower critical threshold. This parameter does not apply to discrete sensors.
managedSensorLowerMajorThreshold	26	FLOAT64	Read- write	The lower major threshold. This parameter does not apply to discrete sensors.
managedSensorLowerMinorThreshold	27	FLOAT64	Read- write	The lower minor threshold. This parameter does not apply to discrete sensors.
managedSensorUpperCriticalThreshold	28	FLOAT64	Read- write	The upper critical threshold. This parameter does not apply to discrete sensors.
managedSensorUpperMajorThreshold	29	FLOAT64	Read- write	The upper major threshold. This parameter does not apply to discrete sensors.
managedSensorUpperMinorThreshold	30	FLOAT64	Read- write	The upper minor threshold. This parameter does not apply to discrete sensors.

Variable	Index	Type	Access mode	Description
managedSensorPositiveHysteresis	31	FLOAT64	Read- write	The positive hysteresis used for deassertions. This parameter does not apply to discrete sensors.
managedSensorNegativeHysteresis	32	FLOAT64	Read- write	The negative hysteresis used for deassertions. This parameter does not apply to discrete sensors.
managedSensorPollInterval	33	INTEGER	Read- write	The sensor polling interval in milliseconds.
managedSensorAssertionDelayCount	34	INTEGER	Read- write	The delay measured in samples before a state is asserted. If the value is zero, then the state is asserted as soon as it is detected; if it is non-zero, say n, then the assertion condition must exist for n+1 consecutive samples before the corresponding assertion event is reported.
managedSensorResourceId	35	INTEGER	Read-only	The resource number of the original physical sensor.
managedSensorExternalSensorNumber	36	INTEGER	Read-only	The sensor number of the original physical sensor.

For example, to retrieve user-defined descriptions of all managed sensors, use the following OID:

<ROOTOID>.1.3.1.4

The controlConfigurationTable exposes parameters of controls that have the following OIDs, where <var> is the variable index described below, <resource> is the resource ID and <control> is the control number.

<ROOTOID>.1.4.1.<var>.<resource>.<control>

Table 17: Control configuration table variables

Variable	Index	Type	Access mode	Description
ctrlResourceId	1	INTEGER	Read-only	The resource number of the control, equal to <resource>.
ctrlId	2	INTEGER	Read-only	The control number, equal to <control>.
ctrlType	3	ControlType	Read-only	The control type. This data type is described in the Table 10.
ctrlOutputType	4	ControlOutput	Read-only	The control output type. This data type is described in the Table 11.
ctrlMaximumValue	5	INTEGER	Read-only	The maximum value of the control.
ctrlMinimumValue	6	INTEGER	Read-only	The minimum value of the control.
ctrlDefaultValue	7	INTEGER	Read-only	The default value of the control.
ctrlDefaultMode	8	INTEGER	Read-only	The default mode of the control: automatic(1), manual(2) or unavailable(-1).
ctrlDefaultModeReadOnly	9	INTEGER (TruthValue)	Read-only	Indicates whether the default control mode is read-only(1) or not(2).
ctrlWriteOnly	10	INTEGER (TruthValue)	Read-only	Indicates whether the control is write-only(1) or not(2).

VARIABLE	INDEX	TYPE	ACCESS MODE	Description
ctrlOem	11	INTEGER	Read-only	An OEM specific value in the control definition.

For example, to retrieve control types of all controls in the system, use the following OID:

<ROOTOID>.1.4.1.3

The logConfiguration sub-branch exposes sensor log parameters with the following OIDs, where <var> is the variable index described in the table below:

<ROOTOID>.1.5.<var>.0

Table 18: Log configuration indices

VARIABLE	INDEX	TYPE	ACCESS MODE	DESCRIPTION
logDataRetrieval	1	INTEGER (TruthValue)	Read- write	Indicates if log data retrieval is enabled(1) or disabled(2).
logMeasurementPeriod	2	INTEGER	Read- write	Data sample collection periodicity in seconds.
logSize	3	INTEGER	Read-only	The number of entries in the sensor log.

For example, to retrieve the current sensor log size, use the following OID:

<ROOTOID>.1.5.3.0

The externalSensorConfigurationTable table exposes means to configure external (physical) sensors. This table is indexed with the resource ID and sensor number. Variables from this table have the following OIDs, where <var> is the variable index from the table below:

<ROOTOID>.1.6.1.<var>.<resource>.<sensor>

Table 19: External Sensor Configuration table variables

VARIABLE	INDEX	TYPE	ACCESS MODE	DESCRIPTION
externalResourceId	1	INTEGER	Read-only	The resource number of the physical sensor.
externalSensorId	2	INTEGER	Read-only	The sensor number, unique for each sensor within a resource.
externalResourceName	3	STRING	Read- write	The name of the resource.
externalSensorName	4	STRING	Read- write	The name of the sensor (e.g. Fan Speed 1).
externalSensorType	5	SensorType	Read-only	The sensor type. This data type is described in the Table 5
externalSensorCategory	6	SensorCategory	Read-only	The sensor category. This data type is described in the Table 7.
externalSensorEnableControl	7	INTEGER (TruthValue)	Read-only	Indicates whether sensor control is enabled(1) or disabled(2).
externalSensorEventControl	8	INTEGER	Read-only	The sensor event control: Per-Event(1), Read-Only Masks(2) or Read-Only(3).
externalSensorAssertEventMask	9	STRING	Read-only	Bitmask of allowed Assertion events from the sensor, e.g. “0x003F”.
externalSensorDeassertEventMask	10	STRING	Read-only	Bitmask of allowed Deassertion events from the sensor, e.g. “0x003F”.

VARIABLE	INDEX	TYPE	ACCESS MODE	DESCRIPTION
externalSensorIsReadingSupported	11	INTEGER (TruthValue)	Read-only	Indicates whether sensor reading is supported(1) or not supported(2).
externalSensorBaseUnit	12	SensorUnit	Read-only	The base units (data type is described in the Table 6). This parameter does not apply to discrete sensors.
externalSensorModifierUnit	13	SensorUnit	Read-only	The sensor modifier unit (data type is described in the Table 6).
externalSensorModifierUse	14	INTEGER	Read-only	A sensor modifier unit use: Basic Over Modifier(1), Basic Times Modifier(2) or None(-1).
externalSensorPercentage	15	INTEGER (TruthValue)	Read-only	Indicated whether the sensor reading is returned in percents (1) or not (2).
externalSensorAccuracy	16	FLOAT64	Read-only	The accuracy: how close (in percents) the measurement is to the actual value. This parameter does not apply to discrete sensors.
externalSensorResolution	17	FLOAT64	Read-only	The resolution: the minimum difference between any two measured values. This parameter does not apply to discrete sensors.
externalSensorTolerance	18	FLOAT64	Read-only	The tolerance: the difference between a sensor value and the actual value. This parameter does not apply to discrete sensors.
externalSensorMaximum	19	FLOAT64	Read-only	The biggest possible value. This parameter does not apply to discrete sensors.
externalSensorMinimum	20	FLOAT64	Read-only	The smallest possible value. This parameter does not apply to discrete sensors.
externalSensorThresholdsIsAccessible	21	INTEGER (TruthValue)	Read-only	Indicates whether sensor thresholds are accessible (1) or not (2).
externalSensorLowerCriticalThreshold	22	FLOAT64	Read-write	The lower critical threshold. This parameter does not apply to discrete sensors.
externalSensorLowerMajorThreshold	23	FLOAT64	Read-write	The lower major threshold. This parameter does not apply to discrete sensors.
externalSensorLowerMinorThreshold	24	FLOAT64	Read-write	The lower minor threshold. This parameter does not apply to discrete sensors.

VARIABLE	INDEX	TYPE	ACCESS MODE	DESCRIPTION
externalSensorUpperCriticalThreshold	25	FLOAT64	Read-write	The upper critical threshold. This parameter does not apply to discrete sensors.
externalSensorUpperMajorThreshold	26	FLOAT64	Read-write	The upper major threshold. This parameter does not apply to discrete sensors.
externalSensorUpperMinorThreshold	27	FLOAT64	Read-write	The upper minor threshold. This parameter does not apply to discrete sensors.
externalSensorPositiveHysteresis	28	INTEGER	Read- write	The positive hysteresis used for deassertions. This parameter does not apply to discrete sensors.
externalSensorNegativeHysteresis	29	INTEGER	Read- write	The negative hysteresis used for deassertions. This parameter does not apply to discrete sensors.
externalSensorPollInterval	30	INTEGER	Read- write	The sensor polling interval in milliseconds.
externalSensorAssertionDelayCount	31	INTEGER	Read- write	The delay measured in samples before a state is asserted. If the value is zero, then the state is asserted as soon as it is detected; if it is non-zero, say n, then the assertion condition must exist for n+1 consecutive samples before the corresponding assertion event is reported.
externalSensorIsManaged	32	INTEGER (TruthValue)	Read-write	Indicates if the sensor is managed (1), or not (2). Set to 1 to manage this sensor, set to 2 to unmanage it.
externalSensorManagedNumber	33	INTEGER	Read-only	The sensor number of the corresponding managed sensor on resource 0 or -1 if the sensor is not managed.

For example, to retrieve names of all physical sensors, use the following OID:

<ROOTOID>.1.6.1.5

```
snmpwalk -v1 -c private 80.240.102.34 SMRC-MIB::externalSensorConfigurationTable
SMRC-MIB::externalResourceId.1000.1 = INTEGER: 1000
SMRC-MIB::externalResourceId.1000.2 = INTEGER: 1000
SMRC-MIB::externalResourceId.1000.3 = INTEGER: 1000
….
SMRC-MIB::externalSensorAssertionDelayCount.4002.3718 = Gauge32: 0
SMRC-MIB::externalSensorAssertionDelayCount.4002.3719 = Gauge32: 0
SMRC-MIB::externalSensorIsManaged.1000.1 = INTEGER: true(1)
```

```
SMRC-MIB::externalSensorIsManaged.1000.2 = INTEGER: true(1)
SMRC-MIB::externalSensorIsManaged.1000.3 = INTEGER: true(1)
SMRC-MIB::externalSensorIsManaged.1000.4 = INTEGER: false(2)
SMRC-MIB::externalSensorIsManaged.1000.5 = INTEGER: false(2)
SMRC-MIB::externalSensorIsManaged.1001.1 = INTEGER: false(2)
SMRC-MIB::externalSensorIsManaged.1001.2 = INTEGER: false(2)
…
```

24.3 Log MIB variables

The log branch exposes sensor log for managed sensors. This branch contains log properties variables and two tables for log timestamps and for managed sensor states. The logProperties variables have the following OIDs:

<ROOTOID>.2.1.<var>

Table 20: Log properties variables

Variable	Index	Type	Access mode	Description
logOldestId	1	INTEGER	Read-only	Index of the oldest sample in the log.
logNewestId	2	INTEGER	Read-only	Index of the newest sample in the log.

The logTimeStampTable contains timestamps or each reading sample. By default the log contains 16 samples. This table has the following OID, where <var> is the variable index and <msensor> is the managed sensor number:

<ROOTOID>.2.2.1.<var>.<entry>

Table 21: Log timestamp table variables

Variable	Index	Type	Access mode	Description
logEntryIdx	1	INTEGER	Read-only	Log entry index, equal to <entry>.
logEntryTimeStamp	2	STRING	Read-only	The time when the data was collected. It is measured in seconds relative to January 1, 1970 (midnight UTC/GMT), i.e. a value of 0 indicates January 1, 1970 (midnight UTC/GMT)

The logManagedSensorTable table contains reading samples for managed sensors. The entries of this table have the following OID, where <var> is the variable index described in the table below and <msensor> is the managed sensor number and <entry> is the number of a specific log entry for the sensor:

<ROOTOID>.2.3.1.<var>.<msensor>.<entry>

Table 22: Log of managed sensors table variables

Variable	Index	Type	Access mode	Description
logManagedSensorDataAvailable	1	INTEGER (TruthValue)	Read-only	Indicates data availability for this sensor: 1 if available, 2 otherwise.
logManagedSensorReadingCount	2	INTEGER	Read-only	The count of successfully obtained sensor readings during the period.
logManagedSensorEventStateCount	3	INTEGER	Read-only	The count of successfully obtained sensor even state words during the period.
logManagedSensorAvgValue	4	FLOAT64	Read-only	The average value across sensor readings for the period.

VARIABLE	INDEX	TYPE	ACCESS MODE	DESCRIPTION
logManagedSensorMinValue	5	FLOAT64	Read-only	The minimum value across sensor readings for the period
logManagedSensorMaxValue	6	FLOAT64	Read-only	The maximum value across sensor readings for the period
logManagedSensorDispValue	7	FLOAT64	Read-only	The dispersion across sensor readings for the period.
logManagedSensorAccState	8	INTEGER	Read-only	The accumulated event state (the logical OR of all sensor event states obtained during the period).

For example, to retrieve average values for logged managed sensor readings, use the following OID:

<ROOTOID>.2.3.1.4

The following command retrieves average readings of managed sensor 1 (a temperature sensor).

```
snmpwalk -v1 -c private 192.168.0.1 SMRC-MIB::logManagedSensorAvgValue.1
SMRC-MIB::logManagedSensorAvgValue.1.1 = Opaque: Float: 29.282292
SMRC-MIB::logManagedSensorAvgValue.1.2 = Opaque: Float: 29.314583
…
```

24.4 Measurements MIB variables

The measurements branch represents all sensor reading in Guardian Management Gateway, including managed sensors i.e. virtual replicas of physical sensors attached to resource 0. This branch contains two tables for managed and physical (external) sensors. The measurementsManagedSensorTable table has the following OID, where <var> is the variable index and <msensor> is the managed sensor number:

<ROOTOID>.3.1.1.<var>.<msensor>

Table 23: Managed sensor table variables

VARIABLE	INDEX	TYPE	ACCESS MODE	DESCRIPTION
measurementsManagedSensorIsAvailable	1	INTEGER (TruthValue)	Read-only	Indicates data availability for the sensor during this measurement period: 1 if available, 2 otherwise.
measurementsManagedSensorState	2	INTEGER	Read-only	The current event state mask for the sensor.
measurementsManagedSensorValue	3	FLOAT64	Read-only	The sensor reading. This parameter does not apply to discrete sensors
measurementsManagedSensorTimeStamp	4	STRING	Read-only	The sensor reading timestamp.

For example, to retrieve readings of all managed sensors, use the following OID:

<ROOTOID>.3.1.1.3

The following command retrieves readings of managed sensors (all three are temperature sensors).

```
snmpwalk -v1 -c private 192.168.0.1 SMRC-MIB::measurementsManagedSensorValue
```

```
SMRC-MIB::measurementsManagedSensorValue.1 = Opaque: Float: 29.300000
SMRC-MIB::measurementsManagedSensorValue.2 = Opaque: Float: 29.300000
SMRC-MIB::measurementsManagedSensorValue.3 = Opaque: Float: 15.300000
```

The measurementsExternalSensorTable table is indexed by resource ID and control number. The entries of this table have the following OID, where <var> is the variable index:

<ROOTOID>.3.2.1.<var>.<resource>.<sensor>

Table 24: External sensor table variables

VARIABLE	INDEX	TYPE	ACCESS MODE	DESCRIPTION
measurementsExternalSensorIsAvailable	1	INTEGER (TruthValue)	Read-only	Indicates data availability for the sensor during this measurement period: 1 if available, 2 otherwise.
measurementsExternalSensorState	2	INTEGER	Read-only	The current sensor state.
measurementsExternalSensorValue	3	FLOAT64	Read-only	The sensor reading.
measurementsExternalSensorTimeStamp	4	STRING	Read-only	The sensor reading timestamp.

For example, to retrieve states of all physical sensors, use the following OID:

<ROOTOID>.3.2.1.2

```
snmpwalk -v1 -c private 80.240.102.34 SMRC-MIB::measurementsExternalSensorTable
SMRC-MIB::measurementsExternalSensorIsAvailable.1000.1 = INTEGER: true(1)
SMRC-MIB::measurementsExternalSensorIsAvailable.1000.2 = INTEGER: true(1)
SMRC-MIB::measurementsExternalSensorIsAvailable.1000.3 = INTEGER: true(1)
SMRC-MIB::measurementsExternalSensorIsAvailable.1000.4 = INTEGER: false(2)
SMRC-MIB::measurementsExternalSensorIsAvailable.1000.5 = INTEGER: false(2)
SMRC-MIB::measurementsExternalSensorIsAvailable.1001.1 = INTEGER: true(1)
SMRC-MIB::measurementsExternalSensorIsAvailable.1001.2 = INTEGER: true(1)
…
```

24.5 Controls MIB variables

All the controls in Smart Rack Controller are exposed in separate SNMP branch named “controls” that contains single table controlsTable indexed by resource ID and control number. The entries of this table have the following OID, where <var> is the variable index:

<ROOTOID>.4.1.<var>.<resource>.<control>

Table 25: Control table variables

VARIABLE	INDEX	TYPE	ACCESS MODE	DESCRIPTION
ctrlMode	1	INTEGER	Read-only	The current control mode: automatic(1), manual(2) or unavailable(-1).
ctrlState	2	INTEGER	Read-write	The current control state.

For example, to retrieve the current state of all controls, use the following OID:

<ROOTOID>.4.1.1.2

The following commands turn MCB “buzzer” on then off.

```
snmpset -v1 -c private 192.168.0.1 SMRC-MIB::ctrlState.3000.1 i 1
SMRC-MIB::ctrlState.3000.1 = INTEGER: 1
snmpset -v1 -c private 192.168.0.1 SMRC-MIB::ctrlState.3000.1 i 0
SMRC-MIB::ctrlState.3000.1 = INTEGER: 0
```

24.6 serverReachability MIB variables

The server reachability variables are represented by a single table with the following OID, where <var> is the index of a particular variable in the table of reachability attributes and <entry> is the number of the table entry.

<ROOTOID>.5.1.<var>.<entry>

Table 26: Server reachability variables

VARIABLE	INDEX	TYPE	ACCESS MODE	DESCRIPTION
serverId	1	INTEGER	Read-only	Table entry index, equal to <entry>.
serverIpAddress	2	STRING	Read-write	Host Name or IP address of the target system.
serverPingEnabled	3	INTEGER (TruthValue)	Read-write	1 – if periodic poll of the target system via the ping command is enabled, 2 – otherwise.
serverReachable	4	INTEGER (TruthValue)	Read-only	1 – if the target system is responding, 2 – otherwise.
serverUnreachable	5	INTEGER (TruthValue)	Read-only	1 – if the target system is not responding, 2 – otherwise.

Normally, this table contains entries for external network servers needed for Smart Rack Controller operations e.g. DNS, NTP and DHCP servers, so that it’s easy to diagnose network issues at Smart Rack Controller via the SNMP interface. For example, to retrieve target addresses from the server reachability table, use the following OID:

<ROOTOID>.5.1.2

The following command retrieves the entire server reachability table.

```
snmpwalk -v1 -c public 192.168.0.1 SMRC-MIB::serverReachabilityTable
SMRC-MIB::serverId.1 = INTEGER: 1
SMRC-MIB::serverId.2 = INTEGER: 2
SMRC-MIB::serverIpAddress.1 = STRING: 192.168.0.1
…
```

24.7 sel MIB variables

The sel branch provides access to the System Event Log parameters and entries and allows clearing the log.

System Event Log parameters have the following OID, where <var> is the variable index:

<ROOTOID>.7.<var>.0

Table 27: System Event Log variables

VARIABLE	INDEX	TYPE	ACCESS MODE	DESCRIPTION
selEntriesCount	1	INTEGER	Read-only	The current number of entries in the SEL.
selSize	2	INTEGER	Read-only	SEL capacity (the maximum number of entries that SEL can contain).
selUpdateTimestamp	3	STRING	Read-only	The timestamp of the latest SEL update.
selCurrentTime	4	STRING	Read-only	The current SEL time.
selEnabled	5	INTEGER (TruthValue)	Read-only	Indicates if the SEL is enabled (1) or disabled (2).
selOverflowFlag	6	INTEGER (TruthValue)	Read-only	Indicates if the SEL is overflown (1) or not (2).
selOverflowAction	7	INTEGER (TruthValue)	Read-only	The overflow mode action for new entries: drop (1) or overwrite (2).

VARIABLE	INDEX	TYPE	ACCESS MODE	DESCRIPTION
selClear	8	INTEGER (TruthValue)	Read-write	Set to 1 to cleat SEL. Value 2 means SEL clear is not in progress.

For example, to retrieve the current number of the system event log entries, use the following OID:

<ROOTOID>.7.1.0

Also, there is the selTable table in this section that contains the log entries with parameters that have the following OIDs, where <var> is the variable index from the table below and <entry> is the entry number:

<ROOTOID>.7.9.1.<var>.<entry>

Table 28: System Event Log table variables

VARIABLE	INDEX	TYPE	ACCESS MODE	DESCRIPTION
selEntryId	1	INTEGER	Read-only	SEL entry number, equal to <entry>
selTimestamp	2	STRING	Read-only	Time of the entry addition into the log.
selEventType	3	EventType	Read-only	The event type (this data type is described in the Table 8).
selResourceId	4	INTEGER	Read-only	Resource ID of the event source.
selEventTimestamp	5	STRING	Read-only	Timestamp of the event generation.
selSeverity	6	SeverityType	Read-only	The event severity (this data type is described in the Table 9).
selEventSubType	7	INTEGER	Read-only	Specific event type for resource events, software events, upgrade status for FUMI events.
selSensorNum	8	INTEGER	Read-only	Sensor number of the event source.
selSensorType	9	SensorType	Read-only	The sensor type (this data type is described in the Table 5).
selEventCategory	10	EventCategory	Read-only	The event category (this data type is described in the Table 7).
selAssertionEvent	11	INTEGER (TruthValue)	Read-only	Indicates if the event is an assertion event (1) or a deassertion event (2).
selEventState	12	INTEGER	Read-only	The specific state of the sensor that triggered the event.
selTriggerReading	13	FLOAT64	Read-only	Sensor reading value that triggered the event.
selTriggerThreshold	14	FLOAT64	Read-only	Sensor threshold value that was crossed at the event.
selPreviousStates	15	INTEGER	Read-only	The mask of previous sensor states (before the event).
selCurrentStates	16	INTEGER	Read-only	The mask of current sensor states (after the event).
selFumiNum	17	INTEGER	Read-only	The FUMI number (for FUMI events, normally 0).
selBankNum	18	INTEGER	Read-only	The FUMI bank number (for FUMI events, normally 0).

For example, to retrieve event severity for all SEL entries, use the following OID:

<ROOTOID>.7.9.1.6

24.8 smrcTrap MIB variables

The SNMP Trap messages are in this section. They are defined in SMRC-MIB as the smrcTrap with the following OID:

<ROOTOID>.8

Currently there is only one supported trap that contains one variable in ASCII text format (JSON format to be more specific) describing an event in system event log. This variable has the following OID:

<ROOTOID>.smrcTrap(8).smrcTextTrap(1)

Depending on network service configuration, SNMP traps can be delivered in either SNMPv1 or SNMPv2 format. Below, you can see an example of such traps collected using the snmptrapd utility from the Net-SNMP package. The first one is in SNMPv1 format and the second is in SNMPv2 format, describing the same event.

```
# snmptrapd -d -f -m SMRC-MIB

Starting snmptrapd 5.0.6


Received 365 bytes from 192.168.0.1

0000: 30 82 01 69  02 01 00 04  06 70 75 62  6C 69 63 A4    0..i.....public.
0016: 82 01 5A 06  0E 2B 06 01  04 01 81 80  0A 02 03 01    ..Z..+.........
0032: 08 00 01 40  04 50 F0 66  22 02 01 06  02 01 63 43    ...@.P.f".....cC
0048: 01 37 30 82  01 37 30 82  01 33 06 0D  2B 06 01 04    .70..70..3..+...
0064: 01 81 80 0A  02 03 01 08  01 04 82 01  20 7B 22 45    ............ {"E
0080: 76 65 6E 74  22 3A 7B 22  53 65 6E 73  6F 72 45 76    vent":{"SensorEv
0096: 65 6E 74 22  3A 7B 22 41  73 73 65 72  74 69 6F 6E    ent":{"Assertion
0112: 22 3A 74 72  75 65 2C 22  45 76 65 6E  74 43 61 74    ":true,"EventCat
0128: 65 67 6F 72  79 22 3A 22  54 68 72 65  73 68 6F 6C    egory":"Threshol
0144: 64 22 2C 22  45 76 65 6E  74 53 74 61  74 65 22 3A    d","EventState":
0160: 22 55 70 70  65 72 4D 69  6E 6F 72 54  68 72 65 73    "UpperMinorThres
0176: 68 6F 6C 64  43 72 6F 73  73 65 64 22  2C 22 53 65    holdCrossed","Se
0192: 6E 73 6F 72  4E 75 6D 62  65 72 22 3A  31 2C 22 53    nsorNumber":1,"S
0208: 65 6E 73 6F  72 54 79 70  65 22 3A 22  54 65 6D 70    ensorType":"Temp
0224: 65 72 61 74  75 72 65 22  2C 22 54 72  69 67 67 65    erature","Trigge
0240: 72 52 65 61  64 69 6E 67  22 3A 32 37  2E 38 31 32    rReading":27.812
0256: 2C 22 54 72  69 67 67 65  72 54 68 72  65 73 68 6F    ,"TriggerThresho
0272: 6C 64 22 3A  30 2E 30 7D  2C 22 53 65  76 65 72 69    ld":0.0},"Severi
0288: 74 79 22 3A  22 4D 69 6E  6F 72 22 2C  22 53 6F 75    ty":"Minor","Sou
0304: 72 63 65 22  3A 31 30 30  30 2C 22 54  69 6D 65 73    rce":1000,"Times
0320: 74 61 6D 70  22 3A 22 32  30 31 38 2D  31 31 2D 31    tamp":"2018-11-1
0336: 33 20 31 38  3A 31 32 3A  30 32 22 2C  22 54 79 70    3 18:12:02","Typ
0352: 65 22 3A 22  53 65 6E 73  6F 72 22 7D  7D             e":"Sensor"}}


192.168.0.1: Enterprise Specific Trap (99) Uptime: 0:00:00.55, SMRC-MIB::smrcTextTrap = STRING:
"{\"Event\":{\"SensorEvent\":{\"Assertion\":true,\"EventCategory\":\"Threshold\",\"EventState\":\"UpperMinorThres
holdCrossed\",\"SensorNumber\":1,\"SensorType\":\"Temperature\",\"TriggerReading\":27.812,\"TriggerThreshold\"
:0.0},\"Severity\":\"Minor\",\"Source\":1000,\"Timestamp\":\"2018-11-13 18:12:02\",\"Type\":\"Sensor\"}}"


Received 394 bytes from 192.168.0.1
```

```
0000: 30 82 01 86  02 01 01 04  06 70 75 62  6C 69 63 A7    0........public.
0016: 82 01 77 02  04 30 C0 B8  C0 02 01 00  02 01 00 30    ..w..0.........0
0032: 82 01 67 30  10 06 08 2B  06 01 02 01  01 03 00 43    ..g0...+.......C
0048: 04 03 3F E3  20 30 1C 06  0A 2B 06 01  06 03 01 01    ..?. 0...+......
0064: 04 01 00 06  0E 2B 06 01  04 01 81 80  0A 02 03 01    .....+..........
0080: 08 00 01 30  82 01 33 06  0D 2B 06 01  04 01 81 80    ...0..3..+......
0096: 0A 02 03 01  08 01 04 82  01 20 7B 22  45 76 65 6E    ......... {"Even
0112: 74 22 3A 7B  22 53 65 6E  73 6F 72 45  76 65 6E 74    t":{"SensorEvent
0128: 22 3A 7B 22  41 73 73 65  72 74 69 6F  6E 22 3A 74    ":{"Assertion":t
0144: 72 75 65 2C  22 45 76 65  6E 74 43 61  74 65 67 6F    rue,"EventCatego
0160: 72 79 22 3A  22 54 68 72  65 73 68 6F  6C 64 22 2C    ry":"Threshold",
0176: 22 45 76 65  6E 74 53 74  61 74 65 22  3A 22 55 70    "EventState":"Up
0192: 70 65 72 4D  69 6E 6F 72  54 68 72 65  73 68 6F 6C    perMinorThreshol
0208: 64 43 72 6F  73 73 65 64  22 2C 22 53  65 6E 73 6F    dCrossed","Senso
0224: 72 4E 75 6D  62 65 72 22  3A 31 2C 22  53 65 6E 73    rNumber":1,"Sens
0240: 6F 72 54 79  70 65 22 3A  22 54 65 6D  70 65 72 61    orType":"Tempera
0256: 74 75 72 65  22 2C 22 54  72 69 67 67  65 72 52 65    ture","TriggerRe
0272: 61 64 69 6E  67 22 3A 32  37 2E 38 31  32 2C 22 54    ading":27.812,"T
0288: 72 69 67 67  65 72 54 68  72 65 73 68  6F 6C 64 22    riggerThreshold"
0304: 3A 30 2E 30  7D 2C 22 53  65 76 65 72  69 74 79 22    :0.0},"Severity"
0320: 3A 22 4D 69  6E 6F 72 22  2C 22 53 6F  75 72 63 65    :"Minor","Source
0336: 22 3A 31 30  30 30 2C 22  54 69 6D 65  73 74 61 6D    ":1000,"Timestam
0352: 70 22 3A 22  32 30 31 38  2D 31 31 2D  31 33 20 31    p":"2018-11-13 1
0368: 38 3A 31 32  3A 35 39 22  2C 22 54 79  70 65 22 3A    8:12:59","Type":
0384: 22 53 65 6E  73 6F 72 22  7D 7D                 "Sensor"}}
```

```
build.auriga.ru [192.168.0.1]: Trap SNMPv2-SMI::mib-2.1.3.0 = Timeticks: (54518560) 6 days, 7:26:25.60, SNMPv2-SMI::snmpModules.1.1.4.1.0 = OID: SMRC-MIB::smrcNotification1, SMRC-MIB::smrcTextTrap = STRING: "{\"Event\":{\"SensorEvent\":{\"Assertion\":true,\"EventCategory\":\"Threshold\",\"EventState\":\"UpperMinorThresholdCrossed\",\"SensorNumber\":1,\"SensorType\":\"Temperature\",\"TriggerReading\":27.812,\"TriggerThreshold\":0.0},\"Severity\":\"Minor\",\"Source\":1000,\"Timestamp\":\"2018-11-13 18:12:59\",\"Type\":\"Sensor\"}}"
```

This trap contains asserted event message from temperature sensor 1 at resource 1000 saying that Upper Minor Threshold value 0 was crossed by reading 27.812 and the severity of this event is minor.

To make Smart Rack Controller send an SNMP trap to the test host (IP address 192.168.0.1 in the example below) it is possible to use the chain of following CLI commands, assuming there is a temperature sensor 1 at resource 1000.

```
smrcli> filter add TestFilter "resource==1000 && sensor_number==1 && assertion==1"
smrcli> action add TestFilter always snmptrap 192.168.0.1
smrcli> sensor threshold set 1000 1 umn 0
smrcli> sensor threshold set 1000 1 umn 50
```

24.9 Accessing Guardian Management Gateway via SNMP

Any SNMP client implementation should be able to access the Guardian Management Gateway variables defined in SMRC-MIB. One specific choice is the Net-SNMP package from: http://net-snmp.sourceforge.net/ that is a part of all popular Linux distributions. This package should be installed on the management (client) system. It provides some basic management tools. To access the SNMP server on a Guardian Management Gateway, the snmpget, snmpset and snmpwalk commands can be used.

To install the MIB file on the management system, follow the instructions supplied with the package e.g. for Net-SNMP the SMRC-MIB.txt file should be placed into the /usr/share/snmp/mibs directory or specified via command line arguments.

After that, use the snmpget or snmpwalk commands to verify access. For SNMPv1 or SNMPv2c access, the community name is either public for read-only access or private for read-write access by default. For SNMPv3 access it is necessary to add SNMPv3 user first (see srmcli user snmp commands). For example you can use the following command to retrieve basic Smart Rack Controller configuration:

```
snmpwalk –v2c -c public <SMRC IP address> SMRC-MIB::firmwareVersion
```

or, if MIB file is not yet installed

```
snmpwalk –v2c -c public <SMRC IP address> .1.3.6.1.4.1.16394.2.3.1.1.1.4
```

The output will be similar to the following:

```
SMRC-MIB::firmwareVersion.0 = STRING: "0.95 63998-20551 AWS.Nov  8 2018.10:10:49"
```

To retrieve the entire SMRC-MIB variables subtree, use the following command:

```
snmpwalk –v2c -c public –t 15 <SMRC IP address> SMRC-MIB::smrc
```

This command takes around 5 minutes.

SNMPv3 access command has username, password and optionally privacy string, instead of community string in SNMPv1 or SNMPv2c, so the same command looks like this:

```
snmpwalk -v3 -l authPriv -a SHA -u myusername -A mypassword -x DES -X myprivacy -t 15 <SMRC IP address> SMRC-MIB::smrc
```

Here is an example of creating a managed sensor from physical sensor 1 at resource 1000:

```
snmpget -v1 -c private <SMRC IP address> SMRC-MIB::externalSensorIsManaged.1000.1
SMRC-MIB::externalSensorIsManaged.1000.1 = INTEGER: false(2)
```

There is no managed sensor yet. Create it now by setting this integer variable to 1(TRUE):

```
snmpset -v1 -c private <SMRC IP address> SMRC-MIB::externalSensorIsManaged.1000.1 i 1
SMRC-MIB::externalSensorIsManaged.1000.1 = INTEGER: true(1)
```

Double-check the result:

```
snmpwalk -v1 -c private <SMRC IP address> SMRC-MIB::externalSensorIsManaged.1000
SMRC-MIB::externalSensorIsManaged.1000.1 = INTEGER: true(1)
SMRC-MIB::externalSensorIsManaged.1000.2 = INTEGER: false(2)
SMRC-MIB::externalSensorIsManaged.1000.3 = INTEGER: false(2)
SMRC-MIB::externalSensorIsManaged.1000.4 = INTEGER: false(2)
SMRC-MIB::externalSensorIsManaged.1000.5 = INTEGER: false(2)
```

25 Technical Data

Technical Data	
Height/Width/Depth	1 U / 250 mm / 1 U
Weight	270 g
Ambient Temperature	5 - 60 °C
Humidity	5 - 90% RH, non condensing
Case Material	Aluminum, powder coated
Power Supply	12 VDC, 20W
Emissions	EN 61000-6-3 including EN 55032 level B, FCC Part 15 pending
Immunity	EN 61000-6-2 (industrial environment)
Safety	EN 62368-1, UL 62368-1 pending

26 References

1. Service Availability™ Forum Hardware Platform Interface Specification, Specification SAI-HPI-B.03.02, August 4, 2009.
2. IPMI – Platform Management FRU Information Storage Definition v1.0, Document Revision 1.1, September 27, 1999.