



A MITEL  
PRODUCT  
GUIDE

# Mitel SIP-DECT 10.0 Event Manager

System Manual  
Version 1.0



### ***NOTICE***

The information contained in this document is believed to be accurate in all respects but is not warranted by Mitel Networks™ Corporation (MITEL®). The information is subject to change without notice and should not be construed in any way as a commitment by Mitel or any of its affiliates or subsidiaries. Mitel and its affiliates and subsidiaries assume no responsibility for any errors or omissions in this document. Revisions of this document or new editions of it may be issued to incorporate such changes.

No part of this document can be reproduced or transmitted in any form or by any means - electronic or mechanical - for any purpose without written permission from Mitel Networks Corporation.

### **TRADEMARKS**

The trademarks, service marks, logos and graphics (collectively "Trademarks") appearing on Mitel's Internet sites or in its publications are registered and unregistered trademarks of Mitel Networks Corporation (MNC) or its subsidiaries (collectively "Mitel") or others. Use of the Trademarks is prohibited without the express consent from Mitel. Please contact our legal department at [legal@mitel.com](mailto:legal@mitel.com) for additional information. For a list of the worldwide Mitel Networks Corporation registered trademarks, please refer to the website: <http://www.mitel.com/trademarks>.

Mitel SIP-DECT 10.0 Event Manager  
System Manual  
Release 10.0 – March 25

®,™ Trademark of Mitel Networks  
Corporation

© Copyright 2025 Mitel Networks  
Corporation All rights reserved

## Table of Contents

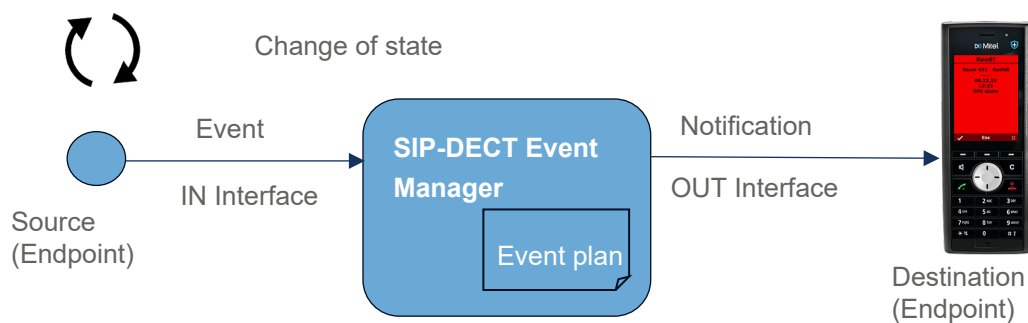
Overview .....	5
Introduction .....	5
Where is the SIP-DECT Event Manager running? .....	7
<i>4th generation RFP</i> .....	7
<i>Linux Server</i> .....	8
Accessing the SIP-DECT Event Manager .....	8
License Requirements for the SIP-DECT Event Manager.....	9
License Requirements for the Locating functionality .....	9
Supported DECT Phones .....	10
Eclipse Mosquitto™ open source MQTT broker on RFP4G.....	11
Using the SIP-DECT Event Manager .....	13
SIP-DECT Event Manager GUI .....	13
<i>Admin view</i> .....	13
<i>Monitor view</i> .....	14
Interfaces .....	14
<i>SIP-DECT (OMM) Interface</i> .....	15
<i>ESPA Interface</i> .....	18
<i>Modbus interface</i> .....	24
<i>SNMP interface</i> .....	27
<i>MQTT interface</i> .....	34
<i>Web-API interface</i> .....	38
Event types .....	41
Notification profiles.....	41
Notification groups .....	42
Event plans .....	42
Locations.....	45
User.....	45
System .....	45
Overview .....	47
Monitor .....	47
Event Log (Summary and Details).....	47
DECT Locating .....	49
Introduction .....	49
Steps for configuration of the locating application .....	50
Backup and restoring the Event Manager data including the installed graphic files .....	53
Quick Start Configuration Guide SIP-DECT Event Manager .....	55

Configuring SOS alarm trigger from a DECT phone.....	55
Configuring an ESPA interface .....	58
Configuring an SNMP interface .....	60
Appendix.....	67
Sitemap .....	67
Web UI Parameter, Action & Status Information overview .....	70
<i>Event Manager without Locating</i> .....	70
<i>Event Manager with Locating</i> .....	84
Recommendation on the procedure for importing log data into Microsoft Excel .....	86

## Overview

### Introduction

The SIP-DECT Event Manager is an integrated software component of a Mitel SIP-DECT system. It is used for the automated processing of incoming events and the sending of outgoing notifications. The SIP-DECT Event Manager can process events from various sources, including SIP-DECT terminals, the SIP-DECT system itself, and other external systems. The processing of the events is carried out according to user-defined rules set by the administrator.



The primary flow is to send notifications as text messages to SIP-DECT phones, which are triggered by incoming events. In this way, SIP-DECT supports customer workflows beyond voice calls, e.g., text messages can be sent to DECT phones to inform about events from nurse call systems without the need for additional hardware.

Processing rules for different types of events consist of event plans, their event phases, notification profiles and different types of confirmation requests.

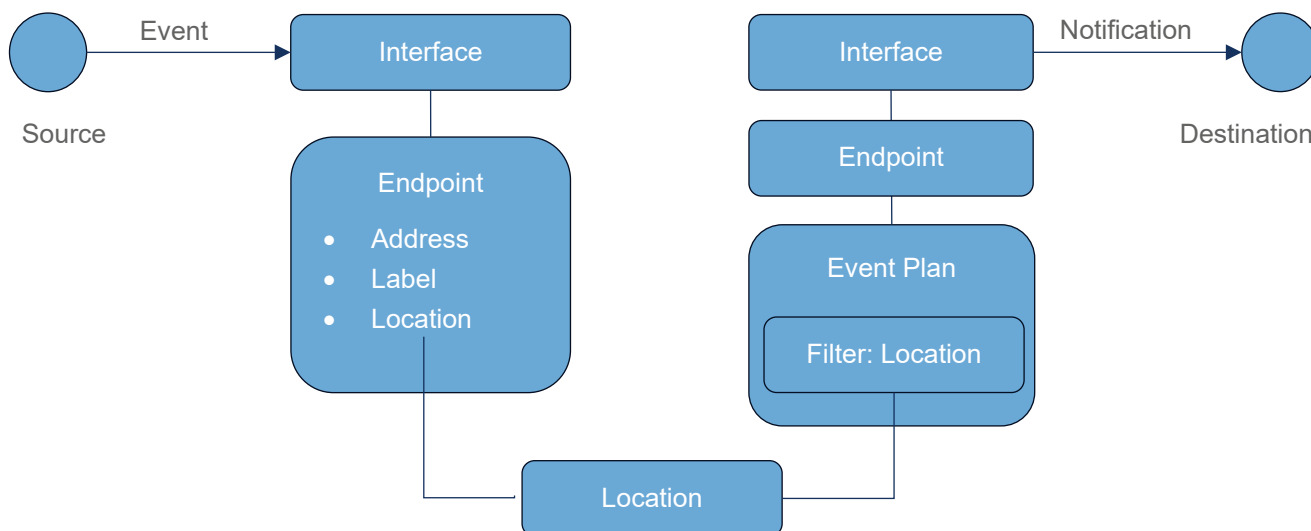
If there is a change in status, e.g., a key press, a source sends an event to the SIP-DECT Event Manager via an input interface. The SIP-DECT Event Manager generates notifications, e.g., text messages, and sends them to destinations, e.g., DECT telephones via outgoing interfaces according to a suitable event plan.

Some interface types are only incoming or only outgoing interfaces, and some can be both incoming and outgoing.

Sources and destinations are called endpoints. They are assigned to the interfaces through which they communicate with the SIP-DECT Event Manager. Endpoints have a unique identification e.g. a telephone number.

Endpoints are also assigned to locations. Depending on the location, a specific event plan can be selected. This allows the same event to be treated differently depending on where it originated.

The following illustration is intended to visualize the relationships between endpoint location and the event plan location filter.




The Event Managers DECT Locating supplements the Event Manager functionality described above with a textual and graphical display of the position of a DECT device based on the DECT radio coverage by a base station (typically approx. 30 to 50 meters in buildings depending on the structural conditions and approx. 300 meters in free field) in the event of an emergency call, triggered by pressing the SOS button on the Mitel DECT telephone (722dt, 732d, 742d, 632d(t) V2) or by a sensor alarm of the DECT device (732d, 742d, 632d(t) V2) as well as feature access codes for customer-specific configurable alarm triggers. In addition, the position of a locatable DECT device can also be queried independently of an event.

The graphical display is provided in a detailed and an overview view.

For the graphical display of the position of a DECT device in case of an event, a suitable event plan must be configured and consequently the triggering DECT phone must be set up as an endpoint.

A locating button  is offered in the monitor in the locating section, which opens the graphical display.

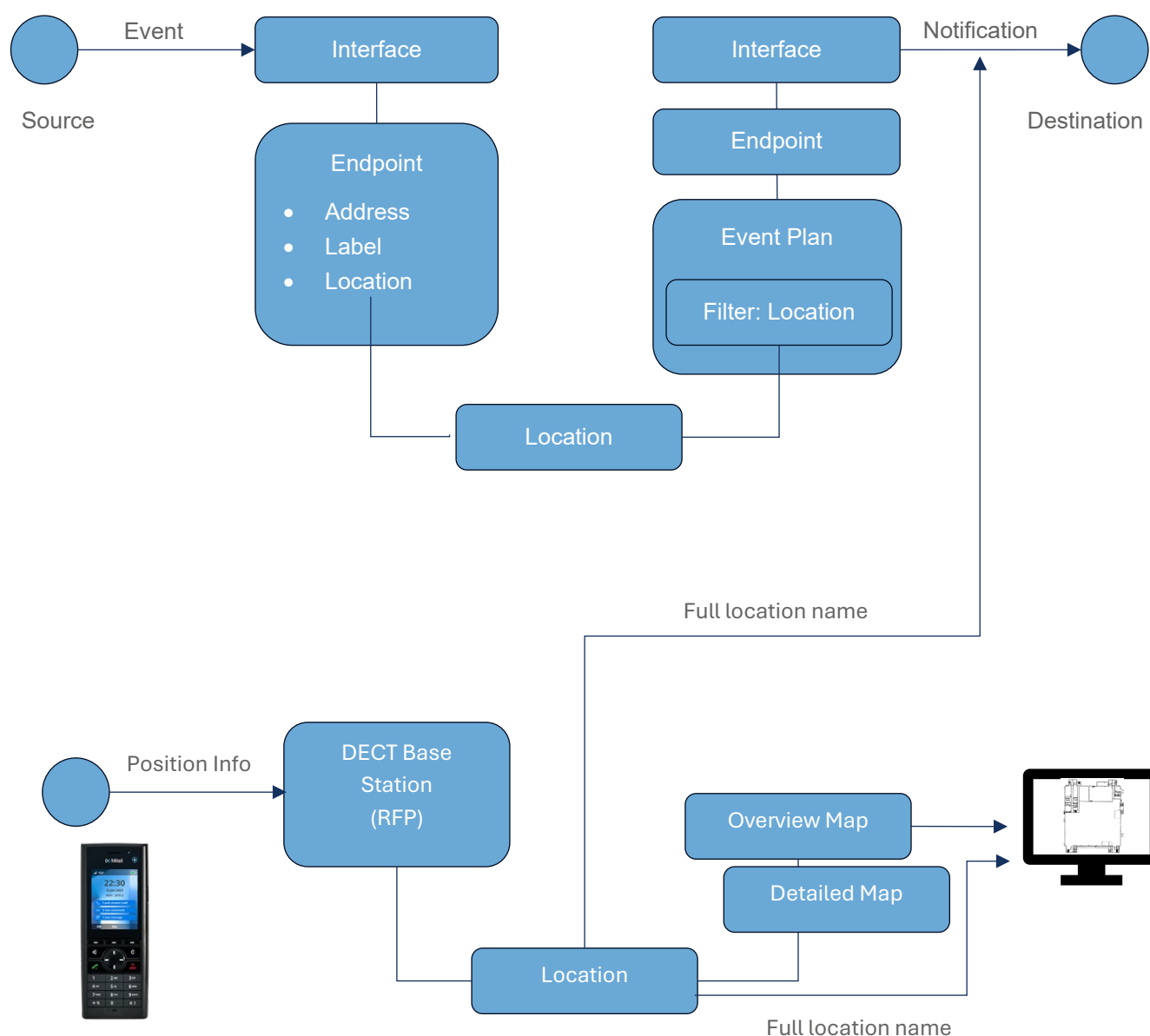
Attention: The appropriate event plan is selected based on the configured location and not on the position determined via DECT.

No event plan is required for an event-independent determination and display of the position of the DECT end device of a locatable and possibly trackable user. The telephone user list in the Locating section, which contains all locatable users, can be used for this purpose. A locating button  is also provided here.

To determine the location of a DECT telephone, the DECT base stations must be assigned to locations. A map must also be assigned to the location and the location must be positioned on a detailed map and on an overview map for graphical representation.

If SIP-DECT locating is used, the full location name from the Event Manager is used in the notification instead of the base station data Site, Building, Corridor etc. configured in the OMM. This ensures that the textual location in notifications matches those in the Event Manager web GUI.

The following figure illustrates the relationships between base stations, maps and locations, as well as the event handling in the Event Manager.



A Linux Server installation of the Event Manager is required to support DECT locating.

Attention: It is recommended not to plan and set up locations too granularly, as DECT works with larger and overlapping radio fields.

## Where is the SIP-DECT Event Manager running?

### 4th generation RFP

The SIP-DECT Event Manager can run on a 4th generation RFP (RFP44, RFP45, RFP47 or RFP48 WLAN) and is part of the iprfp4G.dnld SW package.

The SIP-DECT administrator determines in the OMC (OM Configurator) on which RFP the SIP-DECT Event

Manager is started. This allows a different RFP to be used than the RFP used by the OMM, so that the OMM and SIP-DECT Event Manager do not compete for the same resources.

This also implies that the SIP-DECT Event Manager RFP (the RFP on which the SIP-DECT Event Manager runs) has a local static IP configuration. This ensures that the SIP-DECT Event Manager can be started independently of other services and is always accessible under the same IP address, as is usual for services. Only one SIP-DECT Event Manager per SIP-DECT installation is supported.

To start the SIP-DECT Event Manager the “Start Event Manager” flag must be set as shown below.

The screenshot shows the Mitel OM Configurator window. At the top, there's a table with columns: MAC address, local config, IP address, Net mask, Router, OMM address, 2nd OMM address, TFTP server, and TFTP file name. Below this, the 'Detail Data 08:00:0f:c3:df:0a' section is open, showing tabs for General, IPv6, OpenMobility, and Other. The General tab is active, displaying fields for OMM address, 2nd OMM address, TFTP server address, TFTP file name, Syslog server address, and Syslog server port. To the right of these fields, there are fields for DNS addresses and RFP configuration file server. A checkbox labeled 'Start Event Manager' is checked and highlighted with a red box. At the bottom, there are 'OK' and 'Cancel' buttons. On the right side of the window, there is a 'Tasks' panel with a list of actions: Scan, Add RFP, Clear List, Edit configuration, Copy Configuration, Paste Configuration, Send Configuration, Factory Reset, Remove selected RFP, Save RFP Config, and Load RFP Config.

If this “Start Event Manager” flag is removed again from an RFP via the OMC, the Event Manager will be stopped, and its database will be removed only during the next start of this RFP.

Please note: The Event Manager on an RFP can only handle configurations within the configuration limits of an RFP OMM, i.e. max. 256 RFPs and max. 1024 DECT users. If the OMM runs on a Linux server, the Event Manager must also run on a Linux server.

## Linux Server

The Event Manager can also be installed as an application on a Rocky Linux® 9. A rpm file is available for this purpose. The rpm file is also part of the SIP-DECT VM images. After the initial start of a VM, the OMM, MOM or the Event Manager can be installed. Information on this can be found in the SIP-DECT LINUX Server Installation administration guide.

The Linux Server installation of the Event Manager supports DECT locating with a textual and graphical representation of the position of a DECT device. Please see section DECT Locating. Otherwise, the EM on a Linux server does not differ from an EM on an RFP with regards to the scope of features.

Since the Mitel CloudLink daemon is not available for server installations of the Event Manager, the remote management is not available in this case.

## Accessing the SIP-DECT Event Manager

The SIP-DECT Event Manager has its own web administration interface which is available via https on



port 8444 - https://<IP address>:8444.

Use **admin** as the username and password to login for the first time. During login for the first time, the user is asked to change the password.



The screenshot shows the Mitel SIP-DECT 10.0 Event Manager web interface. The header includes the Mitel logo, the title "SIP-DECT 10.0 Event Manager - EM-37-184", the user "admin", and buttons for "Logout" and "EN". The left sidebar contains a menu with "Interfaces", "Event types", "Notification profiles", "Notification groups", "Event plans", "Locations", "Users" (highlighted), "System", "Overview", and "Monitor". The main content area shows a table for user configuration with columns: Name, Permission, Password, and Password confirmation. The "Name" column contains "admin", the "Permission" column contains "Configuration", and the "Password" and "Password confirmation" columns contain masked passwords. There are also buttons for adding (+) and refreshing (↺) the user list.

Name	Permission	Password	Password confirmation
admin	Configuration	*****	*****

### License Requirements for the SIP-DECT Event Manager

The SIP-DECT Event Manager requires a license for the configured and activated endpoints. There is a built-in license available already for 5 endpoints.

For additional endpoint licenses a SIP-DECT license is required which covers the amount of configured SIP-DECT Event Manager endpoints. It is strongly recommended to import this license into the OMM before the configuration of the Event Manager.

If the number of configured SIP-DECT Event Manager endpoints exceed the number of licensed endpoints, a warning is displayed on the administrator web interface and notifications are sent to various randomly selected SIP-DECT endpoints every 15 minutes. These notification messages are not monitored by the Event Manager and could not be deleted from within the application (also in case the license would be updated to cover the configured number of endpoints). The notifications will be visible on the SIP-DECT terminals as long they are not read and deleted on the terminals itself.

The SIP-DECT Event Manager uses advanced SIP-DECT messaging and alerting features without requiring a "Mitel SIP-DECT Messaging & Alerting License Enterprise" license.

The SIP-DECT Event Manager provides location information for SIP-DECT alarm trigger e.g. SOS-Key or Man-Down automatically without requiring locating license "Mitel SIP-DECT Locating User License XXX". For this purpose, the Event Manager uses the site, building, corridor etc. information of the base station configured in the OMM.

### License Requirements for the Locating functionality

To use the locating functionality the following SIP-DECT licenses are required

- Mitel SIP-DECT Locating User License XXX

- Mitel SIP-DECT Locating Server License

The screenshot shows the Mitel SIP-DECT 10.0 EM Web GUI. The top bar includes the Mitel logo, 'SIP-DECT 10.0', and navigation links for 'Advanced', 'DE', 'EN', 'ES', 'FR', and 'Logout'. The left sidebar lists various system components. The main content area displays system status and licenses. A red box highlights the 'Locating' license section, which includes the following details:

Locating		
Number of users allowed to be located	50	Mitel SIP-DECT Locating User License XXX
OM Locating application	✓	Mitel SIP-DECT Locating Server License
License key	7F44B-XN85R-AERKW-X5TCL-M63E9	

© 2006-2025 Mitel Networks Corporation

The Mitel SIP-DECT Locating Server license must be imported into the OMM before the locating functionality is visible on the EM Web GUI.

As soon as the license has been applied and made available to the EM application, then name changes in the top bar, Locating appears in the navigation bar and allows the access to the locating functionality such as a list of locatable users.

The screenshot shows the Mitel SIP-DECT 10.0 EM Web GUI with the 'Locating & EM' section active. The top bar shows 'User: admin' and 'Logout' buttons. The left sidebar lists various system components, with 'Locating' highlighted. The main content area displays the 'Users' tab, which shows a table of locatable users. A red box highlights the 'Locating' menu item in the left sidebar. Another red box highlights the 'Users' tab, which displays the following table:

Name	Phone number	Location	On	Last action	Description 1	Description 2
SIP-Test-1	322*476	root/Europa/Berlin/Zeughofstr 10/Floor 4	✓	2/5/2025, 4:18:18 PM	R&D	TES2

© 2024-2025 Mitel Networks Corporation. Endpoints: 50 licensed / 7 activated

Please note that only locatable users are displayed and that the Mitel SIP-DECT Locating User License is required for them.

As long as users are not to trigger events or receive notifications, they do not need to have been imported from the OMM into the EM and exist as an endpoint in the SIP DECT interface. They still appear in the list of locatable users. This means that no endpoint license is required for these users.

If users have been imported as endpoints but are only to be located without triggering events or receiving notifications, these endpoints can be set to inactive. They are then still listed in the list of locatable users, but are not counted towards the endpoint license.

## Supported DECT Phones

The SIP-DECT Event Manager supports the 700d DECT phone family. The SIP-DECT 600d V2 DECT phone family is also generally supported. Older generations of the 600d device family or their older SW

versions may not support all SIP-DECT messaging features and may therefore have limitations. Please also note the information in the Mitel 600/700 DECT Phone Messaging and Alerting Applications user guide.

## Eclipse Mosquitto™ open source MQTT broker on RFP4G

It is possible to start a functionally restricted Eclipse Mosquitto™ open source MQTT broker on a RFP4G. This allows the operation of MQTT in conjunction with the Event Manager and MQTT-capable devices mainly for testing purposes.

The MQTT broker is started via an OM Configurator setting.

The screenshot shows the Mitel OM Configurator window. The 'General' tab is selected. The 'Detail Data 14:00:e9:01:a5:fe' section is expanded, showing the 'OpenMobility' sub-tab. In the 'OpenMobility' section, the 'Start Mosquitto MQTT Broker' checkbox is checked and highlighted with a red rectangle. Other settings include OMM address (192.168.2.41), DNS addresses (192.168.2.1), TFTP server address (0.0.0.0), and TFTP file name (none). The 'Tasks' panel on the right lists various actions like Scan, Add RFP, Clear List, Edit configuration, Copy Configuration, Paste Configuration, Send Configuration, Factory Reset, Remove selected RFP, Save RFP Config, and Load RFP Config. The status bar at the bottom indicates the interface is Realtek USB GbE Family Controller and there is no proxy.

If the following restrictions are acceptable, use in operational environments is possible.

- Max 150 clients in parallel are supported.
- No support for retained messages (clients which set the retain flag in publish messages will be disconnected).
- QoS 0 is recommended. Please avoid MQTT QoS level 1 and 2 as additional restrictions apply.
- The maximum packet size for single MQTT messages is 4096 Byte (clients sending larger packets will be disconnected). A MQTT message size of ~1200 Byte is recommended to avoid fragmentation and additional CPU and memory load.
- No support for persistent sessions.
- No support for WebSocket connections.
- No support for TLS, only port 1883 supported.
- No client authentication, anonymous access possible.

The Broker should not run together with the OMM or the Event Manager on a 4G RFP. If a sufficient number of RFPs is available, the Broker should be activated on a separate RFP.

Additional hints:

The mosquitto broker publishes statistics and usage information under the topic hierarchy '\$SYS/broker/#' every 10s.

The tool MQTT-Explorer (<https://mqtt-explorer.com/>) displays this information by default.

With `mosquitto_sub` the information can also be retrieved and stored in a file:

```
mosquitto_sub -h <broker-ip-address> -p 1883 -t '$SYS/#' -v
```

The broker logging can be accessed under the topic hierarchy '\$SYS/broker/log/#'. The messages are sent here by the broker when the corresponding event occurs. It is not possible to retrieve log messages for events in the past, the broker does not save this information.

Only the log messages of the broker are retrieved with the following command.

```
mosquitto_sub -h <broker-ip-address> -p 1883 -t '$SYS/broker/log/#' -v
```

Note: MQTT-Explorer and `mosquitto_sub` can be run in parallel on the same broker, the MQTT-Explorer is well suited to display the status and statistics information and `mosquitto_sub` can be used to record the log output of the broker.

## Using the SIP-DECT Event Manager

To take the first practical steps with the SIP-DECT Event Manager as quickly as possible, you can start with the section [Quick Start Configuration Guide SIP-DECT Event Manager](#).

### SIP-DECT Event Manager GUI

#### Admin view

The SIP-DECT Event Manager has its own web administration interface which is available via <https://<IP address>:8444>. The web interface consists of a series of web pages that are used to configure the various settings of the SIP-DECT Event Manager and can be accessed from any computer or device with a web browser on the same network or via Remote Management (if configured). The web service is implemented as a single-page application (SPA).



#### 1 Login Area

##### Language Selection

The following languages are available: German, English, French and Spanish. When creating the configuration there are numbers of standard values (e.g. event types) set up in the language selected at this time. These values contained in the configuration are not affected by switching the language.

Use 'admin' as the username and password to login for the first time. During login for the first time, the user is asked to change the password.

#### 2 Configuration panes

The SIP-DECT Event Manager includes multiple panes that contain different information about the SIP-DECT Event Manager.

Configuration Pane	Description
<b>Interfaces</b>	The Interfaces pane provides an overview of the status of systems that are connected to the SIP-DECT Event Manager. Interfaces, their endpoints and interface-specific settings can be managed here.
<b>Event types</b>	The Event types pane allows to create new or change existing Event types. There are <b>5 default</b> Event types ('Man Down', 'No Move', 'Escape', 'SOS-key' and 'System Info') available. These types cannot be deleted.

Configuration Pane	Description
	The Event type serves as a kind of filter in an Event plan to control the escalation of an event. Based on the assigned priority, the system can be informed in which order the event should be processed.
Notification profiles	The display and acoustic signaling of an event on the SIP-DECT terminals can be configured within a notification profile.
Notification groups	Endpoints that can receive notifications (e.g. SIP-DECT terminals) can be combined into a notification group. This simplifies the configuration.
Event plans	The Event plans pane allows to create, edit and delete event plans. An Event plan specifies how received events should be handled depending on the location of the originating endpoint. The plan specifies which endpoints should receive notifications and how to react if acknowledgements are not received. An event plan can include one or more event types and one or more locations. It means that the event plan will only be used for events of the configured type and if the originating endpoint belongs to the specified location.
Locations	The Event Manager supports the management of locations to which endpoints are assigned as sources of events. Locations are assigned to event plans too.  This allows the location-specific definition of event plans, i.e. it is possible to notify different recipients depending on the location of the sender of an event.
User	The Users pane allows to create, edit and delete users. The default user <b>admin</b> cannot be deleted.
System	The System pane includes different tabs for naming the system, showing the current software version, for the configuration of a Watchdog and to activate the CloudLink daemon for the remote management. Here can also be performed functions such as Restart, Restart with factory defaults, Export log, Import config, and Export config. The import of SSL certificates is available as well as the configuration of a security level and the used Cipher Suites. In case of activated CloudLink daemon the detailed configuration and displaying the status of it is available.
Overview	The Overview pane shows a short summary of the Event Manager configuration (included are different tables regarding event flow, notification groups, MQTT mappings and interface to endpoint relations).
Monitor	The Monitor pane shows a list of active event handlings and allows the administrator to end a single event or all events.

### Monitor view

The monitor view is the view for users with the permission 'Monitor'. In this view no configuration is possible. The only purpose of this view is to display running event flows. The user may cancel one running single event plan or all running event plans.

### Interfaces

Interfaces connect the SIP-DECT Event Manager to other devices and services. Depending on the type, these interfaces support receiving events or sending notifications, sometimes both.

Depending on the interface type, a certain number of interface instances can be set up until the maximum

number of **10** interfaces is reached.

The following types of interfaces can be configured:

Type	Maximum number
SIP-DECT (OMM)	1
ESPA	4
Modbus (e.g. WAGO or MOXA)	2
SNMP	2
MQTT	2
Web-API	4

Under the **Interfaces** configuration pane, all configured interfaces are displayed, and can be selected and edited.

Interfaces

Event types

Notification profiles

Notification groups

Event plans

Locations

Users

System

Overview

Monitor

+

↺

Active	State	Label ↑	Description	Type	Endpoints	
✓	●	<a href="#">ESPA-37-79-10004</a>	ESPA-37-79-10004 (IF2)	ESPA	1	<div><div></div><div></div></div>
✓	●	<a href="#">MODBUS-MOXA-33-116</a>	MODBUS-MOXA-33-116	Modbus	9	<div><div></div><div></div></div>
✓	●	<a href="#">MODBUS-WAGO-33-109</a>	MODBUS-WAGO-33-109 (IF5)	Modbus	6	<div><div></div><div></div></div>
✓	●	<a href="#">MQTT-33-120</a>	MQTT-Box 10.103.33.120 (TLS)	MQTT	0	<div><div></div><div></div></div>
✓	●	<a href="#">MQTT-Broker-31-88</a>	running on RFP 10.103.31.88 (Nano-31-89, Shelly-31-87, Tasmota-31-124)	MQTT	5	<div><div></div><div></div></div>
✓	●	<a href="#">PC-OMM-37-80</a>	PC-OMM-37-80	SIP-DECT	4	<div><div></div><div></div></div>
✓	●	<a href="#">SNMP-37-79</a>	SNMP-37-79, Receiver-31-89	SNMP	2	<div><div></div><div></div></div>
✓	●	<a href="#">WAPI-Tester</a>	WAPI-Tester-37-79 (Python)	Web-API	1	<div><div></div><div></div></div>
✓	●	<a href="#">WAPI-WF</a>	WAPI for events from and notifications to Workflow	Web-API	4	<div><div></div><div></div></div>

## SIP-DECT (OMM) Interface

The SIP-DECT (OMM) interface is already created by default and cannot be deleted. It contains the following tabs:

### General Tab

The **General** tab is used to configure the OMM IP address(es), user and password. With this configuration the SIP-DECT Event Manager will be able to connect with the OMM. A successful connection is indicated by the interface status turning to green in the interfaces overview tab. The 'User defined event text' box must be selected to take effect the settings under the tab 'User defined event text'.

<

Interface: OMM-37-182

General

Endpoints

User defined event text

Import endpoints

Save

Refresh

OMM 1

10.103.37.182

OMM 2

User

omm

Password

●●●●●●●●

User defined event text

☒





## Endpoints Tab

The **Endpoints** tab is used to define the destinations or receivers of messages in the SIP-DECT event. To simplify the setting up of the endpoints on the SIP-DECT interface, these endpoints can be imported via the 'Import endpoints' tab.

Please be aware that an endpoint which is not marked as active, cannot be used to trigger an alarm and is not counted as a licensed endpoint. Inactive endpoints are marked with (\*) in other configuration panes as shown below.

< Interface: defaultOMM

General Endpoints User defined event text Import endpoints

Active	Address (Phone number) ↑	Label	Location	
✗	118	User 118		 
✓	120	User 120		 

< Location: root

Endpoints assigned	Endpoints available
defaultOMM / User 118 (*) / 118	
defaultOMM / User 120 / 120	
defaultOMM / User 126 / 126	
defaultOMM / User 141 / 141	
ESPA -IF-1 / ESPA EP 9000 / 9000	

## User defined event text Tab

The **User defined event text** tab is used to customize special types of text to be sent to the DECT phones when an event is handled.

This function allows organizations, agencies, or individuals to create and send messages with specific details or instructions that are relevant for a special situation.

The texts defined in this section only take effect when the checkbox 'User defined event text' under tab 'General' is selected.

The message text is normally made up of the event type and the location of the originating endpoint. The composition of alarm texts can be flexibly configured for each interface with user defined alarm texts.

The text delivered by the interface during the triggering of the event can be changed before the further editing by replacing individual character strings. The character strings to be replaced should be entered in 'Text' and 'Replaced by'.

Up to four texts can be used for the composition of the final alarm text. A maximum length should be defined for each of these texts. Either a space or a line feed can be used as a spacer between these texts. Since line feeds cannot be displayed on all endpoints, they are automatically replaced with spaces where necessary.

The following texts are available:

- Event type
- Event type short – max 8 characters
- Priority – Priority of the alarm defined by the alarm type
- Originating endpoint (name) – Name of the endpoint at which the alarm has been triggered
- Originating endpoint (address) – Address (e.g. phone number) for the endpoint at which the alarm has



been triggered

- Location of originating endpoint – Environment to which the alarm which has been triggered is assigned by the configuration or by DECT locating
- Received text from interface – Permits the use of composed alarm texts based on special interface settings (e.g. ESPA)
- Event phase – The designation of the current escalation phase

### ***Import endpoints Tab***

The **Import endpoints** tab allows the automatic import of the SIP-DECT devices configured in the SIP-DECT system as endpoints to the SIP-DECT Event Manager configuration. This function can only be used if a connection has been established between the SIP-DECT Event Manager and the SIP-DECT system (OMM).

If the number of endpoints permitted by the license is exceeded during the import, a warning will be displayed.

Only those endpoints should be imported that are really needed.

The imported endpoints can be deleted under the Endpoints tab.

## ESPA Interface

The ESPA interface enables the connection of devices that support data exchange in accordance with the ESPA 4.4.4 protocol. This protocol was defined by the European Selective Paging Manufacturer's Association for controlling radio paging equipment and for connecting fire alarm and light signaling systems.

The SIP-DECT Event Manager supports the ESPA 4.4.4 protocol over IP. This permits the exchange of messages with fire alarm systems, light signaling systems, radio paging equipment and similar systems which also support this interface. An ESPA interface can only operate as an input interface (where the SIP-DECT Event Manager receives messages) and not as an output interface (where the SIP-DECT Event Manager sends messages).

If supported by the other side, the SIP-DECT Event Manager facilitates monitoring of the ESPA connection protocol-wise.

Components are connected directly via TCP/IP byte stream or via RS-232 / IP converter. The SIP-DECT Event Manager acts as a TCP client in an ESPA slave mode.

An ESPA message contains information organized in numbered fields. The following fields are important for configuring the SIP-DECT Event Manager

No.	Designation	ESPA Standard Designation	Remarks
1	Call address	Call Address	16 characters max.
2	Display message	Display Message	128 characters max.
3	Ringtone	Beep coding	
4	Ring type	Call type	
6	Priority	Priority	

Please note: ESPA messages in a wrong format will not be processed. Unknown fields will be ignored. 'Call address' (1) and 'Display message' (2) must always be present in an ESPA record.

The fields 'Beep coding' (3), 'Call type' (4), and 'Priority' (6) have no direct influence on the notifications to the SIP-DECT phones. They are only used to select the right event type.

The ESPA interface contains the following tabs:

- General
- Endpoints
- User defined event text
- Event assignment
- Simulator/Trace

### General Tab

The **General** tab allows configuring the basic settings of the ESPA interface. The following settings can be configured:

- **IP address:** IP address to which the SIP-DECT Event Manager should connect to
- **IP port:** The IP port to which the SIP-DECT Event Manager should connect to
- **Interface supervision:** Select this check box if this interface should be supervised.
- **Determine endpoint by:** Select the method for determining the endpoint. Available options are 'Call address' (which is the default setting) and 'Message text'.

- **Default event type:** Select the default event type. A specific event type must be created for it in the Event type section. This default event type is used as fallback if nothing else is defined in the Event assignment tab or if nothing fits to the made configuration.
- **Call type 1 (Field 4) terminates event:** Select this check box to terminate the event.
- **User defined event text:** Select this checkbox if this feature should be used!

< Interface: ESPA -IF-1

General Endpoints User defined event text Event assignment Simulator/Trace

Save Refresh

IP address 192.168.2.71

IP port 10001

Interface supervision ☒

Determine endpoint by Call address ▼

Default event type ESPA-Event ▼

Call type 1 (Field 4) terminates event ☐

User defined event text ☒

### Endpoints Tab

The **Endpoints** tab allows the definition of senders of ESPA messages. The assignment of an endpoint to an ESPA message is done based on the call address. The call address can be determined either by the ESPA field 1 (Call address) or by the ESPA field 2 (Message text). If 'Determine endpoint by: Message Text' is set, the message text must contain only the call address and nothing else.

### User defined event text Tab

In the **User defined event text** tab, it is possible to define special content for the notification messages to addressed endpoints (e.g. SIP-DECT terminals). If this feature is not enabled in the **General** tab, the ESPA field 2 (Message text) is used for the notification message. There are two tables available under this tab where a simple text replacement and/or a complete text definition depending on some known parameters is possible.

<

Interface: ESPA

General

Endpoints

User defined event text

Event assignment

Simulator/Trace

Text replacement (not for event type, priority and phase)

Text	Replace by	
<div>ESPA EVENT TEXT</div>	<div>ESPA event text</div>	<div><div></div><div></div></div>
		<div><div></div><div></div></div>
		<div><div></div><div></div></div>

Text	Max. length	Spacer	
	<div>20</div>		<div><div></div><div></div></div>
	<div>20</div>		<div><div></div><div></div></div>
	<div>20</div>		<div><div></div><div></div></div>
	<div>20</div>		<div><div></div><div></div></div>

Simple Text replacement

In the table at the top of this tab the received text (field 2) from the ESPA message can be modified.

Text (field 2) of the ESPA message	Replacement rule	Resulting notification text
ESPA EVENT TEXT	ul	ESPA event text

Compose a new event text based on an ESPA message

In the table at the bottom of this tab the event text can be recomposed from up to 4 elements. These 4 elements can be selected from 8 different event information elements. These information elements are shown in the following example.

<

Interface: ESPA

General

Endpoints

User defined event text

Event assignment

Simulator/Trace

Text replacement (not for event type, priority and phase)

Text	Replace by	
		<div><div></div><div></div></div>

Text	Max. length	Spacer	
<div></div>	<div>20</div>	<div></div>	<div><div></div><div></div></div>
<div>Event type</div>	<div>20</div>		<div><div></div><div></div></div>
<div>Event type short (max. 8)</div>	<div>20</div>		<div><div></div><div></div></div>
<div>Priority</div>	<div>20</div>		<div><div></div><div></div></div>
<div>Originating endpoint (name)</div>	<div>20</div>		<div><div></div><div></div></div>
<div>Originating endpoint (address)</div>			
<div>Location of originating endpoint</div>			
<div>Phase</div>			
<div>Received text from interface</div>			

Event assignment Tab

The **Event assignment** tab allows to define the process of designating or assigning specific tasks, roles, or responsibilities to individuals or teams in response to an emergency event. It is a crucial part of coordinating

an effective response to emergencies.

An event type is assigned for incoming ESPA messages based on the Ringtone (field 3), Priority (field 6) or Text (field 2). In addition, a Default event type must be configured for non-assigned types in the **General** tab.

< **Interface: ESPA**

General Endpoints User defined event text Event assignment Simulator/Trace

Save Refresh

IP address 192.168.2.71

IP port 10001

Interface supervision ☒

Determine endpoint by Message text

Default event type ESPA

Call type 1 (Field 4) terminates event Please select

User defined event text System Info  
SOS-Key  
Man Down  
New ESPA Type  
ESPA

Rules can be defined in the **Event assignment** tab of the ESPA interface configuration, as following shown.

< **Interface: ESPA**

General Endpoints User defined event text Event assignment Simulator/Trace

+ ↺

	Ringtone (3)	or Priority (6)	or Text (2)	Event type
1			TEST2	TEST_TEXT_LONG
2			TEST	TEST_TEXT_SHORT
3		1		TEST_PRIO_1
4		2		TEST_PRIO_2
5	1			TEST_BEEP_1
6	*			TEST_BEEP_*

Rules are displayed in the order of their creation and are also processed in this order (top down). The first matching rule will be applied. Hence, the more specific rules need to be configured first.

The fields are linked 'OR', not 'AND'!

A '\*' can be used as a wildcard in the fields 'Ringtone' and 'Priority'. The assignment is then made for all values used in these fields.

Leading or trailing spaces in the Text field will be removed automatically.

The search for an event will be done in the following order:

1. A search is made for matching values without wildcards.
2. If no such rule applies, the system then searches for wildcards in the 'Ringtone' and 'Priority' fields.
3. If it is also then not possible to assign an event type, the default event type is used.

For example, a rule with 'TEST2' as text is more specific than a rule with text 'TEST'. To avoid that the 'TEST' will always be applied before 'TEST2', the rule with text 'TEST2' needs to be configured first as shown below.

The following table shows how these rules are applied to some ESPA message input examples.

ESPA message input			Matching rule			Resulting event type	Comment
Ringtone (3)	Priority (6)	Text (2)	Ringtone	Priority	Text		
Any or not provided	Any or not provided	TEST2			TEST2	TEST_TEXT_LONG	Rule 1
Any or not provided	Any or not provided	TEST3			TEST	TEST_TEXT_SHORT	Rule 2
1	1	Hello!		1		TEST_PRIO_1	Rule 3
1	3	Hello!	1			TEST_BEEP_1	Rule 5
Any, except 1	Any (except 1 and 3) or not provided	Hello!	*			TEST_BEEP_*	Rule 6
Not provided	Not provided	Hello!				ESPA	no match, default event type

### Event Text Replacement

Normally the 'Message text' (field 2) of an ESPA message is used as the notification text. Leading and trailing spaces in this text field are not supported and will be removed automatically during the configuration.

If there is an event text defined, then the event text will replace the content of the received 'Message text' (field 2) of the ESPA message.

If 'text position > 0' is set, then the 'Message text' (field 2) of the ESPA message is also included in the notification text starting at the specified text position.

If there is additionally a text length set, then only the specified portion of the 'Message text' (field 2) of the ESPA message is also included in the notification text.

< Interface: ESPA

General Endpoints User defined event text Event assignment Simulator/Trace

+ ↺

	Ringtone (3)	or Priority (6)	or Text (2)	Event type	Text position	Text length	Event text	Separator	
0	5	1	ESPA EVENT TEXT	New ESPA Type	0	0	Replacement	#	

Settings – Text position, Text length and Event text					Resulting notification text
Text (2)	Event type	Text position	Text length	Event text	Replacement
ESPA EVENT TEXT	New ESPA Type	0	0	Replacement	

Settings – Text position, Text length and Event text					Resulting notification text
Text (2)	Event type	Text position	Text length	Event text	ESPA EVENT TEXT
ESPA EVENT TEXT	New ESPA Type	0	0		
Text (2)	Event type	Text position	Text length	Event text	Addition - ESPA EVENT TEXT
ESPA EVENT TEXT	New ESPA Type	1	0	Addition -	
Text (2)	Event type	Text position	Text length	Event text	Addition - EVENT TEXT
ESPA EVENT TEXT	New ESPA Type	6	0	Addition -	
Text (2)	Event type	Text position	Text length	Event text	Addition - EVENT
ESPA EVENT TEXT	New ESPA Type	6	5	Addition -	
Text (2)	Event type	Text position	Text length	Event text	EVENT
ESPA EVENT TEXT	New ESPA Type	6	5		

### Simulator/Trace Tab

The **Simulator** function can be used to check if a received ESPA message would be escalated correctly. The ESPA interface itself does not need to be running (state: green) for the Simulator function to work. There must only have been created an ESPA endpoint with a location, and in the **General** tab, a Default event type must be selected, and any IP address and port must be configured.

The communication between the SIP-DECT Event Manager and the ESPA interface can be recorded at the protocol level as needed. The **Trace** function can be used to monitor the data sent and received by the ESPA interface. The trace functionality can be started and stopped by the same button.

Interfaces

Event types  
Notification profiles  
Notification groups  
Event plans  
Locations  
User  
System  
Monitor

< Interface: ESPA-IF-1

General
Endpoints
User defined event text
Event assignment
Simulator/Trace

Simulator

Send

Call address (1) 9000  
Display message (2) Room 123  
Ringtone (3) Optional  
Call type (4) Optional  
Priority (6) Optional

Trace

Stop
Clear

Data received ☒  
Data sent ☒  
Vital sign ☒  
View Hex ☐

19-02-2024 08:51:40:709 R 1 ENQ 2 ENQ  
19-02-2024 08:51:40:709 T ACK  
19-02-2024 08:51:40:709 R SOH 1 STX 1 US 9000 RS 2 US Room 123 ETX 0B  
19-02-2024 08:51:40:710 T ACK

## Modbus interface

The Modbus interface enables the connection of devices e.g. WAGO or MOXA which provides input ports (e.g. buttons or switches) and output ports (e.g. lights) via Modbus-TCP protocol. The Modbus protocol is a client / server data protocol in the application layer of the OSI model which was originally published by Modicon (now Schneider Electric) in 1979 for use with programmable logic controllers via RS232/RS485 interfaces (Modbus-RTU). For data transmission over Ethernet the protocol was adapted to Modbus-TCP. Meanwhile Modbus has become a de facto standard communication protocol for communication between industrial electronic devices in a wide range of buses and networks.

Reading digital input ports and setting digital output ports of Modbus-TCP devices is supported by the Event Manager.

The following devices have been approved for correct interoperability with the Event Manager:

- WAGO I/O System 750 ("Fieldbus Coupler Modbus TCP 4th generation" Item no. 750-362)
- MOXA ioLogik E1200 Series (ioLogik E1212)

Analog inputs and outputs and other sensor ports are not supported by the Event Manager.

**Note: Functionality cannot be guaranteed with other devices and must be checked separately before use. The following conditions must be observed.**

- Only digital inputs/outputs supported (no analog inputs/outputs or other sensors)
- IO addresses must not be remapped by device configuration, Event Manager only supports address range starting with address 1 for input/output ports.

### General Tab

The **General** tab is used for configuration of the IP address and port of the Modbus-TCP device which is connected through the interface.

< **Interface: MODBUS-WAGO-33-109**

General

Endpoints

✓ Save

↻ Refresh

IP address

10.103.33.109

IP port

502



## Endpoints Tab

The Endpoints tab is used for configuration of incoming and outgoing endpoints. Incoming endpoints correspond to digital inputs of Modbus-TCP devices and outgoing endpoints correspond to digital outputs of Modbus-TCP devices. For WAGO devices the incoming ports 1-256 are valid addresses, for MOXA only the addresses 1-16.

< Interface: MODBUS-WAGO-33-109

General Endpoints

Active	Direction	Address ↑	Label	Location	
✓	Incoming	1	WAGO-33-109-IN-I1-Switch	root/Lab-TES1	
✓	Incoming	2	WAGO-33-109-IN-I2-Button	root/Lab-TES1	
✓	Outgoing	2	WAGO-33-109-OUT-O2-White-Light	root/Lab-TES1	
✓	Outgoing	3	WAGO-33-109-OUT-O3-Red-Light	root/Lab-TES1	
✓	Outgoing	4	WAGO-33-109-OUT-O4-Green-Light	root/Lab-TES1	

In the Endpoints configuration (reached by the link in the overview) some special settings for the endpoint can be configured. Mandatory are 'direction' and 'event type', optionally some special settings can be configured: 'Idle current' or 'Working current' is used on the connected device, a 'Alarm delay in seconds' and the 'Behavior when returning to normal state' (not terminate, terminate immediately or terminate at the end of the current alarm phase). For outgoing endpoints can be configured no special settings.

< Interface: MODBUS-MOXA-33-116 / Endpoint: 1

✓ Save Refresh

Direction: Incoming

Event type: Fire alarm

Idle current: ☐

Release delay (sec): 0

Behavior when returning to normal state: Do not terminate event

Do not terminate event

Terminate event immediately

Terminate event at the

< Interface: MODBUS-MOXA-33-116 / Endpoint: 2

✓ Save Refresh

Direction: Incoming

Event type: WC-Call

Idle current: ☐

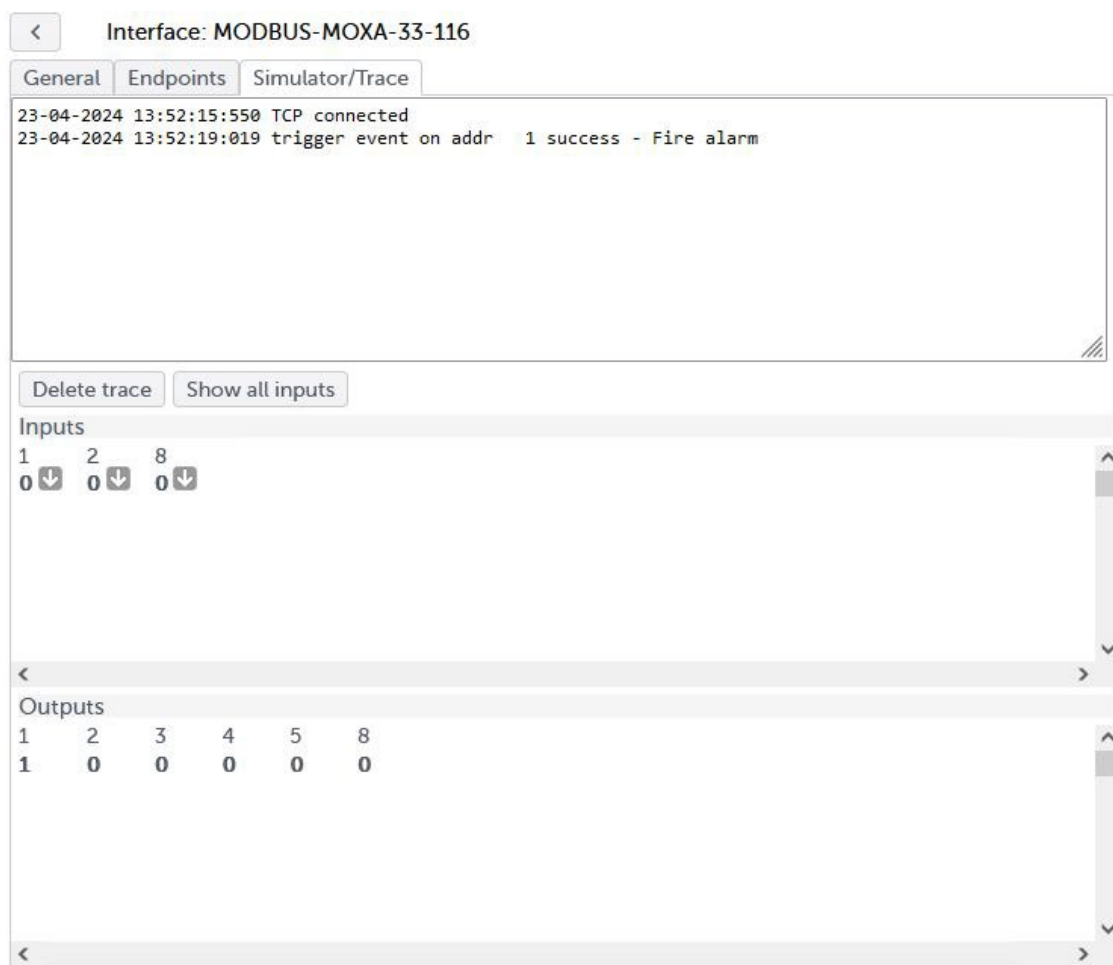
Release delay (sec): 0

Behavior when returning to normal state: Terminate event at the end of phase

## Simulator/Trace Tab

The **Simulator/Trace** tab is used for simulation of the Modbus interface endpoints and for tracing changes on input/output ports. Each time the tab will be opened the trace window will show TCP/IP related connection information and the simulation window will show the actual status of the configured ports. By pressing the "Show all inputs" button the state of all inputs between address 1 and highest configured incoming endpoint address is shown. It is not recommended to open more than one browser window with active Simulator/Trace tab. Only one session is handled by the system.

For configured incoming endpoints a small button is drawn beside each input state. If such button is pressed, the event configured for this endpoint will be generated and processed with defined event plan, Please note that the configured endpoint attributes "Alarm delay", "Idle Current" and "Behavior when returning to normal state" don't apply if this button is pressed, the configured event will be generated immediately. If needed the executed event plan can be canceled via Monitor section of the Event Manager web frontend.



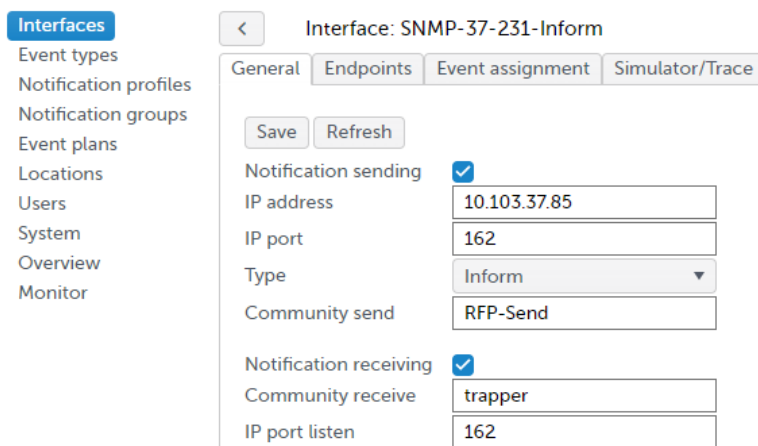
In the Outputs part of the **Simulator/Trace** tab the activity on output port 1 (in this example there is a light connected) is visible and in the trace part of the tab the handled trigger event at the incoming port 1 (triggered by the switch connected physically to this port or by pressing the button 1 in the Inputs part) is documented.

For simulation of the Modbus interface without a connection to a physical device it is possible to configure the interface with local host IP address (127.0.0.1).

## SNMP interface

### General Information

The SNMP interface enables the Event Manager to send and receive SNMP notifications to and from configured IP addresses with correct community strings. Sent notifications as well as received ones may only be Traps or Inform-Requests. Only SNMP v2c is supported for sending notifications while SNMP v1 and SNMP v2c is supported for receiving notifications.



**Interfaces**

- Event types
- Notification profiles
- Notification groups
- Event plans
- Locations
- Users
- System
- Overview
- Monitor

**Interface: SNMP-37-231-Inform**

General | Endpoints | Event assignment | Simulator/Trace

Save Refresh

Notification sending ☒

IP address 10.103.37.85

IP port 162

Type Inform

Community send RFP-Send

Notification receiving ☒

Community receive trapper

IP port listen 162

### Notification sending

In order to send notifications, “IP address”, “IP port”, “Type” and “Community send” need to be correctly configured. Should the selected SNMP interface only send notifications, you may uncheck “Notification receiving”.

“IP address” and “IP port” determine where a notification shall be sent to.

“Type” tells the interface whether to send Traps or Inform-Requests. Traps are notifications that are sent once without the Event Manager checking whether the configured recipient has received them. In the case of Inform-Requests however, the Event Manager waits for a correct Get-Response from the target. Should a correct Get-Response not be received after 5 seconds, the Inform-Request will be resent. The Event Manager will only resend an Inform-Request once (so twice total) before timing out.

“Community send” sets the sent notifications community string. This community string must match whatever community string the configured recipient has configured. Otherwise, the recipient will not process our sent notification.

**Interface: SNMP-37-79**

General | Endpoints | Event assignment

Active	Address	Label	Location
✓	SNMP-37-79	SNMP system endpoint 6	

Once an SNMP interface has been added, a system endpoint for sending notifications is automatically created. This endpoint will count towards the licensed endpoint count for as long as the checkbox “Notification

**Event plan: SOS Button / Phase: Phase 1**

Endpoints/Notification groups | Settings

Endpoints assigned	Endpoints available	Notification profile
Miller / 1036	Evans / 1037	Please select
Smith / 1038	SNMP system endpoint 3 / SNMP-37-231-Infi	

sending” in that same SNMP interface remains checked. This endpoint cannot be edited or deleted in any way and can’t be assigned to a location.

To give the SNMP interface the ability to send notifications, you need to add this system endpoint into an event plan’s phase like any other notification endpoint. Once that phase is activated, the corresponding SNMP interface will send a matching notification to its configured recipient.

### Interface Status Change

Should the event plan be triggered by the predefined event type “System Info”, the notification will contain data about the interface that triggered it and the current status of it. A “System Info” event is triggered by any interface when its status changes. This event is always triggered in the location “root”. If an SNMP interface is supposed to send notifications about interface status changes, an event plan handling the predefined “System Info” should be configured in that location with a phase containing the SNMP system endpoint as an assigned endpoint. Modifying the event type “System Info” has no influence on this functionality.

Notification name	Data field name	Object Identifier (OID)	Comment
interfaceStatusChange	---	.1.3.6.1.4.1.1027.4.1.1337.0.4	the snmpTrapOID value
	interfaceType	.1.3.6.1.4.1.1027.4.1.1337.1.1.3.1.4	the interface’s type
	interfaceLabel	.1.3.6.1.4.1.1027.4.1.1337.1.1.3.1.2	the interface’s name
	interfaceState	.1.3.6.1.4.1.1027.4.1.1337.1.1.3.1.6	the state the interface has now changed to
	InterfaceDescription	.1.3.6.1.4.1.1027.4.1.1337.1.1.3.1.3	description of the interface

### Event Plan Processing

When a phase with an SNMP endpoint is activated, the corresponding SNMP interface will send a notification to the configured target. This notification will contain a notification ID, the event text, data about what triggered the plan and information on the triggered plan and phase. Once the phase has ended by any means, an additional notification with a matching notification ID will be sent to the target, informing about the end of the phase. This notification does not contain the reason for ending the event phase and/or event plan. The current implementation is offered for evaluation of use cases. Accordingly, this functionality may be further developed and may be subject to technical changes in future software updates.

Notification name	Data field name	Object Identifier (OID)	Comment
activateEventPhase	---	.1.3.6.1.4.1.1027.4.1.1337.0.5	the snmpTrapOID value exact same fields as deactivateEventPhase
deactivateEventPhase	---	.1.3.6.1.4.1.1027.4.1.1337.0.6	the snmpTrapOID value exact same fields as activateEventPhase
	trapEventID	.1.3.6.1.4.1.1027.4.1.1337.0.3.1	this ID matches in corresponding activate and deactivate notifications
	trapEventText	.1.3.6.1.4.1.1027.4.1.1337.0.3.2	the event text
	locationLabel	.1.3.6.1.4.1.1027.4.1.1337.2.1.3.1.2	location where the event plan was triggered
	endpointLabel	.1.3.6.1.4.1.1027.4.1.1337.4.1.3.1.5	name of the endpoint that triggered the event
	endpointCallNumber	.1.3.6.1.4.1.1027.4.1.1337.4.1.3.1.3	call number of the endpoint that

Notification name	Data field name	Object Identifier (OID)	Comment
			triggered the event
	eventTypeLabel	.1.3.6.1.4.1.1027.4.1.1337.3.1.3.1.2	name of the event type
	eventPlanLabel	.1.3.6.1.4.1.1027.4.1.1337.6.1.3.1.2	name of the event plan
	phaseLabel	.1.3.6.1.4.1.1027.4.1.1337.6.1.4.1.3.1.2	name of the phase
	phaseDuration	.1.3.6.1.4.1.1027.4.1.1337.6.1.4.1.3.1.6	Duration of phase in seconds

### ***coldStart Notification***

Once an SNMP interface is correctly configured, it will send a coldStart notification to its configured target. This notification will be sent every time the SNMP interface is modified in such a way that it can work correctly or when it is activated after being switched off. This notification will also be sent when the event manager starts or gets rebooted, if they are configured correctly. These coldStart notifications make the interface visible to SNMP management systems. They are however only supposed to inform the recipient that the SNMP interface itself is configured correctly and ready to send notifications. They do not yield concrete information about the state of the event manager itself or other interfaces. Furthermore, the Event Manager does not send warmStart notifications, even if the configuration of the interface did not change.

### ***Additional Notification Fields***

Each notification contains MIB undefined data fields in addition to their defined ones. These notification fields include information about the Event Manager itself and data that is too specific for the more generic notification type. They are appended after the MIB defined data fields.

Notification name	Data field name	Object Identifier (OID)	Comment
Additional fields	---	---	data fields that get appended to notifications after their MIB defined data types
	sysName	.1.3.6.1.2.1.1.3	appended to all notifications, EVP's name
	systemIPAddress	.1.3.6.1.4.1.1027.4.1.1337.10.3	appended to all notifications, EVP's IP address
	systemMACAddress	.1.3.6.1.4.1.1027.4.1.1337.10.4	appended to all notifications, EVP's MAC address
	systemVersion	.1.3.6.1.4.1.1027.4.1.1337.10.2	appended to all notifications, EVP's version number
	snmpTrapEnterprise	.1.3.6.1.6.3.1.1.4.3	always last data field, contains MITEL's Enterprise OID

### ***Management Information Base***

In order to interpret these messages and their data fields correctly, two MIB files are supplied together with the Event Manager. The first Management Information Base (MIB) is MITEL's root MIB file (MITEL-MIB.mib). It is necessary for the second MIB, the MITEL-EVP-MIB.mib, to work. Both .mib files together contain all the proprietary information that an SNMP agent needs to correctly interpret the specific data and notifications of the Event Manager.

Other RFC defined MIB files that the Event Manager utilizes are SNMPv2-SMI (RFC 2578), SNMPv2-TC (RFC 2579), SNMPv2-CONF (RFC-2580) and SNMPv2-MIB (RFC 3418).

## Receiving notifications

In order to receive and process SNMP notifications, “Notification receiving” must be checked and the fields “Community receive” and “IP port listen” need to be configured. Should this interface only receive notifications, “Notification sending” may be unchecked.

“Community receive” configures the community string all received notifications need to have in order to be processed. In case of wrong community strings the Event Manager will ignore the associated notification and no further processing will take place.

“IP port listen” is the port on which this SNMP interface will listen to Traps/Inform-Requests. Should the SNMP interface not be able to open said port for any reason, its status will change to “Inactive” (red). If that is the case, please select a different listening port. Be aware that two different SNMP interfaces may not use the same listening port!



The Event Manager will process received notifications (Traps and Inform-Requests) in order to trigger events. Received Inform-Requests will be answered with correct Get-Responses and received Traps will not be answered and only processed. Any other type of Request or PDU will be ignored, won’t trigger an event and will go unanswered.

In order for notifications to be processed into events, a receiving endpoint has to be configured. Only notifications from configured and active endpoints will be processed. The “Address” field contains the IP address of the SNMP notification sender that you wish to process notifications from. Incoming notifications that do not come from a configured endpoint will not be processed into an event.

< Interface: SNMP-37-231-Inform

General Endpoints Event assignment Simulator/Trace

+ ↺ 🔍 🗑️



Active	Address ↑	Label	Location	
✓	10.103.31.89	Inveo Thermometer	root	 

If a notification is received from a configured and active endpoint with a correct community string an event will be triggered in the endpoint’s assigned location. The triggered event type is determined by the first matching event assignment. Should no valid Event assignment be found, no event will be triggered.

< Interface: SNMP-37-231-Inform

General Endpoints Event assignment Simulator/Trace

+ ↺

	Label	Object identifier	Ignore indicies	Event type	Re-trigger event timeout	Units	Display hint	
1	inveo alarm	.1.3.6.1.4.1.42814.14.3.5.2.0	0	Temperatur Alarm	10 min	°C	Automatic	 

Field name	Explanation
Nr.	The order in which the event assignments have been created, with the lowest number being the earliest created. The first matching event assignment triggers the corresponding event, starting from the lowest number.
Label	The name of this event assignment.
Object Identifier	The object identifier (OID) this event assignment corresponds to. Should a received SNMP notification contain a field with this <b>exact</b> OID or should its 2nd field be snmpTrapOID (defined: SNMPv2-MIB) and contain that <b>exact</b> OID as its value, this event assignment is chosen and its corresponding event will be triggered in the receiving endpoint’s location.
Ignore Indices	The amount of OID indices from the end (right) that will be ignored on <b>incoming</b> notification’s Object Identifiers. The shortened received OID must still <b>exactly</b> match the configured OID in the field “Object

Field name	Explanation
	Identifier".
Event type	The event type to be triggered if this event assignment is selected.
Re-trigger event timeout	The amount of time an event will <b>NOT</b> be triggered again by the same endpoint should this event assignment be chosen. This is especially useful if an SNMP notification sender sends way too many SNMP notifications in a short amount of time. All timeouts will reset if the corresponding interface is disabled, enabled or changed in any way.
Units	A short text which is appended to the defined OIDs interpreted data. Matches the UNITS clause inside MIB-definitions.
Display-Hint	Select how the defined OIDs value is supposed to be displayed inside the generated event text. Values that would lead to useless results are discarded upon event text generation. Matches the DISPLAY-HINT clause inside MIB-definitions but has been simplified to a dropdown menu. It is recommended to be left on "Automatic" unless you are 100% sure about what value you will receive after this OID. "Text" = 'a'; "Decimal" = 'd'; "Decimal with decimal places: X" = 'd-X'

A valid event assignment is determined by trying to match its configured OID with all received OIDs as well as the OID inside the predefined snmpTrapOID value field. This is the 2<sup>nd</sup> field in any SNMP v2c notification with the OID .1.3.6.1.6.3.1.1.4.1(.0). The first matching event assignment will determine the triggered event and the first matching OID inside the received notification will have its value displayed inside the event text.

The event text contains the triggered event type, the endpoint that triggered it and its address, the chosen event assignment's label and the interpreted value behind the event assignment's "Object Identifier"-field according to its "Display-Hint"-field with the "Units"-field simply appended.

Should the "Object Identifier" field be a snmpTrapOID, the event text will indicate that the received value was a "TRAP TYPE" instead of the interpreted value.

Here are some examples of how event assignments are chosen to more easily visualize them.

Received OIDs	Received values	Event assignment	What gets checked *)	Final result
.1.3.6.1.2.1.1.3.0 .1.3.6.1.6.3.1.1.4.1.0 .7.6.4.12.5.9.8.8	37652723 .1.3.6.1.4.5.5.2.4 "Example Text"	OID: .1.3.6.1.2.1.1.3 Ignore indices: 0	.1.3.6.1.2.1.1.3.0 .1.3.6.1.6.3.1.1.4.1.0 .1.3.6.1.4.5.5.2.4 .7.6.4.12.5.9.8.8	<ul style="list-style-type: none"> <li>No exact match</li> <li>No event trigger</li> <li>The next event assignment will be tried</li> </ul>
.1.3.6.1.2.1.1.3.0 .1.3.6.1.6.3.1.1.4.1.0 .7.6.4.12.5.9.8.8	37652723 .1.3.6.1.4.5.5.2.4 "Example Text"	OID: .1.3.6.1.2.1.1.3 Ignore indices: 1	<u>.1.3.6.1.2.1.1.3.0</u> .1.3.6.1.6.3.1.1.4.1.0 .1.3.6.1.4.5.5.2.4 .7.6.4.12.5.9.8.8	<ul style="list-style-type: none"> <li>Exact match because 1 index ignored</li> <li>Event trigger!</li> </ul>
.1.3.6.1.2.1.1.3.0 .1.3.6.1.6.3.1.1.4.1.0 .7.6.4.12.5.9.8.8	37652723 .1.3.6.1.4.5.5.2.4 "Example Text"	OID: .1.3.6.1.2.1.1.3 Ignore indices: 2	.1.3.6.1.2.1.1.3.0 .1.3.6.1.6.3.1.1.4.1.0 .1.3.6.1.4.5.5.2.4 .7.6.4.12.5.9.8.8	<ul style="list-style-type: none"> <li>No exact match</li> <li>No event trigger</li> <li>The next event assignment will be tried</li> </ul>
.1.3.6.1.2.1.1.3.0 .1.3.6.1.6.3.1.1.4.1.0 .7.6.4.12.5.9.8.8	37652723 .1.3.6.1.4.5.5.2.4 "Example Text"	OID: .7.6.4.12.5.9.8.8 Ignore indices: 0	.1.3.6.1.2.1.1.3.0 .1.3.6.1.6.3.1.1.4.1.0 .1.3.6.1.4.5.5.2.4 <u>.7.6.4.12.5.9.8.8</u>	<ul style="list-style-type: none"> <li>Exact match</li> <li>Event trigger!</li> </ul>



Received OIDs	Received values	Event assignment	What gets checked *)	Final result
.1.3.6.1.2.1.1.3.0 .1.3.6.1.6.3.1.1.4.1.0 .7.6.4.12.5.9.8.8 .1.3.6.1.2.1.1.4.0	37652723 .1.3.6.1.4.5.5.2.4 "Example Text" 50	OID: .1.3.6.1.2.1.1 Ignore indices: 2	<u>.1.3.6.1.2.1.1</u> .3.0 <u>.1.3.6.1.6.3.1.1.4</u> .1.0 <u>.1.3.6.1.4.5.5.2.4</u> <u>.7.6.4.12.5.9.8.8</u> <u>.1.3.6.1.2.1.1</u> .4.0	<ul style="list-style-type: none"> <li>Exact match</li> <li>The first matching OID's value will be used for the event text</li> <li>Event trigger</li> </ul>

\*) Red numbers inside received OIDs are what gets matched with the event assignment OID. Black numbers inside received OIDs are ignored when trying to match with the event assignment OID. Underlined OIDs are matching with the event assignment OID. Bold underlined OIDs are the ones used for the event text.

### Simulator/Trace

The Simulator/Trace tab is used to simulate receiving and sending traps.

< Interface: SNMP-37-231-Inform

General Endpoints Event assignment Simulator/Trace

**Simulator**

Type: Coldstart

Send

Endpoint IP address

SysUpTime (cs)

TrapOID

OID	Value

Receive

**Trace**

Start Clear

Data received ☒

Data sent ☒

Additional info ☒

Status

Simulator for sending

Simulator for receiving

**Text output field**

Trace; adjust what gets put into the Text output field manually request this interface's status

The Trace is used to display what the SNMP interface is sending, receiving as well as other information related to internal activities. "Start" starts the trace output and gets replaced by "Stop", which in turn stops the trace output. "Clear" clears the entire trace output. "Status" prints this SNMP interface's status into the text output field. The checkboxes "Data received", "Data sent" and "Additional info" decide what information is automatically printed into the text output field if the trace has been started. "Data received" enables showing received traps and info relating to it, "Data sent" enables showing sent traps and info relating to it and "Additional info" enables showing info relating to how data is processed and what the result was. Error messages are always printed, regardless of which checkboxes are enabled or disabled, so long as the trace has been started.

The Simulator allows you to force the SNMP interface to send predefined traps with the interface's configuration and allows you to test what happens if the SNMP interface receives a customizable trap.

In order to send a predefined trap, select the type of trap this interface is supposed to send and then press "Send" button. The Event Manager will then send that trap type in accord to its own configuration.



**Simulator**

Type Coldstart ▼

Send

Coldstart

Event (man down)

Status change (current)

In order to see what the Event Manager sends, start the trace and check “Data sent”. This is useful to test if this SNMP interface is correctly configured for sending traps as well as to check if the trap receiver outside the Event Manager handles traps correctly. Data sent by the Simulator is in the correct format, but data itself may or may not be correct.

Endpoint IP address	10.103.31.81
SysUpTime (cs)	2057209
TrapOID	.1.3.1.4.5.10
OID	Value
.1.2.3.4.5.6.7.0	test text
.7.6.5.4.3.2.1.0	1902
.8.8.8.8.8.8	

Receive

The Simulator may also be used in order to simulate receiving a trap to test if the “Event assignment” and “Endpoints” have been done correctly.

Firstly, you need to enter an IP address from which the trap was supposedly sent from.

Secondly, the mandatory SNMP fields “SysUpTime” and “snmpTrapOID” need a valid centisecond value and a correctly formatted OID respectively. The OIDs do not need to be real or MIB defined.

Lastly, you may add up to 3 additional OID-value pairs to the simulated trap. Simply write a real or imagined OID into the left column and a corresponding value into the right column.

When pressing the button “Receive”, this interface will generate a trap with the given values (if possible) and send it to itself. In order to see this generated trap, start the trace and check “Data receiving”. The output window will display the generated trap as well as the result of processing this trap. Should an event be triggered, an event plan must exist in the correct location in order to catch it and be triggered.

## MQTT interface

The MQTT interface connects the Event Manager with an MQTT broker. The interface allows the subscription of user defined topics to the MQTT broker to receive messages from IoT devices that publish their events to this broker. The Event Manager process MQTT messages received from the MQTT broker and trigger events if a match of a user defined condition on an assigned topic is found in the Event Manager configuration. The interface is also able to publish messages to the MQTT broker generated by the Event Manager notification mechanism to trigger actions on other IoT devices that are connected to the same MQTT broker. Up to two MQTT interfaces may be configured.

Only in secure inhouse environments (due to the lack of TLS support and user authentication) an internal MQTT broker may be run as internal application on a dedicated RFP4G of the SIP-DECT system. The performance of this internal broker is sufficient for QoS 0 and usual IoT device traffic (short messages every few seconds), the use of QoS 1 and 2 is not recommended, as there is a higher risk for dropped messages by the broker application in case of high load situations. Therefore the broker configuration is limited by default.

### General Tab

The **General** tab allows configuring the basic settings of the MQTT interface. The following settings can be configured:

- **IP address:** IP address of MQTT broker
- **IP port:** IP port of the MQTT broker (default: 1883)
- **User:** username configured on the broker
- **Password:** password of the user configured on the broker
- **Use TLS:** set this, if TLS should be used as protocol (default IP port: 8883)
- **Validate certificates:** activate, if server certificates of the broker should be validated
- **User defined event text:** Select this check box if 'User defined event text' should be used.

When the correct settings have been configured, the Event Manager will connect to the MQTT broker.

**Please note:** *The internal MQTT broker does not support TLS and is limited by the default configuration.*

**Please note:** *If "Use TLS" is not activated, only unencrypted connections without authentication (usually Port 1883 is used by broker for such connection). Eventually the broker setup must be changed to allow such connections. It is not recommended to setup connections to a MQTT broker outside the LAN due to unencrypted data transfer.*

### Endpoints Tab

The **Endpoints** tab allows the creation of those IoT devices which shall interact with the Event Manager via the MQTT broker.

### User defined event text Tab

In the **User defined event text** tab, it is possible to define special content for the notification messages to addressed endpoints (e.g. SIP-DECT terminals). If this feature is not enabled in the **General** tab, the notification text generated by the MQTT interface consist of the endpoint label and the event type description (or the event type name if the label is empty) and will be used for the notification message. Usually the MQTT payload from IoT devices is not intended to be readable for humans. Therefore this setting and configuration can be used to extract special parts of the received messages and to generate more readable notifications for the receiving SIP-DECT endpoints.

### **Topics Tab**

The **Topics** tab allows the creation of topics that the Event Manager shall subscribe at the MQTT broker or which shall be used on publish for notifications. In the column 'Type' can be selected if the topic is to be used for subscription on the MQTT broker or for publish messages on Event Manager notifications. Each topic must be assigned to a previously created MQTT endpoint. It is possible to configure several topics for an endpoint. All topics must be unique for one interface, it is not possible to create a second entry with the same topic assigned to another endpoint.

MQTT in general allows the use of single level ('+') and multilevel ('#') wildcards in topics on subscription to a broker.

On the Event Manager it is possible to configure any text string as topic including wildcards.

But it is in the responsibility of the Event Manager administrator to configure only valid topic(s) matching the complete topic(s) on which a specific device will publish its data.

**A mapping of MQTT messages resulting from a subscribed topic with wildcards is not possible as the received topic is different from the subscribed topic.**

Temporary it might be useful to configure a wildcard topic for a specific device to get knowledge about topics and payloads that a specific device is publishing in Event Manager trace (a trace window in the GUI is planned but currently not yet available).

If used at all it is strongly recommended to restrict a wildcard topic to a specific device, otherwise the Event Manager might be flooded by MQTT messages from a lot of devices and get unstable if there are many devices connected to the MQTT broker.

### **Subscribe mapping Tab**

The **Subscribe mapping** tab allows the configuration of mappings for received payloads of the MQTT messages to event types. For each MQTT topic can be added one or more mappings with a condition for the payload. A condition is used to decide whether an event trigger shall be generated or not. Different conditions for the same MQTT topic are used to generating different event triggers on different MQTT payload content.

On reception of a MQTT message at first the topic of the message must match a configured topic in the Event Manager (which must not be disabled in the configuration). Additionally, there must exist a 'Subscribe mapping' for this topic which contains a condition to be used for checking the payload of the MQTT message. The assigned event type will only be triggered if the condition match and is not waiting for leaving the configured hysteresis range or a retrigger event timeout.

On reception of a MQTT message all configured conditions mapped to the received topic will be checked. If more than one condition matches the received message also more than one events might be triggered in case of only one received MQTT message.

Depending the configuration of a 'json\_key' for a condition either the json value of the json attribute specified by the key or the complete payload of the MQTT message is checked with the conditions. To access attributes in nested json structures, multiple attribute names can be concatenated by '/', similar to the syntax used for MQTT topics (see the following example):

Examples:

```
Json key:      'foo'
Json data:     {"foo":"bar"}
result:        "bar" will be processed by the Event Manager condition
```

```
Json key:      'foo/bar'  
Json data:     {"foo":{"bar":10.27}}  
result:        10.27 will be processed by the Event Manager condition
```

Please note, that Json arrays are not supported by the Event Manager!

A condition can be one of:

- Same text  
Content to be checked matches the given text exactly
- Contains text  
The given text is part of the content to be checked
- Value equal  
The content to be checked is assumed to be a numerical value and on successful conversion checked to be equal the configured value  
The event will be triggered after at least once the received value is not equal to the configured value and become equal again with a later message or if the time is over which is configured by the 'Re-trigger event timeout'.
- Value smaller/greater  
The content to be checked is assumed to be a numerical value and on successful conversion checked to be smaller/greater the configured value. If selecting this type of condition, a hysteresis value must be configured. A new event normally will not be triggered on each reception of a MQTT message (which might occur quite often). The trigger shall be executed once the condition matches the first time. For enabling retrigger of the same event, a message containing a value above/below hysteresis value of the condition must be received.

For each of the conditions can be configured a 'Re-trigger event timeout' with preconfigured values between 1 minute and 2 hours. In these cases, a timer will be started on each generation of the configured event type. A new event will only be triggered if this timer has already expired.

### ***Publish mapping Tab***

The **Publish mapping** tab allows the configuration of MQTT topics and payloads which shall be added to a publish message during a notification to a MQTT endpoint depending on the event type which has triggered the event plan generating the notification.

In a second configuration step the payload for the publish message must be configured. For a given topic there can be configured several payloads which are selected by the event type and which will trigger the event plan to generate the notification (also to MQTT endpoints). Since no more than exactly one publish message for a dedicated event type is executable, it is not useful to map the same event type and payload with different topics on the same MQTT interface. In those cases, only the first found publish mapping would result in an outgoing publish notification. To avoid those conflicts, it might be useful to configure different endpoints related to more specific topics (see the following example):

Endpoints: tasmota\_AF7B08\_P1, tasmota\_AF7B08\_P2 and tasmota\_AF7B08\_P3

Publish with different publish notifications for POWER1, POWER2 and POWER3 (payload may be 'ON' or 'OFF')

Normally the notification text message generated by the Event Manager is intended to be read by humans and it will not make much sense to use it as payload in a MQTT message in most cases.

If there is a consumer client connected to the MQTT broker which is able and customized to process the notification text messages generated by the Event Manager (e.g. Node Red) than an MQTT topic can be

configured to use the notification text message as payload instead of a payload given by a 'Publish mapping'. To use the notification text message as payload for the MQTT publish message the flag 'Message as payload' must have been activated in the 'Topic' configuration.

***Deletion of MQTT interfaces, topics and endpoints***

If MQTT endpoints, topics and interfaces are deleted by the administrator the following rules apply:

- An MQTT interface can only be deleted if no endpoints with assignment to a location are configured for that interface
- On deletion of an MQTT interface all related endpoints, topics, subscribe and publish mappings are deleted implicitly
- On deletion of an MQTT endpoint all related topics, subscribe and publish mappings are deleted implicitly
- On deletion of an MQTT topic all related subscribe and publish mappings are deleted implicitly

## Web-API interface

The SIP-DECT Event Manager provides a Web-API that allows other applications, including Mitel CloudLink Workflow, easily to interact with the Event Manager and e.g. trigger events or receive notifications from the Event Manager.

The following event-related actions are supported

- Sending an event to the Event Manager ("reqType":"**event**") and thereby triggering the execution of an event plan
- Canceling the execution of an event plan ("reqType":"**eventcancel**")
- Receiving the result of an executed event plan from the Event Manager ("reqType":"**eventresult**")

The following notification-related actions are supported

- Receive notification from the event manager ("reqType":"**notification**")
- Confirmation of a notification to the event manager ("reqType":"**confirmation**")
- Cancellation of a notification by the event manager ("reqType":"**cancel**"), e.g. if all required confirmations have been received, the event plan has been canceled or there is a timeout

Mitel CloudLink Workflow communicates with the Event Manager via the Mitel CloudLink Daemon, which is integrated into the 4th generation base station. The Mitel CloudLink Daemon is currently not yet available for server installations of the Event Manager, i.e. Workflow cannot reach the Event Manager if it is installed on a Rocky Linux® server for DECT Locating, for example.

The Web-API supports incoming web requests with an URL in one of the following forms:

- <https://<event manager IP>:8444/wapi/v1/request>
- <https://<CLD tunnel>/wapi/v1/request>

The Event Manager accepts http GET and POST requests.

The Json body definition is available via the EM Web GUI by clicking on the “Show API” button for the respective request.

The screenshot shows the Mitel Event Manager Web GUI. On the left is a sidebar with a menu: Interfaces, Event types, Notification profiles, Notification groups, Event plans, Locations, Users, System, Overview, and Monitor. The 'Interfaces' section is active, showing a list of interfaces. The 'Interface: WAPI' is selected, and the 'Endpoints' tab is active. The 'Incoming URL' field is set to 'https://<event manager IP>:8444 OR <CLD tunnel> /wapi/v1/request'. Below this, there are four 'Incoming URL' fields with 'Show API' buttons next to them. The first 'Show API' button is highlighted with a red box. A modal window titled 'URL: event' is open, showing a JSON body definition for the 'URL: event' request. The JSON body is highlighted with a red box.

```

{
  "reqType": "event",
  "apiKey": "text",
  "eventRespKey": "text",
  "eventName": "text",
  "sourceEndpoint": {
    "sourceEndpointAddress": "text",
    "sourceEndpointLocation": "optional, reserved for future use"
  },
  "callbackAddress": "text",
  "autoCallback": "optional, 1 - active, else - inactive",
  "xmlApp": "optional, app:n:remaining path?additional parameter=value list",
  "xmlAppName": "optional, app name",
  "eventText": "text"
}

```

There is also a simplified form for triggering an event in which the mandatory parameters are added to the request as URL parameters and a Json body is not necessary. This means that an event can even be triggered by a web browser, e.g. for testing purposes.

```
https://192.168.2.41:8444/wapi/v1/request?type=event&apiKey=5gDem3N3QS6XcTtViujWwiiO5usOJhDoIQ5NocONjMQMmvwezUEFrIntsTjPFGyz&eventName=SOS&eventText=Test&sourceEndpointAddress=118
```

Example of a request with URL parameters to trigger an event instead of

```
{
  "reqType": "event",
  "apiKey": "5gDem3N3QS6XcTtViujWwiiO5usOJhDoIQ5NocONjMQMmvwezUEFrIntsTjPFGyz",
  "eventName": "SOS",
  "sourceEndpoint": {
    "sourceEndpointAddress": "118"
  },
  "eventText": "Test"
}
```

Json body example for the request <https://192.168.2.41:8444/wapi/v1/request> Content-Type application/json with mandatory parameters only

The incoming requests (**event**, **eventcancel**, **confirmation**) require an API key which can be copied to clipboard by clicking on the “Copy to clipboard” button.

The screenshot shows the 'Interface: WAPI' configuration page. On the left is a sidebar with navigation links: Interfaces, Event types, Notification profiles, Notification groups, Event plans, Locations, Users, System, Overview, and Monitor. The main area has tabs for 'General' and 'Endpoints'. Under 'Endpoints', there are 'Save' and 'Refresh' buttons. Below these are several 'Incoming URL' fields with corresponding 'Show API' buttons. The 'API key' field is highlighted with a red box, and next to it is a 'Copy to clipboard' button (also highlighted with a red box) and a 'Renew' button. At the bottom, there is a 'Validate certificates' checkbox.

Outgoing request (**eventresult**, **notification**, **cancel**) are sent as POST requests with the Json body, whose definition is available via the EM Web GUI by clicking on the related “Show API” button.

The notification Json body contains Event Manager CloudLink Daemon information which are required to satisfy the CloudLink tunnel API which is required to send confirmations back to the Event Manager. They are not relevant for other applications attached to the Event Manager via the Web-API.

```

"eventManager": {
  "eventManagerUUID": "text",
  "eventManagerName": "text",
  "component_id": "text",
  "platform_id": "text"
},

```

POST
https://tunnel.dev.api.mitel.io/wapi/v1/request

Headers

Body

Authorization

Testing

Key	Value
<a href="#">Content-Type</a>	<a href="#">application/json</a>
<a href="#">x-mitel-tunnel-service</a>	<a href="#">adminportal</a>
<a href="#">x-mitel-tunnel-platform-id</a>	<a href="#">{{eventManagerPlatformID}}</a>
<a href="#">x-mitel-tunnel-component-id</a>	<a href="#">{{eventManagerComponentId}}</a>
<a href="#">x-mitel-tunnel-component</a>	<a href="#">dectevp</a>

The “Validate certificates” options can be used to activate the validation of the certificates of the servers to which the outgoing requests are sent. Further information on handling certificates can be found in the section System / Security Tab.

### General Tab

The **General** tab allows configuring the basic settings of the Web-API interface. The following settings can be configured:

- **URL: event result:** URL for outgoing responses to the event requests
- **URL: notification:** URL of an external web application (e.g. Workflow) as receiver of notifications from the Event Manager
- **URL: cancel:** URL of an external web application as receiver of Event Manager notifications

Examples:

- for Workflow-API:  
https://workflow.eu.dev.api.mitel.io/2017-09-01/webhooks/accounts/ba8750cb-3032-4015-8fde-feddf81da52f/activities/420ed198-5c77-4c14-9117-7330d64b3343/workers
- for WAPI-Tester (a Python application on Windows or Linux to test the Web-API interface;):  
<http://10.103.37.79:8000>

The tool can be made available on request for testing purposes without any warranty or support.

The Json body definitions for the requests are available via the respective “Show API” buttons.

The “Copy to clipboard” and “Renew” buttons can be used here to copy or renew the API key that is used for authentication with the web API for incoming requests.

The “Validate certificates” options activates the validation of the certificates of the servers to which the



outgoing requests are sent.

### Endpoints Tab

The **Endpoints** tab allows you to create endpoints that can act as event requesters or notification recipients.

### Event types

There are five default Event types ('Escape', 'Man Down', 'No Movement', 'SOS-Key' and 'System Info') available. These types can be changed but cannot be deleted. The default Event types 'Man Down', 'No Move', 'Escape' and 'SOS-Key' correspond to the Alarm triggers which are also available as standard in SIP-DECT.

To handle additional Alarm triggers that may be defined in SIP-DECT OMP, Event types with the same name or short name as the name of the Trigger ID in OMP must be configured in the SIP-DECT Event Manager. All Event types serve as a kind of filter in an Event plan to control the escalation of an event. Based on the assigned priority the system knows in which order the events should be processed. Important events should therefore be configured with a higher priority.

Note: An event displayed on a SIP-DECT terminal will be overwritten by a higher priority event.

### Notification profiles

Notification profiles determine how a notification should be presented to the recipient (DECT phone). It is assigned to the receiving endpoint or notification group within event phases. Only one notification and only that one with the highest priority (Event type priority) is displayed on a DECT phone. Notifications with lower priority are not transmitted to the DECT phone if a message with higher priority is to be displayed. If there are several messages with the same priority at the same time, they will be transmitted one after the other to the DECT phone, with each message being displayed for at least 20 seconds before it is replaced by the next one. One notification profile ('normal') is available by default, this profile cannot be deleted. Click the link under the column 'Label' to change the profile settings (Melody, Ringtone, Volume, etc.) for a profile.

<

Notification profile: WC-Alarm2-Emerg10-km

SIP-DECT

Save

Refresh

Ringtone group

Alarm

Ringtone

Alarm 2

Priority

Emergency

Ring volume

10

Increasing ring volume

☐

Vibration

☐

No alert tone during call

☐

Message logging

☒

Disconnect existing call

☐

Font color

Background color

A Ringtone group is a set or collection of ringtones that can be assigned to specific contacts, groups, or categories. Ringtone groups are used to customize the incoming call alert sounds for different callers or types of calls. The ringtone group can be specifically selected from all the ringtones available from SIP-DECT.

If the 'Increasing ring volume' option is used, the ringtone starts quietly and then gradually reaches the ring volume set. In addition, notification can also be signaled by telephone vibration (if supported by the phone type).

If the 'No alert tone during call' option is active, a notification is delivered without acoustic signaling while the terminal is on a call. If 'Disconnect existing call' is selected, an existing call will be disconnected at the time of the notification.

If 'Message logging' flag is enabled, answered notifications (accepted or rejected) will remain available in the text messages list on the Mitel DECT phone for up to fifteen messages. Further messages will overwrite the oldest message entries in the list. Not answered messages (neither accepted nor rejected) will not be logged in the text messages list on the Mitel DECT phone.

If the telephone supports 'Font color' and 'Background color', the font and color display of the message can be controlled by the SIP-DECT Event Manager.

Restrictions and behavior:

- Settings not supported by the used telephone are ignored.
- 'Priority Low': 'Ringtone group', 'Ringtone', 'Ring volume' and 'Increasing ring volume' has no effect.
- 'Priority Emergency': Pop-up window during call only available with this priority
- Further information about the behavior of displayed messages: Please see the document 'Mitel 600/700 DECT Phone Messaging and Alerting Applications'!

## Notification groups

Endpoints that can receive notifications can be combined into a notification group. This simplifies the configuration regarding the escalation of an event. If the assigned notification group address matches with the source endpoint address then the "Use call address" feature of the event phase can be used.

## Event plans

Event plans describe how to react to certain types of events that occur at different locations. Event plans can consist of up to 10 escalation phases and define the process for handling these events and the resulting notifications in the different phases.

<div>Interfaces</div> <div>Event types</div> <div>Notification profiles</div> <div>Notification groups</div> <div><b>Event plans</b></div> <div>Locations</div> <div>Users</div> <div>System</div> <div>Overview</div> <div>Monitor</div>	<div> <div>+</div> <div>↺</div> <div>🔍</div> </div>				
	Active	Label ↑	Description	Restart plan after completion	Continue running plan on same event
	✓	<a href="#">EP-Escape</a>	EP for Escape alarm trigger	✗	✗
	✓	<a href="#">EP-Fire</a>	EP for Fire alarms	✗	✓
	✓	<a href="#">EP-Mandown</a>	EP for Mandown alarm trigger	✗	✗
	✓	<a href="#">EP-Nano-Sensor</a>	EP for Nano Temperature Sensor	✗	✗

A running event plan is terminated and restarted if the same event is sent from the same endpoint. This can prevent the execution of further phases. With SIP-DECT 10.0 the option is introduced that a running event plan continues to run and further events of the same type from the same endpoint are ignored until the running plan is terminated.

The following settings can be carried out in the **Event plans** configuration pane:

### Filter: Event type Tab

Different types of events can be assigned here to the Event plan. At least the following default Event types are available: **System Info**, **SOS-Key**, **Man Down**, **No Movement**, and **Escape**.

< Event plan: EP-Fire

Filter: Event type Filter: Location Phase

Event types assigned	Event types available
	<div> <div>System Info</div> <div>SOS-Key</div> <div>Man Down</div> <div>No Movement</div> <div>Escape</div> <div>Fire alarm</div> </div>

### Filter: Location Tab

Formerly created locations (to which endpoints are assigned) can be assigned here to the Event plan.

< Event plan: Plan

Filter: Event type Filter: Location Phase

Locations assigned	Locations available
	<div> <div>root</div> </div>

### Phase Tab

Up to 10 phases can be added to an Event plan in the Phase tab with the following configurations:

Interfaces  
Event types  
Notification profiles  
Notification groups  
**Event plans**  
Locations  
User  
System  
Monitor

< Event plan: EP-1

Filter: Event type Filter: Location Phase

	Label	Description	Use call address	with Notification profile	
1	<a href="#">EP1-PH1</a>	Phase 1 of Plan 1	✗		
2	<a href="#">EP1-PH2</a>	Phase 2 of Plan 1 (notification group)	✗		

By editing the phase settings, the 'Use call address' flag can be enabled, and a notification profile may be assigned. With this kind of configuration, a direct assignment of call addresses to a notification group with this address can be realized. In the incoming interface (e.g. ESPA) an endpoint with this call address must be configured.

## Endpoints/Notification groups Tab

Up to 1000 endpoints and/or up to 50 notification groups can be added to a phase or deleted from a phase in the **Endpoints/Notification groups** tab. To each endpoint or notification group a formerly created notification profile can be assigned here also.

## Settings Tab

The following settings can be carried out in the **Settings** tab for a phase:

- The duration in seconds for this phase
- Number of retries (repetitions of this phase)
- Number of confirmations (needed for successful ending of the phase)

Note: 'Individual' implies that all to this phase assigned endpoints must confirm the received notification before the phase ends successfully. If the number of confirmations is not reached, it moves on to the next phase (if configured), is repeated (if configured) or is terminated after the phase has expired.

Note: If there are assigned outgoing endpoints like Modbus or SNMP to a phase, the setting for the number of confirmations should not be set to 'Individual' to avoid unsuccessful phases (because those types of endpoints will never be able to confirm received messages).

## Locations


By defining the locations, a spatial environment can be mapped in a tree structure. A location means the origin of an event. Endpoints that should be used to trigger an event can be assigned to a location here. Endpoints that are not assigned to a location cannot trigger an event.

Interfaces	+ ↺ 🔍		
Event types			
Notification profiles			
Notification groups			
Event plans			
<b>Locations</b>			
Users			
System			
Overview			
Monitor			

Location	Label	Description	
root	<a href="#">root</a>		

The root location is always present and cannot be deleted.

To create a new location, a table line must be selected, and the button  must be pressed. The new location is then based on the location that was selected before.

All endpoints can be assigned to a desired location by following the link under the column 'Label'. The assignment can also be changed via the **Endpoints** tab in the **Interfaces** configuration pane.

## User

The **User** pane allows to create, edit and delete users and to change the passwords of the users. The default user 'admin' with permission 'Configuration' cannot be deleted. There are two other user permissions available for 'Monitoring' and 'Locating' which can be used to add users with those different permissions

## System

The **System** pane consist of the following tabs:

### General Tab

The following settings can be carried out in the **General** tab of the system:

- A system name which subsequently will be displayed also in the headline of the Event manager web application.
- CloudLink daemon can be enabled here (for remote management of the Event Manager)
- CloudLink status is shown here (running or not running)
- The version of the running Event manager application is shown here
- An external IP-Watchdog outside of the system can be configured here which observes a ping from the Event Manager (normally sent at regular interval every 30 seconds as long as it is working correctly). The IP-Watchdog can trigger an alert by Email, SMS or SNMP Trap, or activate a relay for interruption of power for the monitored device to restart the RFP where the Event Manager is configured in case of missing ping from the monitored device.

### Backup/Restart Tab

The following actions can be carried out in the **Backup/Restart** tab of the system:

- **Restart:** The SIP-DECT Event Manager can be re-started with this menu item. The SIP-DECT Event Manager is briefly unavailable.
- **Restart with factory defaults:** All data and settings on the SIP-DECT Event Manager are irreversibly deleted when the factory defaults are restored.
- **Export log:** Log files will be downloaded from the SIP-DECT Event Manager. The log files consist of

two csv files which contain the event summary and the event execution details. Depending on the traffic on the Event Manager there are saved the logs from the last days or weeks (maximum size of the details log is 6 MByte).

- **Export config:** A running configuration of the SIP-DECT Event Manager will be downloaded and saved on the local computer of the administrator.
- **Import config:** Allows to restore an existing configuration to the SIP-DECT Event Manager as zipped file (extension “.gz”) but also as normal text file. A validity check is conducted before activation. A configuration recognized as defective or incomplete will not be activated. During the import of an existing configuration the user data will be used from the running SIP-DECT Event Manager system. If the configuration file was recognized as complete the SIP-DECT Event Manager system will be restarted automatically to activate the imported data backup.

### Security Tab

The following actions can be carried out in the **Security** tab of the system:

- The import of additional trusted certificates which are needed to validate certificates used in the SIP-DECT OMM (for future use) or for interfaces like MQTT or Web-API.
- The import of a local certificate chain and private key (with or without a password) for the SIP-DECT Event Manager which will then be used for the web access to the Event Manager application.
- Via a ‘Delete’ button formerly installed certificates and private keys can be deleted at once.
- Via a dedicated ‘Restart’ button the activation of newly imported certificates or private key into the system will be finalized (import into web server configuration).

If a trusted certificate or a local chain certificate has been installed the number of those certificates will be displayed. There is also visible if a private key has been imported. The names of files with trusted certificate(s) are also displayed in a separate table on this page. Trusted certificates can be deleted again from within this extra table. The local certificate chain and private key can only be deleted again together.

***If a local certificate chain was imported, the corresponding private key (and configuration of needed password) must strongly be done also before a restart of the SIP-DECT Event Manager. Otherwise the system will possibly be unreachable for further configuration via the web admin.***

### Security level Tab

The following actions can be carried out in the **Security level** tab of the system:

- Setting of a security level for the Event manager application (High, Medium, Legacy)
- Configuration of ‘Used Cipher Suites’ for the different Security levels

Normally there is configured as a default the security level ‘High’ and a default setting for ‘Used Cipher Suites’. These settings may be modified here carefully. Therefor a list of the currently configured and of the general configurable cipher suites is shown here. The addition of cipher suites into the ‘Used cipher suites’ could be managed by selecting the cipher suites name from the table entry ‘Supported cipher suites’ with a semicolon in front of it at the end of the listed cipher suites in the upper list entry (Used cipher suites). An entry can simply be deleted from the ‘Used cipher suites’ by editing the table entry after deselection of the ‘Use defaults’ checkbox. In all cases of changing Cipher Suites, the configuration must be finished by pressing the ‘Save’ button.

### CloudLink Tab

The **CloudLink** tab is only visible if the CloudLink daemon has been enabled before in the General tab. Via

this tab a detailed CloudLink Daemon window will be available to connect the Event Manager with the CloudLink portal and to start the tunnel for the remote access to the Event Manager.

Information about the CloudLink Daemon portal and system inventory in the CloudLink Portal will be available with the CloudLink documentation on the Document Center at <https://www.mitel.com/document-center/technology/cloudlink>.

An account with SIP-DECT integration is needed on the CloudLink portal.

Before removing the OMM or Event Manager from an RFP, the tunnels must be stopped and the CloudLink Daemon must be disconnected from CloudLink.

The CloudLink Daemon connects to \*.mitel.io services via https (port 443).

## Overview

The Overview pane displays the currently configured event flow, notification groups, MQTT mappings and interface endpoint relations.

## Monitor

The **Monitor** pane shows a table with the currently active event handlings. Single event lines from this table or all active event handlings can be canceled from this point.

Interfaces	Cancel all						
Event types	Priority	Type	Text	Endpoint	Phase	Confirmations	
Notification profiles	3	SOS-Key	SOS - SDT-732d-247 (247), RFP48-02	SDT-732d-247	EP2-SOS-Phase1	0 / 1	
Notification groups							
Event plans							
Locations							
User							
System							
Monitor							

## Event Log (Summary and Details)

The event logs summary and details can be downloaded via the web admin as .csv files

As of SIP-DECT 10.0, the information has been improved so that there is now a clear indication that a notification has been received by the DECT phone.

Column	Information	Meaning
Status	Notify	Notification was sent to the DECT phone
	Notification received	Notification was received by the DECT phone
	Confirmed	User has confirmed the message (positive or negative)
	Notification terminated	Notification was terminated by the EM
Confirmation	Accepted	User has positively confirmed the notification
	Rejected	User has negatively confirmed the notification
	Not confirmed	User has not yet responded to the notification
	Not received	Notification has not (yet) been received by the DECT phone

In addition, the column headings have been largely adapted to the terms on the EM Web interface, where appropriate.

Time	Event-Id	Phase-Id	Notification-Id	Status	Source	Address	Event	Priority	Text	Location	Plan	Phase	Phase-Count	Destination	Address	Profile	Confirmation
27.01.2025 13:48:14	2			New Event	Patient 118	118 SOS			2 Emergency Call	root	SOS	Phase 1	1				
27.01.2025 13:48:14	2	1		New Phase	Patient 118	118 SOS			2 Emergency Call	root	SOS	Phase 1	1	Supervisor 1	120 SOS		
27.01.2025 13:48:14	2	1	4	Notify	Patient 118	118 SOS			2 Emergency Call	root	SOS	Phase 1	1	Caregiver 1	118 SOS		
27.01.2025 13:48:14	2	1	5	Notify	Patient 118	118 SOS			2 Emergency Call	root	SOS	Phase 1	1	Caregiver 2	119 SOS		
27.01.2025 13:48:14	2	1	6	Notify	Patient 118	118 SOS			2 Emergency Call	root	SOS	Phase 1	1	Supervisor 1	120 SOS		
27.01.2025 13:48:16	2	1	4	Notification received	Patient 118	118 SOS			2 Emergency Call	root	SOS	Phase 1	1	Caregiver 1	118 SOS		
27.01.2025 13:48:16	2	1	5	Notification received	Patient 118	118 SOS			2 Emergency Call	root	SOS	Phase 1	1	Supervisor 1	120 SOS		
27.01.2025 13:48:18	2	1	4	Confirmed	Patient 118	118 SOS			2 Emergency Call	root	SOS	Phase 1	1	Caregiver 1	118 SOS		Accepted
27.01.2025 13:51:14	2	1	5	Notification terminated	Patient 118	118 SOS			2 Emergency Call	root	SOS	Phase 1	1	Caregiver 1	118 SOS		Not confirmed
27.01.2025 13:51:14	2	1	6	Notification terminated	Patient 118	118 SOS			2 Emergency Call	root	SOS	Phase 1	1	Caregiver 2	119 SOS		Not received
27.01.2025 13:51:14	2			Event Finished: Timeout	Patient 118	118 SOS			2 Emergency Call								
27.01.2025 14:13:45	3			New Event	Patient 118	118 SOS			2 Emergency Call								
27.01.2025 14:13:45	3	1		New Phase	Patient 118	118 SOS			2 Emergency Call	root	SOS	Phase 1	1				
27.01.2025 14:13:45	3	1	7	Notify	Patient 118	118 SOS			2 Emergency Call	root	SOS	Phase 1	1	Supervisor 1	120 SOS		
27.01.2025 14:13:45	3	1	8	Notify	Patient 118	118 SOS			2 Emergency Call	root	SOS	Phase 1	1	Caregiver 1	118 SOS		
27.01.2025 14:13:45	3	1	9	Notify	Patient 118	118 SOS			2 Emergency Call	root	SOS	Phase 1	1	Caregiver 2	119 SOS		
27.01.2025 14:13:47	3	1	7	Notification received	Patient 118	118 SOS			2 Emergency Call	root	SOS	Phase 1	1	Supervisor 1	120 SOS		
27.01.2025 14:13:47	3	1	8	Notification received	Patient 118	118 SOS			2 Emergency Call	root	SOS	Phase 1	1	Caregiver 1	118 SOS		
27.01.2025 14:13:51	3	1	7	Confirmed	Patient 118	118 SOS			2 Emergency Call	root	SOS	Phase 1	1	Supervisor 1	120 SOS		Rejected
27.01.2025 14:13:53	3	1	8	Confirmed	Patient 118	118 SOS			2 Emergency Call	root	SOS	Phase 1	1	Caregiver 1	118 SOS		Rejected
27.01.2025 14:16:45	3	1	9	Notification terminated	Patient 118	118 SOS			2 Emergency Call	root	SOS	Phase 1	1	Caregiver 2	119 SOS		Not received
27.01.2025 14:16:45	3			Event Finished: Timeout	Patient 118	118 SOS			2 Emergency Call								



## DECT Locating

### Introduction

The Event Managers DECT Locating supplements the Event Manager functionality described above with a textual and graphical display of the position of a DECT device based on the DECT radio coverage by a base station (typically approx. 30 to 50 meters in buildings depending on the structural conditions and approx. 300 meters in free field) in the event of an emergency call, triggered by pressing the SOS button on the Mitel DECT telephone (722dt, 732d, 742d, 632d(t) V2) or by a sensor alarm of the DECT device (732d, 742d, 632d(t) V2) as well as feature access codes for customer-specific configurable alarm triggers. In addition, the position of a locatable DECT device can also be queried independently of an event.

The main prerequisites for the locating application are:

- Installation of the Event Manager on a Rocky Linux server (in a Microsoft® Hyper-V server environment, a VMware® environment, or in a KVM/QEMU based virtualization environment with the installation type EM).
- Upload of a locating application license and the locating license for a number of locatable DECT users in the Open Mobility Manager
- Configuration of the attributes 'DECT locatable' (and 'Trackable') for those users which should be tracked and/or located by the locating application
- The configuration requires the provision of building and floor plans in the form of graphic files (file formats ".png" and ".jpg" are supported).

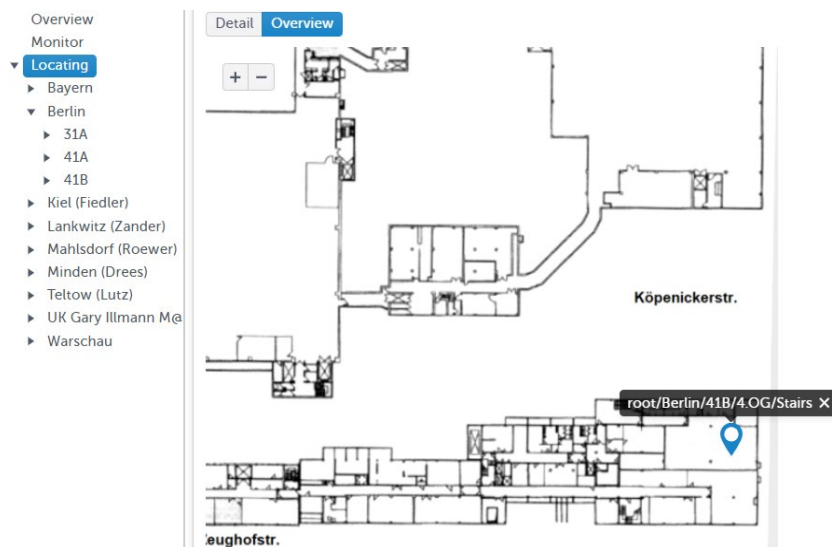
Since the Mitel CloudLink daemon is not available for server installations of the Event Manager, the remote management of the Event Manager and the DECT Locating application is not available in this case.

The graphical representation is available from the locating monitor and from the locating user list in a detailed and in an overview view if the maps have been uploaded to the server and the locations have been placed on these maps.

The screenshot displays the Mitel SIP-DECT 10.0 Locating & EM web interface. The top navigation bar includes 'Interfaces', 'Event types', 'Notification profiles', 'Notification groups', 'Event plans', 'Locations', 'Users', 'System', 'Overview', 'Monitor', and 'Locating'. The 'Locating' menu item is highlighted. The main content area shows a table of users with the following data:

Name	Phone number	Location	On	Last action
Smith, Jerry	322*476	root/Berlin/41B/4.OG/Stairs	On	3/11/2025, 2:07:10 PM

Below the table, a detailed floor plan map is shown for the location 'root/Berlin/41B/4.OG/Stairs'. The map includes labels for various areas: 'Treppenhaus', 'Flur', 'Nutzungsbereich A', 'Nutzungsbereich B', 'Drucker', 'Teeküche', 'Standort', and 'Treppenhaus'. A red location pin is placed on the map, indicating the current location of the device.



## Steps for configuration of the locating application

The configuration must be executed by an administrator user of the Event Manager. The additional menu entry 'Locating' a new page with different tabs (Monitor, Users, Maps, and RFPs) is available only if the Event Manager is running on a Linux server and a locating application license is available in the connected SIP-DECT system.

Interfaces

Event types

Notification profiles

Notification groups

Event plans

Locations

Users

System

Overview

Monitor

Locating

MonitorUsersMapsRFPs

Cancel all

Priority	Type	Text	Endpoint	Phase	Confirmations	
10	Test2	Test2 - Smith, Jerry (322*476), Berlin/41B/4.OG/Stairs	Smith, Jerry	1	0 / 1	<div><div></div><div></div></div>

In the RFPs tab are visible all configured Radio Fixed Parts of the SIP-DECT system. They are automatically imported into the database of the Event Manager. The table includes the name and MAC address of the Radio Fixed Parts as they have been imported from the OMM.

Monitor Users Maps RFPs						
<div> <input type="text"/> <input type="button" value="Import locations"/> </div>						
Name ↑	MAC address	Location	Detail	Overview		
OMM-RFP47-00	08:00:0F:C3:DF:1B		×	×		
RFP35-01	00:30:42:25:83:4F		×	×		
RFP48-02	08:00:0F:C3:DE:C1		×	×		

A location can be assigned here to each of the RFPs or be imported via 'Import locations' button for all RFP. The result is visible in the picture below. The red crosses in the columns for 'Detail' and Overview' show, that these locations are positioned neither on a detail nor an overview map at this time.

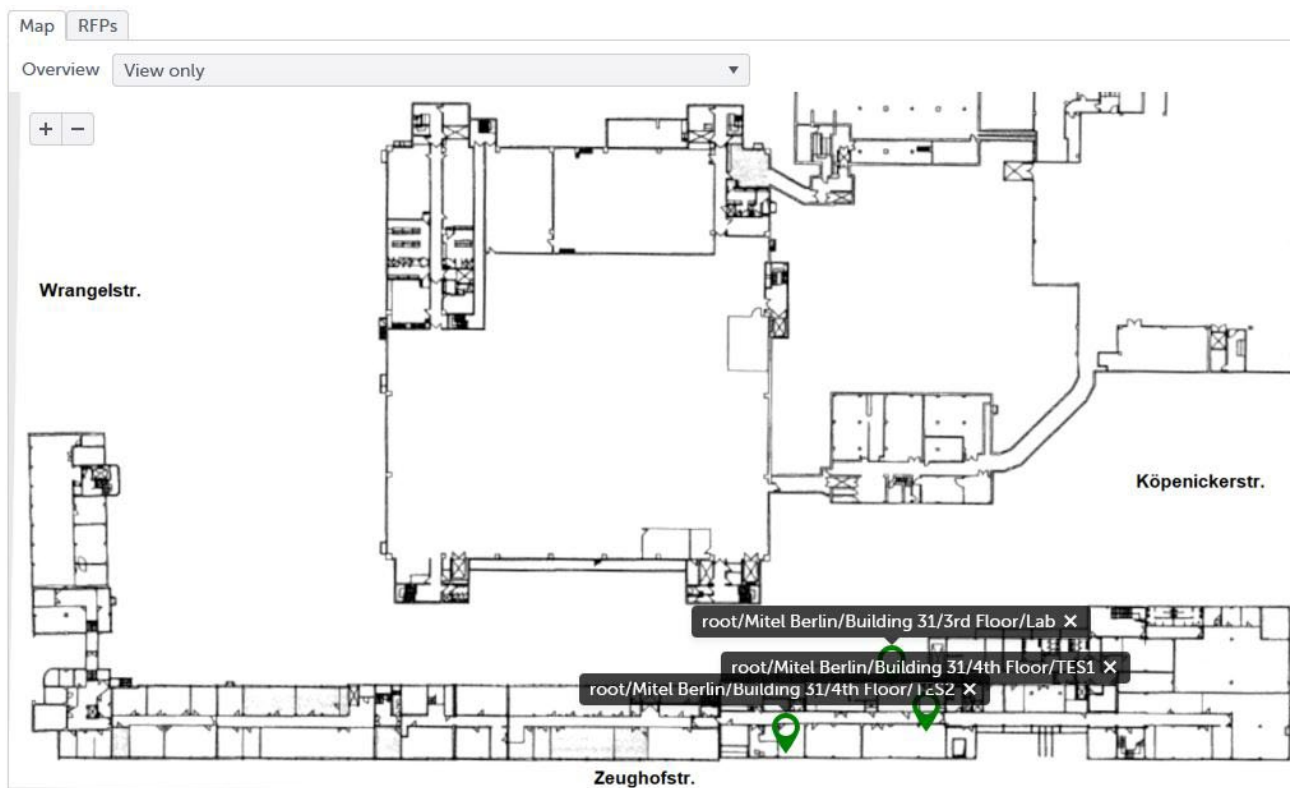
In the next step these necessary maps must be uploaded into the Event Manager via the Maps tab. It is recommended to upload at least one overview map e.g. for the campus and as much detail maps for special floors or building parts. Supported graphic formats are PNG and JPG with resolutions of 1024, 2048, 4096 or 8192 pixel, which leads to zoom levels 1, 2, 3 or 4.

Monitor Users Maps RFPs						
<div> <div> <div></div> <div></div> </div> <div>Import locations</div> </div>						
Name ↑	MAC address	Location	Detail	Overview		
OMM-RFP47-00	08:00:0F:C3:DF:1B	root/Mitel Berlin/Building 41/4th Floor/TES1	×	×		^
RFP35-01	00:30:42:25:83:4F	root/Mitel Berlin/Building 31/4th Floor/Lab	×	×		
RFP48-02	08:00:0F:C3:DE:C1	root/Mitel Berlin/Building 41/4th Floor/TES2	×	×		
Monitor Users Maps RFPs						
<div> <div> <div></div> <div></div> <div></div> </div> </div>						
Label ↑	Image	Zoom level	Location			
Campus Berlin		2	root/Mitel Berlin			^
Monitor Users Maps RFPs						
<div> <div> <div></div> <div></div> <div></div> </div> </div>						
Label	Image	Zoom level	Location			
Campus Berlin		2	root/Mitel Berlin			^
Building 31, Floor 3		1	root/Mitel Berlin/Building 31/3rd Floor			
Building 41, Floor 4		2	root/Mitel Berlin/Building 31/4th Floor			

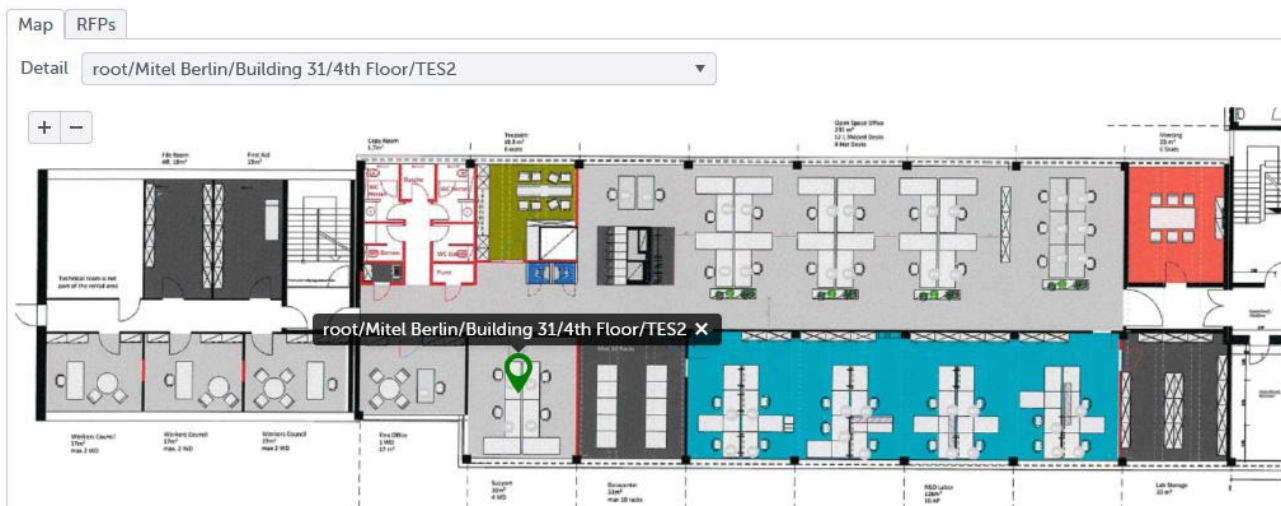
During the upload of the maps the direct assignment to a dedicated location (configured already or imported via the RFPs tab in the step before) is possible, otherwise this must be done in a separate step after the map upload. As a result, the final table shows then all available maps with links to the uploaded images, with value of zoom levels available (depending on the resolution of the uploaded maps) and with the assigned location.

Now the locations must still be positioned on the maps. This is done in the location tree below the menu entry 'Locating' on the different levels (in the example from above these are the entries 'Campus Berlin' for the overview and 'Building 31, 3<sup>rd</sup> Floor' and 'Building 31, 4<sup>th</sup> Floor'.

For the overview there must be selected the dedicated locations step by step from the dropdown menu and then the position marker must be set on the map. At the end the overview (view only) will look like the following picture.



A similar handling is needed for setting of the location positions on the detailed maps, e.g. for a special floor.



When all locations are assigned with their position marks on the detailed and overview maps the table behind the RFP tab in the 'Locating' menu entry will show now green ticks in the columns for 'Detail' and 'Overview' like it is visible in the following picture. If there is still visible a red cross instead of a green tick in one of the columns, this means that there is still missing the location mark on the mentioned map for the dedicated

location.

Monitor Users Maps RFPs					
<input type="button" value="Refresh"/> <input type="text"/> <input type="button" value="Import locations"/>					
Name ↑	MAC address	Location	Detail	Overview	
OMM-RFP47-00	08:00:0F:C3:DF:1B	root/Mitel Berlin/Building 31/4th Floor/TES1	✓	✓	
RFP35-01	00:30:42:25:83:4F	root/Mitel Berlin/Building 31/3rd Floor/Lab	✓	✓	
RFP48-02	08:00:0F:C3:DE:C1	root/Mitel Berlin/Building 31/4th Floor/TES2	✓	✓	

When all locations have the green ticks in the columns 'Detail' and 'Overview', the configuration is completed, and the 'Users' tab in the 'Locating' menu entry will show now the completed list of all those SIP-DECT users that are configured in the SIP-DECT system at least with the 'DECT locatable' attribute and probably also with the 'Trackable' attribute.

The table show the attributes 'Name', 'Phone number', 'Location' (only for trackable users), a graphical link to the location map, the status of the user's phone, and also the timestamp of its last action together with two description fields imported from the SIP-DECT system with information like department or team.

Monitor Users Maps RFPs							
<input type="button" value="Refresh"/> <input type="text"/>							
Name	Phone number	Location		On	Last action	Description 1	Description 2
Andreas Gutschick	325447	root/Campus Kreuzberg/Geb. 41C/4. Flur/TES2		✓	3/11/2025, 10:06:52 AM	R&D	TES1
Frank-Horst Müller	323351	root/Campus Kreuzberg/Geb. 41C/4. Flur/TES1		✓	3/11/2025, 10:00:22 AM	R&D	TES1
Boris Genow	323498	root/Campus Kreuzberg/Geb. 41C/4. Flur/TES1		✓	3/11/2025, 10:06:22 AM	R&D	TES1
Andreas Belz	324498	root/Campus Kreuzberg/Geb. 41C/4. Flur/Druckerraum		✓	3/11/2025, 9:49:11 AM	R&D	TES1
Jörg Tielmann	325459	root/Campus Kreuzberg/Geb. 41C/4. Flur/TES1		✓	3/11/2025, 10:02:45 AM	R&D	TE
Michael Mende	322480	root/Campus Kreuzberg/Geb. 41A/4. Flur/Labor-TEQ hinten		✓	3/11/2025, 10:21:39 AM	R&D	TEQ
Sven Longolius	324235			✓	3/11/2025, 9:14:08 AM	R&D	TES1
Thomas Kloos	323341	root/Campus Kreuzberg/Geb. 41C/4. Flur/TES2		✓	3/11/2025, 10:11:33 AM	R&D	TEQ
René Vieweg	323493	root/Campus Kreuzberg/Geb. 41A/4. Flur/Labor-TEQ hinten		✓	3/11/2025, 10:00:21 AM	R&D	TEQ
Joachim Esper	324417	root/Campus Kreuzberg/Geb. 41C/4. Flur/TES2		✓	3/11/2025, 8:41:12 AM	R&D	TES1

The content of the table will be updated automatically due to actions of the user's phones.

In the 'Monitor' tab of the 'Locating' menu entry additionally to the 'normal' monitor will be visible a location link in case of an event generated by a locatable user, e.g. in case of SOS key or Man Down alarm initiated by a SIP-DECT phone.

Monitor Users Maps RFPs						
<input type="button" value="Cancel all"/>						
Priority	Type	Text	Endpoint	Phase	Confirmations	
3	SOS-Key	SOS - User-245 (245), Mitel Berlin/ Building 31/4th Floor/TES2	User-245	EP-SOS-P1	0 / 1	

## Backup and restoring the Event Manager data including the installed graphic files

The Event Manager database, which can be backed up and restored via the EM web service, does not contain the uploaded graphic files for locating, as the graphic files may be very large.

However, as there is a dependency between the configuration data and the graphic files, these must be backed up and restored together, e.g. when transferring an existing configuration to a new installation.

In addition, the EM application should not be running during the backup and restore process to avoid creating unwanted inconsistencies through parallel activities.

So that these processes do not have to be carried out manually, the Event Manager automatically provides two shell scripts that allow the simple creation of a complete backup and restore. Both scripts must be executed on the command line interface by the root user.

The script `sip-dect-em-create-backup.sh` is used to create a data backup. The script requires as an argument a target directory in which the data backup is to be stored. This directory must already exist. The generated file will then have the name `sip-dect-em-backup_<timestamp>.tar.gz` with the current timestamp from date and time e.g. `20250121_162259`. During the execution of the script the `sip-dect-em` service will be terminated, the user is asked for confirmation again to prevent accidental termination. After completion of creating the backup the `sip-dect-em` service is restarted automatically.

```
[root@deberndws5090 10.0]$ sip-dect-em-create-backup.sh /root/Downloads/
User: root
OK, you are root
check service sip-dect-em:
active
The service 'sip-dect-em' is running. Would you like to stop it? (y/Y): y
Service sip-dect-em successfully stopped.
Create archive: /root/Downloads//sip-dect-em-backup_20250121_162259.tar.gz
...
Archive /root/Downloads//sip-dect-em-backup_20250121_162259.tar.gz created
Start service sip-dect-em
[root@deberndws5090 10.0]$
```

The script `sip-dect-em-restore-backup.sh` is used to restore a data backup. The script requires the name of the backup file as the first argument and a target path as the second. The target path is always the root directory / unless you want the backup to be unpacked at a different location.

```
[root@deberndws5090 10.0]$ sip-dect-em-restore-backup.sh /root/Downloads/sip-dect-em-backup_20250121_162259.tar.gz /
User: root
OK, you are root
OK. Unpack file /root/Downloads/sip-dect-em-backup_20250121_162259.tar.gz to target directory /
check service sip-dect-em:
active
The service 'sip-dect-em' is running. Would you like to stop it? (y/Y): y
Service sip-dect-em successfully stopped.
restore backup from /root/Downloads/sip-dect-em-backup_20250121_162259.tar.gz to /
OK. Unpack file /root/Downloads/sip-dect-em-backup_20250121_162259.tar.gz to target directory /
...
Start service sip-dect-em
[root@deberndws5090 10.0]$
```

To back up the data in the long term, it is recommended to copy the data backup to an external backup medium. As no other services are pre-installed on the secure virtualization images, an external copy protocol client, for example SCP, must be used for this.



## Quick Start Configuration Guide SIP-DECT Event Manager

The following steps need to be followed to get a basic working configuration. There are two basic scenarios.

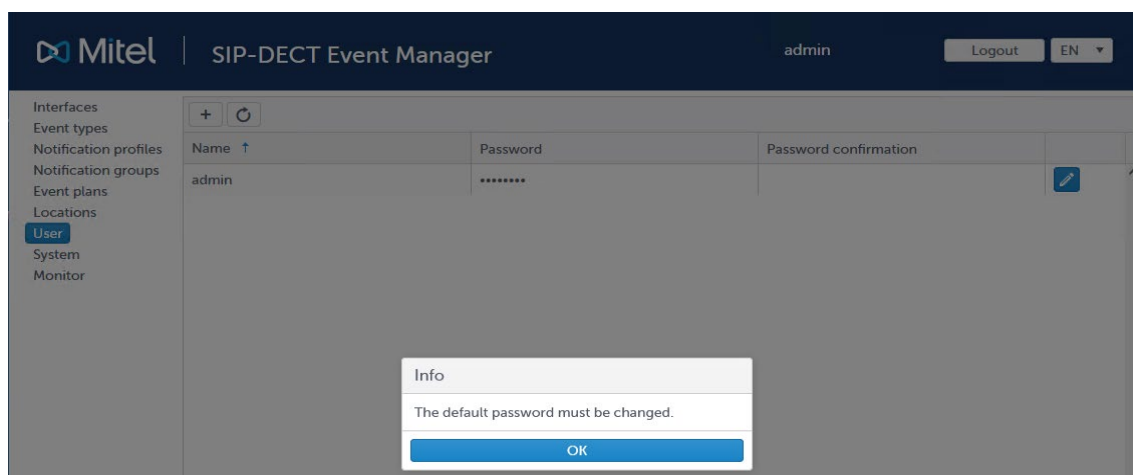
- Configuring a SOS alarm trigger from a SIP-DECT phone
- Configuring an ESPA message

The prerequisite for the following steps is a functioning SIP DECT installation with several Mitel SIP-DECT 602d v2 / 700d terminals. The SIP-DECT terminals are already updated to the SW provided with the SIP-DECT system.

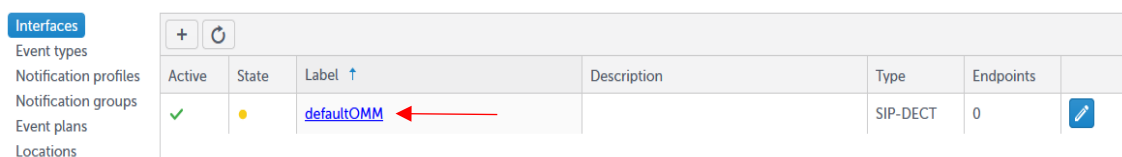
The SIP-DECT Event Manager was started on an RFP using the OM Configurator (OMC) and has the default configuration.

### Configuring SOS alarm trigger from a DECT phone

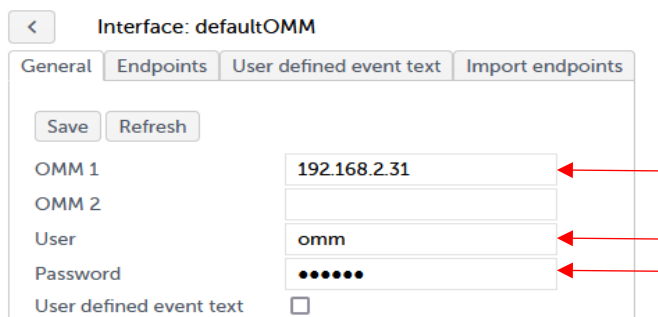
1. Log in to the SIP-DECT Event Manager web service <https://<RFP IP address>:8444> with default login “admin” and password “admin”.
2. Change the default password.



3. Open OMM interface configuration dialog by selecting the link as shown below.



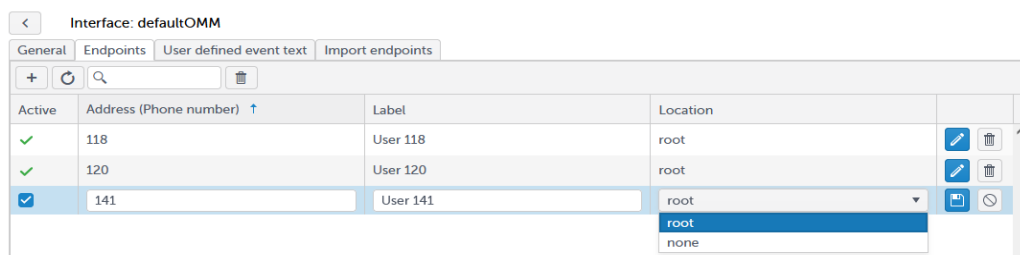
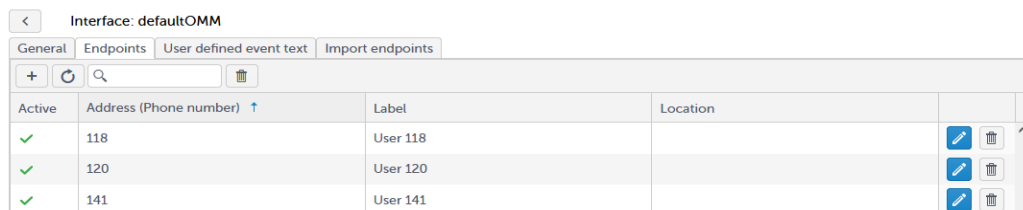
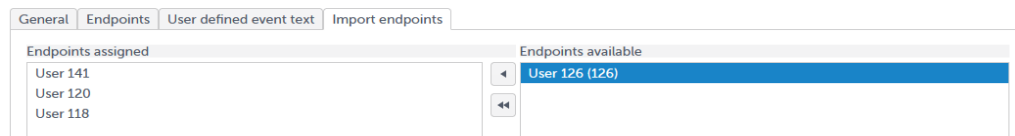
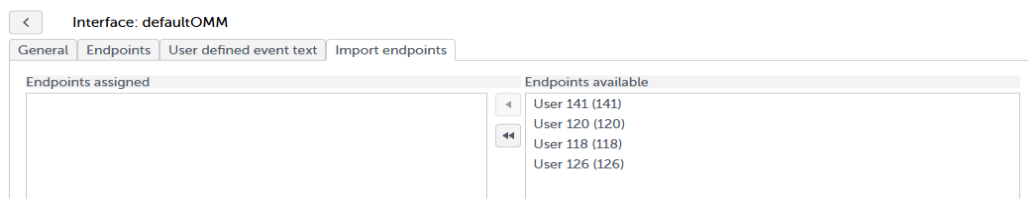
4. Enter the OMM IP address(es), user and password and confirm with ‘Save’. Return to the interface overview by clicking the Back button [←](#).



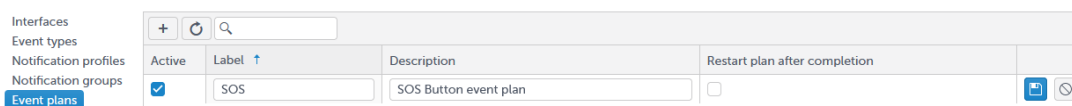
- The interface status should change to green, indicating that the SIP-DECT Event Manager could connect with the OMM.



- Go back into the OMM interface configuration dialog, click the Import endpoints tab and transfer the SIP-DECT endpoints into the SIP-DECT Event Manager configuration by selecting one by one and clicking or all by clicking . As a result the endpoints should now appear in the endpoints list.
- Assign the endpoints to the default location root as shown below.

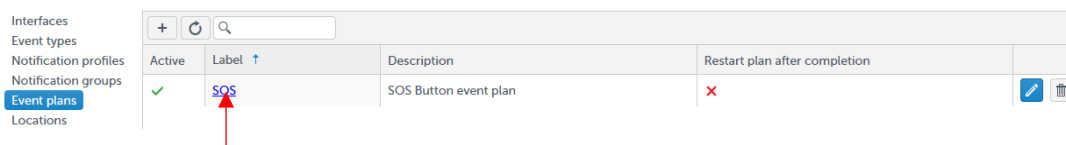


- Click the **Event plans** configuration pane and create a new event plan by clicking . Set a name and description for the plan and confirm with .

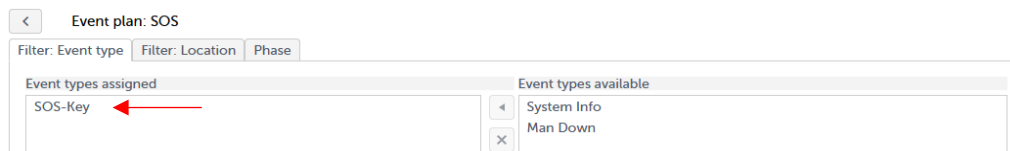




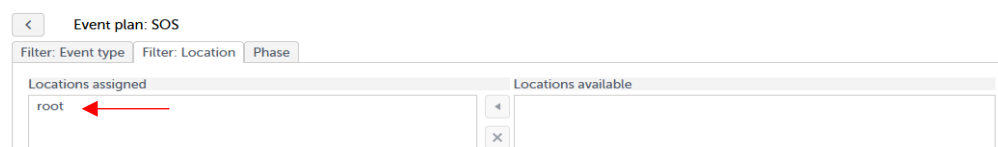
- Click on the newly created plan.



- Under the **Filter: Event type** tab, add the default event type SOS-Key to the event type filter.



- Click **Filter: Location** tab and add the default location root to the location filter.



- Click the **Phase** tab and create a phase for the event plan by clicking New. Set a name and description for the phase and confirm with .



- Open the Phase configuration dialog by selecting the link as shown below.



- Transfer the endpoints you want to be notified into the endpoints list by selecting one by one (to select more than one use additionally the Ctrl key) and press . The default notification profile 'Normal' will automatically be assigned.




No further phase settings need to be changed. Return to the main level dialog by .

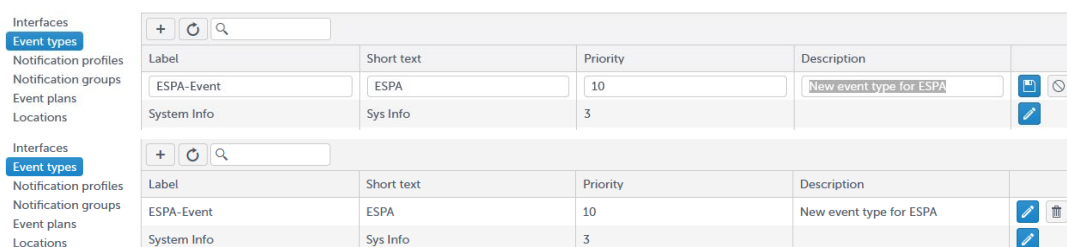
- If now the SOS button is pressed on one of the Mitel SIP-DECT phones, a notification should appear on those SIP-DECT terminals that have been assigned as endpoints to the event plans phase.




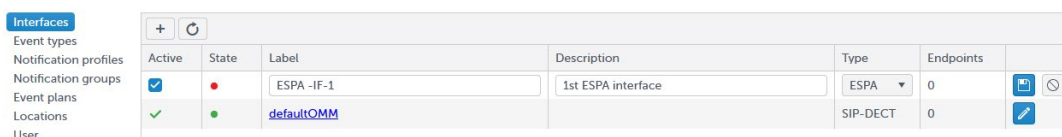
## Configuring an ESPA interface

Execute the same steps to setup the ESPA interface, add the ESPA interface endpoints and assign the default location “root” as described in the Configuring SOS alarm trigger from a DECT phone section. Before a new event plan is created, the ESPA interface must be set up and a new event type must be created.

- Click the **Event types** configuration pane.
- Add a new entry by clicking **+**. Set a unique label and short text and confirm with .



- Click the **Interfaces** configuration pane.
- Add a new entry by clicking **+**. Set a unique label and description and confirm with . Ensure that the interface type ‘ESPA’ is selected under ‘Type’.



- Open the Interface configuration dialog by selecting the created link.



- Enter the IP address and port that the ESPA 4.4.4 of the SIP-DECT Event Manager should connect to, select the Default event type and confirm with Save.

Interface: ESPA-IF-1

General Endpoints User defined event text Event assignment Simulator/Trace

Save Refresh

IP address 192.168.2.71

IP port 10001


Interface supervision ☒

Determine endpoint by Call address

Default event type Please select

Call type 1 (Field 4) terminates event Please select



User defined event text System Info  
SOS-Key  
Man Down  
ESPA-Event

7. Add an ESPA endpoint in the **Endpoints** tab. Set the endpoint address (ESPA field 1 – Call address), assign a name and the default location 'root' and confirm with .

Interface: ESPA-IF-1

General Endpoints User defined event text Event assignment Simulator/Trace

+ ↻ 🔍 🗑️

Active	Address (Field 1)	Label	Location	
<input checked="" type="checkbox"/>	9000	ESPA EP 9000	root	 

8. Return to the interfaces overview by clicking . If the SIP-DECT Event Manager could connect with the nurse call system or similar the interface status turns to green.

Active	State	Label	Description	Type	Endpoints	
<input checked="" type="checkbox"/>		defaultOMM		SIP-DECT	4	
<input checked="" type="checkbox"/>		ESPA-IF-1	1st ESPA interface	ESPA	1	 

9. Create an event plan. Follow steps 8-15 as described in the Configuring SOS alarm trigger from a DECT phone section. However, this time the newly created event type should be assigned to the ESPA interface as the default event type to use.

Interfaces

Event types

Notification profiles

Notification groups

Event plans

Locations

User

System

Monitor

Event plan: ESPA event plan

Filter: Event type Filter: Location Phase

Event types assigned

ESPA-Event

Event types available

System Info

SOS-Key

Man Down

10. To trigger an event even without a connected system, there is useable the simulator function of the ESPA interface

Interfaces

Event types

Notification profiles

Notification groups

Event plans

Locations

User

System

Monitor

Interface: ESPA-IF-1

General Endpoints User defined event text Event assignment Simulator/Trace

Simulator

Send

Call address (1) 9000

Display message (2) Room 123

Ringtone (3) Optional

Call type (4) Optional

Priority (6) Optional

Trace

Stop Clear

Data received ☒

Data sent ☒

Vital sign ☒

View Hex ☐

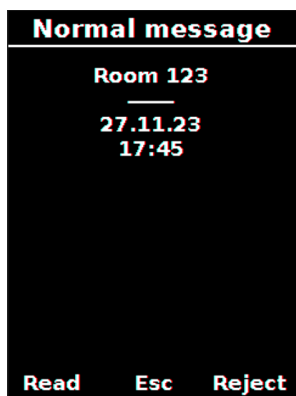
19-02-2024 08:51:40:709 R 1 ENQ 2 ENQ

19-02-2024 08:51:40:709 T ACK

19-02-2024 08:51:40:709 R SOH 1 STX 1 US 9000 RS 2 US Room 123 ETX 0B

19-02-2024 08:51:40:710 T ACK

- When an ESPA message is received, a notification with the received text message should now appear on the Mitel SIP-DECT terminal assigned to the event plans phase.



## Configuring an SNMP interface

This chapter will explain how to configure an SNMP interface to send and receive Traps & Inform-Requests step-by-step. Before following this guide, make sure to have a working SIP-DECT interface with endpoints. As an example for a Trap sender, whose notifications the Event Manager receives and processes, we will be using the Inveo Nano Temperature Sensor.

- Enter the “Interfaces” dialogue. Then create and name a new SNMP interface. Make sure the interface is set to active.

Interfaces

Event types

Notification profiles

Notification groups

Event plans

Locations

Users

System



1

+

↺

Active	State	Label	Description	Type	Endpoints	
<input checked="" type="checkbox"/>	<span style="color: red;">●</span>	SNMP interface		SNMP	0	<div>2</div> <div> <div>3</div> <div>↗</div> </div>
<input checked="" type="checkbox"/>	<span style="color: green;">●</span>	OMM connection		SIP-DECT	3	<div>↗</div> <div>↗</div>
<input checked="" type="checkbox"/>	<span style="color: red;">●</span>	ESPA interface		ESPA	250	<div>↗</div> <div>↗</div>

- Click onto the newly created SNMP interface’s name. You should now be inside the tab “General”. Tick the “Notification sending” box and enter the IP address and IP port of the trap receiver you wish to send traps to into their corresponding fields. Select if you want to send Inform-Requests or simple Traps in the dropdown menu “Type” and proceed to enter a valid community string in the field “Community send”. Make sure the box “Notification receiving” is unchecked for now. Press the button “Save” at the top left.

✓
●
SNMP interface
←
SNMP
1



<
 Interface: SNMP interface

General
 Endpoints
 Event assignment
 Simulator/Trace

Save

Refresh

Notification sending

☒

IP address

10.103.37.85

IP port

162

Type

Inform

Community send

Inform Send

Notification receiving

☐

Community receive

IP port listen

162

- Click the arrow at the top left. You should now be back in the “Interfaces” dialogue from the very start. Make sure the SNMP interface’s status is now active (green). If you did something wrong or forgot to save, it should either be red (inactive) or yellow (misconfigured). If it is green, proceed with the next instructions. If it is any other color, repeat 2.

The screenshot shows the 'Interface: SNMP interface' configuration window. The 'General' tab is selected, showing fields for 'Notification sending' (checked), 'IP address' (10.103.37.85), 'IP port' (162), 'Type' (Inform), 'Community send' (Inform Send), 'Notification receiving' (unchecked), 'Community receive', and 'IP port listen' (162). To the right, the interface status bar shows a green checkmark, a green dot (highlighted with a red box), and the text 'SNMP interface'.

- After clicking onto the SNMP interface’s name and entering its configuration window, click onto the tab “Endpoints”. An Endpoint with the label “SNMP system endpoint X” (with X being a number) should be there and it should be active. If it is not active, check if you have ticked “Notification sending” inside the “General” tab. If you want the SNMP interface to send Traps/Inform-Requests, you will need to add this system endpoint into an event plan’s phase. When that event plan is triggered by an event and it reaches the phase with the SNMP system endpoint in it, the interface will send a notification to its configured destination. In this example, we are adding the SNMP system endpoint into an event plan in the location “root” which gets triggered by the predefined event type “System Info”. This results in our SNMP interface sending interfaceStatusChange notifications when any interface changes its status.

< Interface: SNMP interface

General Endpoints Event assignment Simulator/Trace

+ ↻ 🔍 🗑️

Active	Address ↑	Label	Location
✓	SNMP interface	SNMP system endpoint 3	

1.

Interfaces  
Event types  
Notification profiles  
Notification groups  
**Event plans**  
Locations  
Users  
System  
Overview  
Monitor

+ ↻ 🔍 🗑️

Active	Label ↑
✓	<a href="#">Event Plan: Sys Info</a>

2.

< Event plan: Event Plan: Sys Info

Filter: Event type Filter: Location Phase

Event types assigned

System Info

3.

< Event plan: Event Plan: Sys Info / Phase: System Info SNMP

Endpoints/Notification groups Settings

Endpoints assigned	Endpoints available
SNMP system endpoint 3 / SNMP interface	<div>Evans / 1037</div> <div>Miller / 1036</div> <div>Smith / 1038</div>

4.

< Event plan: Event Plan: Sys Info

Filter: Event type Filter: Location Phase

Locations assigned

root

5. Next, we will configure notification receiving and processing. To start off, we will once again enter the “General” tab of our SNMP interface. Tick the box “Notification receiving”, enter the community string we expect to receive in the text field “Community receive” and enter the IP port this SNMP interface is supposed to listen for notifications on into the field “IP port listen”. Press “Save”.

< Interface: SNMP interface

General Endpoints Event assignment Simulator/Trace

! Save Refresh

Notification sending ☒

IP address 10.103.37.85

IP port 162

Type Inform ▼

Community send Inform Send

Notification receiving ☒

Community receive recvCom

IP port listen 162

6. Leave the “General” tab through the arrow at the top left. In the interface overview window, check if the SNMP interface is still active (green). If it is active, proceed with the next set of instructions. If it is red (inactive) that means the IP port you tried to configure for listening is already taken by some other process or interface. It cannot be used. Re-enter the SNMP interface’s configuration, select a different port and press “Save”. Recheck the interface’s status. Repeat until the SNMP interface is active (green).

Interface: SNMP interface

General Endpoints Event assignment Simulator/Trace

Save Refresh

Notification sending ☒

IP address 10.103.37.85

IP port 162

Type Inform

Community send Inform Send

Notification receiving ☒

Community receive recvCom

IP port listen 162

✓ ● [SNMP interface](#)

7. Now, configure the device you wish to receive notifications from to be able to send Traps/Inform-Requests to the Event Manager's SNMP interface. In this example, we are configuring the Inveo Nano Temperature Sensor to send Traps. This step may look vastly different in your use case with your device. Please follow the instructions of the manufacturer of the device you are configuring and ask them for help if you encounter any problems. Make sure that the trap community that the sending device sends, is the same as that one the Event Manager's SNMP interface has configured in the field "Community receive".

**Read Community :** recvCom

**Write Community:** recvCom

**Trap IP Address 1:** 10.103.37.68

☒ **Enable Trap 1**

8. In order to process the received SNMP notifications, an Endpoint with the sender's IP address needs to be created as well as an Event assignment which reacts to the correct Object Identifier (OID). If you already know the IP address of your SNMP notification sender and what OIDs it is sending in its notifications, you may skip step 9.
9. To process received SNMP notifications an SNMP endpoint with the sender's IP address as well as a matching event assignment need to be created. In order to figure these out easily, enter the SNMP interface's tab "Simulator/Trace". Tick the boxes for "Data received" and "Additional info" and untick the box "Data sent" at the very bottom under the headline "Trace". Now press "Start". The trace window to the right will now display any and all incoming notifications on this interface. In order to figure out the sending devices' IP address as well as the OIDs it is supplying in its sent SNMP notifications, make it send a notification to the Event Manager, read out the displayed IP address and decide which OID you wish to assign an event to. The event manager is incapable of knowing what any received Object Identifier means. This information is contained inside the MIB files of the notification sending device and need to be read out on your own. In this example, the OID ".1.3.6.1.4.1.42814.3.5.2.0" contains the current temperature, which is sent by the Inveo Nano Temperature Sensor if it is too hot or too cold according to its configuration. Once you are done, press "Stop" to deactivate the trace functionality.

**Trace**

Start Clear

Data received ☒

Data sent ☐

Additional info ☒

Status

```



08-01-2025 09:40:40:358
Sender: 10.103.31.89, Endpoint: NO ENDPOINT!
Community: recvCom, Version: v2c, Type: Trap-v2
IN  <- 1 - [ .1.3.6.1.2.1.1.3.0]: Timeticks: (133422) 0:22:14.22
IN  <- 2 - [ .1.3.6.1.6.3.1.1.4.1.0]: OID: .1.3.6.1.4.1.42814.14
IN  <- 3 - [ .1.3.6.1.4.1.42814.14.3.5.2.0]: INTEGER: 22
Could not find an endpoint with a matching IP address on this SNMP interface.
    
```

10. Now that we have the sender's IP address, we will create an SNMP endpoint with the IP in the "Address" field and an easily recognizable label. This can be done inside the "Endpoints" tab inside the SNMP interface. Furthermore, we will assign it the location "root". You may add it to a different, more fitting location.

< **Interface: SNMP interface**

General Endpoints Event assignment Simulator/Trace

+ ↺ 🔍 🗑️

Active	Address	Label	Location	
<input checked="" type="checkbox"/>	10.103.31.89	Inveo Nano	root	 
<input checked="" type="checkbox"/>	SNMP interface	SNMP system endpoint 3		

11. Create a new event type which fits whatever information you are receiving from the SNMP notification sender.

Interfaces

**Event types**

Notification profiles

Notification groups

Event plans








Locations

Users

System

Overview

Monitor

Label	Short text	Priority	Description	
Temperature Alarm	Temp	10	too hot/cold	 
System Info	Sys Info	3		
SOS-Key	SOS-Key	3		
Man Down	Man Down	1		
No Movement	No Move	1		
Escape	Escape	1		

12. Create an event assignment with the correct Object Identifier. Note: The first 2 OIDs of an SNMPv2 notification are the same in all SNMPv2 notifications. Creating an event





assignment that matches with the first 2 OIDs (.1.3.6.1.2.1.1.3.0 & .1.3.6.1.6.3.1.1.4.1.0) will match with ALL correct v2 notifications. Since the first matching event assignment is chosen, this will lead to all SNMP notifications triggering the same event.

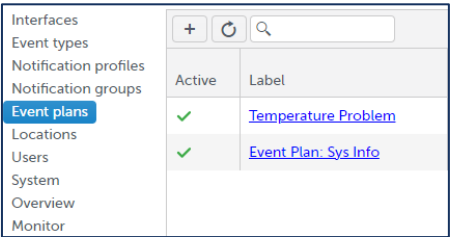
< Interface: SNMP interface

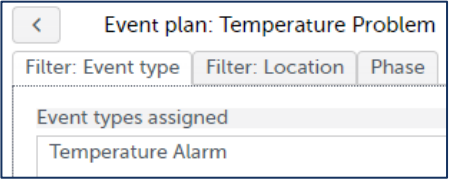
General Endpoints Event assignment Simulator/Trace

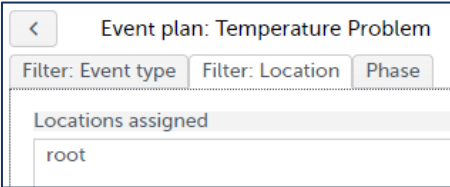
+ ↺

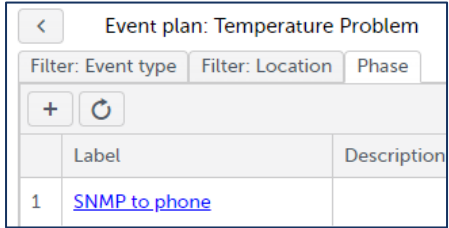
	Label	Object identifier	!!! Ignore indices	Event type	Re-trigger event timeout	Units	Display hint	
0	Temperature Alarm	.1.3.6.1.4.1.42814.3.5.2.0	0	Temperature Alarm	10 min	°C	Automatic	 

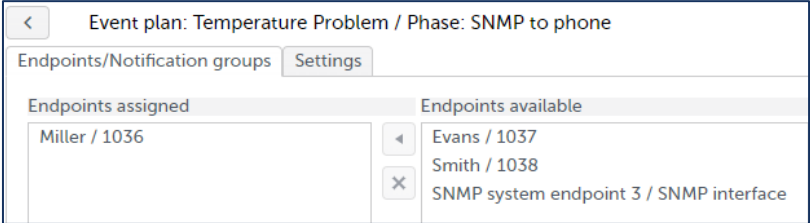
13. Add an event plan into the location you assigned the endpoint to ("root" in this example). This event plan should react to the event type you used inside the "Event assignment" tab of the SNMP interface. Add a phase to the event plan as well as SIP-DECT phones as recipients inside said phase.

1. 

2. 

3. 

4. 

5. 

14. Once the SNMP notification sender sends a trap/inform-request our way and everything was set up correctly, the following message should be displayed inside SIP-DECT phones assigned to a phase of the newly created event plan.



15. In case this did not happen, you may enter the SNMP interface's "Simulator/Trace" tab, tick the boxes "Data received" and "Additional Info" under "Trace" and start the trace. Once an SNMP notification has been received, a message will be displayed after the received notification, telling you what ended up happening while processing the notification. This may indicate what went wrong while setting up the SNMP interface. If the message says that everything went well, but you still do not see a message inside the desired SIP-DECT phone, the problem may lie inside the SIP-DECT interface or within the event plan you set up.

## Appendix

### Sitemap

The following table provides an overview of the Event Manager Web service structure.

Interfaces			
Interfaces	Interface SIP-DECT	General	
		Endpoints	
		User defined event text	
			Text replacement Event text structure
	Interface ESPA	Import endpoints	
			Endpoints assigned Endpoints available
		General	
		Endpoints	
		User defined event text	
			Text replacement Event text structure
	Interface SNMP	Event assignment	
		Simulator/Trace	
			Simulator Trace
		General	
	Interface Modbus	Endpoints	
		Event assignment	
			Endpoint config
		Simulator/Trace	
	Interface MQTT		Inputs Outputs
		General	
		Endpoints	
		User defined event text	
			Text replacement

		Event text structure	
		Topics	
		Subscribe mapping	
		Publish mapping	
Interface Web-API		General	
		Endpoints	
Event types			
Notification profiles			
		SIP-DECT profile	
Notification groups			
		Notification group	
		Endpoints assigned	
		Endpoints available	
Event plans			
		Plan	
		Filter: Event type	
			Event types assigned
			Event types available
		Filter: Location	
			Locations assigned
			Locations available
		Phase	
			Endpoints
			Endpoints assigned
			Endpoints available
			Notification groups assigned
			Notification groups available
		Settings	
Locations			
		Location	
		Endpoints assigned	
		Endpoints available	
Users			
		Name	
		Permission	
		Password	

	Password confirmation		
System	General		
	Backup/Restart		
	Security		
	Security level	Security level	
		Cipher suites	Used cipher suites
			Supported cipher suites
	CloudLink		
Overview			
Monitor			

## Web UI Parameter, Action & Status Information overview

### Event Manager without Locating

Web UI Parameter, Action & Status Information		Description
<b>Interfaces</b>		Configuration pane to administrate the Event Manager's interfaces. Up to 5 interfaces are supported. There is always one SIP-DECT interface which cannot be deleted. Up to 4 incoming ESPA interfaces can be configured.
	<b>Active</b>	Switch to activate or deactivate the interface
	<b>State</b>	Shows the state of the interface (running, misconfigured, inactive)
	<b>Label</b>	Name to identify the interface
	<b>Description</b>	Additional information
	<b>Type</b>	SIP-DECT, ESPA, SNMP, MODBUS
	<b>Endpoints</b>	Shows the number of configured endpoints for the interface. Up to 2000 endpoints in total are supported across all interfaces.
<b>Type SIP-DECT</b>		There is one interface to connect with the SIP-DECT OMM. Standby-OMM configuration is supported. Via this interface, messages are sent to SIP-DECT telephones, confirmations as well as alarm triggers are received from telephones, e.g., SOS, Man Down or Alarm Trigger.
	<b>General</b>	General settings for the SIP-DECT interface
	<b>OMM 1</b>	OMM IP address
	<b>OMM 2</b>	Standby OMM IP address
	<b>User</b>	Username to authenticate with the OMM
	<b>Password</b>	Password to authenticate with the OMM
	<b>User defined event text</b>	Switch to activate or deactivate the user defined event text function
	<b>Endpoints</b>	Via SIP-DECT reachable endpoints (SIP-DECT users)
	<b>Active</b>	Switch to activate or deactivate the endpoint
	<b>Address</b>	Endpoint identifier e.g., telephone number
	<b>Label</b>	Endpoint name
	<b>Location</b>	Location to which the endpoint is assigned
	<b>User defined event text</b>	The user defined event text feature allows to modify or replace the received event text to generate an appropriate notification.
	<b>Text replacement</b>	Simple text replacement function. Up to 10 text replacement rules can be defined.
	<b>Text</b>	Text to be replaced

Web UI Parameter, Action & Status Information		Description
	<b>Replace by</b>	Replacing text
	<b>Event text structure</b>	Function to create a new text from predefined elements. The user defined event text can be composed of up to 4 elements.
	<b>Text</b>	One of the following elements: Event type, Event type short, Priority, Originating endpoint (name), Originating endpoint (address), Location of originating endpoint, Event phase, Received text from interface
	<b>Max. length</b>	Maximum length of text to be inserted
	<b>Spacer</b>	Separator to separate the text elements
	<b>Import endpoints</b>	Function to simplify the setup of SIP-DECT endpoints
	<b>Endpoints assigned</b>	Endpoints which are already imported from SIP-DECT into EVP
	<b>Endpoints available</b>	SIP-DECT endpoints that can still be imported
<b>Type ESPA</b>	Incoming Interface to connect with a nurse call system, fire alarm system or similar via ESPA 4.4.4 over IP.	
	<b>General</b>	General settings for the ESPA interface.
	<b>IP address</b>	IP address of the nurse call system or similar or of the serial IP converter to connect with
	<b>IP port</b>	IP port of the nurse call system or similar or of the serial IP converter to connect with
	<b>Interface supervision</b>	Switch to enable or disable interface monitoring
	<b>Determine endpoint by</b>	Switch for defining the method for determining the endpoint. One of the two options: Call address, Message text
	<b>Default event type</b>	Event type that should be used if no other event type was determined
	<b>Call type 1 (Field 4) terminates event</b>	Switch to activate or deactivate the option that Call type 1 (ESPA Field 4) shall terminate the event
	<b>User defined event text</b>	Switch to activate or deactivate the user defined event text function
	<b>Endpoints</b>	Endpoints that can send events to the Event Manager via the ESPA interface.
	<b>Active</b>	Switch to activate or deactivate the endpoint
	<b>Address</b>	Endpoint identifier e.g., ESPA call address
	<b>Label</b>	Name to identify the endpoint
	<b>Location</b>	Location to which the endpoint is assigned

Web UI Parameter, Action & Status Information		Description
	<b>User defined event text</b>	The user defined event text feature allows to modify or replace the received event text to generate an appropriate notification.
	<b>Text replacement</b>	Simple text replacement function. Up to 10 text replacement rules can be defined (not usable for event type, priority and phase)
	<b>Text</b>	Text to be replaced
	<b>Replace by</b>	Replacing text
	<b>Event text structure</b>	Function to create a new text from predefined elements. The user defined event text can be composed of up to 4 elements.
	<b>Text</b>	One of the following elements: Event type, Event type short, Priority, Originating endpoint (name), Originating endpoint (address), Location of originating endpoint, Phase, Received text from interface
	<b>Max. length</b>	Maximum length of text to be inserted
	<b>Spacer</b>	Separator to separate the text elements
	<b>Event assignment</b>	Function for assigning an event type based on different ESPA 4.4.4 message contents.
	<b>Position</b>	Position of the rule in the list of rules. First matching rule will be applied.
	<b>Ringtone (3)</b>	Ringtone value (ESPA field 3) which should be mapped to the specified event type.
	<b>Priority (6)</b>	Priority value (ESPA field 6) which should be mapped to the specified event type.
	<b>Text (2)</b>	Text value (ESPA field 2) which should be mapped to the specified event type.
	<b>Event type</b>	Event type to be used.
	<b>Text position</b>	Start position in the received event text from which the event text should be copied. 0 - the original event text will be used.
	<b>Text length</b>	Number of characters that should be taken over from the received event text from the start position.
	<b>Event text</b>	Alternative text to replace or add the original event message text.
	<b>Separator</b>	Delimiter which will be followed by a phone number, e.g. for callback
	<b>Simulator/Trace</b>	



Web UI Parameter, Action & Status Information		Description
	<b>Simulator</b>	The simulator function allows to send ESPA messages into the Event Manager to emulate traffic even when the interface is not connected to another system.
	<b>Call address</b>	ESPA Field 1 Call address (mandatory field)
	<b>Display message</b>	ESPA Field 2 Display message (mandatory field)
	<b>Ring tone</b>	ESPA Field 3 Ringtone
	<b>Call type</b>	ESPA Field 4 Call type
	<b>Priority</b>	ESPA Field 6 Priority (1 – alarm, 2 – high, 3 – normal)
	<b>Trace</b>	Function to display traffic on the ESPA interface
	<b>Data received</b>	Switch to enable display of received data
	<b>Data sent</b>	Switch to enable display of sent data
	<b>Vital sign</b>	Switch to enable display of keep alive messages / ESPA polling messages
	<b>View Hex</b>	Switch to enable display of data additionally in hexadecimal format
	<b>Trace window</b>	ESPA traffic display window
<b>Type SNMP</b>	The SNMP interface allows to send SNMPv2c traps or inform-requests to a trap destination and to receive trap messages from SNMP clients.	
	<b>General</b>	General settings of the SNMP interface.
	<b>Notification sending</b>	Switch to activate/deactivate the SNMP sender
	<b>IP address</b>	IP address of the SNMP server (trap receiver).
	<b>IP port</b>	IP port address of the SNMP server (trap receiver) (default: 162).
	<b>Type</b>	Either trap or inform message can be selected.
	<b>Community send</b>	SNMP trap community for SNMP message sending, e.g. 'public'.
	<b>Notification receiving</b>	Switch to activate/deactivate the SNMP receiver
	<b>Community receive</b>	SNMP trap community for SNMP message receiving, e.g. 'trapper'.
	<b>IP port listen</b>	IP port address of the SNMP listener (default: 162).
	<b>Endpoints</b>	Endpoints of the SNMP interface.
	<b>Active</b>	Switch to activate or deactivate the endpoint

Web UI Parameter, Action & Status Information		Description
	<b>Address</b>	Endpoint identifier e.g., SNMP receiver call address or SNMP endpoint IP address from which the Event Manager will receive SNMP notifications
	<b>Label</b>	Name to identify the endpoint
	<b>Location</b>	Location to which the endpoint is assigned
	<b>Event assignment</b>	Function for assigning an event type based on different Object identifiers received in SNMP notifications from SNMP endpoints
	<b>Label</b>	Name to identify the endpoint
	<b>Object identifier</b>	MQTT object identifier
	<b>Ignore indices</b>	Number of bytes to ignore in an object identifier
	<b>Event type</b>	Event type to be triggered when the object identifier is received.
	<b>Re-trigger event timeout</b>	Timeout before retriggering the same event due to received object identifier in a SNMP notification
	<b>Units</b>	Short text to be appended to the defined OIDs interpreted data; matches the UNITS clause inside MIB definitions.
	<b>Display hint</b>	Select how defined OIDs value is supposed to be displayed inside the generated notification event text. Values that would lead to useless results are discarded upon event text generating; matches the DISPLAY-HINT clause inside MIB definitions (simplified to a drop-down menu here).
	<b>Simulator/Trace</b>	
	<b>Simulator</b>	The simulator function allows to send SNMP messages into the Event Manager or to receive SNMP messages to emulate traffic even when the interface is not connected to another system.
	<b>Send</b>	
		Type: Coldstart, Event (Man Down) or Status change (current)
	<b>Receive</b>	
		Endpoint IP address
		SysUpTime (cs)
		TrapOID

Web UI Parameter, Action & Status Information		Description
		OID
		Value
	<b>Trace</b>	Function to display traffic on the SNMP interface
	<b>Data received</b>	Switch to enable display of received data
	<b>Data sent</b>	Switch to enable display of sent data
	<b>Additional info</b>	Switch to enable display of data additionally in hexadecimal format
	<b>Trace window</b>	SNMP traffic display window
<b>Type Modbus</b>	The Modbus interface allows to connect external devices (WAGO/MOXA) with incoming and outgoing ports.	
	<b>General</b>	General settings of the Modbus interface.
	<b>IP address</b>	IP address of the Modbus device.
	<b>IP port</b>	IP port address of Modbus device.
	<b>Endpoints</b>	Endpoints of the Modbus interface.
	<b>Active</b>	Switch to activate or deactivate the endpoint
	<b>Outgoing</b>	Endpoints to which the Event Manager can send messages
	<b>Incoming</b>	Endpoints from which the Event Manager can receive messages
	<b>Event type</b>	Event type to be used
	<b>Idle current</b>	Switch to activate or deactivate idle current for this endpoint
	<b>Alarm delay (sec)</b>	How long the endpoint needs to be activated in order to trigger an event in seconds
	<b>Behavior when returning to normal state</b>	Select the behavior of this endpoint when it returns to its normal state (e.g. "Do not terminate event", "Terminate event immediately" & "Terminate event at the end of phase")
	<b>Address</b>	Endpoint identifier e.g., MODBUS call address
	<b>Label</b>	Name to identify the endpoint
	<b>Location</b>	Location to which the endpoint is assigned
	<b>Simulator/Trace</b>	

Web UI Parameter, Action & Status Information		Description
	<b>Trace</b>	The trace window shows if connection to a Modbus device could be established or not (errors) and if it is possible to received trigger events from incoming endpoints.
	<b>Simulator</b>	The simulator function allows simulation of events on incoming endpoints into the Event Manager to emulate traffic even when the interface is not connected to anything. The status of incoming and outgoing endpoints from a real connected Modbus device will also be shown.
<b>Type MQTT</b>	Interface to connect with a MQTT broker via MQTT protocol.	
	<b>General</b>	General settings for the MQTT interface.
	<b>IP address</b>	IP address of the MQTT broker to connect with
	<b>IP port</b>	IP port of the MQTT broker to connect with
	<b>User defined event text</b>	Switch to activate or deactivate the user defined event text function
	<b>Endpoints</b>	IoT devices from which the Event Manager can receive events via the MQTT interface to the MQTT broker.
	<b>Active</b>	Switch to activate or deactivate the endpoint
	<b>Address</b>	Endpoint identifier e.g., identifier of the IoT device publishing events to the same MQTT broker
	<b>Label</b>	Name to identify the endpoint
	<b>Location</b>	Location to which the endpoint is assigned
	<b>User defined event text</b>	The user defined event text feature allows to modify or replace the received event text to generate an appropriate notification.
	<b>Text replacement</b>	Simple text replacement function. Up to 10 text replacement rules can be defined (not usable for event type, priority and phase)
	<b>Text</b>	Text to be replaced
	<b>Replace by</b>	Replacing text
	<b>Event text structure</b>	Function to create a new text from predefined elements. The user defined event text can be composed of up to 4 elements.
	<b>Text</b>	One of the following elements: Event type, Event type short, Priority, Originating endpoint (name), Originating endpoint (address), Location of originating endpoint, Phase, Received text from interface

Web UI Parameter, Action & Status Information		Description
	<b>Max. length</b>	Maximum length of text to be inserted
	<b>Spacer</b>	Separator to separate the text elements
	<b>Topics</b>	MQTT topic for subscription or publishing
	<b>Active</b>	Switch to activate or deactivate the topic
	<b>Type</b>	Type of topic (Subscribe or Publish)
	<b>Message as payload</b>	Switch to select if notification message should be sent as payload or not in a publish message to the MQTT broker
	<b>Endpoint</b>	Label of the endpoint to which the topic is assigned to
	<b>Subscribe mapping</b>	MQTT topic for subscription or publishing
	<b>Active</b>	Switch to activate or deactivate the subscribe mapping
	<b>Event type</b>	Type of event which shall be triggered by the received MQTT message
	<b>Condition</b>	Condition which will be checked to trigger an event when a MQTT message for a subscribed topic is received (Same text, Contain text, Value equal, Value greater, Value smaller)
	<b>Publish mapping</b>	Mapping of an event type to a publish topic with payload content
	<b>Topic</b>	Publish topic to be send to the IoT device via the MQTT broker
	<b>Event type</b>	Type of event that will trigger the publish message to MQTT message
<b>Type Web-API</b>	Interface to deal with a Web application via HTTPS protocol (RESTapi).	
	<b>General</b>	General settings for the Web-API interface.
	<b>Incoming URL</b>	Fix: https://<IP address of the EM> or <CLD tunnel> /wapi/v1/request
	<b>URL: event</b>	Incoming URL
	<b>URL: event result</b>	URL for outgoing responses to requested events
	<b>URL: event cancel</b>	Incoming URL
	<b>URL: notification</b>	URL for sending of outgoing notifications
	<b>URL: notification</b>	Incoming URL
	<b>URL: cancel</b>	URL for canceling of outgoing notifications
	<b>API key</b>	Buttons for 'Copy to clipboard' and 'Renew' of API key (CloudLink-API)
	<b>Validate certificates</b>	Switch to enable certificate validation during outgoing messages

Web UI Parameter, Action & Status Information		Description
	<b>Endpoints</b>	Internal endpoints for sending/receiving Web-API notifications/requests.
	<b>Active</b>	Switch to activate or deactivate the endpoint
	<b>Address</b>	Endpoint identifier e.g., identifier of the Web-API device requesting events or receiving notifications
	<b>Label</b>	Name to identify the endpoint
	<b>Location</b>	Location to which the endpoint is assigned
<b>Event types</b>	Configuration pane to administrate up to 100 event types. Individual events are mapped to these event types for further processing.	
	<b>Label</b>	Event type name
	<b>Short text</b>	Short (max. 8 character long) event type name
	<b>Priority</b>	Event priority
	<b>Description</b>	Additional information
<b>Notification profiles</b>	Configuration pane to administrate up to 50 notification profiles. Notification profiles define the way notifications are presented by the receiving device.	
	<b>Label</b>	Notification profile name
	<b>Description</b>	Additional information
	<b>SIP-DECT profile</b>	The profile contains various parameter to control the way a notification is indicated on the Mitel 6x2d/700d DECT phone.
	<b>Ringtone group</b>	The Event Manager can control the ringtone to alert the message received on the DECT phone. Various options are available: a) <b>Not to be used for now: None</b> b) Using the device settings with selection of a specific melody setting: Local settings c) Selecting a ringtone from a group: one of the available ringtone groups
	<b>Ringtone</b>	a) If the ringtone group is set to “Local settings”, a specific melody setting of the device can be selected. B) If a ringtone group is set, a melody or sound effect can be selected.
	<b>Priority</b>	SIP-DECT message priority: Low, Normal, High, Emergency
	<b>Ring volume</b>	Ring tone volume which shall be used to indicate the notification.

Web UI Parameter, Action & Status Information		Description
	<b>Increasing ring volume</b>	Enables the automatic volume increase
	<b>Vibration</b>	Enables the vibration function if not automatically activated by the phone based on the message priority.
	<b>No alert tone during call</b>	Switch to turn off the audible indication (in-band) of the received message.
	<b>Message logging</b>	Switch to turn on the message logging on the phone for accepted and rejected messages.
	<b>Disconnect existing call</b>	If activated, ends an existing telephone conversation when the message arrives.
	<b>Font color</b>	Display color of the message text
	<b>Background color</b>	Background color of the message text
<b>Notification groups</b>	Configuration pane to administrate up to 50 notification groups. (maximum 2000 endpoints in total across all groups). Notification groups group endpoints to be notified for easier management. Groups can be assigned to phases of event plans instead of individual endpoints. In addition, notification groups can have addresses to use the "Use call address" function in event plans.	
	<b>Label</b>	Notification group name
	<b>Description</b>	Additional information
	<b>Address</b>	Unique id e.g., telephone number / extension number
	<b>Endpoints assigned</b>	List of endpoints assigned to this group (Label/Address)
	<b>Endpoints available</b>	List of endpoints which could be assigned to this group (Label/Address)
<b>Event plans</b>	Configuration pane to administrate up to 500 event plans. Event plans define processes for handling received events sent by endpoints at the various locations to notify receiving endpoints	
	<b>Active</b>	Switch to activate or deactivate the event plan.
	<b>Label</b>	Event plan name
	<b>Description</b>	Additional information

Web UI Parameter, Action & Status Information		Description
	<b>Restart plan after completion</b>	Switch to enable or disable the restart of the plan after completion (default: off)
	<b>Continue running plan on same event</b>	Switch to enable or disable the continuation of the plan by the same event (default: off)
	<b>Filter: Event type</b>	
	<b>Event types assigned</b>	List of Event types for which the plan is applied, i.e., should be executed.
	<b>Event types available</b>	List of Event types that have not yet been assigned to the plan, i.e., to which the plan is not applied
	<b>Filter: Location</b>	
	<b>Locations assigned</b>	List of Locations to which the plan applies, i.e., the plan is applied to events sent from endpoints assigned to these locations.
	<b>Locations available</b>	List of Locations that have not yet been assigned to the plan, i.e., to which the plan does not apply
	<b>Phase</b>	Event plan phases: up to 10 phases in a single plan and up to 1000 phases in total across all event plans.
	<b>Label</b>	Phase name
	<b>Description</b>	Additional description for the phase.
	<b>Use call address</b>	Option to enable selecting a notification group based on the receiving endpoints address. A notification group with the same address must exist.
	<b>With Notification profile</b>	If the notification group is selected by the endpoints call address, then the specified notification profile will be applied when processing this phase.
	<b>Endpoints/Notification groups</b>	Tab in which endpoints or notification groups to be notified are assigned to the phase.
	<b>Endpoints assigned</b>	Endpoints assigned to this phase (Label / Address).
	<b>Endpoints available</b>	Endpoints which could assigned to this phase.
	<b>Notification profile</b>	Notification profile to be used in this phase for the selected assigned endpoint
	<b>Notification groups assigned</b>	Notification groups assigned to this phase (Label/Address).



Web UI Parameter, Action & Status Information		Description
	<b>Notification groups available</b>	Notification group which could assigned to this phase (Label/Address).
	<b>Notification profile</b>	Notification profile to be used in this phase for the selected assigned group
	<b>Settings</b>	Tab for configuring general phase settings.
	<b>Duration</b>	Duration in seconds
	<b>Number of retries</b>	Never / Permanently / 1..49
	<b>Number of confirmations</b>	Individual (each endpoint) or value between 1 and 49
<b>Locations</b>	Configuration pane to administrate up to 500 endpoint locations. These locations can be assigned here endpoints that probably will send events to the Event Manager. The locations can be used as filter in Event plans so that location-dependent processes can be defined. If different locations are configured, they are displayed in the menu as an expandable tree.	
	<b>Location</b>	Complete location information with parent locations
	<b>Label</b>	Location name
	<b>Description</b>	Additional information
	<b>Endpoints assigned</b>	List of endpoints assigned to this location (Label/Address).
	<b>Endpoints available</b>	List of endpoints which are not assigned to any location and could assigned to this location (Label/Address).
<b>Users</b>	Configuration pane to administrate up to 10 users who have access to the Event Manager's Web service.	
	<b>Name</b>	Username, login name
	<b>Permission</b>	Permission of the user (Configuration, Monitor, Locating)
	<b>Password</b>	User password
	<b>Password confirmation</b>	User password confirmation
<b>System</b>	Administration pane for various administrative activities for the operation of the Event Manager.	
	<b>General</b>	General system settings
	<b>System name</b>	System name
	<b>CloudLink enabled</b>	Switch to enable or disable the CloudLink daemon
	<b>CloudLink status</b>	Displays the status of the CloudLink daemon

Web UI Parameter, Action & Status Information		Description
	<b>Version</b>	Running software version is shown here
	<b>Watchdog</b>	Switch to enable or disable the Watchdog functionality
	<b>Watchdog IP address</b>	IP address of the watchdog that is to be triggered
	<b>Backup/Restart</b>	Options to restart the Event Manager, backup the configuration and the event log.
	<b>Restart</b>	Restart the Event Manager
	<b>Restart with factory defaults</b>	Restart the Event Manager and resets the Event Manager configuration to default
	<b>Export log</b>	Allows to store the alarm log on the PC as a <date>-<time>_evp_summary_log.csv file and <date>-<time>_evp_details_log.csv file
	<b>Export config</b>	Allows to store the Event Manager's configuration on the PC as a <date>-<time>_evp_conf.gz file
	<b>Import config</b>	Allows to restore the Event Manager's configuration from a PC
	<b>Security</b>	Options to import trusted certificate, local certificate chain and private key (with or without password).
	<b>Trusted certificate(s)</b>	Displays how many certificates the Event Manager has
	<b>Local certificate chain</b>	Displays how many local certificate chains the Event Manager has
	<b>Private key</b>	Display if the Event Manager has a working private key
	<b>Private key: password</b>	Enter the password for the imported private key
	<b>Private key: password confirmation</b>	Confirm the password for the imported private key
	<b>Import PEM file with</b>	Define the type of PEM file to be imported
	<b>Import PEM file</b>	Import a PEM file
	<b>Delete certificates/key</b>	Delete all imported certificates and keys
	<b>Make it work</b>	Restart the Event Manager to apply changes
	<b>Security level</b>	Options to configure the security level used by the Event Manager and the used cipher suites for AXI and HTTPS connections.
	<b>Security Level</b>	Select the security level ("Legacy", "Medium" or "High")
	<b>Cipher suites of security level</b>	Select the cipher suites security level ("Legacy", "Medium" or "High")

Web UI Parameter, Action & Status Information		Description
	<b>Use defaults</b>	Switch to use or not use the default settings
	<b>Used cipher suites</b>	List of all used cipher suites (can be edited)
	<b>Supported cipher suites</b>	List of all supported cipher suites
	<b>CloudLink</b>	Shows the current configuration of the CloudLink Daemon and allows to configure connection to CloudLink portal and for Remote Management.
<b>Overview</b>	Area to display the currently configured event flow, notification groups, MQTT mappings and interface endpoint relations.	
<b>Monitor</b>	Area to display the currently active event processing activities and their status and an option to terminate them.	
	<b>Cancel all</b>	Cancel all active event plans
	<b>Priority</b>	Event type priority
	<b>Type</b>	Event type
	<b>Text</b>	Event message text
	<b>Endpoint</b>	Endpoint that triggered the event
	<b>Phase</b>	Current event plan phase
	<b>Confirmations</b>	Received Confirmations/Required Confirmations
	<b>Cancel</b>	Cancel a single active event plan

## Event Manager with Locating

In case of an Event Manager running as PC application on a Linux server and connected to an OMM with an installed 'Mitel SIP-DECT Locating Server License' an additional menu entry in the menu tree is available: Locating.

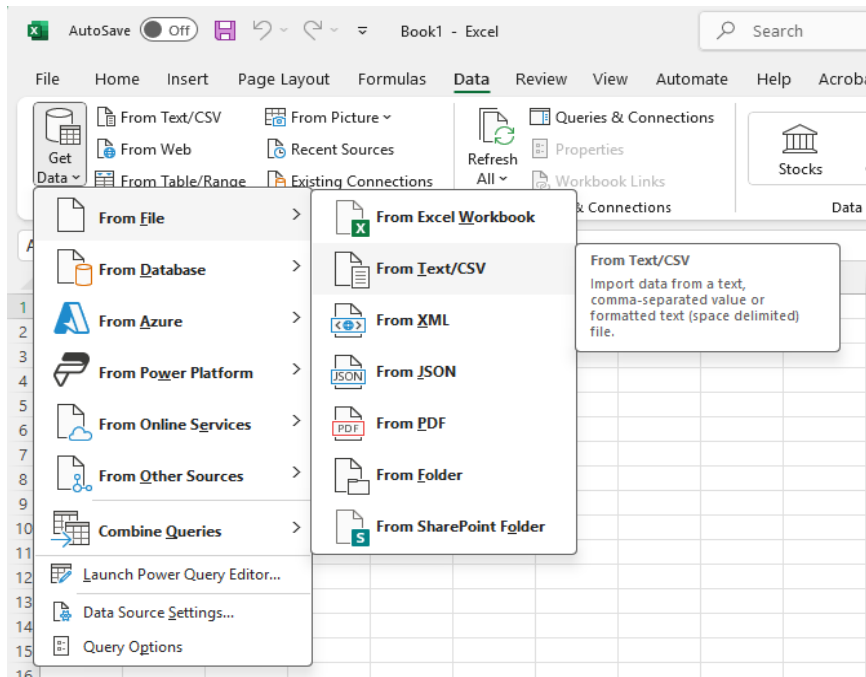
This menu entry is realized as an expandable tree with locations configured in the Event Manager and includes different tabs for Monitor, Users, Maps and Radio Fixed Parts (RFPs). Here is also available a button 'Import locations' to import locations that are already defined in the OMM.

Web UI Parameter, Action & Status Information		Description
<b>Locating</b>	Configuration pane realized as an expandable tree with locations configured in the Event Manager	
	<b>Monitor</b>	Area to display the currently active event processing activities and their status and an option to terminate them.
	<b>Cancel all</b>	Cancel all active event plans
	<b>Priority</b>	Event type priority
	<b>Type</b>	Event type
	<b>Text</b>	Event message text
	<b>Endpoint</b>	Event type priority
	<b>Phase</b>	Current event plan phase
	<b>Confirmations</b>	Received Confirmations/Required Confirmations
	<b>Cancel</b>	Cancel a single active event plans
	<b>Users</b>	List of SIP-DECT users with activated locatable and/or trackable feature
	<b>Name</b>	Username assigned to a locatable DECT device (in SIP-DECT)
	<b>Phone number</b>	Phone number of the DECT device (in SIP-DECT)
	<b>Location</b>	Actual location of the DECT device (based on events from OMM)
		Link to a map showing the current location of the DECT device
	<b>On</b>	Icon to show if the DECT device position has been received already before
	<b>Last action</b>	Date and time of the last known position of a DECT device (based on events from the OMM)
	<b>Description 1</b>	Username assigned to a locatable DECT device (in SIP-DECT)
	<b>Description 2</b>	Cancel a single active event plan
	<b>Maps</b>	Username assigned to a locatable DECT device (in SIP-DECT)

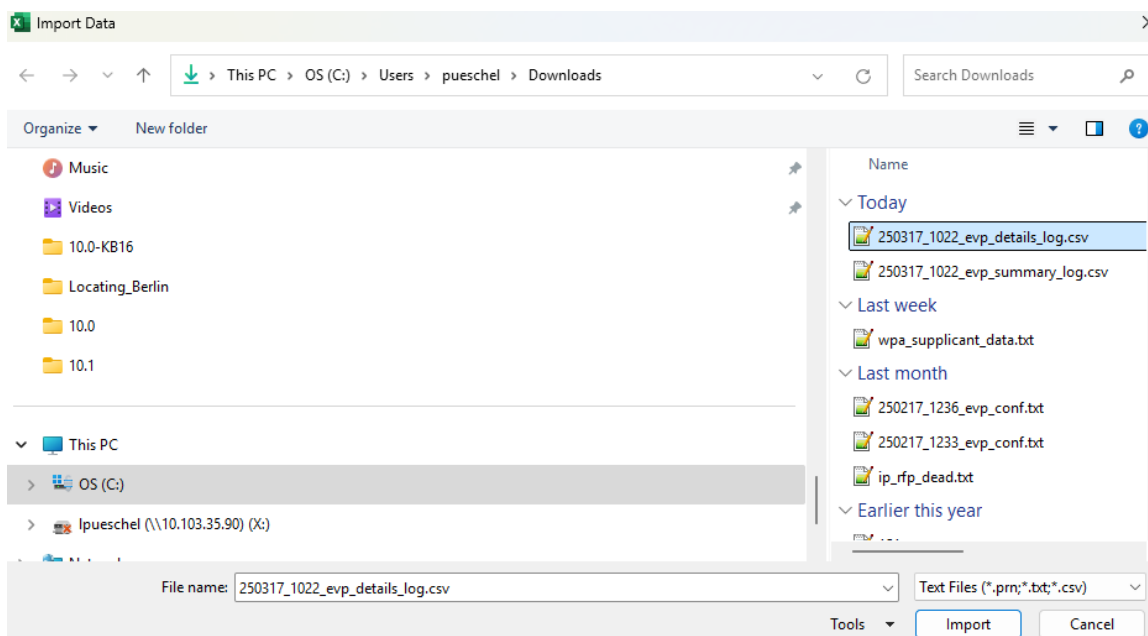
Web UI Parameter, Action & Status Information		Description
	<b>Label</b>	Description for the loaded map
	<b>Image</b>	Link to the loaded map
	<b>Zoom level</b>	Available zoom levels of a loaded map
	<b>Location</b>	Username assigned to a locatable DECT device (in SIP-DECT)
	<b>Save / Delete</b>	Buttons for Saving / Deleting an entry
	<b>RFPs</b>	List of SIP-DECT Radio Fixed Parts (RFP) (automatically imported from the OMM)
	<b>Name</b>	Username assigned to a locatable DECT device (in SIP-DECT)
	<b>MAC address</b>	MAC address of the RFP
	<b>Location</b>	Assigned location of the RFP
	<b>Detail</b>	Icon that shows if the RFP is already positioned on a loaded detail map
	<b>Overview</b>	Icon that shows if the RFP is already positioned on a loaded overview map
	<b>Save</b>	Button to save the table entry

## Recommendation on the procedure for importing log data into Microsoft Excel

A convenient way to import the log data from the Event Manager into a Microsoft Excel file while maintaining the correct formatting of the data is the data import 'Get data from file'.



Select the file you want to import.



Select “Unicode (UTF-8)” for the encoding so that non-ASCII characters are displayed correctly.

Select “Semicolon” as delimiter.

250317\_1022\_evp\_details\_log.csv

File Origin: 65001: Unicode (UTF-8) | Delimiter: Semicolon | Data Type Detection: Based on first 200 rows

Time	Event-Id	Phase-Id	Source	Address	Event	Priority
28-01-2025 11:35:24	1	1	Patient 118	118	SOS	2
28-01-2025 11:35:24	1	1	Patient 118	118	SOS	2
28-01-2025 11:35:24	1	1	Patient 118	118	SOS	2
28-01-2025 11:35:24	1	1	Patient 118	118	SOS	2
28-01-2025 11:35:24	1	1	Patient 118	118	SOS	2
28-01-2025 11:35:26	1	1	Patient 118	118	SOS	2
28-01-2025 11:36:33	1	1	Patient 118	118	SOS	2
28-01-2025 11:38:24	1	1	Patient 118	118	SOS	2
28-01-2025 11:38:24	1	1	Patient 118	118	SOS	2
28-01-2025 11:38:24	1	1	Patient 118	118	SOS	2
28-01-2025 11:38:24	1	1	Patient 118	118	SOS	2
29-01-2025 16:13:02	1	1	Caregiver 1	118	SOS	2
29-01-2025 16:13:02	1	1	Caregiver 1	118	SOS	2
29-01-2025 16:13:02	1	1	Caregiver 1	118	SOS	2
29-01-2025 16:13:02	1	1	Caregiver 1	118	SOS	2
29-01-2025 16:13:02	1	1	Caregiver 1	118	SOS	2
29-01-2025 16:13:03	1	1	Caregiver 1	118	SOS	2
29-01-2025 16:13:05	1	1	Caregiver 1	118	SOS	2
29-01-2025 16:16:02	1	1	Caregiver 1	118	SOS	2
29-01-2025 16:16:02	1	1	Caregiver 1	118	SOS	2
29-01-2025 16:16:02	1	1	Caregiver 1	118	SOS	2
30-01-2025 16:16:02	1	1	Caregiver 1	118	SOS	2
30-01-2025 16:15:06	1	1	Patient 118	118	SOS	2

Then confirm with “Load” to load the data.

The data should then be displayed in in this way.

Time	Event-Id	Phase-Id	Notification-Id	Status	Source	Address	Event	Priority	Text	Location	Plan	Phase	Phase-Count	Destination	Address_1	Profile	Confirmation
03-02-2025 15:42:59	2	1		New Event	Patient 118	118	SOS	2	Emergency Call	root	SOS	Phase 1					
03-02-2025 15:42:59	2	1		New Phase	Patient 118	118	SOS	2	Emergency Call	root	SOS	Phase 1	1				
03-02-2025 15:42:59	2	1	4	Notify	Patient 118	118	SOS	2	Emergency Call	root	SOS	Phase 1	1	Supervisor 1	120	SOS	
03-02-2025 15:42:59	2	1	5	Notify	Patient 118	118	SOS	2	Emergency Call	root	SOS	Phase 1	1	Caregiver 1	118	SOS	
03-02-2025 15:42:59	2	1	6	Notify	Patient 118	118	SOS	2	Emergency Call	root	SOS	Phase 1	1	Caregiver 2	119	SOS	
03-02-2025 15:43:00	2	1	4	Notification received	Patient 118	118	SOS	2	Emergency Call	root	SOS	Phase 1	1	Supervisor 1	120	SOS	
03-02-2025 15:43:00	2	1	5	Notification received	Patient 118	118	SOS	2	Emergency Call	root	SOS	Phase 1	1	Caregiver 1	118	SOS	
03-02-2025 15:43:01	2	1	6	Notification received	Patient 118	118	SOS	2	Emergency Call	root	SOS	Phase 1	1	Caregiver 2	119	SOS	
03-02-2025 15:43:02	2	1	5	Confirmed	Patient 118	118	SOS	2	Emergency Call	root	SOS	Phase 1	1	Caregiver 1	118	SOS	Accepted
03-02-2025 15:43:03	2	1	4	Confirmed	Patient 118	118	SOS	2	Emergency Call	root	SOS	Phase 1	1	Supervisor 1	120	SOS	Accepted
03-02-2025 15:43:03	2	1	6	Confirmed	Patient 118	118	SOS	2	Emergency Call	root	SOS	Phase 1	1	Caregiver 2	119	SOS	Accepted
03-02-2025 15:43:03	2			Event Finished: Confirmed	Patient 118	118	SOS	2	Emergency Call								

If the time does still not contain seconds, the format of the cells must be adjusted. To do this, select the user-defined format “d/m/yyyy hh:mm” and add “:ss” so that the time consists of hours:minutes:seconds (“d/m/yyyy hh:mm:ss”)

The screenshot shows an Excel spreadsheet with columns A through G. Column A contains dates and times. A 'Format Cells' dialog box is open, showing the 'Number' category and 'Time' type. The 'Type' field is set to 'd/m/yyyy hh:mm:ss', which is highlighted with a red box. The dialog box also shows a list of other time formats and a 'Delete' button.

Since the data is linked to the source file, the above steps do not have to be repeated each time. If updated

The screenshot shows the Excel ribbon with the 'Data' tab selected. The 'Queries & Connections' group is visible, and the 'Refresh All' button is highlighted. Below the ribbon, a table of event data is shown, similar to the one in the previous image.

logs are copied to the same location under the same file name, a refresh of the data is sufficient.

The changed data appears after the refresh.



# Mitel SIP-DECT 10.0 Event Manager System Manual

<div> <div>FileHomeInsertPage LayoutFormulasDataReviewViewAutomateHelpAcrobatPower PivotTable DesignQuery</div> <div> <div>GetFrom DataFrom WebFrom Table/RangeFrom PictureRecent SourcesExisting Connection</div> <div> <div>Get &amp; Transform Data</div> <div> <div>RefreshAll</div> <div>Queries &amp; Connections</div> <div>Properties</div> <div>Workbook Links</div> <div>Queries &amp; Connections</div> </div> </div> </div> <div> <div>StocksCurrenciesGeography</div> <div>Data Types</div> <div>Sort &amp; Filter</div> <div>ClearReapply</div> <div>FilterAdvanced</div> <div>Text to ColumnsFlash FillRemove DuplicatesData ValidationConsolidateData ModelWhat-If AnalysisForecastSheetGroupUngroup</div> <div>Data Tools</div> <div>Outlin</div> </div> </div> <div> <div>A1</div> <div>fx</div> </div>																	
Time	Event-Id	Phase-Id	Notification-Id	Source	Address	Event	Priority	Text	Location	Plan	Phase	Phase-Count	Destination	Address 1	Profile	Confirmation	
30-01-2025 16:15:06	1			New Event	Patient 118	118 SOS		2 SOS Alarm from desk phone 118	root	SOS	Phase 1						
30-01-2025 16:15:06	1	1		New Phase	Patient 118	118 SOS		2 SOS Alarm from desk phone 118	root	SOS	Phase 1	1					
30-01-2025 16:15:06	1	1	1	1 Notify	Patient 118	118 SOS		2 SOS Alarm from desk phone 118	root	SOS	Phase 1		1 Supervisor 1		120 SOS		
30-01-2025 16:15:06	1	1	2	2 Notify	Patient 118	118 SOS		2 SOS Alarm from desk phone 118	root	SOS	Phase 1		1 Caregiver 1		118 SOS		
30-01-2025 16:15:06	1	1	3	3 Notify	Patient 118	118 SOS		2 SOS Alarm from desk phone 118	root	SOS	Phase 1		1 Caregiver 2		119 SOS		
30-01-2025 16:15:07	1	1		2 Notification received	Patient 118	118 SOS		2 SOS Alarm from desk phone 118	root	SOS	Phase 1		1 Caregiver 1		118 SOS		
30-01-2025 16:15:07	1	1	1	1 Notification received	Patient 118	118 SOS		2 SOS Alarm from desk phone 118	root	SOS	Phase 1		1 Supervisor 1		120 SOS		
30-01-2025 16:15:08	1	1	2	2 Notification received	Patient 118	118 SOS		2 SOS Alarm from desk phone 118	root	SOS	Phase 1		1 Caregiver 2		119 SOS		
30-01-2025 16:15:15	1	1	2	2 Confirmed	Patient 118	118 SOS		2 SOS Alarm from desk phone 118	root	SOS	Phase 1		1 Caregiver 1		118 SOS	Accepted	
30-01-2025 16:15:18	1	1	3	3 Confirmed	Patient 118	118 SOS		2 SOS Alarm from desk phone 118	root	SOS	Phase 1		1 Caregiver 2		119 SOS	Accepted	
30-01-2025 16:15:20	1	1	1	1 Confirmed	Patient 118	118 SOS		2 SOS Alarm from desk phone 118	root	SOS	Phase 1		1 Supervisor 1		120 SOS	Accepted	
30-01-2025 16:15:20	1			Event Finished: Confirmed	Patient 118	118 SOS		2 SOS Alarm from desk phone 118									
03-02-2025 15:42:49	1			New Event	Patient 118	118 SOS		2 Emergency Call	root	SOS	Phase 1						
03-02-2025 15:42:49	1	1		New Phase	Patient 118	118 SOS		2 Emergency Call	root	SOS	Phase 1	1					
03-02-2025 15:42:49	1	1	1	1 Notify	Patient 118	118 SOS		2 Emergency Call	root	SOS	Phase 1		1 Supervisor 1		120 SOS		
03-02-2025 15:42:49	1	1	2	2 Notify	Patient 118	118 SOS		2 Emergency Call	root	SOS	Phase 1		1 Caregiver 1		118 SOS		
03-02-2025 15:42:49	1	1	3	3 Notify	Patient 118	118 SOS		2 Emergency Call	root	SOS	Phase 1		1 Caregiver 2		119 SOS		
03-02-2025 15:42:50	1	1	1	1 Notification received	Patient 118	118 SOS		2 Emergency Call	root	SOS	Phase 1		1 Supervisor 1		120 SOS		
03-02-2025 15:42:50	1	1	2	2 Notification received	Patient 118	118 SOS		2 Emergency Call	root	SOS	Phase 1		1 Caregiver 1		118 SOS		
03-02-2025 15:42:51	1	1	3	3 Notification received	Patient 118	118 SOS		2 Emergency Call	root	SOS	Phase 1		1 Caregiver 2		119 SOS		
03-02-2025 15:42:52	1	1	1	1 Confirmed	Patient 118	118 SOS		2 Emergency Call	root	SOS	Phase 1		1 Supervisor 1		120 SOS	Accepted	
03-02-2025 15:42:54	1	1	3	3 Confirmed	Patient 118	118 SOS		2 Emergency Call	root	SOS	Phase 1		1 Caregiver 2		119 SOS	Accepted	
03-02-2025 15:42:55	1	1	2	2 Confirmed	Patient 118	118 SOS		2 Emergency Call	root	SOS	Phase 1		1 Caregiver 1		118 SOS	Accepted	
03-02-2025 15:42:55	1			Event Finished: Confirmed	Patient 118	118 SOS		2 Emergency Call									