



HIRSCHMANN

A **BELDEN** BRAND

Hirschmann Automation and Control GmbH

Eagle40-4F HiSecOS Rel. 05101

Reference Manual
Graphical User Interface

User Manual
Configuration



HIRSCHMANN

A **BELDEN** BRAND

Reference Manual

Graphical User Interface

Industrial Firewall

EAGLE404F

The naming of copyrighted trademarks in this manual, even when not specially indicated, should not be taken to mean that these names may be considered as free in the sense of the trademark and tradename protection law and hence that they may be freely used by anyone.

© 2024 Hirschmann Automation and Control GmbH

Manuals and software are protected by copyright. All rights reserved. The copying, reproduction, translation, conversion into any electronic medium or machine scannable form is not permitted, either in whole or in part. An exception is the preparation of a backup copy of the software for your own use.

The performance features described here are binding only if they have been expressly agreed when the contract was made. This document was produced by Hirschmann Automation and Control GmbH according to the best of the company's knowledge. Hirschmann reserves the right to change the contents of this document without prior notice. Hirschmann can give no guarantee in respect of the correctness or accuracy of the information in this document.

Hirschmann can accept no responsibility for damages, resulting from the use of the network components or the associated operating software. In addition, we refer to the conditions of use specified in the license contract.

You find the latest user documentation for your device at: doc.hirschmann.com

Hirschmann Automation and Control GmbH
Stuttgarter Str. 45-51
72654 Neckartenzlingen
Germany

Contents

	Safety instructions	7
	About this Manual	9
	Key	10
	Notes on the Graphical User Interface	11
	Banner	11
	Menu pane	13
	Dialog area	15
1	Basic Settings	19
1.1	System	19
1.2	Network	23
1.2.1	Global	24
1.2.2	IPv4	26
1.3	Software	27
1.4	Load/Save	31
1.5	External Memory	41
1.6	Port	44
1.7	Restart	49
2	Time	51
2.1	Basic Settings	51
2.2	NTP	55
2.2.1	Global	56
2.2.2	Server	58
3	Device Security	61
3.1	User Management	61
3.2	Authentication List	66
3.3	LDAP	68
3.3.1	LDAP Configuration	69
3.3.2	LDAP Role Mapping	75
3.4	Management Access	77
3.4.1	Server	78
3.4.2	IP Access Restriction	90
3.4.3	Web	93
3.4.4	Command Line Interface	94
3.4.5	SNMPv1/v2 Community	96
3.5	Pre-login Banner	97
4	Network Security	99
4.1	Network Security Overview	99
4.2	RADIUS	100
4.2.1	RADIUS Global	101
4.2.2	RADIUS Authentication Server	102
4.2.3	RADIUS Authentication Statistics	104

4.3	Asset	105
4.4	Protocol	109
4.5	Packet Filter	112
4.5.1	Routed Firewall Mode	112
4.5.1.1	Global	114
4.5.1.2	Firewall Learning Mode	116
4.5.1.3	Packet Filter Rule	122
4.5.1.4	Packet Filter Assignment	128
4.5.1.5	Packet Filter Overview	131
4.5.2	Transparent Firewall Mode	132
4.5.2.1	Packet Filter Global	134
4.5.2.2	Packet Filter Rule	136
4.5.2.3	Packet Filter Assignment	144
4.5.2.4	Packet Filter Overview	147
4.6	Deep Packet Inspection	149
4.6.1	Deep Packet Inspection - Modbus Enforcer	150
4.6.2	Deep Packet Inspection - OPC Enforcer	156
4.6.3	Deep Packet Inspection - DNP3 Enforcer	159
4.6.3.1	DNP3 Profile	160
4.6.3.2	DNP3 Object	165
4.6.4	Deep Packet Inspection - IEC104 Enforcer	187
4.6.5	Deep Packet Inspection - AMP Enforcer	194
4.6.5.1	AMP Global	195
4.6.5.2	AMP Profile	198
4.6.6	Deep Packet Inspection - ENIP Enforcer	205
4.6.6.1	ENIP Profile	207
4.6.6.2	ENIP Object	211
4.7	DoS	240
4.7.1	DoS Global	241
5	Virtual Private Network	245
5.1	VPN Overview	245
5.2	VPN Certificates	253
5.3	VPN Connections	257
6	Switching	281
6.1	Switching Global	281
6.2	Rate Limiter	283
6.3	Filter for MAC Addresses	285
6.4	QoS/Priority	286
6.4.1	QoS/Priority Global	288
6.4.2	QoS/Priority Port Configuration	289
6.4.3	802.1D/p Mapping	290
6.5	VLAN	291
6.5.1	VLAN Global	292
6.5.2	VLAN Configuration	293
6.5.3	VLAN Port	295

7	Routing	297
7.1	Routing Global	297
7.2	Routing Interfaces	299
7.2.1	Routing Interfaces Configuration	300
7.2.2	Routing Interfaces Secondary Interface Addresses	306
7.3	ARP	307
7.3.1	ARP Global	308
7.3.2	ARP Current	310
7.3.3	ARP Static	312
7.4	Open Shortest Path First	313
7.4.1	OSPF Global	315
7.4.2	OSPF Areas	323
7.4.3	OSPF Stub Areas	325
7.4.4	OSPF Not So Stubby Areas	327
7.4.5	OSPF Interfaces	330
7.4.6	OSPF Virtual Links	335
7.4.7	OSPF Ranges	338
7.4.8	OSPF Diagnostics	340
7.5	Routing Table	351
7.6	L3 Relay	355
7.7	Loopback Interface	359
7.8	L3-Redundancy	361
7.8.1	VRRP	361
7.8.1.1	VRRP Configuration	362
7.8.1.2	VRRP Statistics	373
7.8.1.3	VRRP Tracking	375
7.9	NAT	376
7.9.1	NAT Global	377
7.9.2	1:1 NAT	380
7.9.2.1	1:1 NAT Rule	382
7.9.3	Destination NAT	385
7.9.3.1	Destination NAT Rule	387
7.9.3.2	Destination NAT Mapping	392
7.9.3.3	Destination NAT Overview	394
7.9.4	Masquerading NAT	395
7.9.4.1	Masquerading NAT Rule	397
7.9.4.2	Masquerading NAT Mapping	400
7.9.4.3	Masquerading NAT Overview	402
7.9.5	Double NAT	404
7.9.5.1	Double NAT Rule	406
7.9.5.2	Double NAT Mapping	409
7.9.5.3	Double NAT Overview	411
8	Diagnostics	413
8.1	Status Configuration	413
8.1.1	Device Status	414
8.1.2	Security Status	418

8.1.3	Alarms (Traps)	423
8.1.3.1	Trap Destinations	424
8.2	System	426
8.2.1	System Information	427
8.2.2	Configuration Check.	428
8.2.3	ARP	430
8.2.4	Selftest	431
8.3	Syslog	433
8.4	Ports	436
8.5	LLDP	436
8.5.1	LLDP Configuration	437
8.5.2	LLDP Topology Discovery	441
8.6	Report	442
8.6.1	Report Global	443
8.6.2	Persistent Logging	447
8.6.3	System Log	450
8.6.4	Audit Trail	451
9	Advanced	453
9.1	DNS	453
9.1.1	DNS Client	453
9.1.1.1	DNS Client Global	454
9.1.1.2	DNS Client Current	455
9.1.1.3	DNS Client Static	456
9.1.2	DNS Cache	457
9.1.2.1	DNS Cache Global.	458
9.2	Tracking	458
9.2.1	Tracking Configuration	460
9.2.2	Tracking Applications	466
9.3	Command Line Interface	466
A	Index	469
B	Technical support	473
C	Readers' Comments	474

Safety instructions

WARNING

UNCONTROLLED MACHINE ACTIONS

To avoid uncontrolled machine actions caused by data loss, configure all the data transmission devices individually.

Before you start any machine which is controlled via data transmission, be sure to complete the configuration of all data transmission devices.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

About this Manual

The “Configuration” user manual contains the information you need to start operating the device. It takes you step by step from the first startup operation through to the basic settings for operation in your environment.

The “Installation” user manual contains a device description, safety instructions, a description of the display, and the other information that you need to install the device.

The “Graphical User Interface” reference manual contains detailed information on using the graphical user interface to operate the individual functions of the device.

The “Command Line Interface” reference manual contains detailed information on using the Command Line Interface to operate the individual functions of the device.


The Industrial HiVision Network Management software provides you with additional options for smooth configuration and monitoring:

- Auto-topology discovery
- Browser interface
- Client/server structure
- Event handling
- Event log
- Simultaneous configuration of multiple devices
- Graphical user interface with network layout
- SNMP/OPC gateway

Key

The designations used in this manual have the following meanings:

	List
	Work step
Link	Cross-reference with link
Note:	A note emphasizes a significant fact or draws your attention to a dependency.
<code>Courier</code>	Representation of a CLI command or field contents in the graphical user interface

 Execution in the Graphical User Interface

 Execution in the Command Line Interface

Notes on the Graphical User Interface

The prerequisite to use the Graphical User Interface of the device is a web browser with HTML5 support.

The responsive Graphical User Interface automatically adapts to the size of your screen. Consequently, you can see more details on a large, high-resolution screen than on a small screen. For example, on a high-resolution screen, the buttons have a label next to the icon. On a screen with a small width, the Graphical User Interface displays only the icon.

Note: On a conventional screen, you click to navigate. On a device with a touchscreen, on the other hand, you tap. For simplicity, we only use "click" in our help texts.

The Graphical User Interface is divided as follows:

- [Banner](#)
- [Menu pane](#)
- [Dialog area](#)

Banner

The banner displays the following information:



Displays and hides the menu. When the web browser window is too narrow, the Graphical User Interface hides the menu pane. The banner displays the button instead.

Brand logo

Click the logo to open the website of the manufacturer of the device in a new window.

Dialog name

Displays the name of the dialog currently displayed in the dialog area.



Displays that the web browser cannot contact the device. The connection to the device is interrupted.



Displays if the settings in the volatile memory ([RAM](#)) differ from the settings of the "Selected" configuration profile in the non-volatile memory ([NVM](#)). The banner displays the icon if you have applied the settings, but not yet saved them in the non-volatile memory ([NVM](#)).



When you click the button, the online help opens in a new window.



When you click the button, a tooltip displays the following information:

- The summary of the *Device status* frame. See the *Basic Settings > System* dialog.
- The summary of the *Security status* frame. See the *Basic Settings > System* dialog.

A red dot next to the icon means that at least one of the values is greater than 0.



When you click the button, a submenu opens with the following menu items:

- User account name
The account name of the user that is currently logged in.
- *Logout* button
When you click the button, this logs out the currently logged in user. Then the login dialog opens.

Menu pane

When the web browser window is too narrow, the Graphical User Interface hides the menu pane.

To display the menu pane, click the  button in the banner.

The menu pane is divided as follows:

- [Icons bar](#)
- [Menu tree](#)

Icons bar

The icons bar displays the following information:


Device software

Displays the version number of the currently running device software that the device loaded during the last system startup.



Displays a text field to search for a keyword. When you enter a character or string, the menu tree displays a menu item only for those dialogs that are related to this keyword.



The menu tree displays a menu item only for those dialogs in which at least one parameter differs from the default setting (*Diff to default*). To display the complete menu tree again, click the  button.



Collapses the menu tree. The menu tree then displays only the menu items of the first level.



Expands the menu tree. The menu tree then displays every menu item on every level.

Menu tree

The menu tree contains one item for each dialog in the Graphical User Interface. When you click a menu item, the dialog area displays the corresponding dialog. You can change the view of the menu tree by clicking the buttons in the icons bar at the top. Furthermore, you can change the view of the menu tree by clicking the following buttons:



Expands the current menu item to display the menu items of the next lower level. The menu tree displays the button next to each collapsed menu item that contains menu items on the next lower level.



Collapses the menu item to hide the menu items of the lower levels. The menu tree displays the button next to each expanded menu item.

Dialog area

The dialog area displays the dialog that you select in the menu tree, including its controls. Here, you can monitor and change the settings of the device depending on your access role.

Below you find useful information on how to use the dialogs.

- [Control elements](#)
- [Modification mark](#)
- [Standard buttons](#)
- [Saving the settings](#)
- [Updating the display](#)
- [Working with tables](#)

Control elements

The dialogs contain different control elements. These control elements are read-only or editable, depending on the parameter and your access role as a user.

The control elements have the following visual properties:

- Input fields
 - An editable input field has a line at the bottom.
 - A read-only input field has no special visual properties.
- Checkboxes
 - An editable checkbox has a bright color.
 - A read-only checkbox has a grey color.
- Radio buttons
 - An editable radio button has a bright color.
 - A read-only radio button has a grey color.

Modification mark

When you modify a value, the corresponding field or table cell displays a red triangle in its top-left corner. The red triangle indicates that you have not yet applied this modification. The modified settings are not yet effective.

Standard buttons

Here you find the description of the standard buttons. The special dialog-specific buttons are described in the corresponding dialog help text.



Applies the settings you modified to the device.

Information on how the device retains the modified settings even after a reboot you find in section [“Saving the settings” on page 16](#).



Undoes the unsaved changes in the current dialog. Resets the values in the fields to the settings applied to the device.

Saving the settings

When applying settings, the device temporarily stores the modified settings. To do this, perform the following step:


Click the  button.

Note: Unintentional changes to the settings can terminate the connection between your PC and the device. To keep the device accessible, enable the *Undo configuration modifications* function in the *Basic Settings > Load/Save* dialog, before changing any settings. Using the function, the device continuously checks if it can still be reached from the IP address of your PC. If the connection is lost, then the device loads the configuration profile saved in the non-volatile memory (NVM) after the specified time. Afterwards, the device can be accessed again.

To keep the modified settings even after restarting the device, perform the following steps:

Open the *Basic Settings > Load/Save* dialog.


In the table, mark the checkbox far left in the table row of the desired configuration profile.

When the checkbox in the *Selected* column is unmarked, click the  button and then the *Select* item.

Click the  button to save your current changes.

Updating the display

If a dialog remains open for a longer time, then the values in the device have possibly changed in the meantime.

To update the display in the dialog, click the  button. Unsaved information in the dialog is lost.

Working with tables

The dialogs display numerous settings in table form. You have the option of customizing the appearance of the tables to fit your needs.

You can find useful information on how to use the tables in the following sections:

- [Filtering table rows](#)
- [Sorting table rows](#)
- [Selecting multiple table rows](#)

Filtering table rows

The filter lets you reduce the number of displayed table rows.



Displays a second table row in the table header containing a text field for every column. When you enter a string in a field, the table displays only the table rows that contain this string in the corresponding column.

Sorting table rows

You can change the order of the table rows. When you click the table header, an icon displays the sorting status.



Displays that the table rows are sorted by a criterion other than the values in this column.

Click the icon to sort the table rows in descending order based on the entries of the corresponding column. You might be able to restore the initial sorting in the table only after logging out and logging in again.



Displays that the table rows are sorted in descending order based on the entries of the corresponding column.

Click the icon to sort the table rows in ascending order based on the entries of the corresponding column. You might be able to restore the initial sorting in the table only after logging out and logging in again.



Displays that the table rows are sorted in ascending order based on the entries of the corresponding column.

Click the icon to sort the table rows in descending order based on the entries of the corresponding column. You might be able to restore the initial sorting in the table only after logging out and logging in again.

Selecting multiple table rows

You have the option of selecting multiple table rows at once and then apply an action to the selected table rows.

To select individual table rows, mark the leftmost checkbox in the desired table row.

To select every table row, mark the leftmost checkbox in the table header.

Once you have selected multiple table rows, you can apply an action to each of these table rows at the same time, for example:

- Entering or changing the values in one table column
- Removing multiple table rows

1 Basic Settings

The menu contains the following dialogs:

- System
- Network
- Software
- Load/Save
- External Memory
- Port
- Restart

1.1 System

[Basic Settings > System]

This dialog displays information about the operating status of the device.

Device status

Device status

Displays the device status and the alarms that currently exist. When at least one alarm is present, the background color changes to red. Otherwise, the background color remains green.

You specify the parameters that the device monitors in the [Diagnostics > Status Configuration > Device Status](#) dialog. If a monitored parameter differs from the desired status, then the device triggers an alarm.

A tooltip displays the cause of the currently existing alarms and the time at which the device triggered each alarm. To display the tooltip, hover the mouse pointer over or tap the field. In the [Diagnostics > Status Configuration > Device Status](#) dialog, the [Status](#) tab displays an overview of the alarms.

Note: If you connect only one power supply unit to a device that supports 2 redundant power supply units, then the device triggers an alarm. To avoid this alarm, deactivate the monitoring of the missing power supply units in the [Diagnostics > Status Configuration > Device Status](#) dialog.

Security status



Security status

Displays the security status and the alarms that currently exist. When at least one alarm is present, the background color changes to red. Otherwise, the background color remains green.

You specify the parameters that the device monitors in the [Diagnostics > Status Configuration > Security Status](#) dialog. If a monitored parameter differs from the desired status, then the device triggers an alarm.

A tooltip displays the cause of the currently existing alarms and the time at which the device triggered each alarm. To display the tooltip, hover the mouse pointer over or tap the field. In the [Diagnostics > Status Configuration > Security Status](#) dialog, the [Status](#) tab displays an overview of the alarms.

System data

The fields in this frame display operating data and information on the location of the device.

System name

Specifies the name by which the device is known in the network.

Possible values:

Alphanumeric ASCII character string with 0..255 characters

The device accepts the following characters:

- 0 . 9
 - a . z
 - A . Z
 - ! # \$ % & ' () * + , - . / : ; < = > ? @ [\] ^ _ ` { } ~
- <device type name>-<MAC address> (default setting)

When generating a digital certificate, the application generating the certificate uses the specified value as the domain name and common name.

The following functions use the specified value as a hostname or Fully Qualified Domain Name (FQDN). For compatibility reasons, it is recommended to use only lowercase letters, as some systems differentiate uppercase from lowercase in the FQDN. Verify that this name is unique in the entire network.

- [Syslog](#)

Location

Specifies the current or planned location.

Possible values:

Alphanumeric ASCII character string with 0..255 characters

Contact person

Specifies the contact person for this device.

Possible values:

Alphanumeric ASCII character string with 0..255 characters

Device type

Displays the product name of the device.

Power supply 1

Power supply 2

Displays the status of the power supply unit at the respective voltage supply connector.

Possible values:

[present](#)

[defective](#)

[not installed](#)

[unknown](#)

Uptime

Displays the time that has elapsed since the device was last restarted.

Possible values:

Time in the format [day\(s\)](#), [... h](#) [... m](#) [... s](#)

Temperature [°C]

Displays the current temperature in the device in °C.

You activate the monitoring of the temperature threshold values in the [Diagnostics > Status Configuration > Device Status](#) dialog.

Upper temp. limit [°C]

Specifies the upper temperature threshold value in °C.

Possible values:

-99 . 99 (integer)

If the temperature in the device exceeds the specified value, then the device displays an alarm.

Lower temp. limit [°C]

Specifies the lower temperature threshold value in °C.

Possible values:

-99 . 99 (integer)

If the temperature in the device falls below the specified value, then the device displays an alarm.

LED status

For further information about the device status LEDs, see the “Installation” user manual.

Status



There is currently no device status alarm. The device status is OK.



There is currently at least one device status alarm. For details, see the [Device status](#) frame.

Power



Device that supports one power supply unit: The supply voltage is active.

Device that supports 2 redundant power supply units: Both supply voltages are active.

Device that supports 2 redundant power supply units: Only one supply voltage is active.

ACA



No external memory is connected.

or


The external memory is connected but not ready for operation.



The external memory is connected and ready for operation.

Port status

This frame displays a simplified view of the device ports at the time of the last display update. You identify the port status from the indicator.

In the initial view, the frame only displays ports with an active link. When you click the  button, the frame displays every port.

- The port speed is displayed next to the port number.
- When you hover the mouse pointer over or tap the appropriate port icon, a tooltip displays detailed port state information.

Green background color

Port with an active link.

Gray background color

Port with an inactive link.

1.2 Network

[Basic Settings > Network]

The menu contains the following dialogs:

Global
IPv4

1.21 Global

[Basic Settings > Network > Global]

This dialog lets you specify the VLAN and HiDiscovery settings required for the access to the device management through the network.

Management interface

This frame lets you specify the VLAN in which the device management can be accessed.


VLAN ID

Specifies the VLAN in which the device management is accessible through the network. The device management is accessible through ports that are members of this VLAN.

Possible values:

1..4042 (default setting: 1)

The prerequisite is that in the [Switching > VLAN > Configuration](#) dialog the VLAN is already set up. Assign a VLAN that is not assigned to any router interface.

When you click the  button after changing the value, the [Information](#) window opens. Select the port, over which you connect to the device in the future. After clicking the [Ok](#) button, the new device management VLAN settings are assigned to the port.

- After that the port is a member of the VLAN and transmits the data packets without a VLAN tag (untagged). See the [Switching > VLAN > Configuration](#) dialog.
- The device assigns the port VLAN ID of the device management VLAN to the port. See the [Switching > VLAN > Port](#) dialog.

After a short time the device is reachable over the new port in the new device management VLAN.

MAC address

Displays the MAC address of the device. The device management is accessible through the network using the MAC address.

HiDiscovery protocol v1/v2

This frame lets you specify settings for the access to the device using the HiDiscovery protocol.

On a PC, the HiDiscovery software displays the Hirschmann devices that can be accessed in the network on which the HiDiscovery function is enabled. You can access these devices even if they have invalid or no IP parameters assigned. The HiDiscovery software lets you assign or change the IP parameters in the device.

Note: With the HiDiscovery software you access the device only through ports that are members of the same VLAN as the device management. You specify which VLAN a certain port is assigned to in the [Switching > VLAN > Configuration](#) dialog.

Operation

Enables/disables the HiDiscovery function in the device.

Possible values:

On (default setting)

The HiDiscovery function is enabled.

You can use the HiDiscovery software to access the device from your PC.

Off

The HiDiscovery function is disabled.

Access

Enables/disables the write access to the device using for the HiDiscovery function.

Possible values:

readWrite (default setting)

The HiDiscovery function has write access to the device. The device lets you change the IP parameters in the device using the HiDiscovery function.

readOnly

The HiDiscovery function has read-only access to the device. The device lets you view the IP parameters in the device using the HiDiscovery function.

Recommendation: Change the setting to the value **readOnly** only after putting the device into operation.

1.22 IPv4

[Basic Settings > Network > IPv4]

This dialog allows you to specify the IPv4 settings required for the access to the device management through the network.

IP parameter

This frame lets you assign the IP parameters manually. If you have selected the [Local](#) radio button in the [Management interface](#) frame, [IP address assignment](#) option list, then these fields can be edited.

IP address

Specifies the IP address under which the device management can be accessed through the network.

Possible values:

Valid IPv4 address

Verify that the IP subnet of the device management does not overlap with any subnet connected to another interface of the device:

- router interface
- loopback interface

Netmask

Specifies the netmask.

Possible values:

Valid IPv4 netmask

Gateway address

Specifies the IP address of a router through which the device accesses other devices outside of its own network.

Possible values:

Valid IPv4 address

If the device does not use the specified gateway, then verify that another *default gateway* is specified. The setting in the following dialog has precedence:

- [Routing > Routing Table](#) dialog, *Next hop IP address* column, if the value in the *Network address* column and in the *Netmask* column is 0. 0. 0. 0

1.3 Software

[Basic Settings > Software]

This dialog lets you update the device software and display information about the device software.

You also have the option to restore a backup of the device software that is saved in the device.

Note: Before you update the device software, follow the version-specific notes in the [Readme](#) text file.

Version

Stored version

Displays the version number and creation date of the device software stored in the flash memory. The device loads the device software during the next system startup.

Running version

Displays the version number and creation date of the currently running device software that the device loaded during the last system startup.

Backup version

Displays the version number and creation date of the device software saved as a backup in the flash memory. The device copied this device software into the backup memory during the last software update or after you clicked the [Restore](#) button.

Restore

The device swaps the device software images and accordingly the values displayed in the fields *Stored version* and *Backup version*.

During the next system startup, the device loads the device software displayed in the *Stored version* field.

Bootcode

Displays the version number and creation date of the boot code.

Software update


The device lets you update the device software at this place, if a suitable device software image is available outside the device. If a suitable device software image is saved on the selected external memory, use the table in the *File system* tab below.

URL

Specifies the path and the file name of the device software image that you use to update the device software.

The device gives you the following options for updating the device software:

- Software update from the PC

Drag and drop the file into the  area from your PC or network drive. As an alternative, click in the area to select the file.

You can also use SFTP or SCP to transfer the file from your PC to the device. Perform the following steps:

On your PC, open an SFTP or SCP client, for example WinSCP.

Use the SFTP or SCP client to open a connection to the device.

Transfer the file onto the device, into the directory */upload/firmware*.

When the file transfer is complete, the device starts updating the device software. If the update was successful, then the device generates an *ok* file in the directory */upload/firmware* and deletes the transferred file.

The device loads the device software during the next system startup.

- Software update from an SCP or SFTP server

If the file is on an SCP or SFTP server, then specify the URL in one of the following forms:

scp: // or sftp: //<IP address>/<path>/<file name>

Click the *Start* button to open the *Credentials* window. In this window, you enter the *User name* and *Password* to log into the server.

scp: // or sftp: //<user>:<password>@<IP address>/<path>/<file name>

Start

Updates the device software.

- To remain logged in to the device management during the software update, move the mouse pointer occasionally. As an alternative, before you start the software update, specify a sufficiently high value in the *Device Security > Management Access > Web* dialog, *Web interface session timeout [min]* field.
- The device transfers the previously used device software to the backup memory.
- The device transfers the selected file to the flash memory, replacing the previously used device software. During the next startup, the device boots with the device software that you have transferred.

[File system]

Table

For information on how to customize the appearance of the table, see “Working with tables” on page 16.

Buttons

Update Firmware

Updates the device software if a suitable device software image is saved on the selected external memory. The prerequisite is that a table row is selected for which the *File location* column displays the value *sd-card* or *usb*.

- Verify that the relevant external memory is selected from the *Selected external memory* drop-down list. See the *Basic Settings > Load/Save* dialog, *External memory* frame.
- To remain logged in to the device management during the software update, move the mouse pointer occasionally. As an alternative, before you start the software update, specify a sufficiently high value in the *Device Security > Management Access > Web* dialog, *Web interface session timeout [min]* field.
- The device transfers the previously used device software to the backup memory.
- The device transfers the selected file to the flash memory, replacing the previously used device software. During the next startup, the device boots with the device software that you have transferred.

File location

Displays the storage location of the device software.

Possible values:

- ram*
Volatile memory of the device
- flash*
Non-volatile memory (NVM) of the device
- sd-card*
External SD memory (ACA31)
- usb*
External USB memory (ACA21/ACA22)

Index

Displays the index of the device software.

The index number of the device software in the flash memory has the following meaning:

- 1*
During the next system startup, the device loads this device software.
- 2*
The device copied this device software into the backup area during the last software update.

File name

Displays the device-internal file name of the device software.

Firmware

Displays the version number and creation date of the device software.

1.4 Load/Save

[Basic Settings > Load/Save]

This dialog lets you save the device settings permanently in a configuration profile.

The device can hold several configuration profiles. When you activate an alternative configuration profile, you change to other device settings. You have the option of exporting the configuration profiles to your PC or to a server. You also have the option of importing the configuration profiles from your PC or from a server to the device.

In the default setting, the device saves the configuration profiles unencrypted. If you enter a password in the *Configuration encryption* frame, then the device saves both the current and the future configuration profiles in an encrypted format.

Unintentional changes to the settings can terminate the connection between your PC and the device. To keep the device accessible, enable the *Undo configuration modifications* function before changing any settings. If the connection is lost, then the device loads the configuration profile saved in the non-volatile memory (NVM) after the specified time.

Table

For information on how to customize the appearance of the table, see [“Working with tables” on page 16](#).

Buttons



Remove

Removes the configuration profile selected in the table from the non-volatile memory (NVM) or from the external memory.

If the configuration profile is designated as "Selected", then the device helps prevent you from removing the configuration profile.



Save

Saves the temporarily applied settings in the configuration profile designated as "Selected" in the non-volatile memory (NVM).

When in the *Basic Settings > External Memory* dialog the checkbox in the *Backup config when saving* column is marked, then the device saves a copy of the configuration profile in the external memory.



Displays a context menu with further functions for the corresponding dialog.

Save as..

Opens the [Save as..](#) window to copy the configuration profile selected in the table and saves it with a user-specified name in the non-volatile memory (NVM).

In the [Profile name](#) field, enter the name under which you want to save the configuration profile.

To save the configuration profile under a new name, click the **+** button.

To overwrite an existing configuration profile, select the corresponding item from the drop-down list.

If in the [Basic Settings > External Memory](#) dialog the checkbox in the [Backup config when saving](#) column is marked, then the device designates the configuration profile of the same name in the external memory as “Selected”.

Note: Before adding additional configuration profiles, decide for or against permanently activated configuration encryption in the device. Save additional configuration profiles either unencrypted or encrypted with the same password.

Activate

Loads the settings of the configuration profile selected in the table to the volatile memory (RAM).

- The device terminates the connection to the Graphical User Interface. To access the device management again, perform the following steps:
 - Reload the Graphical User Interface.
 - Log in again.
- The device immediately uses the settings of the configuration profile on the fly.

Enable the [Undo configuration modifications](#) function before you activate another configuration profile. If the connection is lost afterwards, then the device loads the last configuration profile designated as “Selected” from the non-volatile memory (NVM). The device can then be accessed again.

If the configuration encryption is inactive, then the device loads an unencrypted configuration profile. If the configuration encryption is active and the password matches the password stored in the device, then the device loads an encrypted configuration profile.

When you activate an older configuration profile, the device takes over the settings of the functions contained in this software version. The device sets the values of new functions to their default value.

Select

Designates the configuration profile selected in the table as “Selected”. In the [Selected](#) column, the checkbox is then [marked](#).

When applying the [Undo configuration modifications](#) function or during the system startup, the device loads the settings of this configuration profile to the volatile memory (RAM).

- If the configuration encryption in the device is disabled, then designate an unencrypted configuration profile only as “Selected”.
- If the configuration encryption in the device is enabled and the password of the configuration profile matches the password saved in the device, then designate an encrypted configuration profile only as “Selected”.

Otherwise, the device is unable to load and encrypt the settings in the configuration profile the next time it restarts. For this case you specify in the [Diagnostics > System > Selftest](#) dialog if the device starts with the default settings or terminates the restart and stops.

Note: You only mark the configuration profiles saved in the non-volatile memory (NVM).

If in the *Basic Settings > External Memory* dialog the checkbox in the *Backup config when saving* column is marked, then the device designates the configuration profile of the same name in the external memory as “Selected”.

Import...

Opens the *Import...* window to import a configuration profile.

The prerequisite is that you have exported the configuration profile using the *Export...* button or using the link in the *Profile name* column.

From the *Select source* drop-down list, select from where the device imports the configuration profile.

PC/URL


The device imports the configuration profile from the local PC or from a remote server.

External memory

The device imports the configuration profile from the selected external memory. See the *External memory* frame.

When **PC/URL** is selected above, in the *Import profile from PC/URL* frame you specify the configuration profile file to be imported.

- Import from the PC

If the file is on your PC or on a network drive, then drag and drop the file into the  area.

As an alternative, click in the area to select the file.

You can also use SFTP or SCP to transfer the file from your PC to the device. Perform the following steps:

On your PC, open an SFTP or SCP client, for example WinSCP.

Use the SFTP or SCP client to open a connection to the device.

Transfer the file onto the device, into the directory */nv/cfg*.

- Import from an SCP or SFTP server

If the file is on an SCP or SFTP server, then specify the URL in one of the following forms:

scp: // or sftp: //<IP address>/<path>/<file name>

Click the *Start* button to open the *Credentials* window. In this window, you enter the *User name* and *Password* to log into the server.

scp: // or sftp: //<user>:<password>@<IP address>/<path>/<file name>

When **External memory** is selected above, in the *Import profile from external memory* frame you specify the configuration profile file to be imported.

From the *Profile name* drop-down list, select the name of the configuration profile to be imported.

In the *Destination* frame you specify where the device saves the imported configuration profile.

In the *Profile name* field you specify the name under which the device saves the configuration profile.

In the *Storage* field you specify the storage location for the configuration profile. The prerequisite is that from the *Select source* drop-down list the **PC/URL** item is selected.

RAM

The device saves the configuration profile in the volatile memory (**RAM**) of the device. This replaces the *running-config*, the device uses the settings of the imported configuration profile immediately. The device terminates the connection to the Graphical User Interface. Reload the Graphical User Interface. Log in again.

NVM

The device saves the configuration profile in the non-volatile memory (**NVM**) of the device.

When you import a configuration profile, the device takes over the settings as follows:

- If the configuration profile was exported on the same device or on an identically equipped device of the same type, then:
The device takes over the settings completely.
- If the configuration profile was exported on another device, then:
The device takes over the settings which it can interpret based on its hardware equipment and software level.
The remaining settings the device takes over from its `running-config` configuration profile.

Regarding configuration profile encryption, also read the help text of the [Configuration encryption](#) frame. The device imports a configuration profile under the following conditions:

- The configuration encryption of the device is inactive. The configuration profile is unencrypted.
- The configuration encryption of the device is active. The configuration profile is encrypted with the same password that the device currently uses.

Export...

Exports the configuration profile selected in the table and saves it as an XML file on a remote server.

To save the file on your PC, click the link in the [Profile name](#) column to select the storage location and specify the file name.

The device gives you the following options for exporting a configuration profile:

- Export to an SCP or SFTP server
To save the file on an SCP or SFTP server, specify the URL for the file in one of the following forms:
scp:// or sftp://<IP address>/<path>/<file name>
Click the [Ok](#) button to open the [Credentials](#) window. In this window, you enter the [User name](#) and [Password](#) to log into the server.
scp:// or sftp://<user>:<password>@<IP address>/<path>/<file name>

Back to factory...

Resets the settings in the device to the default values.

- The device deletes the saved configuration profiles from the volatile memory ([RAM](#)) and from the non-volatile memory ([NVM](#)).
- The device deletes the digital certificate used by the web server in the device.
- The device deletes the RSA key (Host Key) used by the SSH server in the device.
- When an external memory is connected, the device deletes the configuration profiles saved in the external memory.
- After a short time, the device reboots and then uses the default settings.

Back to default

Deletes the current operating (`running-config`) settings from the volatile memory ([RAM](#)).

Storage

Displays the storage location of the configuration profile.

Possible values:

[RAM](#) (volatile memory of the device)

In the volatile memory, the device stores the settings for the current operation.


NVM (non-volatile memory of the device)

When applying the [Undo configuration modifications](#) function or during the system startup, the device loads the “Selected” configuration profile from the non-volatile memory.

The non-volatile memory provides space for multiple configuration profiles, depending on the number of settings saved in the configuration profile. The device manages a maximum of 20 configuration profiles in the non-volatile memory.

You can load a configuration profile into the volatile memory (**RAM**). To do this, perform the following steps:

Select the table row of the configuration profile.

Click the  button and then the [Activate](#) item.

ENMM (external memory)

In the external memory, the device saves a backup copy of the “Selected” configuration profile. The prerequisite is that in the [Basic Settings > External Memory](#) dialog the [Backup config when saving](#) checkbox is marked.

Profile name

Displays the name of the configuration profile.

Possible values:


[running-config](#)

Name of the configuration profile in the volatile memory (**RAM**).


[config](#)

Name of the factory setting configuration profile in the non-volatile memory (**NVM**).

User-defined name

The device lets you save a configuration profile with a user-specified name. To do this, select the table row of an existing configuration profile in the table, click the  button and then the [Save as..](#) item.

To export the configuration profile as an XML file on your PC, click the link. Then you select the storage location and specify the file name.


To save the file on a remote server, click the  button and then the [Export...](#) item.

Last modified (UTC)

Displays the Universal Time Coordinated (UTC) time a user last saved the configuration profile.

Selected


Displays if the configuration profile is designated as “Selected”.

The device lets you designate another configuration profile as “Selected”. To do this, select the desired configuration profile in the table, click the  button and then the [Activate](#) item.

Possible values:

marked

The configuration profile is designated as “Selected”.

- When applying the [Undo configuration modifications](#) function or during the system startup, the device loads the configuration profile into the volatile memory ([RAM](#)).
- When you click the  button, the device saves the temporarily applied settings in this configuration profile.

unmarked

Another configuration profile is designated as “Selected”.

Encryption

Displays if the configuration profile is encrypted.

Possible values:

marked

The configuration profile is encrypted.

unmarked

The configuration profile is unencrypted.

You activate/deactivate the encryption of the configuration profile in the [Configuration encryption](#) frame.

Verified

Displays if the password of the encrypted configuration profile matches the password stored in the device.

Possible values:

marked

The passwords match. The device is able to unencrypt the configuration profile.

unmarked

The passwords are different. The device is unable to unencrypt the configuration profile.

Note: The device applies script files additionally to the current settings. Verify that the script file does not contain any parts that conflict with the current settings.

Software version

Displays the version number of the device software that the device ran while saving the configuration profile.

Fingerprint

Displays the checksum saved in the configuration profile.

When saving the settings, the device calculates the checksum and inserts it into the configuration profile.

Verified

Displays if the checksum saved in the configuration profile is valid.

The device calculates the checksum of the configuration profile marked as “Selected” and compares it with the checksum saved in this configuration profile.

Possible values:

[marked](#)

The calculated and the saved checksum match.

The saved settings are consistent.

[unmarked](#)

For the configuration profile marked as “Selected” applies:

The calculated and the saved checksum are different.

The configuration profile contains modified settings.

Possible causes:

- The file is damaged.
- The file system in the external memory is inconsistent.
- A user has exported the configuration profile and changed the XML file outside the device.

For the other configuration profiles the device has not calculated the checksum.

The device verifies the checksum correctly only if the configuration profile has been saved before as follows:

- on an identical device
- with the same software version, which the device is running

Note: This function identifies changes to the settings in the configuration profile. The function does not provide protection against operating the device with modified settings.

External memory

Selected external memory

Specifies the external memory that the device uses for file operations.

This setting has the following effects:

- For example, the device stores a copy of the device configuration files on the selected external memory.
- The device lets you conveniently update the device software if a suitable device software image is saved on the selected external memory. See the [Basic Settings > Software](#) dialog.

Possible values:

[sd](#)

External SD memory (ACA31)

[usb](#)

External USB memory (ACA21/ACA22)

Status

Displays the operating state of the selected external memory.

Possible values:

[not Present](#)

No external memory is connected.

[removed](#)

Someone has removed the external memory from the device during operation.

[ok](#)

The external memory is connected and ready for operation.

[outOfMemory](#)

The memory space is occupied in the external memory.

[generalError](#)

The device has detected an error.

Configuration encryption

Active

Displays if the configuration encryption is active/inactive in the device.

Possible values:

[marked](#)

The configuration encryption is active.

If the configuration profile is encrypted and the password matches the password stored in the device, then the device loads a configuration profile from the non-volatile memory (NVM).

[unmarked](#)

The configuration encryption is inactive.

If the configuration profile is unencrypted, then the device loads a configuration profile from the non-volatile memory (NVM) only.

If in the [Basic Settings > External Memory](#) dialog, the [Config priority](#) column has the value [first](#) or [second](#) and the configuration profile is unencrypted, then the [Security status](#) frame in the [Basic Settings > System](#) dialog displays an alarm.

In the [Diagnostics > Status Configuration > Security Status](#) dialog, [Global](#) tab, [Monitor](#) column you specify if the device monitors the [Load unencrypted config from external memory](#) parameter.

Set password

Opens the [Set password](#) window that helps you to enter the password needed for the configuration profile encryption. Encrypting the configuration profiles makes unauthorized access more difficult. To do this, perform the following steps:

When you are changing an existing password, enter the existing password in the [Old password](#) field. To display the password in plain text instead of ***** (asterisks), mark the [Display content](#) checkbox.

In the [New password](#) field, enter the password.

To display the password in plain text instead of ***** (asterisks), mark the [Display content](#) checkbox.

Mark the [Save configuration afterwards](#) checkbox to use encryption also for the "Selected" configuration profile in the non-volatile memory (NVM) and in the external memory.

Note: If a maximum of one configuration profile is stored in the non-volatile memory (NVM) of the device, then use this function only. Before adding additional configuration profiles, decide for or against permanently activated configuration encryption in the device. Save additional configuration profiles either unencrypted or encrypted with the same password.

If you are replacing a device with an encrypted configuration profile, for example due to an inoperable device, then perform the following steps:

Restart the new device and assign the IP parameters.

Open the [Basic Settings > Load/Save](#) dialog on the new device.

Encrypt the configuration profile in the new device. See above. Enter the same password you used in the inoperable device.

Install the external memory from the inoperable device in the new device.

Restart the new device.

During the next system startup, the device loads the configuration profile with the settings of the inoperable device from the external memory. The device copies the settings into the volatile memory (RAM) and into the non-volatile memory (NVM).

Note: The prerequisite for loading a configuration profile from the external memory is that in the [Basic Settings > External Memory](#) dialog the *Config priority* column displays the value *first* or *second*. This value is set as the default setting.

Delete

Opens the [Delete](#) window which helps you to cancel the configuration encryption in the device. To cancel the configuration encryption, perform the following steps:

In the *Old password* field, enter the existing password.

To display the password in plain text instead of ***** (asterisks), mark the [Display content](#) checkbox.

Mark the [Save configuration afterwards](#) checkbox to remove the encryption also for the "Selected" configuration profile in the non-volatile memory (NVM) and in the external memory.

Note: If you keep additional encrypted configuration profiles in the memory, then the device helps prevent you from activating or designating these configuration profiles as "Selected".

Undo configuration modifications

Operation

Enables/disables the [Undo configuration modifications](#) function. Using the function, the device continuously checks if it can still be reached from the IP address of your PC. If the connection is lost, after a specified time period the device loads the "Selected" configuration profile from the non-volatile memory (NVM). Afterwards, the device can be accessed again.

Possible values:

On

The function is enabled.

- You specify the time period between the interruption of the connection and the loading of the configuration profile in the [Timeout \[s\] to recover after connection loss](#) field.
- When the non-volatile memory (NVM) contains multiple configuration profiles, the device loads the configuration profile designated as "Selected".

Off (default setting)

The function is disabled.

Disable the function again before you close the Graphical User Interface. You thus help prevent the device from restoring the configuration profile designated as "Selected".

Note: Before you enable the function, save the settings in the configuration profile. The device thus maintains the current settings, that are only temporarily saved.

Timeout [s] to recover after connection loss

Specifies the time in seconds after which the device loads the "Selected" configuration profile from the non-volatile memory (NVM) if the connection is lost.

Possible values:

30 . 600 (default setting: 600)

Specify a sufficiently large value. Take into account the time when you are viewing the dialogs of the Graphical User Interface without changing or updating them.

Watchdog IP address

Displays the IP address of the PC on which you have enabled the function.

Possible values:

IPv4 address (default setting: 0.0.0.0)

Information

NVM in sync with running config


Displays if the settings in the volatile memory (RAM) differ from the settings of the "Selected" configuration profile in the non-volatile memory (NVM).

Possible values:

marked

The settings match.

unmarked

The settings differ. Additionally, the Banner displays the icon .

External memory in sync with NVM

Displays if the settings of the "Selected" configuration profile in the external memory (ACA) differ from the settings of the "Selected" configuration profile in the non-volatile memory (NVM).

Possible values:

marked

The settings match.

unmarked

The settings differ.

Possible causes:

- No external memory is connected to the device.
- In the *Basic Settings > External Memory* dialog, the *Backup config when saving* function is disabled.

1.5 External Memory

[Basic Settings > External Memory]

This dialog lets you activate functions that the device automatically executes in combination with the external memory. The dialog also displays the operating state and identifying characteristics of the external memory.

Table

For information on how to customize the appearance of the table, see [“Working with tables” on page 16](#).

Type

Displays the type of the external memory.

Possible values:

`sd`

External SD memory (ACA31)

`usb`

External USB memory (ACA21/ACA22)

Status

Displays the operating state of the external memory.

Possible values:

`notPresent`

No external memory is connected.

`removed`

Someone has removed the external memory from the device during operation.

`ok`

The external memory is connected and ready for operation.

`outOfMemory`

The memory space is occupied in the external memory.

`genericErr`

The device has detected an error.

Writable

Displays if the device has write access to the external memory.

Possible values:

`marked`

The device has write access to the external memory.

`unmarked`

The device has read-only access to the external memory. Possibly the write protection is activated in the external memory.

Software auto update

Activates/deactivates the automatic device software update during the system startup.

Possible values:

marked (default setting)

The device updates the device software when the following files are located in the external memory:

- the device software image file
- a text file `startup.txt` with the content `autoUpdate=<software_image_file_name>.bin`

unmarked

No automatic device software update during the system startup.

Config priority

Specifies the memory from which the device loads the configuration profile upon reboot.

Possible values:

disable

The device loads the configuration profile from the non-volatile memory (NVM).

first, second

The device loads the configuration profile from the external memory designated as **first**. When the device does not find a configuration profile there, it loads the configuration profile from the external memory designated as **second**, and so on.

When the device does not find a configuration profile in the external memory, it loads the configuration profile from the non-volatile memory (NVM).

Note: When loading the configuration profile from the external memory (EMM), the device overwrites the settings of the “Selected” configuration profile in the non-volatile memory (NVM).

If the *Config priority* column has the value **first** or **second** and the configuration profile is unencrypted, then the *Security status* frame in the *Basic Settings > System* dialog displays an alarm.


In the *Diagnostics > Status Configuration > Security Status* dialog, *Global* tab, *Monitor* column you specify if the device monitors the *Load unencrypted config from external memory* parameter.

Backup config when saving

Activates/deactivates saving a copy of the configuration profile in the external memory.

Possible values:

marked (default setting)

Saving a copy is activated. When you click in the *Basic Settings > Load/Save* dialog the  button, the device saves a copy of the configuration profile on the active external memory.

unmarked

Saving a copy is deactivated. The device does not save a copy of the configuration profile.

Manufacturer ID

Displays the name of the memory manufacturer.

Revision

Displays the revision number specified by the memory manufacturer.

Version

Displays the version number specified by the memory manufacturer.

Name

Displays the product name specified by the memory manufacturer.

Serial number

Displays the serial number specified by the memory manufacturer.

1.6 Port

[Basic Settings > Port]

This dialog lets you specify settings for the individual ports. The dialog also displays the operating mode, connection status, bit rate and duplex mode for every port.

The dialog contains the following tabs:

[\[Configuration\]](#)

[\[Statistics\]](#)

[Configuration]

Table

For information on how to customize the appearance of the table, see [“Working with tables” on page 16](#).

Port

Displays the port number.

Name

Name of the port.

Possible values:

Alphanumeric ASCII character string with 0..64 characters

The device accepts the following characters:

- <space>
- 0 . 9
- a . z
- A . Z
- ! # \$ % & ' () * + , - . / : ; < = > ? @ [\] ^ _ ` { } ~

Port on

Activates/deactivates the port.

Possible values:

marked (default setting)

The port is active.

unmarked

The port is inactive. The port does not send or receive any data.

State

Displays if the port is currently physically enabled or disabled.

Possible values:

[marked](#)

The port is physically enabled.

[unmarked](#)

The port is physically disabled.

Autoneg

Activates/deactivates the automatic selection of the operating mode for the port.

Possible values:

[marked](#) (default setting)

The automatic selection of the operating mode is active.

The port negotiates the operating mode independently using auto-negotiation and automatically detects the assignment of the twisted-pair port connectors (auto cable crossing). This setting has priority over the manual setting of the port.

Elapse several seconds until the port has set the operating mode.

[unmarked](#)

The automatic selection of the operating mode is inactive.

The port operates with the values you specify in the [Manual configuration](#) column and in the [Manual cable crossing](#) column.

Grayed-out display

No automatic selection of the operating mode.

Manual configuration

Specifies the operating mode of the ports when the [Autoneg](#) function is disabled.

Possible values:

[10M HDX](#)

Half-duplex connection

[10M FDX](#)

Full-duplex connection

[100M HDX](#)

Half-duplex connection

[100M FDX](#)

Full-duplex connection

[1G FDX](#)

Full-duplex connection

Note: The operating modes of the port actually available depend on the device hardware.

Link/Current settings

Displays the operating mode which the port currently uses.

Possible values:

-

No cable connected, no link.

[10M HDX](#)

Half-duplex connection

10M FDX
Full-duplex connection
100M HDX
Half-duplex connection
100M FDX
Full-duplex connection
1G FDX
Full-duplex connection

Note: The operating modes of the port actually available depend on the device hardware.

Manual cable crossing

Specifies the devices connected to a twisted-pair port.

The prerequisite is that the *Autoneg* function is disabled.

Possible values:

mdi

The device interchanges the send- and receive-line pairs on the port.

mdi x (default setting on twisted-pair ports)

The device helps prevent the interchange of the send- and receive-line pairs on the port.

auto-mdi x

The device detects the send and receive line pairs of the connected device and automatically adapts to them.

Example: When you connect an end device with a crossed cable, the device automatically resets the port from *mdi x* to *mdi*.

unsupported (default setting on optical ports or twisted-pair SFP ports)

The port does not support this function.

Flow control

Activates/deactivates the flow control on the port.

Possible values:

marked (default setting)

The Flow control on the port is active.

The sending and evaluating of pause packets (full-duplex operation) or collisions (half-duplex operation) is activated on the port.

To enable the flow control in the device, also activate the *Flow control* function in the *Switching > Global* dialog.

Activate the flow control also on the port of the device that is connected to this port.

On an uplink port, activating the flow control can possibly cause undesired sending interruptions in the higher-level network segment (“wandering backpressure”).

unmarked

The Flow control on the port is inactive.

If you are using a redundancy function, then you deactivate the flow control on the participating ports. If the flow control and the redundancy function are active at the same time, it is possible that the redundancy function operates differently than intended.

Send trap

Activates/deactivates the sending of SNMP traps when the device detects a change in the link up/down status on the port.

Possible values:

marked (default setting)

The sending of SNMP traps is active. The prerequisite is that in the [Diagnostics > Status Configuration > Alarms \(Traps\)](#) dialog the [Alarms \(Traps\)](#) function is enabled and at least one trap destination is specified.

When the device detects a link up/down status change, the device sends an SNMP trap.

unmarked

The sending of SNMP traps is inactive.

Power state

Specifies if the port is physically enabled or disabled after you deactivated the port in the [Port on](#) column.

Possible values:

marked

The device keeps the port physically enabled when the [Port on](#) checkbox is unmarked. A device connected to this port continues to detect the link status as active.

unmarked (default setting)

The port is physically disabled. The physical status of the port is controlled only by the setting in the [Port on](#) column.

Power save

Specifies how the port behaves when no cable is connected.

Possible values:

no-power-save (default setting)

The port remains activated.

auto-power-down

The port changes to the energy-saving mode.

unsupported

The port does not support this function and remains activated.

[Statistics]

This tab displays the following overview per port:

- Number of data packets/bytes received by the device
 - [Received packets](#)
 - [Received octets](#)
 - [Received unicasts](#)
 - [Received multicasts](#)
 - [Received broadcasts](#)
- Number of data packets/bytes sent or forwarded by the device
 - [Transmitted packets](#)
 - [Transmitted octets](#)
 - [Transmitted unicasts](#)
 - [Transmitted multicasts](#)
 - [Transmitted broadcasts](#)

- Number of errors detected by the device
 - [Received fragments](#)
 - [Detected CRC errors](#)
 - [Detected collisions](#)
- Number of data packets per size category received by the device
 - [Packets 64 bytes](#)
 - [Packets 65 to 127 bytes](#)
 - [Packets 128 to 255 bytes](#)
 - [Packets 256 to 511 bytes](#)
 - [Packets 512 to 1023 bytes](#)
 - [Packets 1024 to 1518 bytes](#)
- Number of data packets discarded by the device
 - [Received discards](#)
 - [Transmitted discards](#)

To sort the table by a specific criterion click the header of the corresponding column.

For example, to sort the table based on the number of received bytes in ascending order, click the header of the [Received octets](#) column once. To sort in descending order, click the header again.

To reset the counter for the port statistics in the table to 0, perform the following steps:

In the [Basic Settings > Port](#) dialog, click the  button.

or

In the [Basic Settings > Restart](#) dialog, click the [Clear port statistics](#) button.

1.7 Restart

[Basic Settings > Restart]

This dialog lets you restart the device, reset port counters and the MAC address table (forwarding database), and delete log files.

Restart

Cold start...

Opens the [Restart](#) window to initiate a restart of the device.

If the configuration profile in the volatile memory ([RAM](#)) and the "Selected" configuration profile in the non-volatile memory ([NVM](#)) differ, then the device displays the [Warning](#) window.

To permanently save the settings, click the [Yes](#) button in the [Warning](#) window.

To discard the changed settings, click the [No](#) button in the [Warning](#) window.

The device restarts and goes through the following phases:

- The device starts the device software that the [Stored version](#) field displays in the [Basic Settings > Software](#) dialog.
- The device loads the settings from the "Selected" configuration profile. See the [Basic Settings > Load/Save](#) dialog.

Note: During the restart, the device does not transfer any data. During this time, the device cannot be accessed by the Graphical User Interface or other management systems.

Buttons

Clear FDB

Removes the MAC addresses from the forwarding table that have in the [Switching > Filter for MAC Addresses](#) dialog the value [Learned](#) in the [Status](#) column.

Clear ARP table

Removes the dynamically set up addresses from the ARP table.

See the [Diagnostics > System > ARP](#) dialog.

Clear port statistics

Resets the counter for the port statistics to 0.

See the [Basic Settings > Port](#) dialog, [Statistics](#) tab.

Clear log file

Removes the logged events from the log file.

See the [Diagnostics > Report > System Log](#) dialog.

Clear persistent log file

Removes the log files from the external memory.

See the [Diagnostics > Report > Persistent Logging](#) dialog.

Clear firewall table

Removes the information about open connections from the state table of the firewall. It is possible that the device interrupts open communication connections.

2 Time

The menu contains the following dialogs:

[Basic Settings](#)
[NTP](#)

2.1 Basic Settings

[Time > Basic Settings]

After a restart, the device initializes its clock to January 1 2024, 01:00 UTC+1. Reset the time if you disconnect the device from the power supply or restart it. As an alternative, you specify that the device automatically obtains the correct time from an [SNTP](#) server or from a PTP clock.

In this dialog, you specify time-related settings independently of the time synchronization protocol specified.

The dialog contains the following tabs:

[\[Global\]](#)
[\[Daylight saving time\]](#)

[Global]

In this tab, you specify the system time and the time zone.

Configuration

System time (UTC)

Displays the date and time in Universal Time Coordinated (UTC) format.

Set time from PC

The device takes over the time from your computer as the system time.

System time

Displays the local date and time: $\text{System time} = \text{System time (UTC)} + \text{Local offset [min]} + \text{Daylight saving time}$

Time source

Displays the time source from which the device obtains the time information.

The device automatically selects the available time source with the greatest accuracy.

Possible values:

[local](#)

System clock of the device.

[ntp](#)

The *NTP* client is enabled, and the device is synchronized by an *NTP* server. See the [Time > NTP](#) dialog.

Local offset [min]

Specifies the difference in minutes between Universal Time Coordinated (UTC) and local time:

$Local\ offset\ [min] = System\ time - System\ time\ (UTC)$

Possible values:

-780 . 840 (default setting: 60)

[Daylight saving time]

In this tab, you enable/disable the [Daylight saving time](#) function. You specify the start and end of summer time using a pre-defined profile. As an alternative, you specify these settings individually. During the summer time, the device advances the local time by one hour.

Operation

Daylight saving time

Enables/disables the [Daylight saving time](#) mode.

Possible values:

[On](#)

The [Daylight saving time](#) mode is enabled.

The device automatically sets the clock forward to summer time and back again.

[Off](#) (default setting)

The [Daylight saving time](#) mode is disabled.

You specify the daylight saving time settings in the [Summertime begin](#) and [Summertime end](#) frames.

Profile...

Opens the [Profile...](#) window to select a pre-defined profile for the start and end of summer time. Selecting a profile overwrites the settings specified in the [Summertime begin](#) and [Summertime end](#) frames.

Possible values:

[EU](#)

Daylight saving time settings as applicable in the European Union.

[USA](#)

Daylight saving time settings as applicable in the United States.

Summertime begin

In this frame, you specify the time at which the device sets the clock forward from standard time to summer time. In the first 3 fields, you specify the day for the start of summer time. In the last field, you specify the time.

Week

Specifies the week in the current month.

Possible values:

- (default setting)

first

second

third

fourth

last

Day

Specifies the day of the week.

Possible values:

- (default setting)

Sunday

Monday

Tuesday

Wednesday

Thursday

Friday

Saturday

Month

Specifies the month.

Possible values:

- (default setting)

January

February

March

April

May

June

July

August

September

October

November

December

System time

Specifies the time at which the device sets the clock forward to summer time.

Possible values:

<HH MM> (default setting: 00:00)

Summertime end

In this frame, you specify the time at which the device resets the clock from summer time to standard time. In the first 3 fields, you specify the day for the end of summer time. In the last field, you specify the time.

Week

Specifies the week in the current month.

Possible values:

- (default setting)

first

second

third

fourth

last

Day

Specifies the day of the week.

Possible values:

- (default setting)

Sunday

Monday

Tuesday

Wednesday

Thursday

Friday

Saturday

Month

Specifies the month.

Possible values:

- (default setting)

January

February

March

April

May

June
July
August
September
October
November
December

System time

Specifies the time at which the device resets the clock to standard time.

Possible values:

<HH MM> (default setting: 00.00)

22 NTP

[Time > NTP]

The device lets you synchronize the system time in the device and in the network using the Network Time Protocol (NTP).

The Network Time Protocol (NTP) is a procedure described in RFC 5905 for time synchronization in the network.

On the basis of a reference time source, NTP defines hierarchy levels for time servers and clients. A hierarchy level is known as a *stratum*. Devices of the 1st level (*stratum 1*) synchronize themselves directly with the reference time source and make the time information available to clients of the 2nd level (*stratum 2*). A GPS receiver or a radio-controlled clock can serve as the reference time source.

The NTP client in the device evaluates the time information of several servers and adjusts its own clock continuously to attain a high level of accuracy. If you also set up the device as an NTP server, then the device distributes time information to the clients in the subordinate network segment.

The menu contains the following dialogs:

Global
Server

2.2.1 Global

[Time > NTP > Global]

In this dialog, you determine if the device functions as an NTP client and server or only as an NTP client.

- As an NTP client, the device obtains the Universal Time Coordinated (UTC) from one or more NTP servers in the network.
- As an NTP server, the device distributes the Universal Time Coordinated (UTC) to NTP clients in the subordinate network segment. The device obtains the Universal Time Coordinated (UTC) from one or more NTP servers in the network, if these were previously specified.

Client only

The device transmits the time information without authentication in the VLAN of the device management as well as in Layer 3 on the IP interfaces set up.

Client

Enables/disables the NTP client in the device.

Possible values:

On

The NTP client is enabled.

The device obtains the time information from one or more NTP servers in the network.

Off (default setting)

The NTP client is disabled.

Note: Before you enable the client, disable the *Server* function in the *Client and server* frame.

Mode

Specifies from where the NTP client takes the time information.

Possible values:

uni cast (default setting)

The NTP client takes the time information from unicast responses of the servers that are indicated as active in the *Time > NTP > Server* dialog.

broadcast

The NTP client takes the time information from Broadcast messages.

Client and server

The device transmits the time information without authentication in the VLAN of the device management as well as in Layer 3 on the IP interfaces set up.

Server

Enables/disables the NTP client and the NTP server in the device.

Possible values:

On

The NTP client and the NTP server are enabled.

The NTP client obtains the time information from one or more NTP servers in the network. The NTP server distributes the time information to the NTP clients in the subordinate network segment.

Off (default setting)

The NTP client and the NTP server are disabled.

Note: If you enable the NTP client and the NTP server, then the device disables the function in the *Client* field in the *Client only* frame.

Mode

Specifies in which mode the NTP server works.

Possible values:

client-server (default setting)

With this setting, the device obtains the time information from NTP servers in the network and distributes it to NTP clients in the subordinate network segment.

- The NTP client takes the time information from the unicast responses of the servers that are indicated as active in the *Time > NTP > Server* dialog.
- The NTP server distributes the time information through unicast to the requesting clients.

symmetric

With this setting you integrate the device in a cluster of redundant NTP servers. The device synchronizes the time information with the other NTP servers in the cluster at intervals of 64 seconds.

In the *Time > NTP > Server* dialog, indicate the NTP servers participating in the cluster as active.

Specify a uniform value for the *stratum* for the NTP servers participating in the cluster.

Stratum

Specifies the hierarchical distance of the device to the referent time source.

Possible values:

1..16 (default setting: 12)

Example: Devices of the first level (*stratum 1*) synchronize themselves directly with the reference time source and make the time information available to the clients of the second level (*stratum 2*).

The device evaluates this value under the following circumstances:

- The NTP server in the device is working in *symmetric* mode.
or
- The device is using the local system clock as the time source. See the *Time source* field in the *Time > Basic Settings* dialog.

2.2.2 Server

[Time > NTP > Server]

In this dialog, you specify the NTP servers.

- The NTP client of the device obtains the time information from the unicast responses of the servers specified here.
- If the NTP server of the device is working in **symmetric** mode, then you specify the servers participating in the cluster here.
- The device lets you specify up to 4 NTP servers.

Table

For information on how to customize the appearance of the table, see [“Working with tables” on page 16](#).

Buttons



Add

Adds a table row.



Remove

Removes the selected table row.

Index

Displays the index number to which the table row relates. The device automatically assigns the value when you add a table row.

When you delete a table row, this leaves a gap in the numbering. When you add a table row, the device fills the first gap.

Active

Activates/deactivates the connection to the NTP server.

Possible values:

marked

The connection to the NTP server is activated.

- The NTP client of the device obtains the time information from the unicast responses of this server.
- If the NTP server of the device is working in **symmetric** mode, then this server participates in a cluster.

unmarked

The connection to the NTP server is deactivated.

IP address

Specifies the IP address of the NTP server.

Possible values:

Valid IPv4 address (default setting: 0.0.0.0)

Initial burst

Activates/deactivates the *Initial burst* mode.

During operation, the NTP client of the device only sends single data packets to request the time information. If the NTP server is unreachable (*Status* column = *not Responding*), then the NTP client of the device sends several data packets at once (burst) to synchronize as soon as possible.

Possible values:

marked

The *Initial burst* mode is active.

- The device sends only once several data packets (burst) when the NTP server is unreachable.
- Only use this setting if you use a private, non-public NTP server as reference time source.
- You use this setting with care to speed up the initial synchronization.

unmarked (default setting)

The *Initial burst* mode is inactive.

Burst

Activates/deactivates the *Burst* mode.

During operation, the NTP client of the device only sends single data packets to request the time information. In the *Burst* mode, the NTP client of the device sends several data packets at once (burst) when the NTP server is reachable and ready for synchronization.

Possible values:

marked

The *Burst* mode is active.

- For each polling interval, the device sends several data packets (burst) when the NTP server is reachable.
- Only use this setting if you use a private, non-public NTP server as reference time source.
- You use this setting with care to improve precision when the connection to the NTP server is unstable.

unmarked (default setting)

The *Burst* mode is inactive.

Preferred

Marks the NTP server as preferred reference time source when multiple NTP servers are specified.

Without marking, the NTP client of the device uses standard algorithms to select the reference time source.

Mark max. 1 sufficiently precise server as *Preferred*.

Possible values:

[marked](#)

The device uses the NTP server as the preferred reference time source. You use this setting to help prevent frequent connection changes between equal NTP servers.

[unmarked](#) (default setting)

No preferred NTP server.

Status

Displays the synchronization status.

Possible values:

[disabled](#)

No server available.

[protocol Error](#)

[not Synchronized](#)

The server is available. The server itself is not synchronized.

[not Responding](#)

The server is available. The device does not receive time information.

[synchronizing](#)

The server is available. The device receives time information.

[synchronized](#)

The server is available. The device has synchronized its clock with the server.

[genericError](#)

Device-internal error.

3 Device Security

The menu contains the following dialogs:

- User Management
- Authentication List
- LDAP
- Management Access
- Pre-login Banner

3.1 User Management

[Device Security > User Management]

If users log into the device management with valid login data, then the device lets them have access to its device management.

In this dialog, you manage the users of the local user management. You also specify the following settings here:

- Settings for the login
- Settings for saving the passwords
- Specify policy for valid passwords

The methods that the device uses for the authentication you specify in the [Device Security > Authentication List](#) dialog.

Configuration

This frame lets you specify settings for the login.

Login attempts

Specifies the number of possible consecutive unsuccessful login attempts when the user accesses the device management using the Graphical User Interface or the Command Line Interface.

Note: When accessing the device management using the Command Line Interface through the serial connection, the number of unsuccessful consecutive login attempts is unlimited.

Possible values:

0 . 5 (default setting: 0)

If the user makes one more consecutive unsuccessful login attempt, then the device locks access for the user.

The device lets only users with the [admini strator](#) authorization remove the lock.

The value 0 deactivates the lock. The user has unlimited attempts to log into the device management.

Min. password length

The device accepts the password if it contains at least the number of characters specified here.

The device checks the password according to this setting, regardless of the setting for the *Policy check* checkbox.

Possible values:

1 . 64 (default setting: 6)

Login attempts period (min.)

Displays the time period before the device resets the counter in the *Login attempts* field.

Possible values:

0 . 60 (default setting: 0)

Password policy

This frame lets you specify the policy for valid passwords. The device checks every new password and password change according to this policy.

The settings effect the *Password* column. The prerequisite is that the checkbox in the *Policy check* column is marked.

Upper-case characters (min.)

The device accepts the password if it contains at least as many upper-case letters as specified here.

Possible values:

0 . 16 (default setting: 1)

The value 0 deactivates this setting.

Lower-case characters (min.)

The device accepts the password if it contains at least as many lower-case letters as specified here.

Possible values:

0 . 16 (default setting: 1)

The value 0 deactivates this setting.

Digits (min.)

The device accepts the password if it contains at least as many numbers as specified here.

Possible values:

0 . 16 (default setting: 1)

The value 0 deactivates this setting.

Special characters (min.)

The device accepts the password if it contains at least as many special characters as specified here.

Possible values:

0 . 16 (default setting: 1)

The value 0 deactivates this setting.

Table

Every user requires an active user account to gain access to the device management. The table lets you set up and manage user accounts. To change settings, click the desired parameter in the table and modify the value.

For information on how to customize the appearance of the table, see [“Working with tables” on page 16](#).

Buttons



Add

Opens the [Create](#) window to add a table row.

- In the [User name](#) field, you specify the name of the user account.
Possible values:
Alphanumeric ASCII character string with 1..32 characters




Remove

Removes the selected table row.

User name

Displays the name of the user account.

To add a user account, click the  button.

Active

Activates/deactivates the user account.

Possible values:

[marked](#)

The user account is active. The device accepts the login of a user, to the device management, with this user name.

[unmarked](#) (default setting)

The user account is inactive. The device rejects the login of a user, to the device management, with this user name.

When one user account exists with the access role [admini str ator](#), this user account is constantly active.

Password

Displays ***** (asterisks) instead of the password with which the user logs into the device management. To change the password, click the relevant field.

Possible values:

Alphanumeric ASCII character string with 6..64 characters

The device accepts the following characters:

- a . z
- A . Z
- 0 . 9
- ! # \$ % & ' () * + , - . / : ; < = > ? @ [\] ^ _ ` { } ~

The minimum length of the password is specified in the [Configuration](#) frame. The device differentiates between upper and lower case.

If the checkbox in the [Policy check](#) column is marked, then the device checks the password according to the policy specified in the [Password policy](#) frame.

The device constantly checks the minimum length of the password, even if the checkbox in the [Policy check](#) column is [unmarked](#).

Role

Specifies the access role that regulates the access of the user to the individual functions of the device.

Possible values:

[unauthorized](#)

The user is blocked, and the device rejects the user login to the device management.

Assign this value to temporarily lock the user account. If the device detects an error when another access role is being assigned, then the device assigns this access role to the user account.

[guest](#) (default setting)

The user is authorized to monitor the device.

[auditor](#)

The user is authorized to monitor the device and to save the log file in the [Diagnostics > Report > Audit Trail](#) dialog.

[operator](#)

The user is authorized to monitor the device and to change the settings – with the exception of security settings for device access.

[administrator](#)

The user is authorized to monitor the device and to change the settings.

The device assigns the Service Type transferred in the response of a RADIUS server as follows to an access role:

- [Administrative-User](#): [administrator](#)
- [Login-User](#): [operator](#)
- [NAS-Prompt-User](#): [guest](#)

User locked

Unlocks the user account.

Possible values:

`marked`

The user account is locked. The user has no access to the device management. If the user makes too many consecutive unsuccessful login attempts, then the device automatically locks the user.

`unmarked` (grayed out) (default setting)

The user account is unlocked. The user has access to the device management.

Policy check

Activates/deactivates the password check.

Possible values:

`marked`

The password check is activated.

When you set up or change the password, the device checks the password according to the policy specified in the *Password policy* frame.

`unmarked` (default setting)

The password check is deactivated.

SNMP auth type

Specifies the authentication protocol that the device applies for user access using SNMPv3.

Possible values:

`hmacmd5` (default setting)

For this user account, the device uses protocol HMACMD5.

`hmacsha`

For this user account, the device uses protocol HMACSHA.

SNMP encryption type

Specifies the encryption protocol that the device applies for user access using SNMPv3.

Possible values:

`none`

No encryption.

`des` (default setting)

DES encryption

`aesCfb128`

AES128 encryption

3.2 Authentication List

[Device Security > Authentication List]

In this dialog, you manage the authentication lists. In an authentication list you specify which method the device uses for the authentication. You also have the option to assign pre-defined applications to the authentication lists.

If users log in with valid login data, then the device lets them have access to its device management. The device authenticates the users using the following methods:

- User management of the device
- LDAP
- RADIUS

In the default setting the following authentication lists are available:

- defaultLoginAuthList
- defaultV24AuthList

Table

For information on how to customize the appearance of the table, see “Working with tables” on page 16.

Note: If the table does not contain a list, then the access to the device management is only possible using the Command Line Interface through the serial interface of the device. In this case, the device authenticates the user by using the local user management. See the [Device Security > User Management](#) dialog.

Buttons



Add

Opens the [Create](#) window to add a table row.

- In the [Name](#) field, you specify the name of the list.
Possible values:
Alphanumeric ASCII character string with 1..32 characters



Remove

Removes the selected table row.



Allocate applications

Opens the [Allocate applications](#) window. The window displays the applications that you can designate to the selected list.

Click and select an item to designate it to the currently selected list.

An application that is already designated to a different list the device designates to the currently selected list, after you click the [Ok](#) button.

Click and deselect an item to undo its designation to the currently selected list.

If you deselect the application [Web Interface](#), then the connection to the device is lost, after you click the [Ok](#) button.

Name

Displays the name of the list.

To add a list, click the  button.

Policy 1
Policy 2
Policy 3
Policy 4
Policy 5

Specifies the authentication policy that the device uses for access using the application specified in the *Dedicated applications* column.

The device gives you the option of a fall-back solution. For this, you specify another policy in each of the policy fields. If the authentication with the specified policy is unsuccessful, then the device can use the next policy, depending on the order of the values entered in each policy.

Possible values:

l o c a l (default setting)

The device authenticates the users by using the local user management. See the *Device Security > User Management* dialog.

You cannot assign this value to the authentication list `defaul tDot 1x8021AuthLi st`.

r a d i u s

The device authenticates the users with a RADIUS server in the network. You specify the RADIUS server in the *Network Security > RADIUS > Authentication Server* dialog.

r e j e c t

The device accepts or rejects the user logging into the device management depending on which policy you try first. The following list contains authentication scenarios:


- If the first policy in the authentication list is **l o c a l** and the device accepts the login credentials of the user, then it logs the user into the device management without attempting the other policies.
- If the first policy in the authentication list is **l o c a l** and the device denies the login credentials of the user, then it attempts to log the user into the device management using the other policies in the order specified.
- If the first policy in the authentication list is **r a d i u s** or **l d a p** and the device rejects a login, then the login is immediately rejected without attempting to log in the user using another policy. If there is no response from the RADIUS or LDAP server, then the device attempts to authenticate the user with the next policy.
- If the first policy in the authentication list is **r e j e c t**, then the devices immediately rejects the user login without attempting another policy.
- Verify that the authentication list `defaul tV24AuthLi st` contains at least one policy different from **r e j e c t**.

l d a p

The device authenticates the users with authentication data and access role saved in a central location. You specify the Active Directory server that the device uses in the *Device Security > LDAP > Configuration* dialog.

Dedicated applications

Displays the dedicated applications. When users access the device with the relevant application, the device uses the specified policies for the authentication.

To allocate another application to the list or remove the allocation, click the  button. The device lets you assign each application to exactly one list.

Active

Activates/deactivates the list.

Possible values:

[marked](#) (default setting)

The list is activated. The device uses the policies in this list when users access the device with the relevant application.

[unmarked](#)

The list is deactivated.

3.3 LDAP

[Device Security > LDAP]

The Lightweight Directory Access Protocol (LDAP) lets you authenticate and authorize the users at a central point in the network. A widely used directory service accessible through LDAP is Active Directory[®].

The device forwards the login data of the user to the authentication server using the Lightweight Directory Access Protocol (LDAP). The authentication server decides if the login data is valid and transfers the authorizations of the user to the device.

Upon successful login, the device caches the login data. This speeds up the login process when users log into the device management again. In this case, no complex LDAP search operation is necessary.

The menu contains the following dialogs:

[LDAP Configuration](#)

[LDAP Role Mapping](#)

3.3.1 LDAP Configuration

[Device Security > LDAP > Configuration]

This dialog lets you specify up to 4 authentication servers. An authentication server authenticates and authorizes the users when the device forwards the login data to the server.

The device sends the login data to the first authentication server. When no response comes from this server, the device contacts the next server in the table.

Operation

Operation

Enables/disables the *LDAP* client.

If in the *Device Security > Authentication List* dialog you specify the value *l d a p* in one of the columns *Policy 1* to *Policy 5*, then the device uses the *LDAP* client. Prior to this, specify in the *Device Security > LDAP > Role Mapping* dialog at least one mapping for this access role *admini s t r a t o r*. This provides you access to the device as administrator after logging into the device management through LDAP.

Possible values:

On

The *LDAP* client is enabled.

Off (default setting)

The *LDAP* client is disabled.

Configuration

Buttons



Flush cache

Removes the cached login data of the successfully logged in users.

Client cache timeout [min]

Specifies for how many minutes after successfully logging into the device management the login data of a user remain valid. When a user logs in again within this time, no complex LDAP search operation is necessary. The login process is much faster.

Possible values:

1..1440 (default setting: 10)

Bind user

Specifies the user ID in the form of the “Distinguished Name” (DN) with which the device logs into the LDAP server.

If the LDAP server requires a user ID in the form of the “Distinguished Name” (DN) for the login, then this information is necessary. In Active Directory environments, this information is unnecessary.

The device attempts to authenticate on the LDAP server with the user ID to find the “Distinguished Name” (DN) for the users logging into the device management. The device conducts the search according to the settings in the *Base DN* and *User name attribute* fields.

Possible values:

Alphanumeric ASCII character string with 0..64 characters

Bind user password

Specifies the password which the device uses together with the user ID specified in the *Bind user* field when logging into the LDAP server.

Possible values:

Alphanumeric ASCII character string with 0..64 characters

Base DN

Specifies the starting point for the search in the directory tree in the form of the “Distinguished Name” (DN).

Possible values:

Alphanumeric ASCII character string with 0..255 characters

User name attribute

Specifies the LDAP attribute which contains a biunique user name. Afterwards, the user uses the user name contained in this attribute to log into the device management.

Often the LDAP attributes *userPrincipalName*, *mail*, *sAMAccountName* and *uid* contain a unique user name.

The device adds the character string specified in the *Default domain* field to the user name under the following condition:

- The user name contained in the attribute does not contain the @ character.
- In the *Default domain* field, a domain name is specified.

Possible values:

Alphanumeric ASCII character string with 0..64 characters
(default setting: *userPrincipalName*)

Default domain

Specifies the character string which the device adds to the user name of the users logging in if the user name does not contain the @ character.

Possible values:

Alphanumeric ASCII character string with 0..64 characters

CA certificate

To establish a secure connection, the device requires to obtain a valid digital certificate to verify the identity of the server. The prerequisite is that you have transferred the public certificate of the server onto the device. Ask the server administrator for a digital certificate in X.509 format. For security reasons, Hirschmann recommends using only digital certificates signed by a Certification Authority (CA).

URL

Specifies the path and file name of the digital certificate.

The device accepts digital certificates with the following properties:

- X.509 format
- . PEMfile name extension
- Base64-coded and enclosed by the lines
 -----BEGIN CERTIFICATE-----
 ...
 -----END CERTIFICATE-----

The device gives you the following options for transferring the file onto the device:

- Import from the PC

When the file is located on your PC or on a network drive, drag and drop it onto the  area.

As an alternative, click in the area to select the file.

You can also use SFTP or SCP to transfer the file from your PC to the device. Perform the following steps:

On your PC, open an SFTP or SCP client, for example WinSCP.

Use the SFTP or SCP client to open a connection to the device.

Transfer the file onto the device, into the directory /upload/ldap-cert.

When the file transfer is complete, the device starts installing the digital certificate. If the installation was successful, then the device generates an ok file in the directory /upload/ldap-cert and deletes the transferred file.

- Import from an SCP or SFTP server

When the file is on an SCP or SFTP server, specify the URL for the file in the following form:

scp: // or sftp: //<IP address>/<path>/<file name>

Click the *Start* button to open the *Credentials* window. In this window, you enter the *User name* and *Password* to log into the server.

scp: // or sftp: //<user>:<password>@<IP address>/<path>/<file name>

Start

Transfers the file specified in the *URL* field onto the device.

For the changes to take effect after transferring a digital certificate into the device, disable and re-enable the *LDAP* function. See the *Operation* frame.

Table

For information on how to customize the appearance of the table, see [“Working with tables” on page 16](#).

Buttons



Add

Adds a table row.



Remove

Removes the selected table row.

Index

Displays the index number to which the table row relates. The device automatically assigns the value when you add a table row.

Description

Specifies the description.

You have the option to describe here the authentication server or note additional information.

Possible values:

Alphanumeric ASCII character string with 0..255 characters

Address

Specifies the IP address or the DNS name of the server.

If in the [Connection security](#) column a value other than [none](#) is specified and the digital certificate contains only DNS names of the server, then specify a DNS name.

Possible values:

Valid IPv4 address (default setting: 0.0.0.0)

DNS name in the format <domain>. <ttl d> or <host>. <domain>. <ttl d>

The prerequisite is that you also enable the [Client](#) function in the [Advanced > DNS > Client > Global](#) dialog.

To establish an encrypted connection using a digital certificate, verify that the [Common Name](#) or [Subject Alternative Name](#) information in the digital certificate that you have transferred onto the device matches the value you specify here. Otherwise, the device will not be able to verify the identity of the server.

[_ldap._tcp.<domain>. <ttl d>](#)

Using this DNS name, the device queries the LDAP server list (SRV Resource Record) from the DNS server.

Destination TCP port

Specifies the TCP Port on which the server expects the requests.

If you have specified the value `_ldap._tcp.domain.tld` in the *Address* column, then the device ignores this value.

Possible values:

`0..65535 (216 - 1)` (default setting: `389`)
Exception: Port `2222` is reserved for internal functions.

Frequently used TCP-Ports:

- LDAP: `389`
- LDAP over SSL: `636`
- Active Directory Global Catalogue: `3268`
- Active Directory Global Catalogue SSL: `3269`

Connection security

Specifies the protocol which encrypts the communication between the device and the authentication server.

Possible values:

`none`

No encryption.

The device establishes an LDAP connection to the server and transmits the communication including the passwords in clear text.

`ssl`

Encryption with SSL.

The device establishes a TLS connection to the server and tunnels the LDAP communication over it.

`startTLS` (default setting)

Encryption with startTLS extension.

The device establishes an LDAP connection to the server and encrypts the communication.

The prerequisite for encrypted communication is that the device uses the correct time. If the digital certificate contains only the DNS names, then you specify the DNS name of the server in the *Address* column. Enable the *Client* function in the *Advanced > DNS > Client > Global* dialog.

If the digital certificate contains the IP address of the server in the *Subject Alternative Name* field, then the device is able to verify the identity of the server without the DNS setting.

Server status

Displays the connection status and the authentication with the authentication server.

Possible values:

`ok`

The server is reachable.

If in the *Connection security* column a value other than `none` is specified, then the device has verified the digital certificate of the server.

`unreachable`

Server is unreachable.

`other`

The device has not established a connection to the server yet.

Active

Activates/deactivates the use of the server.

Possible values:

`marked`

The device uses the server.

`unmarked` (default setting)

The device does not use the server.

3.3.2 LDAP Role Mapping

[Device Security > LDAP > Role Mapping]

This dialog lets you set up to 64 mappings to assign an access role to users.

In the table you specify if the device assigns an access role to the user based on an attribute with a specific value or based on the group membership.

- The device searches for the attribute and the attribute value within the user object.
- By evaluating the “Distinguished Name” (DN) contained in the member attributes, the device checks group the membership.

When a user logs into the device management, the device searches for the following information on the LDAP server:

- In the related user project, the device searches for attributes specified in the mappings.
- In the group objects of the groups specified in the mappings, the device searches for the member attributes.

On this basis, the device checks any mapping.

- Does the user object contain the required attribute?
or
- Is the user member of the group?

If the device does not find a match, then the user does not get access to the device.

If the device finds more than one mapping that applies to a user, then the setting in the *Matching policy* field decides. The user either obtains the access role with the more extensive authorizations or the 1st access role in the table that applies.

Configuration

Matching policy

Specifies which access role the device applies if more than one mapping applies to a user.

Possible values:

highest (default setting)

The device applies the access role with more extensive authorizations.

first

The device applies the rule which has the lower value in the *Index* column to the user.

Table

For information on how to customize the appearance of the table, see “Working with tables” on page 16.

Buttons



Add

Opens the [Create](#) window to add a table row.

- In the [Index](#) field, you specify the index number.

Possible values:

1 . 64



Remove

Removes the selected table row.

Index

Displays the index number to which the table row relates. You specify the index number when you add a table row.

Role

Specifies the access role that regulates the access of the user to the individual functions of the device.

Possible values:

[unauthorized](#)

The user is blocked, and the device rejects the user login.

Assign this value to temporarily lock the user account. If an error is detected when another role is being assigned, then the device assigns this access role to the user account.

[guest](#) (default setting)

The user is authorized to monitor the device.

[auditor](#)

The user is authorized to monitor the device and to save the log file in the [Diagnostics > Report > Audit Trail](#) dialog.

[operator](#)

The user is authorized to monitor the device and to change the settings – with the exception of security settings for device access.

[administrator](#)

The user is authorized to monitor the device and to change the settings.

Type

Specifies if a group or an attribute with an attribute value is specified in the [Parameter](#) column.

Possible values:

[attribute](#) (default setting)

The [Parameter](#) column contains an attribute with an attribute value.

[group](#)

The [Parameter](#) column contains the “Distinguished Name” (DN) of a group.

Parameter

Specifies a group or an attribute with an attribute value, depending on the setting in the *Type* column.

Possible values:

Alphanumeric ASCII character string with 0..255 characters

The device differentiates between upper and lower case.

- If in the *Type* column the value *attribute* is specified, then you specify the attribute in the form of *Attribute_name=Attribute_value*.
Example: *l=Germany*
- If in the *Type* column the value *group* is specified, then you specify the “Distinguished Name” (DN) of a group.
Example: *CN=admi n- users, OU=Groups, DC=exampl e, DC=com*

Active

Activates/deactivates the role mapping.

Possible values:

marked (default setting)

The role mapping is active.

unmarked

The role mapping is inactive.

3.4 Management Access

[Device Security > Management Access]

The menu contains the following dialogs:

[Server](#)

[IP Access Restriction](#)

[Web](#)

[Command Line Interface](#)

[SNMPv1/v2 Community](#)

3.4.1 Server

[Device Security > Management Access > Server]

This dialog lets you set up the server services which enable users or applications to access the management of the device.

The dialog contains the following tabs:

- [Information]
- [SNMP]
- [SSH]
- [HTTP]
- [HTTPS]

[Information]

This tab displays as an overview which server services are enabled.

Table

For information on how to customize the appearance of the table, see [“Working with tables” on page 16](#).

SNMPv1

Displays if the server service is active or inactive, which authorizes access to the device using SNMP version 1. See the [SNMP](#) tab.

Possible values:

- [marked](#)
Server service is active.
- [unmarked](#)
Server service is inactive.

SNMPv2

Displays if the server service is active or inactive, which authorizes access to the device using SNMP version 2. See the [SNMP](#) tab.

Possible values:

- [marked](#)
Server service is active.
- [unmarked](#)
Server service is inactive.

SNMPv3

Displays if the server service is active or inactive, which authorizes access to the device using SNMP version 3. See the [SNMP](#) tab.

Possible values:

[marked](#)

Server service is active.

[unmarked](#)

Server service is inactive.

SSH server

Displays if the server service is active or inactive, which authorizes access to the device using Secure Shell (SSH). See the [SSH](#) tab.

Possible values:

[marked](#)

Server service is active.

[unmarked](#)

Server service is inactive.

HTTP server

Displays if the server service is active or inactive, which authorizes access to the device using the Graphical User Interface through HTTP. See the [HTTP](#) tab.

Possible values:

[marked](#)

Server service is active.

[unmarked](#)

Server service is inactive.

HTTPS server

Displays if the server service is active or inactive, which authorizes access to the device using the Graphical User Interface through HTTPS. See the [HTTPS](#) tab.

Possible values:

[marked](#)

Server service is active.

[unmarked](#)

Server service is inactive.

[SNMP]

This tab lets you specify settings for the SNMP agent of the device and to enable/disable access to the device with different SNMP versions.

The SNMP agent enables access to the device management with SNMP-based applications.

Configuration

SNMPv1

Activates/deactivates the access to the device with SNMP version 1.

Possible values:

marked

SNMP version 1 access is active.

- You specify the community names in the [Device Security > Management Access > SNMPv1/v2 Community](#) dialog.

unmarked (default setting)

SNMP version 1 access is inactive.

SNMPv2

Activates/deactivates the access to the device with SNMP version 2.

Possible values:

marked

SNMP version 2 access is active.

- You specify the community names in the [Device Security > Management Access > SNMPv1/v2 Community](#) dialog.

unmarked (default setting)

SNMP version 2 access is inactive.

SNMPv3

Activates/deactivates the access to the device with SNMP version 3.

Possible values:

marked (default setting)

Access is activated.

unmarked

Access is deactivated.

Network management systems like Industrial HiVision use this protocol to communicate with the device.

UDP port

Specifies the number of the UDP port on which the SNMP agent receives requests from clients.

Possible values:


1.. 65535 (2¹⁶ - 1) (default setting: 161)

Exception: Port 2222 is reserved for internal functions.

To enable the SNMP agent to use the new port after a change, you proceed as follows:

Click the  button.

Select in the [Basic Settings > Load/Save](#) dialog the active configuration profile.

Click the  button to save the current settings.

Restart the device.

[SSH]

This tab lets you enable/disable the SSH server in the device and specify its settings required for SSH. The server works with SSH version 2.

The SSH server enables access to the device management remotely through the Command Line Interface. SSH connections are encrypted.

To access the device and the connected external memory using SFTP or SCP, you also need access to the SSH server. With an SFTP or SCP client, for example WinSCP, you have the option of transferring configuration files or an updated device software onto the device.

The SSH server identifies itself to the clients using its public RSA key. When first setting up the connection, the client program displays the user the fingerprint of this key. The fingerprint contains a Base64-coded character sequence that is easy to check. When you make this character sequence available to the users through a reliable channel, they have the option to compare both fingerprints. If the character sequences match, then the client is connected to the correct server.

The device lets you generate the private and public keys (host keys) required for RSA directly in the device. As an alternative, transfer your own host key in PEM format onto the device.

As an alternative, the device lets you load the RSA key (host key) from an external memory during the system startup. You activate this function in the [Basic Settings > External Memory](#) dialog, [SSH key auto upload](#) column.

Operation

SSH server

Enables/disables the SSH server.

Possible values:

On (default setting)

The SSH server is enabled.

The access to the device management is possible through the Command Line Interface using an encrypted SSH connection.

You can start the server only if there is an RSA signature in the device.

Off

The SSH server is disabled.

When you disable the SSH server, the existing connections remain established. However, the device helps prevent new connections from being set up.

Note: If you disable the [SSH](#) server, then the access to the Command Line Interface is only possible through the serial interface of the device.

Configuration

TCP port

Specifies the number of the TCP port on which the device receives SSH requests from clients.

Possible values:

- 1.. 65535 (2¹⁶ - 1) (default setting: 22)
- Exception: Port 2222 is reserved for internal functions.

The server restarts automatically after the port is changed. Existing connections remain in place.

Sessions

Displays how many SSH connections are currently established to the device.

Sessions (max.)

Specifies the maximum number of SSH connections to the device that can be set up simultaneously.

When you access the device using Command Line Interface, SFTP or SCP, each of these applications establishes a separate SSH connection to the device.

Possible values:

- 1.. 5 (default setting: 5)

Session timeout [min]

Specifies the timeout in minutes. After the user logged into the device management has been inactive for this time, the device ends the connection.

A change in the value takes effect the next time a user logs into the device management.

Possible values:

- 0
Deactivates the function. The connection remains established in the case of inactivity.
- 1.. 160 (default setting: 5)

Signature

RSA present

Displays if an RSA host key is present in the device.

Possible values:

- marked
A key is present.
- unmarked
No key is present.

Create

Generates a host key in the device. The prerequisite is that the [SSH](#) server is disabled.

Length of the key generated:

- 2048 bit (RSA)

To get the SSH server to use the generated host key, restart the SSH server.

As an alternative, transfer your own host key in PEM format onto the device. See the [Key import](#) frame.

Delete

Removes the host key from the device. The prerequisite is that the SSH server is disabled.

Oper status

Displays if the device currently generates a host key.

It is possible that another user triggered this action.

Possible values:

[rsa](#)

The device currently generates an RSA host key.

[none](#)

The device does not generate a host key.

Fingerprint

The fingerprint is an easy to verify string that uniquely identifies the host key of the SSH server.

After importing a new host key, the device continues to display the existing fingerprint until you restart the server.

Fingerprint type

Specifies which fingerprint the [RSA fingerprint](#) field displays.

Possible values:

[md5](#)



The [RSA fingerprint](#) field displays the fingerprint as hexadecimal MD5 hash.

[sha256](#) (default setting)

The device does not support this setting. The [RSA fingerprint](#) field retains the previous display.

RSA fingerprint

Displays the fingerprint of the public host key of the SSH server.

When you change the settings in the [Fingerprint type](#) field, click afterwards the  button and then the  button to update the display.

Key import

URL


Specifies the path and file name of your own RSA host key.

The device accepts the RSA key if it has the following key length:

- 2048 bit (RSA)

The device gives you the following options for transferring the file onto the device:

- Import from the PC

When the file is located on your PC or on a network drive, drag and drop it onto the  area. As an alternative, click in the area to select the file.

You can also use SFTP or SCP to transfer the file from your PC to the device. Perform the following steps:

On your PC, open an SFTP or SCP client, for example WinSCP.

Use the SFTP or SCP client to open a connection to the device.

Transfer the file onto the device, into the directory `/upload/ssh-key`.

When the file transfer is complete, the device starts installing the key. If the installation was successful, then the device generates an `ok` file in the directory `/upload/ssh-key` and deletes the transferred file.

- Import from an SCP or SFTP server

When the file is on an SCP or SFTP server, specify the URL for the file in the following form:

`scp: // or sftp: //<IP address>/<path>/<file name>`

Click the **Start** button to open the **Credentials** window. In this window, you enter the **User name** and **Password** to log into the server.

`scp: // or sftp: //<user>:<password>@<IP address>/<path>/<file name>`

Start

Transfers the file specified in the **URL** field onto the device.


For the changes to take effect after transferring a digital certificate onto the device, disable and re-enable the **SSH server** function. See the **Operation** frame.

[HTTP]

This tab lets you enable/disable the Hypertext Transfer Protocol (HTTP) for the web server and specify the settings required for HTTP.

The web server provides the Graphical User Interface through an unencrypted HTTP connection. For security reasons, disable the Hypertext Transfer Protocol (HTTP) and use the Hypertext Transfer Protocol Secure (HTTPS) instead.

The device supports up to 10 simultaneous connections using HTTP or HTTPS.

Note: If you change the settings in this tab and click the  button, then the device ends the session and disconnects every opened connection. To continue working with the Graphical User Interface, log in again.

Operation

HTTP server

Enables/disables the *HTTP* function for the web server.

Possible values:

On (default setting)

The *HTTP* function is enabled.

The access to the device management is possible through an unencrypted *HTTP* connection. When the *HTTPS* function is also enabled, the device automatically redirects the request for a *HTTP* connection to an encrypted *HTTPS* connection.

Off

The *HTTP* function is disabled.

When the *HTTPS* function is enabled, the access to the device management is possible through an encrypted *HTTPS* connection.

Note: If the *HTTP* and *HTTPS* functions are disabled, then you can enable the *HTTP* function using the Command Line Interface command `http server` to get to the Graphical User Interface.

Configuration

TCP port

Specifies the number of the TCP port on which the web server receives HTTP requests from clients.

Possible values:

1.. 65535 (2¹⁶ - 1) (default setting: 80)

Exception: Port 2222 is reserved for internal functions.


[HTTPS]

This tab lets you enable/disable the Hypertext Transfer Protocol Secure(HTTPS) for the web server and specify the settings required for HTTPS.

The web server provides the Graphical User Interface through an encrypted HTTP connection.

A digital certificate is required for the encryption of the HTTP connection. The device lets you generate this digital certificate yourself or to transfer an existing digital certificate onto the device.

The device supports up to 10 simultaneous connections using HTTP or HTTPS.

Note: If you change the settings in this tab and click the  button, then the device ends the session and disconnects every opened connection. To continue working with the Graphical User Interface, log in again.

Operation

HTTPS server

Enables/disables the [HTTPS](#) function for the web server.

Possible values:

[On](#) (default setting)

The [HTTPS](#) function is enabled.

The access to the device management is possible through an encrypted [HTTPS](#) connection. When there is no digital certificate present, the device generates a digital certificate before it enables the [HTTPS](#) function.

[Off](#)

The [HTTPS](#) function is disabled.

When the [HTTP](#) function is enabled, the access to the device management is possible through an unencrypted [HTTP](#) connection.

Note: If the [HTTP](#) and [HTTPS](#) functions are disabled, then you can enable the [HTTPS](#) function using the Command Line Interface command `https server` to get to the Graphical User Interface.

Configuration

TCP port

Specifies the number of the TCP port on which the web server receives [HTTPS](#) requests from clients.

Possible values:

1.. 65535 ($2^1 - 1$) (default setting: 443)

Exception: Port 2222 is reserved for internal functions.

Certificate

If the device uses a digital certificate not signed by a Certification Authority (CA) known to the web browser, then the web browser may display a warning message before loading the Graphical User Interface.

To address the warning, you have the following possibilities:

- Transfer a digital certificate onto the device whose Certification Authority (CA) is known to your web browser. This may additionally require you to make the Certification Authority (CA) known to your web browser or operating system.
- As a workaround, you can also add an exception rule for the existing device certificate in your web browser.

Present

Displays if a digital certificate is present in the device.

Possible values:

[marked](#)

A digital certificate is present.

[unmarked](#)

The digital certificate has been removed.

Create

Generates a digital certificate in the device.

Until restarting the web server uses the previous certificate.

To get the web server to use the newly generated digital certificate, restart the web server. Restarting the web server is possible only through the Command Line Interface.

As an alternative, transfer your own digital certificate onto the device. See the [Certificate import](#) frame.

Delete

Deletes the digital certificate.

Until restarting the web server uses the previous certificate.

Oper status

Displays if the device currently generates or deletes a digital certificate.

It is possible that another user has triggered the action.

Possible values:

[none](#)

The device does currently not generate or delete a digital certificate.

[delete](#)

The device currently deletes a digital certificate.

[generate](#)

The device currently generates a digital certificate.

Fingerprint

The fingerprint is an easily verified hexadecimal number sequence that uniquely identifies the digital certificate of the HTTPS server.

After importing a new digital certificate, the device displays the current fingerprint until you restart the server.

Fingerprint type

Specifies which fingerprint the *Fingerprint* field displays.

Possible values:

sha1

The *Fingerprint* field displays the SHA1 fingerprint of the digital certificate.

sha256 (default setting)

The *Fingerprint* field displays the SHA256 fingerprint of the digital certificate.

Fingerprint

Hexadecimal character sequence of the digital certificate used by the server.

When you change the settings in the *Fingerprint type* field, click afterwards the ✓ button and then the ↻ button to update the display.

Certificate import

URL


Specifies the path and file name of the digital certificate.

The device accepts digital certificates with the following properties:

- X.509 format
- . PEMfile name extension
- Base64-coded and enclosed by the lines
 - -----BEGIN PRIVATE KEY-----
 - ...
 - END PRIVATE KEY-----
 - OR
 - -----BEGIN CERTIFICATE-----
 - ...
 - END CERTIFICATE-----
- RSA key with 2048 bit length

The device gives you the following options for transferring the file onto the device:

- Import from the PC

When the file is located on your PC or on a network drive, drag and drop it onto the  area. As an alternative, click in the area to select the file.

You can also use SFTP or SCP to transfer the file from your PC to the device. Perform the following steps:

On your PC, open an SFTP or SCP client, for example WinSCP.

Use the SFTP or SCP client to open a connection to the device.

Transfer the file onto the device, into the directory `/upload/https-cert`.

When the file transfer is complete, the device starts installing the certificate. If the installation was successful, then the device generates an `ok` file in the directory `/upload/https-cert` and deletes the transferred file.

- Import from an SCP or SFTP server

When the file is on an SCP or SFTP server, specify the URL for the file in the following form:

- `scp: // or sftp: //<IP address>[: port] /<path>/<file name>`

Click the *Start* button to open the *Credentials* window. In this window, you enter the *User name* and *Password* to log into the server.

- `scp: // or sftp: //<user>: <password>@<IP address>[: port] /<path>/<file name>`

Start

Transfers the file specified in the *URL* field onto the device.

For the changes to take effect after transferring a digital certificate onto the device, disable and re-enable the *HTTPS server* function. See the *Operation* frame.

3.4.2 IP Access Restriction

[Device Security > Management Access > IP Access Restriction]

This dialog lets you restrict access to the device management from a specific IP address range or through a specific physical interface for selected applications.

- If the function is disabled, then access to the device management is unrestricted. Everyone can access the device management from any IP address or through any physical interface using any application.
- If the function is enabled, then access is restricted. Everyone can access the device management only under the following conditions:
 - At least one rule is active.
and
 - You access the device with a permitted application from a permitted IP address range or through a permitted physical interface specified in the rule.

Operation

Operation

Enables/disables the *IP Access Restriction* function.

Possible values:

On

The *IP Access Restriction* function is enabled.

The access to the device management is restricted.

Note: Before you enable the function, verify that the table contains at least one active rule that grants you access to the device management. Otherwise, access to the device management is only possible using the Command Line Interface through the serial connection.

Off (default setting)

The *IP Access Restriction* function is disabled.

Table

You have the option of defining up to 16 table rows and activating them separately.

For information on how to customize the appearance of the table, see [“Working with tables” on page 16](#).

Buttons



Add

Adds a table row.



Remove

Removes the selected table row.

Index

Displays the index number to which the table row relates. The device automatically assigns the value when you add a table row.

The priority of access to the device management is based on the index number.

When you delete a table row, this leaves a gap in the numbering. When you add a table row, the device fills the first gap.

Possible values:

1..16

Interface

Specifies the physical interface through which users have access to the device management.

The prerequisite is that in the *Address* and *Netmask* columns, the value 0.0.0.0 is specified.

Possible values:

All (default setting)

Users can have restricted access to the device management through every interface based on the IP address specified in the *Address* column.

<Port number >

Users can have restricted access to the device management only through the specified interface.

The device supports the *IP Access Restriction* function only on physical interfaces, not on logical interfaces.

Address

Specifies the IP address of the network from which you allow the access to the device management. You specify the network range in the *Netmask* column.

The prerequisite is that in the *Interface* column the value All is specified.

Possible values:

Valid IPv4 address (default setting: 0.0.0.0)

Netmask

Specifies the range of the network specified in the *Address* column.

The prerequisite is that in the *Interface* column the value All is specified.

Possible values:

Valid netmask (default setting: 0.0.0.0)

Example: To restrict access from a single IP address, specify the value as 255.255.255.255.

HTTP

Activates/deactivates the HTTP access.

Possible values:

`marked` (default setting)

HTTP access is active. Access is possible from the adjacent IP address range or through the adjacent physical interface.

`unmarked`

HTTP access is inactive.

HTTPS

Activates/deactivates the HTTPS access.

Possible values:

`marked` (default setting)

HTTPS access is active. Access is possible from the adjacent IP address range or through the adjacent physical interface.

`unmarked`

HTTPS access is inactive.

SNMP

Activates/deactivates the SNMP access.

Possible values:

`marked` (default setting)

SNMP access is active. Access is possible from the adjacent IP address range or through the adjacent physical interface.

`unmarked`

SNMP access is inactive.

SSH

Activates/deactivates the SSH access.

Possible values:

`marked` (default setting)

SSH access is active. Access is possible from the adjacent IP address range or through the adjacent physical interface.

`unmarked`

SSH access is inactive.

Active

Activates/deactivates the table row.

Possible values:

`marked`

The table row is active. The device restricts the access to the device management from the specified IP address range or through the specified interface for the selected applications.

`unmarked` (default setting for new table row)

The table row is inactive. The device does not restrict access to the device management from the specified IP address range or through the specified interface for the selected applications.

3.4.3 Web

[Device Security > Management Access > Web]

In this dialog, you specify settings for the Graphical User Interface.

Configuration

Web interface session timeout [min]

Specifies the timeout in minutes. After the device has been inactive for this time, it ends the session for the user logged into the device management.

Possible values:

0 . 160 (default setting: 5)

The value 0 deactivates the function, and the user remains logged in when inactive.

3.4.4 Command Line Interface

[Device Security > Management Access > CLI]

In this dialog, you specify settings for the Command Line Interface. For further information about the Command Line Interface, see the “Command Line Interface” reference manual.

The dialog contains the following tabs:

- [Global]
- [Login banner]

[Global]

This tab lets you change the prompt in the Command Line Interface and specify the automatic closing of sessions through the serial interface when they have been inactive.

The device has the following serial interfaces.

- V.24 interface

Configuration

Login prompt

Specifies the character string that the device displays in the Command Line Interface at the start of every command line.

Possible values:

Alphanumeric ASCII character string with 0..128 characters
(0x20..0x7E) including space characters

Wildcards

- %d date
- %i IP address
- %m MAC address
- %p product name
- %t time

Default setting: (EAGLE)

Changes to this setting are immediately effective in the active Command Line Interface session.

Serial interface timeout [min]

Specifies the time in minutes after which the device automatically closes the session of an inactive user logged into the device management with the Command Line Interface through the serial interface.

Possible values:

0..160 (default setting: 5)

The value 0 deactivates the function, and the user remains logged into the device management when inactive.

A change in the value takes effect the next time a user logs into the device management.

For the *SSH* server, you specify the timeout in the *Device Security > Management Access > Server* dialog.

[Login banner]

In this tab you replace the start screen of the Command Line Interface with your own text.

In the default setting, the start screen displays information about the device, such as the software version and the device settings. With the function in this tab, you deactivate this information and replace it with an individually specified text.

To display your own text in the Command Line Interface and in the Graphical User Interface before the login, you use the *Device Security > Pre-login Banner* dialog.

Operation

Operation

Enables/disables the *Login banner* function.

Possible values:

On

The *Login banner* function is enabled.

The device displays the text information specified in the *Banner text* field to the users that log into the device management through the Command Line Interface.

Off (default setting)

The *Login banner* function is disabled.

The start screen displays information about the device. The text information in the *Banner text* field is kept.

Banner text

Banner text

Specifies the character string that the device displays in the Command Line Interface at the start of every session.

Possible values:

Alphanumeric ASCII character string with 0..1024 characters

(*0x20* . *0x7E*) including space characters

<Tab>

<Li ne break>

3.4.5 SNMPv1/v2 Community

[Device Security > Management Access > SNMPv1/v2 Community]

In this dialog, you specify the community name for SNMPv1/v2 applications.

Applications send requests using SNMPv1/v2 with a community name in the SNMP data packet header. Depending on the community name (see [Community](#) column), the application gets *read-only* authorization or *read and write* authorization.

You activate the access to the device using SNMPv1/v2 in the [Device Security > Management Access > Server](#) dialog.

Table

For information on how to customize the appearance of the table, see [“Working with tables” on page 16](#).

Community

Displays the authorization for SNMPv1/v2 access to the device.

Possible values:

[Write](#)

For requests with the community name entered, the application receives *read and write* authorization.

[Read](#)

For requests with the community name entered, the application receives *read-only* authorization.

Name

Specifies the community name for the adjacent authorization.

Possible values:

Alphanumeric ASCII character string with 0..64 characters

The device accepts the following characters:

- <space>
- 0 . 9
- a . z
- A . Z
- ! " # \$ % & ' () * + , - . / : ; <=> ? @ [\] ^ _ ` { | } ~

`private` (default setting for *read and write* authorization)

`public` (default setting for *read-only* authorization)

3.5 Pre-login Banner

[Device Security > Pre-login Banner]

This dialog lets you display a greeting or information text to users before they log into the device management.

The users see this text in the login dialog of the Graphical User Interface and of the Command Line Interface. Users logging into the device management with SSH see the text - regardless of the client used - before or during the login.

To display the text only in the Command Line Interface, use the settings in the [Device Security > Management Access > CLI](#) dialog.

Operation

Operation

Enables/disables the [Pre-login Banner](#) function.

Using the [Pre-login Banner](#) function, the device displays a greeting or information text in the login dialog of the Graphical User Interface and of the Command Line Interface.

Possible values:

On

The [Pre-login Banner](#) function is enabled.

The device displays the text specified in the [Banner text](#) field in the login dialog.

Off (default setting)

The [Pre-login Banner](#) function is disabled.

The device does not display a text in the login dialog. When you enter a text in the [Banner text](#) field, the device saves this text.

Banner text

Banner text

Specifies information text that the device displays in the login dialog of the Graphical User Interface and of the Command Line Interface.

Possible values:

Alphanumeric ASCII character string with 0..512 characters
(0x20 . 0x7E) including space characters

<Tab>

<Line break>

4 Network Security

The menu contains the following dialogs:

- [Network Security Overview](#)
- [RADIUS](#)
- [Asset](#)
- [Protocol](#)
- [Packet Filter](#)
- [Deep Packet Inspection](#)
- [DoS](#)

4.1 Network Security Overview

[Network Security > Overview]

This dialog displays an overview over the network security rules used in the device.

Overview

The top level displays:

- The ports to which a network security rule is assigned
- The VLANs to which a network security rule is assigned

The subordinate levels display:

- The set-up [Packet filter L3](#) rules
See the [Network Security > Packet Filter > Routed Firewall Mode](#) dialog.
- The set-up [Packet filter L2](#) rules
See the [Network Security > Packet Filter > Transparent Firewall Mode](#) dialog.
- The set-up [Destination NAT](#) rules
See the [Routing > NAT > Destination NAT](#) dialog.
- The set-up [Double NAT](#) rules
See the [Routing > NAT > Double NAT](#) dialog.
- The set-up [Masquerading NAT](#) rules
See the [Routing > NAT > Masquerading NAT](#) dialog.
- The set-up [1:1 NAT](#) rules
See the [Routing > NAT > 1:1 NAT](#) dialog.

Buttons



Displays a text field to search for a keyword. When you enter a character or string, the overview displays only items related to this keyword.



Collapses the levels. The overview then displays only the first level of the items.



Expands the levels. The overview then displays every level of the items.



Expands the current item and displays the items of the next lower level.



Collapses the item and hides the items of the underlying levels.

4.2 RADIUS

[Network Security > RADIUS]

With its factory settings, the device authenticates users based on the local user management. However, as the size of a network increases, it becomes more difficult to keep the login data of the users consistent across the devices.

RADIUS (Remote Authentication Dial-In User Service) lets you authenticate and authorize the users at a central point in the network. A RADIUS server performs the following tasks here:

- Authentication
The authentication server authenticates the users when the RADIUS client at the access point forwards the login data of the users to the server.
- Authorization
The authentication server authorizes logged in users for selected services by assigning various parameters for the relevant end device to the RADIUS client at the access point.

If you assign the `radius` policy to an application in the [Device Security > Authentication List](#) dialog, then the device operates in the role of the RADIUS client. The device forwards the login data of the users to the primary authentication server. The authentication server decides if the login data is valid and transfers the authorizations of the users to the device.

The device assigns the Service Type transferred in the response of a RADIUS server as follows to an access role existing in the device:

- Administrative-User: `administrator`
- Login-User: `operator`
- NAS-Prompt-User: `guest`

The menu contains the following dialogs:

- [RADIUS Global](#)
- [RADIUS Authentication Server](#)
- [RADIUS Authentication Statistics](#)

4.21 RADIUS Global

[Network Security > RADIUS > Global]

This dialog lets you specify basic settings for RADIUS.

RADIUS configuration

Buttons



Deletes the statistics in the [Network Security > RADIUS > Authentication Statistics](#) dialog.

Retransmits (max.)

Specifies how many times the device retransmits an unanswered request to the authentication server before the device sends the request to an alternative authentication server.

Possible values:

1..15 (default setting: 4)

Timeout [s]

Specifies how many seconds the device waits for a response after a request to an authentication server before it retransmits the request.

Possible values:

1..30 (default setting: 5)

NAS IP address (attribute 4)

Specifies the IP address that the device transfers to the authentication server as attribute 4. Specify the IP address of the device or another available address.

Note: The device only includes the attribute 4 if the packet was triggered by the *802.1X* authentication request of an end device (supplicant).

Possible values:

Valid IPv4 address (default setting: 0.0.0.0)

In many cases, there is a firewall between the device and the authentication server. In the Network Address Translation (NAT) in the firewall changes the original IP address, and the authentication server receives the translated IP address of the device.

The device transfers the IP address in this field unchanged across the Network Address Translation (NAT).

4.2.2 RADIUS Authentication Server

[Network Security > RADIUS > Authentication Server]

This dialog lets you specify up to 8 authentication servers. An authentication server authenticates and authorizes the users when the device forwards the login data to the server.

The device sends the login data to the specified primary authentication server. When the server does not respond, the device contacts the specified authentication server that is highest in the table. When no response comes from this server either, the device contacts the next server in the table.

Table

For information on how to customize the appearance of the table, see [“Working with tables” on page 16](#).

Buttons



Add

Opens the [Create](#) window to add a table row.

- In the [Index](#) field, you specify the index number.
- In the [Address](#) field, you specify the IP address of the server.



Remove

Removes the selected table row.

Index

Displays the index number to which the table row relates. You specify the index number when you add a table row.

Name

Displays the name of the server. To change the value, click the relevant field.

Possible values:

Alphanumeric ASCII character string with 1..32 characters
(default setting: [Default-RADIUS-Server](#))

You can specify the same name for several servers. When several servers have the same name, the setting in the [Primary server](#) column applies.

Address

Specifies the IP address of the server.

Possible values:

Valid IPv4 address

Destination UDP port

Specifies the number of the UDP port on which the server receives requests.

Possible values:

0 . 65535 (2¹⁶ - 1) (default setting: 1812)

Exception: Port 2222 is reserved for internal functions.

Secret

Displays ***** (asterisks) when you specify a password with which the device logs into the server. To change the password, click the relevant field.

Possible values:

Alphanumeric ASCII character string with 1..16 characters

You get the password from the administrator of the authentication server.

Primary server

Specifies the authentication server as primary or secondary.

Possible values:

marked

The server is specified as the primary authentication server. The device sends the login data for authenticating the users to this authentication server.

This setting applies only if more than one server in the table has the same value in the [Name](#) column.

unmarked (default setting)

The server is the secondary authentication server. When the device does not receive a response from the primary authentication server, the device sends the login data to the secondary authentication server.

Active

Activates/deactivates the connection to the server.

The device uses the server, if you specify in the [Device Security > Authentication List](#) dialog the value [radius](#) in one of the columns [Policy 1](#) to [Policy 5](#).

Possible values:

marked (default setting)

The connection is active. The device sends the login data for authenticating the users to this server if the preconditions named above are fulfilled.

unmarked

The connection is inactive. The device does not send any login data to this server.

4.2.3 RADIUS Authentication Statistics

[Network Security > RADIUS > Authentication Statistics]

This dialog displays information about the communication between the device and the authentication server. The table displays the information for each server in a separate table row.

To delete the statistic, click in the [Network Security > RADIUS > Global](#) dialog the  button.

Table

For information on how to customize the appearance of the table, see [“Working with tables” on page 16](#).

Name

Displays the name of the server.

IP address

Displays the IP address of the server.

Round trip time

Displays the time interval in hundredths of a second between the last response received from the server (Access Reply/Access Challenge) and the corresponding data packet sent (Access Request).

Access requests

Displays the number of access data packets that the device sent to the server. This value does not take repetitions into account.

Retransmitted access requests

Displays the number of access data packets that the device retransmitted to the server.

Access accepts

Displays the number of access accept data packets that the device received from the server.

Access rejects

Displays the number of access reject data packets that the device received from the server.

Access challenges

Displays the number of access challenge data packets that the device received from the server.

Malformed access responses

Displays the number of malformed access response data packets that the device received from the server (including data packets with an invalid length).

Bad authenticators

Displays the number of access response data packets with an invalid authenticator that the device received from the server.

Pending requests

Displays the number of access request data packets that the device sent to the server to which it has not yet received a response from the server.

Timeouts

Displays how many times no response to the server was received before the specified waiting time elapsed.

Unknown types

Displays the number data packets with an unknown data type that the device received from the server on the authentication port.

Packets dropped

Displays the number of data packets that the device received from the server on the authentication port and then discarded them.

4.3 Asset

[Network Security > Asset]

This dialog lets you specify the settings for managing the assets. An asset can represent a physical device, such as a PLC (Programmable Logic Controller), a computer or a network device. An asset can also represent a virtual object, such as a multicast address range, or a multicast address. Assets provide flexibility when setting up and maintaining firewall rules. The device lets you set up to 100 assets.

Table

For information on how to customize the appearance of the table, see [“Working with tables” on page 16](#).

Buttons



Add

Opens the [Create](#) window to add a table row.

- In the [Name](#) field, you specify a unique name for the asset.

Possible values:

Alphanumeric ASCII character string with 1..32 characters, excluding the character [any](#)

When you click the [Ok](#) button, the device adds the table row. The device assigns the name specified in the [Name](#) field to the table row.



Remove

Removes the selected table row.

Index

Displays the sequential number of the asset to which the table row relates. The device automatically assigns the value when you add a table row.

Name

Specifies a unique name for the asset.

Possible values:

Alphanumeric ASCII character string with 1..32 characters, excluding the character [any](#)

Description

Specifies a description for the asset.

Possible values:

Alphanumeric ASCII character string with 0..128 characters

Type

Specifies the type of the asset.

Possible values:

[computer](#) (default setting)

[control l er](#)

[devi ce](#)

[netw ork](#)

[netw ork-equi pment](#)

[broadcast](#)

[mul ti cast](#)

Manufacturer

Specifies the manufacturer of the asset.

Possible values:

Alphanumeric ASCII character string with 0..128 characters

Model

Specifies the model of the asset.

Possible values:

Alphanumeric ASCII character string with 0..128 characters

General location

Specifies a general location for the asset.

Possible values:

Alphanumeric ASCII character string with 0..128 characters

Specific location

Specifies a specific location for the asset.

Possible values:

Alphanumeric ASCII character string with 0..128 characters

Asset tag

Specifies a tag for the identification of the user-defined asset.

Possible values:

Alphanumeric ASCII character string with 0..128 characters

IP address

Specifies the IP address of the asset.

Possible values:

[any](#) (default setting)

The device accepts any IP address associated with the asset.

Valid IPv4 address

The device applies the specified IP address to the asset.

Valid IPv4 address and netmask in CIDR notation

The device applies the specified IP address in the specified subnet to the asset.

Example: [192.168.112.0/25](#)

An exclamation mark (!) preceding the IP address reverses the expression into its opposite. The device accepts any IP address or the subnet associated with the asset excluding the specified IP address or the subnet.

Example: [!1.1.1.1](#) or [!192.168.112.0/25](#)

MAC address

Specifies the MAC address of the asset.

Possible values:

[any](#) (default setting)

The device accepts any MAC address associated with the asset.

Valid MAC address

The device applies the specified MAC address to the asset.

4.4 Protocol

[Network Security > Protocol]

This dialog lets you specify basic settings for the user-defined protocol. The device lets you set up to 50 user-defined protocols.

Table

For information on how to customize the appearance of the table, see [“Working with tables” on page 16](#).

Buttons



Opens the [Create](#) window to add a table row. In the [Protocol name](#) field, you specify a unique name for the protocol.

Possible values:

Alphanumeric ASCII character string with 1..32 characters, excluding the following characters:

- any
- i cmp
- i gmp
- i pi p
- tcp
- udp
- esp
- ah
- i cmpv6

When you click the [Ok](#) button, the device adds the table row. The device assigns the name specified in the [Protocol name](#) field to the table row.



Removes the selected table row.

Index

Displays the sequential number of the protocol to which the table row relates. The device automatically assigns the value when you add a table row.

Protocol name

Specifies a unique name for the protocol.

Possible values:

Alphanumeric ASCII character string with 1..32 characters, excluding the following characters:

- any
- i cmp
- i gmp
- i pi p
- tcp

- udp
- esp
- ah
- icmpv6

Description

Specifies a description for the protocol.

Possible values:

Alphanumeric ASCII character string with 0..128 characters

Protocol type

Specifies the protocol type for the user-defined protocol, which the device applies in the packet filter rule.

Possible values:

- any (default setting)
- ethernet
- icmp
- tcp
- udp

Ethertype

Specifies the *Ethertype* keyword of the data packets, which the Layer 2 packet filter uses.

Possible values:

- custom (default setting)
- atalk
- arp
- brna
- ipv4
- ipv6
- ipxd
- mplsncast
- mplsucast
- netbios
- novell
- pppoe-disc
- rarp
- pppoe-session
- ipxnew
- profinet
- powerlink
- ethercat
- vlan8021q

Ethertype custom value

Specifies the *Ethertype* value of the data packets in a decimal notation, which the Layer 2 packet filter uses. The prerequisite is that in the *Ethertype* column the value *custom* is specified.

Possible values:

1536 . 65535 ($2^1 - 1$) (default setting: 0)

Protocol number

Specifies the protocol number for the user-defined protocol which the IPv4 header uses. The prerequisite is that in the *Protocol type* column a value other than *ether net* is specified.

Possible values:

any (default setting)

0 . 255

Port

Specifies the destination port that the device evaluates in the data packet. The prerequisite is that in the *Protocol type* column the value *TCP* or *UDP* is specified.

Possible values:

any (default setting)

The device applies the rule to every data packet without evaluating the destination port.

1 . 65535 ($2^1 - 1$)

The device applies the rule only to data packets containing the specified destination port.

The field lets you specify the following options:

- You specify a port with a single numerical value, for example 21.
 - You specify multiple individual ports with numerical values separated by commas, for example 21, 80, 110.
 - You specify a port range with numerical values connected by dashes, for example 2000-3000.
 - You can also combine ports and port ranges, for example 21, 2000-3000, 65535.
- The field lets you specify up to 15 numerical values. When you enter 21, 2000-3000, 65535, for example, you use 4 of 15 numerical values.

4.5 Packet Filter

[Network Security > Packet Filter]

In this menu, you specify the settings for the *Packet Filter* functions.

The menu contains the following dialogs:

- [Routed Firewall Mode](#)
- [Transparent Firewall Mode](#)

4.5.1 Routed Firewall Mode

[Network Security > Packet Filter > Routed Firewall Mode]

In this menu, you specify the settings for the *Routed Firewall Mode* packet filter.

The *Routed Firewall Mode* packet filter contains rules which the device applies successively to the data stream on its router interfaces. The *Routed Firewall Mode* packet filter evaluates the data stream statefully and filters undesired data packets selectively. The device evaluates the status of the connection, and also determines if the data packets belong to a specific connection (*Stateful Packet Inspection*).

If a data packet matches the criteria of one or more rules, then the device applies the action specified in the first applicable rule to the data stream. The device ignores the rules that follow the first applicable rule.

If no rule matches, then the device applies the default rule. In the default setting, the default rule has the value *accept*. The device lets you change the default rule in the *Network Security > Packet Filter > Routed Firewall Mode > Global* dialog.

The device provides a multi-step approach to set up and apply the *Packet Filter* rules:

- You add a rule.
- You assign the rule to a router interface.
Up to this step, changes have no effect on the behavior of the device and the data stream.
- You apply the rule to the data stream.

The data packets go through the filter functions of the device in the following sequence:

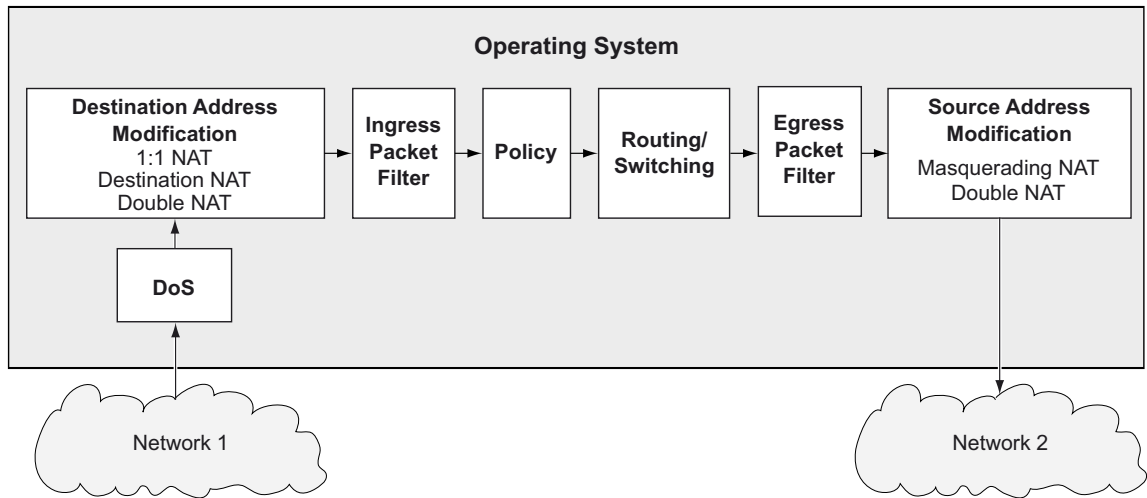


Figure 1: Processing sequence of the data packets in the device

The menu contains the following dialogs:

- [Global](#)
- [Firewall Learning Mode](#)
- [Packet Filter Rule](#)
- [Packet Filter Assignment](#)
- [Packet Filter Overview](#)

4.5.1.1 Global

[Network Security > Packet Filter > Routed Firewall Mode > Global]

In this dialog, you specify the global settings for the *Routed Firewall Mode* packet filter.

Configuration

Buttons

 Commit changes

Applies the rules saved in the device to the data stream.

In the process, the device also removes the state information from the packet filter. This includes potential *DCE RPC* information of the *OPC Enforcer* function. In the process, the device interrupts open communication connections.

Note: While the device is activating the saved rules, you cannot establish any new communication connections.

Allowed rules for L3 firewalling (max.)

Displays the maximum number of allowed firewall rules for data packets.

Default policy

Specifies how the firewall processes data packets if no rule applies.

Possible values:

[accept](#) (default setting)

The device accepts the data packets.

[drop](#)

The device discards the data packets.

[reject](#)

The device discards the data packet and sends an *ICMP Admin Prohibited* message to the sender.

Validate checksum

Specifies how the firewall handles *connection tracking* on the basis of data packet checksum.

Possible values:

[marked](#) (default setting)

The device evaluates the *checksum* in the data packet. If the value is invalid, then the device drops the data packet.

[unmarked](#)

The device ignores the *checksum*. The device forwards the data packet even if the value is invalid.


Information

Uncommitted changes present

Displays if the *Packet Filter* rules applied to the data stream differ from the *Packet Filter* rules saved in the device.

Possible values:

marked

At least one of the *Packet Filter* rules saved in the device contains modified settings. When you click the  button, the device applies the *Packet Filter* rules to the data stream.

unmarked

The device applies the saved *Packet Filter* rules to the data stream.

4.5.1.2 Firewall Learning Mode

[Network Security > Packet Filter > Routed Firewall Mode > FLM]

This dialog lets you specify the connections which you allow to have access to the network.

The maximum number of rules that you can specify using the *FLM* function depends on the number of rules already set up in the *Network Security > Packet Filter > Routed Firewall Mode > Rule* dialog. The device lets you specify up to 2048 rules.

The *FLM* function only applies to packets that pass through the device matching the *FORWARD* chain. The *FLM* function does not apply to the packets that the device receives on the *INPUT* chain and to those that the device generates on the *OUTPUT* chain. During the learning phase the device retains SSH, SNMP, and GUI access.

The *FLM* function requires you to set up and select at least 2 router interfaces in the device.

The maximum number of connections that the *FLM* function can learn is 65535.

Note: During the learning phase the network is temporarily exposed, because the *FLM* function sets up rules to accept every data packet on the selected ports.

Note: If you enable the *VRRP* function on a router interface, then the *FLM* function is ineffective on this router interface.

The dialog contains the following tabs:

[Configuration]

[Rules]

[Configuration]

The tab lets you enable the *FLM* function. The device monitors up to 4 interfaces to discover what type of data packets the device forwards through the interfaces into the network.

Operation

Operation

Enables/disables the *FLM* function.

Possible values:

On

The *FLM* function is enabled.

Off (default setting)

The *FLM* function is disabled.

Information

Buttons

 Start

Starts the learning phase. The device filters the data packets on the active interfaces.

 Stop

Stops the learning phase.

 Clear

Clears the memory. Learned data can be cleared only when the *FLM* function is stopped.

Status

Displays the state of the running *FLM* application.

Possible values:

off

The function is inactive.

stopped-data-not present

stopped-data-present

The device stopped the learning mode. Check the *Rule* tab for learned data.

learning

The device is learning data.

pending

The device is busy processing learned data.

Information

Displays the status of *FLM* application memory.

Interfaces selected for learning

Displays the interfaces that the *FLM* function actively monitors. The maximum number of interfaces that the device monitors is 4.

Additional information

Displays a special status message.

Learned entries

Displays the number of Layer 3 entries in the connection table.

Free memory for learning data [%]

Displays the percentage of free memory available for learning data.

[Rules]

This tab displays the type of data that is traversing the selected ports. You can add rules to manage the data stream traversing the device. Using the data displayed in the [Learned entries](#) table you can accept or reject data as required.

The tab is active after the device forwards one data packet and the [FLM](#) function is disabled again.

Learned entries table

Buttons



Add

Opens the [Create](#) window to add a rule when the [Learned entries](#) table displays at least one table row. The [Packet filter rules](#) table displays the added rule:

- In the [Description](#) field, you specify a name for the rule.
- In the [Source address](#) field, you specify the source address of the data packets.
- In the [Destination address](#) field, you specify the destination address of the data packets.
- From the [Protocol](#) drop-down list, you select the protocol type of the data packets.
- In the [Destination port](#) field, you specify the destination port of the data packets.
- In the [Ingress interface](#) field, you specify if the device applies the rule to data packets received or sent on a router interface.

Source address

Displays the source address of the packets.

Destination address

Displays the destination address of the packet.

Protocol

Displays the IP protocol, based on RFC 791, for protocol filtering.

Destination port

Displays the destination port of the packet.

Ingress interface

Displays the interface that received the packet.

Egress interface

Displays the interface that sent the packet.

First occurrence

Displays the first time that the device has determined the packet.

Connections by Rule Set

Displays the number of connections that match the rules set in the table below.

Connections by Selection

Displays the number of connections that match the selections in the table below.

Packet filter rules table

Buttons



Removes the selected table row.



Opens the [Edit](#) window to edit the parameters of the selected table row.

Rule index

Displays the sequential number of the [Packet Filter](#) rule.

Description

Specifies a name for the rule.

Possible values:

Alphanumeric ASCII character string with 0..32 characters

Source address

Specifies the source address of the data packets to which the device applies the rule.

Possible values:

[any](#) (default setting)

The device applies the [Packet Filter](#) rule to data packets with any source address.

Valid IPv4 address

The device applies the rule to data packets with the specified source address.

Valid IPv4 address and netmask in CIDR notation

The device applies the rule to data packets with the specified source address in the specified subnet.

Destination address

Specifies the destination address of the data packets to which the device applies the rule.

Possible values:

[any](#) (default setting)

The device applies the [Packet Filter](#) rule to data packets with any destination address.

Valid IPv4 address

The device applies the rule to data packets with the specified destination address.

Valid IPv4 address and netmask in CIDR notation

The device applies the rule to data packets with the specified destination address in the specified subnet.

Protocol

Specifies the protocol type of the data packets to which the device applies the rule. The device applies the rule only to data packets that contain the specified value in the *Protocol* field.

Possible values:

[any](#) (default setting)

The device applies the rule to every data packet without evaluating the protocol.

[i cnp](#)

Internet Control Message Protocol (RFC 792)

[i gmp](#)

Internet Group Management Protocol

[i pi p](#)

IP in IP tunneling (RFC 2003)

[t cp](#)

Transmission Control Protocol (RFC 793)

[udp](#)

User Datagram Protocol (RFC 768)

[esp](#)

IPsec Encapsulated Security Payload (RFC 2406)

[ah](#)

IPsec Authentication Header (RFC 2402)

[i cnpv6](#)

Internet Control Message Protocol for IPv6

Destination port

Specifies the destination port of the data packets to which the device applies the rule. The prerequisite is that in the *Protocol* column the value [TCP](#) or [UDP](#) is specified.

Possible values:

[any](#) (default setting)

The device applies the *Packet Filter* rule to every data packet without evaluating the destination port.

[1..65535 \(2¹⁶ - 1\)](#)

The device applies the *Packet Filter* rule only to data packets containing the specified destination port.

The field lets you specify the following options:

- You specify a port with a single numerical value, for example [21](#).
- You specify multiple individual ports with numerical values separated by commas, for example [21, 80, 110](#).
- You specify a port range with numerical values connected by dashes, for example [2000-3000](#).
- You can also combine ports and port ranges, for example [21, 2000-3000, 65535](#).
The field lets you specify up to 15 numerical values. When you enter [21, 2000-3000, 65535](#), for example, you use 4 of 15 numerical values.

Action

Specifies how the device handles received data packets when the device applies the rule.

Possible values:

`accept` (default setting)

The device accepts the data packets according to the ingress rules. Afterwards, the device applies the egress rules before the port sends the data packets.

`drop`

The device discards the data packet without informing the sender.

`reject`

The device discards the data packet and informs the sender.

`enforce-modbus`

The device applies the rule specified in the *DPI profile index* column to the data packets.

`enforce-opc`

The device applies the rule specified in the *DPI profile index* column to the data packets.

`enforce-dnp3`

The device applies the rule specified in the *DPI profile index* column to the data packets.

`enforce-iec104`

The device applies the rule specified in the *DPI profile index* column to the data packets.

`enforce-ethernetip`

The device applies the rule specified in the *DPI profile index* column to the data packets.

Ingress interface

Displays if the device applies the *Packet Filter* rule to data packets received or sent on a router interface.

Possible values:

`ingress`

The device applies the *Packet Filter* rule to data packets received on the router interface.

`egress`

The device applies the *Packet Filter* rule to data packets sent on the router interface.

Active

Activates/deactivates the rule.

Possible values:

`marked`

The rule is active.

`unmarked` (default setting)

The rule is inactive.

4.5.1.3 Packet Filter Rule

[Network Security > Packet Filter > Routed Firewall Mode > Rule]

This dialog lets you set up rules for the packet filter. You assign the rules specified here to the desired router interface in the [Network Security > Packet Filter > Routed Firewall Mode > Assignment](#) dialog.

Table

For information on how to customize the appearance of the table, see [“Working with tables” on page 16](#).

Buttons



Add

Adds a table row.



Remove

Removes the selected table row.

Rule index

Displays the sequential number of the [Packet Filter](#) rule. The device automatically assigns the value when you add a table row.

Description

Specifies a name for the rule.

Possible values:

Alphanumeric ASCII character string with 0..32 characters

Source address

Specifies the asset name or the source address of the data packets to which the device applies the rule. Select an item from the drop-down list or specify the source address. You specify the asset name in the [Network Security > Asset](#) dialog.

Possible values:

[any](#) (default setting)

The device applies the rule to data packets with any asset name or source address.

Valid IPv4 address

The device applies the rule to data packets with the specified source address.

Valid IPv4 address and netmask in CIDR notation

The device applies the rule to data packets with the specified source address in the specified subnet.

Example: [192.168.112.0/25](#)

An exclamation mark (!) preceding the IP address reverses the expression into its opposite. The device applies the rule to data packets with any source address or subnet excluding the specified source address or the subnet.

Example: ! 1. 1. 1. 1 or ! 192. 168. 112. 0/25

Name of the asset

Alphanumeric ASCII character string with 1..32 characters

Destination address

Specifies the asset name or the destination address of the data packets to which the device applies the rule. Select an item from the drop-down list or specify the destination address. You specify the asset name in the [Network Security > Asset](#) dialog.

Possible values:

[any](#) (default setting)

The device applies the rule to data packets with any asset name or destination address.

Valid IPv4 address

The device applies the rule to data packets with the specified destination address.

Valid IPv4 address and netmask in CIDR notation

The device applies the rule to data packets with the specified destination address in the specified subnet.

Example: 192. 168. 112. 0/25

An exclamation mark (!) preceding the IP address reverses the expression into its opposite. The device applies the rule to data packets with any destination address or subnet excluding the specified destination address or the subnet.

Example: ! 1. 1. 1. 1 or ! 192. 168. 112. 0/25

Name of the asset

Alphanumeric ASCII character string with 1..32 characters

Protocol

Specifies the IP protocol or Layer 4 protocol type of the data packets to which the device applies the rule. The device applies the rule only to data packets that contain the specified value in the *Protocol* field.

Possible values:

[any](#) (default setting)

The device applies the rule to every data packet without evaluating the protocol.

[i cnp](#)

Internet Control Message Protocol (RFC 792)

[i grp](#)

Internet Group Management Protocol

[i pi p](#)

IP in IP tunneling (RFC 2003)

[t cp](#)

Transmission Control Protocol (RFC 793)

[udp](#)

User Datagram Protocol (RFC 768)

[esp](#)

IPsec Encapsulated Security Payload (RFC 2406)

[ah](#)

IPsec Authentication Header (RFC 2402)

i cnpv6

Internet Control Message Protocol for IPv6 (RFC 4443)

<user-defined protocols>

The device also processes user-defined protocols. You specify user-defined protocols in the *Network Security > Protocol* dialog.

Source port

Specifies the L4 source port of the data packets to which the device applies the rule. The prerequisite is that in the *Protocol* column the value **tcp** or **udp** is specified.

Possible values:

any (default setting)

The device applies the *Packet Filter* rule to every data packet without evaluating the L4 source port.

1..65535 (2¹⁶ - 1)

The device applies the *Packet Filter* rule only to data packets containing the specified L4 source port.

The field lets you specify the following options:

- You specify a port with a single numerical value, for example **21**.
- You specify multiple individual ports with numerical values separated by commas, for example **21, 80, 110**.
- You specify a port range with numerical values connected by dashes, for example **2000-3000**.
- You can also combine ports and port ranges, for example **21, 2000-3000, 65535**.
The field lets you specify up to 15 numerical values. When you enter **21, 2000-3000, 65535**, for example, you use 4 of 15 numerical values.

Destination port

Specifies the L4 destination port of the data packets to which the device applies the rule. The prerequisite is that in the *Protocol* column the value **tcp** or **udp** is specified.

Possible values:

any (default setting)

The device applies the *Packet Filter* rule to every data packet without evaluating the L4 destination port.

1..65535 (2¹⁶ - 1)

The device applies the *Packet Filter* rule only to data packets containing the specified L4 destination port.

The field lets you specify the following options:

- You specify a port with a single numerical value, for example **21**.
- You specify multiple individual ports with numerical values separated by commas, for example **21, 80, 110**.
- You specify a port range with numerical values connected by dashes, for example **2000-3000**.
- You can also combine ports and port ranges, for example **21, 2000-3000, 65535**.
The field lets you specify up to 15 numerical values. When you enter **21, 2000-3000, 65535**, for example, you use 4 of 15 numerical values.

Parameters

Specifies additional parameters for this rule.

Enter parameters in the form **<param>=<val >**. If you enter multiple parameters, then separate them using a comma. If you enter multiple values, then separate them using a vertical bar.

Some parameters are valid when you use a specific protocol. Exception: the value `mac` is valid independently of the protocol. You also have the option of entering a combination of valid rules and protocol-specific rules.

Possible values:

`none` (default setting)

You have not specified any additional parameters for this rule.

`mac=de: ad: de: ad: be: ef`

This rule applies to packets with the source MAC address `de: ad: de: ad: be: ef`.

`type=<0 . 255>`

This rule applies to packets with a specific ICMP type. Enter exactly one value (for the meaning of these values see RFC 792).

`code=<0 . 255>`

This rule applies to packets with a specific ICMP code. Enter exactly one value (for the meaning of these values see RFC 792).

`frags=<true| false>`

When `true`, this rule applies to fragmented packets for which you set specific rules.

`flags=<syn| ack| fin>`

This rule applies to packets for which you set specific flags.

`flags=syn`

This rule applies to packets for which you set the `syn` flag.

`flags=syn| ack| fin`

This rule applies to packets for which you set the `syn`, `ack`, or `fin` flag.

`mac=de: ad: de: ad: be: ef, state=new|rel, flags=syn`

This rule applies to packets that come from the `de: ad: de: ad: be: ef` MAC address, are in a new or relative connection, and for which you set the `syn` flag.

Action

Specifies how the device processes received data packets when the device applies the rule.

Possible values:

`accept` (default setting)

The device accepts the data packets according to the ingress rules. Afterwards, the device applies the egress rules before the port sends the data packets.

`drop`

The device discards the data packet without informing the sender.

`reject`

The device discards the data packet and informs the sender.

`enforce-modbus`

The device applies the rule specified in the *DPI profile index* column to the data packets. The prerequisite is that in the *Source address*, *Destination address* and *Destination port* columns a value other than `any` is specified.

The value is only available in the software level IN/SU/UN. Refer to the *Software level* characteristic value in the product code.

`enforce-opc`

The device applies the rule specified in the *DPI profile index* column to the data packets. The prerequisite is that in the *Source address*, *Destination address* and *Destination port* columns a value other than `any` is specified.

The value is only available in the software level IN/UN. Refer to the *Software level* characteristic value in the product code.

enforce-dnp3

The device applies the rule specified in the *DPI profile index* column to the data packets. The prerequisite is that in the *Source address*, *Destination address* and *Destination port* columns a value other than *any* is specified.

The value is only available in the software level SU/UN. Refer to the *Software level* characteristic value in the product code.

enforce-iec104

The device applies the rule specified in the *DPI profile index* column to the data packets. The prerequisite is that in the *Source address*, *Destination address* and *Destination port* columns a value other than *any* is specified.

The value is only available in the software level SU/UN. Refer to the *Software level* characteristic value in the product code.

enforce-ethernetip

The device applies the rule specified in the *DPI profile index* column to the data packets. The prerequisite is that in the *Source address*, *Destination address* and *Destination port* columns a value other than *any* is specified.

The value is only available in the software level IN/UN. Refer to the *Software level* characteristic value in the product code.

Log

Activates/deactivates the logging in the log file.

Possible values:

marked

Logging is active.

When the device applies the *Packet Filter* rule to a data packet, the device places an entry in the log file. See the *Diagnostics > Report > System Log* dialog.

unmarked (default setting)

Logging is inactive.

Trap

Activates/deactivates the sending of SNMP traps when the device applies a *Packet Filter* rule to a data packet.

Possible values:

marked

The sending of SNMP traps is active. The prerequisite is that in the *Diagnostics > Status Configuration > Alarms (Traps)* dialog the *Alarms (Traps)* function is enabled and at least one trap destination is specified.

If the device applies the *Packet Filter* rule to a data packet, then the device sends an SNMP trap.

unmarked (default setting)

The sending of SNMP traps is inactive.

DPI profile index

Specifies which rule the device applies to the data packets.

The prerequisite is that in the *Action* column one of the following values is specified:

- **enforce-modbus**
- **enforce-opc**
- **enforce-dnp3**
- **enforce-iec104**
- **enforce-ethernetip**

Possible values:

0 (default setting)

The device does not apply any rule to the data packets.


1..32

The device applies the rule with the specified Index number to the data packets.

Active

Activates/deactivates the rule.

To apply the settings to the data stream, perform the following steps:

Click the  button to save the current settings.

Open the [Network Security > Packet Filter > Routed Firewall Mode > Global](#) dialog, or the [Network Security > Packet Filter > Routed Firewall Mode > Assignment](#) dialog.

Click the  button.

Possible values:

marked

The rule is active.

unmarked (default setting)

The rule is inactive.

4.5.1.4 Packet Filter Assignment

[Network Security > Packet Filter > Routed Firewall Mode > Assignment]

This dialog lets you assign one or more *Packet Filter* rules to the router interfaces of the device. You set up router interfaces in the *Routing > Interfaces > Configuration* dialog.

Information

Assignments


Displays how many rules are active for the ports.

Uncommitted changes present

Displays if the *Packet Filter* rules applied to the data stream differ from the *Packet Filter* rules saved in the device.

Possible values:

marked

At least one of the *Packet Filter* rules saved in the device contains modified settings. When you click the  button, the device applies the *Packet Filter* rules to the data stream.

unmarked

The device applies the saved *Packet Filter* rules to the data stream.

Table

For information on how to customize the appearance of the table, see “[Working with tables](#)” on [page 16](#).

Buttons



Add

Opens the *Create* window to assign a rule to a router interface.

- From the *Rule index* drop-down list, you select the rule that you assign to the router interface.
- From the *Direction* drop-down list, you select if the device applies the rule to received or sent data packets or to both.
- From the *Interface* drop-down list, you select the router interface on which the device applies the rule.



Remove

Removes the selected table row.



Commit changes

Applies the rules saved in the device to the data stream.

In the process, the device also removes the state information from the packet filter. This includes potential *DCE RPC* information of the *OPC Enforcer* function. In the process, the device interrupts open communication connections.

Note: While the device is activating the saved rules, you cannot establish any new communication connections.

Description

Displays the name of the rule. You specify the description in the [Network Security > Packet Filter > Routed Firewall Mode > Rule](#) dialog.

Rule index

Displays the sequential number of the [Packet Filter](#) rule. You specify the rule index when you add a table row.

Interface

Displays the router interface on which the device applies the rule. You specify the interface number when you add a table row.

Direction

Displays if the device applies the [Packet Filter](#) rule to received or sent data packets or to both.

Possible values:

[i ngress](#)

The device applies the [Packet Filter](#) rule to data packets received on the router interface.

[e gress](#)

The device applies the [Packet Filter](#) rule to data packets sent on the router interface.

[bot h](#)

The device applies the [Packet Filter](#) rule to data packets sent and received on the router interface.

Priority

Specifies the priority of the [Packet Filter](#) rule.

Using the priority, you specify the sequence in which the device applies the rules to the data stream. The device applies the rules in ascending order which starts with priority 0.


Possible values:

0 . 4294967295 ($2^{32} - 1$) (default setting: 1)

Active

Activates/deactivates the rule.

To apply the settings to the data stream, perform the following steps:

Click the  button to save the current settings.

Open the [Network Security > Packet Filter > Routed Firewall Mode > Global](#) dialog, or the [Network Security > Packet Filter > Routed Firewall Mode > Assignment](#) dialog.

Click the  button.

Possible values:

[marked](#)

The rule is active.

[unmarked](#) (default setting)

The rule is inactive.

4.5.1.5 Packet Filter Overview

[Network Security > Packet Filter > Routed Firewall Mode > Overview]

This dialog gives you an overview of the specified *Packet Filter* rules.

Table

For information on how to customize the appearance of the table, see “Working with tables” on page 16.

Description

Displays the name of the rule. You specify the description in the *Network Security > Packet Filter > Routed Firewall Mode > Rule* dialog.

Rule index

Displays the sequential number of the *Packet Filter* rule.

Interface

Displays the router interface on which the device applies the rule.

Direction

Displays if the device applies the *Packet Filter* rule to received or sent data packets or to both.

Possible values:

ingress

The device applies the *Packet Filter* rule to data packets received on the router interface.

egress

The device applies the *Packet Filter* rule to data packets sent on the router interface.

both

The device applies the *Packet Filter* rule to data packets sent and received on the router interface.

Priority

Displays the priority of the *Packet Filter* rule. The device applies the rules in ascending order which starts with priority 0.

Source address

Displays the asset name or the source address of the data packets to which the device applies the rule.

Source port

Displays the source TCP or UDP port of the data packets to which the device applies the rule.

Destination address

Displays the asset name or the destination address of the data packets to which the device applies the rule.

Destination port

Displays the destination TCP or UDP port of the data packets to which the device applies the rule.

Protocol

Displays the IP protocol to which the *Packet Filter* rule is restricted. The device applies the *Packet Filter* rule only to data packets with the specified IP protocol.

Parameters

Displays additional parameters for this rule.

Action

Displays how the device processes received data packets when the device applies the rule.

DPI profile index

Displays the profile index of the *DPI enforcer* function. You specify the profile index in the *Network Security > Packet Filter > Routed Firewall Mode > Rule* dialog.

Log

Displays if the device places an entry in the log file when the device applies the rule to a data packet.

Trap

Displays if the device sends an SNMP trap when the device applies the rule to a data packet.

4.5.2 Transparent Firewall Mode

[Network Security > Packet Filter > Transparent Firewall Mode]

In this menu, you specify the settings for the *Transparent Firewall Mode* packet filter. The *Transparent Firewall Mode* packet filter contains rules which the device applies successively to the data stream on its non-routing ports or VLAN interfaces. The *Transparent Firewall Mode* packet filter evaluates every data packet that passes through the firewall based on the connection status as mentioned below:

- For IPv4, evaluation is *stateful*.
- For other Layer 2 and Layer 3 protocols, evaluation is *stateless*.

The device filters the undesired data packets selectively while the connection is unknown.

- If a data packet matches the criteria of one or more rules, then the device applies the action specified in the first applicable rule to the data stream. The device ignores the rules that follow the first applicable rule.
- If no rule matches, then the device applies the default rule. In the default setting, the default rule has the value `accept`. The device lets you change the default rule in the [Network Security > Packet Filter > Transparent Firewall Mode > Global](#) dialog.

The device provides a multi-step approach to set up and apply the `Packet Filter` rules:

- You add a rule.
- You assign the rule to a non-routing port or VLAN.
Up to this step, changes have no effect on the behavior of the device and the data stream.
- You apply the rule to the data stream.

The device processes data packets in the following sequence:

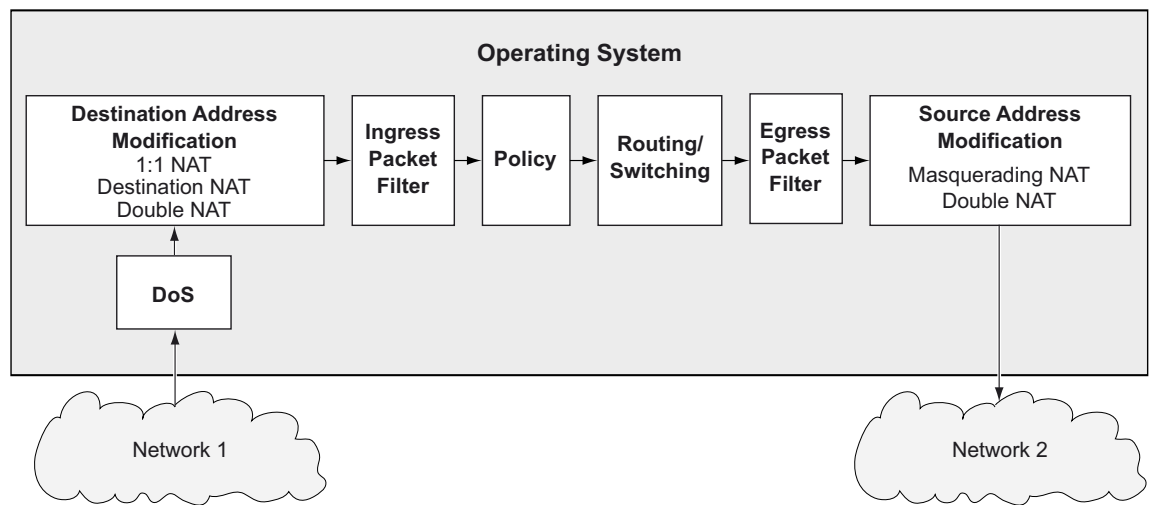


Figure 2: Processing sequence of the data packets in the device

The menu contains the following dialogs:

- [Packet Filter Global](#)
- [Packet Filter Rule](#)
- [Packet Filter Assignment](#)
- [Packet Filter Overview](#)

4.5.21 Packet Filter Global

[Network Security > Packet Filter > Transparent Firewall Mode > Global]

In this dialog, you specify the global settings for the *Transparent Firewall Mode* packet filter.

Configuration

Buttons

 Commit changes

Applies the rules saved in the device to the data stream.

Note: While the device is activating the saved rules, you cannot establish any new communication connections.

Allowed rules for L2 firewalling (max.)

Displays the maximum number of allowed firewall rules for data packets.

Default policy

Specifies how the firewall processes data packets if no rule applies.

Possible values:

accept (default setting)

The device accepts the data packets.

drop

The device discards the data packets.

In further progress, note when you assign any rule to a port or VLAN interface: The device accepts ARP packets implicitly, regardless of the data packet type.

Validate FCS

Specifies if the firewall evaluates the *Frame Check Sequence* of data packets.

Possible values:

marked (default setting)

The device evaluates the *Frame Check Sequence* in the data packet. If the value is invalid, then the device drops the data packet.

unmarked

The device ignores the *Frame Check Sequence*. The device forwards the data packet even if the value is invalid.


Information

Uncommitted changes present

Displays if the *Packet Filter* rules applied to the data stream differ from the *Packet Filter* rules saved in the device.

Possible values:

marked

At least one of the *Packet Filter* rules saved in the device contains modified settings. When you click the  button, the device applies the *Packet Filter* rules to the data stream.

unmarked

The device applies the saved *Packet Filter* rules to the data stream.

4.5.2.2 Packet Filter Rule

[Network Security > Packet Filter > Transparent Firewall Mode > Rule]

This dialog lets you set up rules for the packet filter. You assign the rules specified here to the desired non-routing ports or VLANs in the [Network Security > Packet Filter > Transparent Firewall Mode > Assignment](#) dialog.

Table

For information on how to customize the appearance of the table, see [“Working with tables” on page 16](#).

Buttons



Add

Adds a table row.



Remove

Removes the selected table row.

Index

Displays the sequential number of the [Packet Filter](#) rule. The device automatically assigns the value when you add a table row.

Description

Specifies a name for the rule.

Possible values:

Alphanumeric ASCII character string with 0..32 characters

Action

Specifies how the device processes received data packets when the device applies the rule.

Possible values:

[accept](#) (default setting)

The device accepts the data packets according to the ingress rules. Afterwards, the device applies the egress rules before the port sends the data packets.

[drop](#)

The device discards the data packet without informing the sender.

[enforce-modbus](#)

The device applies the rule specified in the [DPI profile index](#) column to the data packets. The prerequisite is that in the [Source IP address](#), [Destination IP address](#) and [Destination port](#) columns a value other than [any](#) is specified.

The value is only available in the software level IN/SU/UN. Refer to the [Software level](#) characteristic value in the product code.

[enforce-opc](#)

The device applies the rule specified in the *DPI profile index* column to the data packets. The prerequisite is that in the *Source IP address*, *Destination IP address* and *Destination port* columns a value other than *any* is specified.

The value is only available in the software level IN/UN. Refer to the *Software level* characteristic value in the product code.

[enforce-iec104](#)

The device applies the rule specified in the *DPI profile index* column to the data packets. The prerequisite is that in the *Source IP address*, *Destination IP address* and *Destination port* columns a value other than *any* is specified.

The value is only available in the software level SU/UN. Refer to the *Software level* characteristic value in the product code.

[enforce-dnp3](#)

The device applies the rule specified in the *DPI profile index* column to the data packets. The prerequisite is that in the *Source IP address*, *Destination IP address* and *Destination port* columns a value other than *any* is specified.

The value is only available in the software level SU/UN. Refer to the *Software level* characteristic value in the product code.

[enforce-ethernetip](#)

The device applies the rule specified in the *DPI profile index* column to the data packets. The prerequisite is that in the *Source IP address*, *Destination IP address* and *Destination port* columns a value other than *any* is specified.

The value is only available in the software level IN/UN. Refer to the *Software level* characteristic value in the product code.

[enforce-amp](#)

The device applies the rule specified in the *DPI profile index* column to the data packets. The prerequisite is that in the *Source IP address*, *Destination IP address* and *Destination port* columns a value other than *any* is specified.

The value is only available in the software level IN/UN. Refer to the *Software level* characteristic value in the product code.

Source MAC address

Specifies the asset name or the source address of the MAC data packets to which the device applies the rule. Select an item from the drop-down list or specify the source address. You specify the asset name in the *Network Security > Asset* dialog.

Possible values:

[any](#) (default setting)

The device applies the rule to MAC data packets with any asset name or source address.

Valid MAC address

The device applies the rule to MAC data packets with the specified source address.

Example: 00:11:22:33:44:55

Name of the asset

Alphanumeric ASCII character string with 1..32 characters

Destination MAC address

Specifies the asset name or the destination address of the MAC data packets to which the device applies the rule. Select an item from the drop-down list or specify the destination address. You specify the asset name in the *Network Security > Asset* dialog.

Possible values:

[any](#) (default setting)

The device applies the rule to MAC data packets with any asset name or destination address.

Valid MAC address

The device applies the rule to MAC data packets with the specified destination address.

Example: 00: 11: 22: 33: 44: 55

Name of the asset

Alphanumeric ASCII character string with 1..32 characters

Ethertype

Specifies the *Ethertype* keyword of the MAC data packets to which the device applies the rule.

Possible values:

`custom` (default setting)

The device applies the value specified in the *Ethertype custom value* column.

`appletalk`

`arp`

`ibnsna`

`ipv4`

`ipv6`

`ipxold`

`mplsncast`

`mplsucast`

`netbios`

`novell`

`pppoedisc`

`rarp`

`pppoessess`

`ipxnew`

`profinet`

`powerlink`

`ethercat`

`vlan8021q`

Ethertype custom value

Specifies the *Ethertype* value of the MAC data packets to which the device applies the rule. The prerequisite is that in the *Ethertype* column the value `custom` is specified.

Possible values:

`0` (default setting)

The device applies the rule to every MAC data packet without evaluating the *Ethertype* value.

`1..5ff`

The device applies the rule to Logical Link Control (LLC) data packets whose length field contains the specified value. These values are available only for port-based rules.

`600..ffff`

The device applies the rule only to MAC data packets that contain the *Ethertype* value specified here.

VLAN ID

Specifies the VLAN ID of the data packets to which the device applies the rule. The prerequisite is that in the *Ethertype* column the value *vlan8021q* is specified.

Possible values:

any (default setting)

The device applies the rule to every data packet without evaluating the VLAN ID.

1..4042

The device applies the rule only to data packets containing the specified VLAN ID.

Source IP address

Specifies the asset name or the source address of the IP data packets to which the device applies the rule. Select an item from the drop-down list or specify the source address. You specify the asset name in the *Network Security > Asset* dialog.

Prerequisites:

- In the *Ethertype* column, the value *ipv4* is specified.
- In the *Action* column, a value other than *enf* or *ce-goose* is specified.

Possible values:

any (default setting)

The device applies the rule to IP data packets with any asset name or source address.

Valid IPv4 address and netmask in CIDR notation

The device applies the rule to data packets with the specified source address in the specified subnet.

Example: *192.168.112.0/25*

An exclamation mark (!) preceding the IP address reverses the expression into its opposite. The device applies the rule to data packets with any source address or subnet excluding the specified source address or the subnet.

Example: *!1.1.1.1* or *!192.168.112.0/25*

Name of the asset

Alphanumeric ASCII character string with 1..32 characters

Destination IP address

Specifies the asset name or the destination address of the IP data packets to which the device applies the rule. Select an item from the drop-down list or specify the destination address. You specify the asset name in the *Network Security > Asset* dialog.

Prerequisites:

- In the *Ethertype* column, the value *ipv4* is specified.
- In the *Action* column, a value other than *enf* or *ce-goose* is specified.

Possible values:

any (default setting)

The device applies the rule to IP data packets with any asset name or destination address.

Valid IPv4 address and netmask in CIDR notation

The device applies the rule to data packets with the specified destination address in the specified subnet.

Example: *192.168.112.0/25*

An exclamation mark (!) preceding the IP address reverses the expression into its opposite. The device applies the rule to data packets with any destination address or subnet excluding the specified destination address or the subnet.

Example: ! 1. 1. 1. 1 or ! 192. 168. 112. 0/25

Name of the asset

Alphanumeric ASCII character string with 1..32 characters

Protocol

Specifies the IP protocol or Layer 4 protocol type of the data packets to which the device applies the rule. The device applies the rule only to data packets that contain the specified value in the *Protocol* field.

Possible values:

[any](#) (default setting)

The device applies the rule to every data packet without evaluating the protocol.

[i cnp](#)

Internet Control Message Protocol (RFC 792)

[i gmp](#)

Internet Group Management Protocol

[i pi p](#)

IP in IP tunneling (RFC 2003)

[t cp](#)

Transmission Control Protocol (RFC 793)

[udp](#)

User Datagram Protocol (RFC 768)

[esp](#)

IPsec Encapsulated Security Payload (RFC 2406)

[ah](#)

IPsec Authentication Header (RFC 2402)

[i cnpv6](#)

Internet Control Message Protocol for IPv6

[<user-defined protocols>](#)

The device also processes user-defined protocols. You specify user-defined protocols in the [Network Security > Protocol](#) dialog.

TOS priority

Specifies the *IP Precedence (ToS)* value in the header of the IP data packets to which the device applies the rule.

Possible values:

[0](#) (default setting)

The device applies the rule to every IP data packet without evaluating the *ToS* value.

[1.. 255](#)

The device applies the rule only to IP data packets containing the specified *ToS* value.

DPI profile index

Specifies which rule the device applies to the data packets.

The prerequisite is that in the *Action* column one of the following values is specified:

- [enforce-modbus](#)
- [enforce-opc](#)

- `enforce-dnp3`
- `enforce-iec104`
- `enforce-amp`
- `enforce-ethernetip`

Possible values:

0 (default setting)

The device does not apply any rule to the data packets.

1..32

The device applies the rule with the specified Index number to the data packets.

Source port

Specifies the TCP or UDP source port of the data packets to which the device applies the rule.

Prerequisites:

- In the *Protocol* column, the value `tcp` or `udp` is specified.
- In the *Action* column, a value other than `enforce-geose` is specified.

Possible values:

`any` (default setting)

The device applies the rule to every data packet without evaluating the source port.

1..65535 (2¹⁶ - 1)

The device applies the rule only to data packets containing the specified source port.

The field lets you specify the following options:

- You specify a port with a single numerical value, for example `21`.
- You specify multiple individual ports with numerical values separated by commas, for example `21, 80, 110`.
- You specify a port range with numerical values connected by dashes, for example `2000-3000`.
- You can also combine ports and port ranges, for example `21, 2000-3000, 65535`.
The column lets you specify up to 15 numerical values. When you enter `21, 2000-3000, 65535`, for example, you use 4 of 15 numerical values.

Destination port

Specifies the TCP or UDP destination port of the data packets to which the device applies the rule.

Prerequisites:

- In the *Protocol* column, the value `tcp` or `udp` is specified.
- In the *Action* column, a value other than `enforce-geose` is specified.

Possible values:

`any` (default setting)

The device applies the rule to every data packet without evaluating the destination port.

1..65535 (2¹⁶ - 1)

The device applies the rule only to data packets containing the specified destination port.

The field lets you specify the following options:

- You specify a port with a single numerical value, for example `21`.
- You specify multiple individual ports with numerical values separated by commas, for example `21, 80, 110`.

- You specify a port range with numerical values connected by dashes, for example [2000-3000](#).
- You can also combine ports and port ranges, for example [21, 2000-3000, 65535](#).
The column lets you specify up to 15 numerical values. When you enter [21, 2000-3000, 65535](#), for example, you use 4 of 15 numerical values.

Rate limit

Specifies the data rate limit for the non-routing port or VLAN. The limit applies to the sum of the sizes of data packets sent and received.

Possible values:

[0](#) (default setting)

No limitation of the data transfer rate.

[1.. 10000000](#) (10)

If the data transfer rate on the port exceeds the value specified, then the device discards superfluous IP data packets. The prerequisite is that in the [Burst size](#) column a value >0 is specified. You specify the measurement unit of the limit in the [Unit](#) column.

Burst size

Specifies the limit in KByte for the data volume during temporary bursts.

Possible values:

[0](#) (default setting)

No limitation of the data volume.

[1.. 128](#)

If during temporary bursts on the port the data volume exceeds the value specified, then the device discards superfluous MAC data packets.

Recommendation:

- If the bandwidth is known:
 $Burst\ size = bandwidth \times allowed\ duration\ of\ a\ burst / 8$
- If the bandwidth is unknown:
 $Burst\ size = 10 \times MTU\ (Maximum\ Transmission\ Unit)\ of\ the\ port$

Unit

Specifies the measurement unit for the data transfer rate specified in the [Rate limit](#) column.

Possible values:

[pps](#) (default setting)

Data packets per second

[kbps](#)

kBytes per second

Trap

Activates/deactivates the sending of SNMP traps when the device applies a *Packet Filter* rule to a data packet.

Possible values:

marked

The sending of SNMP traps is active. The prerequisite is that in the *Diagnostics > Status Configuration > Alarms (Traps)* dialog the *Alarms (Traps)* function is enabled and at least one trap destination is specified.

If the device applies the *Packet Filter* rule to a data packet, then the device sends an SNMP trap.

unmarked (default setting)

The sending of SNMP traps is inactive.

Log

Activates/deactivates the logging in the log file.

Possible values:

marked

Logging is active.

When the device applies the *Packet Filter* rule to a data packet, the device places an entry in the log file. See the *Diagnostics > Report > System Log* dialog.


unmarked (default setting)

Logging is inactive.

Active

Activates/deactivates the rule.

To apply the settings to the data stream, perform the following steps:

Click the  button to save the current settings.

Open the *Network Security > Packet Filter > Transparent Firewall Mode > Global* dialog, or the *Network Security > Packet Filter > Transparent Firewall Mode > Assignment* dialog.

Click the  button.

Possible values:

marked

The rule is active.

unmarked (default setting)

The rule is inactive.

4.5.23 Packet Filter Assignment

[Network Security > Packet Filter > Transparent Firewall Mode > Assignment]

This dialog lets you assign one or more *Packet Filter* rules to the non-routing ports or VLANs.

Information

Assignments


Displays how many rules are active for the non-routing ports or VLANs.

Uncommitted changes present

Displays if the *Packet Filter* rules applied to the data stream differ from the *Packet Filter* rules saved in the device.

Possible values:

marked

At least one of the *Packet Filter* rules saved in the device contains modified settings. When you click the  button, the device applies the *Packet Filter* rules to the data stream.

unmarked

The device applies the saved *Packet Filter* rules to the data stream.

Table

For information on how to customize the appearance of the table, see [“Working with tables” on page 16](#).

Buttons



Add

Opens the *Create* window to assign a rule to a non-routing port or VLAN.

- From the *Port/VLAN* drop-down list, you select the non-routing port or the VLAN to which the device applies the rule. If you select the *VLAN 1* item from the drop-down list, the device applies the rule to all the ports associated with VLAN 1.
- From the *Direction* drop-down list, you select if the device applies the rule to received or sent data packets.
- From the *Index* drop-down list, you select the rule that you assign to the non-routing port or VLAN.



Remove

Removes the selected table row.



Commit changes

Applies the rules saved in the device to the data stream.

Note: While the device is activating the saved rules, you cannot establish any new communication connections.

Description

Displays the name of the rule. You specify the description in the [Network Security > Packet Filter > Transparent Firewall Mode > Rule](#) dialog.

Index

Displays the sequential number of the [Packet Filter](#) rule. You specify the index number when you add a table row.

Type

Displays where the device applies the rule to.

Possible values:


Port

The device already applies the [Packet Filter](#) rule to a non-routing port. You find the corresponding port number in the [Port/VLAN](#) column.

VLAN

The device already applies the [Packet Filter](#) rule to a non-routing VLAN interface. You find the corresponding VLAN ID in the [Port/VLAN](#) column.

Port/VLAN

Displays the number of the non-routing port or the VLAN to which the device applies the rule. To specify the port number or VLAN ID, click the  button.

Possible values:

<Port number >

Number of the non-routing port.

VLAN: <VLAN I D>

ID of the VLAN.

Direction

Displays if the device applies the *Packet Filter* rule to received or sent data packets.

Possible values:

ingress

The device applies the *Packet Filter* rule to data packets received on the non-routing port or VLAN interface.

egress

The device applies the *Packet Filter* rule to data packets sent on the non-routing port or VLAN interface.

Priority

Specifies the priority of the *Packet Filter* rule.

Using the priority, you specify the sequence in which the device applies the rules to the data stream. The device applies the rules in ascending order which starts with priority 0.


Possible values:

0 . 4294967295 ($2^{32} - 1$) (default setting: 1)

Active

Activates/deactivates the rule.

To apply the settings to the data stream, perform the following steps:

Click the  button to save the current settings.

Open the *Network Security > Packet Filter > Transparent Firewall Mode > Global* dialog, or the *Network Security > Packet Filter > Transparent Firewall Mode > Assignment* dialog.

Click the  button.

Possible values:

marked

The rule is active.

unmarked (default setting)

The rule is inactive.

4.5.24 Packet Filter Overview

[Network Security > Packet Filter > Transparent Firewall Mode > Overview]

This dialog gives you an overview of the specified *Packet Filter* rules.

Table

For information on how to customize the appearance of the table, see [“Working with tables” on page 16](#).

Description

Displays the name of the rule. You specify the description in the *Network Security > Packet Filter > Transparent Firewall Mode > Rule* dialog.

Index

Displays the sequential number of the *Packet Filter* rule.

Direction

Displays if the device applies the *Packet Filter* rule to received or sent data packets.

Possible values:

ingress

The device applies the *Packet Filter* rule to data packets received on the non-routing port or VLAN interface.

egress

The device applies the *Packet Filter* rule to data packets sent on the non-routing port or VLAN interface.

Priority

Displays the priority of the *Packet Filter* rule. The device applies the rules in ascending order which starts with priority 0.

Type

Displays where the device applies the rule to.

Port/VLAN

Displays the number of the non-routing port or the VLAN to which the device applies the rule.

Source MAC address

Displays the asset name or source address of the MAC data packets to which the device applies the rule.

Destination MAC address

Displays the asset name or destination address of the MAC data packets to which the device applies the rule.

Ethertype

Displays the *Ethertype* keyword of the MAC data packets to which the device applies the rule.

Ethertype custom value

Displays the *Ethertype* value of the MAC data packets to which the device applies the rule. The prerequisite is that in the *Ethertype* column the value *custom* is specified.

Source IP address

Displays the asset name or source address of the IP data packets to which the device applies the rule.

Destination IP address

Displays the asset name or destination address of the IP data packets to which the device applies the rule.

Protocol

Displays the IP protocol to which the *Packet Filter* rule is restricted. The device applies the *Packet Filter* rule only to data packets of the specified IP protocol.

TOS priority

Displays the *IP Precedence (ToS)* value in the header of the IP data packets to which the device applies the rule.

Action

Displays how the device processes received data packets when the device applies the rule.

DPI profile index

Displays the profile index of the *DPI enforcer* function. You specify the profile index in the [Network Security > Packet Filter > Transparent Firewall Mode > Rule](#) dialog.

Source port

Displays the source TCP or UDP port of the data packets to which the device applies the rule.

Destination port

Displays the destination TCP or UDP port of the data packets to which the device applies the rule.

Rate limit

Displays the data rate limit for the non-routing port or VLAN. The limit applies to the sum of the sizes of data packets sent and received.

Burst size

Displays the limit in KByte for the data volume during temporary bursts.

Unit

Displays the measurement unit for the data transfer rate specified in the *Rate limit* column.

Trap

Displays if the device sends an SNMP trap when the device applies the rule to a data packet.

Log

Displays if the device places an entry in the log file when the device applies the rule to a data packet.

Active

Displays if the rule is active or inactive.

4.6 Deep Packet Inspection

[Network Security > DPI]

The *DPI* function lets you monitor and filter data packets. The function supports you in protecting the network from undesirable content, such as spam or viruses.

The *DPI* function inspects data packets for undesirable characteristics and protocol violations. The protocol inspects the header and the payload of the data packets.

This dialog lets you specify the *DPI* settings. The device blocks the data packets that violate the specified profiles. If an error is detected, then the device terminates the data connection upon user request.

The menu contains the following dialogs:

- [Deep Packet Inspection - Modbus Enforcer](#)
- [Deep Packet Inspection - OPC Enforcer](#)
- [Deep Packet Inspection - DNP3 Enforcer](#)
- [Deep Packet Inspection - IEC104 Enforcer](#)
- [Deep Packet Inspection - AMP Enforcer](#)
- [Deep Packet Inspection - ENIP Enforcer](#)

4.6.1 Deep Packet Inspection - Modbus Enforcer

[Network Security > DPI > Modbus Enforcer]

This dialog lets you specify the *Modbus Enforcer* settings and define the *Modbus TCP* specific profiles.

The profiles specify *function codes* and register or coil addresses. The *function code* in the protocol Modbus TCP specifies the purpose of the data transfer. The device blocks the data packets that violate the specified profiles. If an error is detected, then the device terminates the data connection upon user request. The predefined *function code* lists and the *function code* generator support you when specifying the *function codes*.

When the *Modbus Enforcer* profile is active (checkbox in the *Profile active* column is marked), the device applies the profiles to the data stream.

- The device permits data packets containing only the *function codes* specified in the *Function code* column.
- The device rejects the data packets containing any other *function codes* that are not specified in the *Function code* column.

Information


Uncommitted changes present

Displays if the *Modbus Enforcer* profiles applied to the data stream differ from the profiles saved in the device.

Possible values:

marked

At least one of the active *Modbus Enforcer* profiles saved in the device contains modified settings.

When you click the  button, the device applies the specified profiles.

unmarked

The *Modbus Enforcer* profiles applied to the data stream match the profiles saved in the device.

Table

For information on how to customize the appearance of the table, see “Working with tables” on page 16.

Buttons



Add

Opens the *Create* window to add a table row.

- In the *Index* field, you specify the number of the profile.

Possible values:

1 . 32

When you click the *Ok* button, the device adds the table row. The device assigns the number specified in the *Index* field to the table row.



Remove

Removes the selected table row.

If you mark the *Profile active* checkbox for the profile, then the device stops you from removing the profile.



Copy

Opens the *Copy* window to copy an existing table row. The prerequisite is that the table row for the profile to be copied is selected.

- In the *Index* field, you specify a new number which identifies the copied profile.
 Possible values:
 1 . 32
 The device adds the table row. The device assigns the number specified in the *Index* field to the table row.



Commit changes

The device applies the specified profiles to the data stream.

If you changed the value in the *Function type* field, then the device applies the change to the *Function code* list and refreshes the display in the *Function code* column.

Index

Displays the sequential number of the profile to which the table row relates. You specify the index number when you add a table row.

Description

Specifies a name for the profile.

Possible values:

Alphanumeric ASCII character string with 0..64 characters
 (default setting: *modbus*)

Function type

Specifies the function type for the *Modbus Enforcer* profile. After clicking the button, the device assigns the corresponding *type IDs*.

Possible values:

readOnly (default setting)

Assigns the *function codes* for the *read* function of the *Modbus TCP* protocol.
 1, 2, 3, 4, 7, 11, 12, 17, 20, 24

readWrite

Assigns the *function codes* for the *read/write* functions of the *Modbus TCP* protocol.
 1, 2, 3, 4, 5, 6, 7, 11, 12, 15, 16, 17, 20, 21, 22, 23, 24

programming

Assigns the *function codes* for the *programming* functions of the *Modbus TCP* protocol.
 1, 2, 3, 4, 5, 6, 7, 11, 12, 15, 16, 17, 20, 21, 22, 23, 24, 40, 42, 90, 125, 126

all

Assigns the *function codes* for every function of the *Modbus TCP* protocol.

1, 2, . . . , 254, 255

advanced

Lets you specify user-defined values in the *Function code* column.

Note: If you have specified the value *advanced*, then for your own security the device does not allow any subsequent changes to be made to the value. The device helps prevent a change to *readOnly*, *readWrite* or *programming*. This helps avoid overwriting the manually specified values in the *Function code* column. To specify a table row with the value *readOnly*, *readWrite* or *programming*, add a table row.

Function code

Displays the *function codes* for the *Modbus Enforcer* profile. The device permits data packets with the specified properties.

The column displays different values depending on the value specified in the *Function type* column:

If in the *Function type* column the value *readOnly*, *readWrite* or *programming* is specified, then the device automatically enters the related *function codes*.

If in the *Function type* column the value *advanced* is specified, then the device lets you specify user-defined *function codes*. To do this, perform the following steps:

For the relevant profile, click into the *Function code* column.

The dialog displays the *Function code* window. See “[Function code]” on page 154.

From the *Function code* drop-down list, select the desired *function code* item.

Click the *Add* button.

To add multiple *function codes*, repeat the previously described steps.

Click the *Ok* button.

Possible values:

<FC> | <AR>, <FC> | <AR>, ...

The device lets you specify multiple *function codes* and for some *function codes* an additional address range. You find the meaning of the numbers in section “Meaning of the Function code values” on page 154.

– *Function code* <FC> = 1 . 255

You separate each *function code* with a comma, for example 1, 2, 3.

For some *function codes* the device lets you specify an additional address range. You separate the address range from the *function code* with a vertical bar (pipe), for example 1 | 128-255.

– *Address range* <AR> = 0 . 65535 or 0 . 65535 | 0 . 65535 (for *function codes* that require read and write address ranges)

You join the start value and end value of the range with a hyphen, for example 128-255.

The device also lets you specify a single value as an address range. For example, specifying the address range 5-5 is equivalent to the single address 5.

Unit identifier

Specifies the *Modbus TCP* identification unit for the *Modbus Enforcer* profile.

Possible values:

`none` (default setting)

The device permits data packets without an identification unit.

`0 . 255`

The device permits data packets with the specified identification unit.

The field lets you specify the following options:

- A single *Modbus TCP* identification unit with a single numerical value, for example `1`.
- Multiple *Modbus TCP* identification units with numerical values separated by a comma, for example `1, 2, 3`.

Sanity check

Activates/deactivates the plausibility check for the data packets.

Possible values:

`marked` (default setting)

The plausibility check is active.

The device checks the plausibility of the data packets regarding format and specification.

`unmarked`

The plausibility check is inactive.

Exception

Activates/deactivates the sending of an *exception* response in case of a protocol violation or if the plausibility check identifies errors.

Possible values:

`marked`

The sending of an *exception* response is active.

If the device identifies a protocol violation or a plausibility check error, then the device sends an *exception* response to the end points and terminates the *Modbus TCP* connection.

`unmarked` (default setting)

The sending of an *exception* response is inactive. The *Modbus TCP* connection remains established.

TCP reset

Activates/deactivates the resetting of the TCP connection in case of a protocol violation or if the plausibility check detects an error.

Possible values:

`marked` (default setting)

The resetting of the TCP connection is active.

If the device identifies a protocol violation or detects a plausibility check error, then the device terminates the TCP connection.

`unmarked`

The resetting of the TCP connection is inactive. The TCP connection remains established.

Profile active

Activates/deactivates the profile.

Possible values:

marked

The profile is active.

The device applies the *Modbus Enforcer* profiles specified in this table row to the data packets.

unmarked (default setting)

The profile is inactive.

[Function code]

Function code

Specifies the *function codes* for the relevant *Modbus Enforcer* profile.

You find the meaning of the numbers in section “[Meaning of the Function code values](#)” on [page 154](#).

Read address range

Specifies the read address range for certain *function codes*. See section “[Meaning of the Function code values](#)” on [page 154](#).

Possible values:

0 . 65535 (2¹⁶ - 1)

Write address range

Specifies the write address range for certain *function codes*. See section “[Meaning of the Function code values](#)” on [page 154](#).

Possible values:

0 . 65535 (2¹⁶ - 1)

Add

Adds the items you selected from the drop-down list to the *Function code* field.



Removes the item from the *Function code* field.

Meaning of the Function code values

#	Meaning	Address range (read)	Address range (write)
1	Read Coils	<0 . 65535>	-
2	Read Discrete Inputs	<0 . 65535>	-
3	Read Holding Registers	<0 . 65535>	-

#	Meaning	Address range (read)	Address range (write)
4	Read Input Registers	<0 . 65535>	-
5	Write Single Coil	-	<0 . 65535>
6	Write Single Register	-	<0 . 65535>
7	Read Exception Status	-	-
8	Diagnostic	-	-
11	Get Comm Event Counter	-	-
12	Get Comm Event Log	-	-
13	Program (584/984)	-	-
14	Poll (584/984)	-	-
15	Write Multiple Coils	-	<0 . 65535>
16	Write Multiple Registers	-	<0 . 65535>
17	Report Slave ID	-	-
20	Read File Record	-	-
21	Write File Record	-	-
22	Mask Write Register	-	<0 . 65535>
23	Read/Write Multiple Registers	<0 . 65535>	<0 . 65535>
24	Read FIFO Queue	<0 . 65535>	-
40	Program (Concept)	-	-
42	Concept Symbol Table	-	-
43	Encapsulated Interface Transport	-	-
48	Advantech Co. Ltd. - Management Functions	-	-
66	Scan Data Inc. - Expanded Read Holding Registers	-	-
67	Scan Data Inc. - Expanded Write Holding Registers	-	-
90	Unity Programming/CFS	-	-
100	Scattered Register Read	-	-
125	Schneider Electric - Firmware	-	-

4.6.2 Deep Packet Inspection - OPC Enforcer

[Network Security > DPI > OPC Enforcer]

This dialog lets you specify the *OPC Enforcer* (*OLE for Process Control Enforcer*) settings and define the *OPC Enforcer* specific profiles.

The *OPC* is an integration protocol for industrial environments. The *OPC Enforcer* is a function that supports the network security. The device blocks the data packets that violate the specified profiles. Upon user request, the device verifies the data packets for their plausibility and their fragment characteristics. The device verifies and observes *OPC* data connections and helps protect against invalid or fake data packets. The function dynamically activates TCP ports for each data connection. When requested by an *OPC* server, the device sets up the data connection only between the *OPC* server and the related *OPC* client.

The prerequisite is that *authentication level 5* or lower is set up in your end device to perform the Deep Packet Inspection (DPI). The end device can be a computer or any other equipment capable of sending *OPC* data packets. The *authentication level* defines the type of authentication required for an *OPC* client to connect with an *OPC* server.


The device removes the state information from the packet filter on the following events:

- When applying the profiles saved in the device to the data stream.
- When activating/deactivating the *Routing* function on a router interface.

This includes potential *DCE RPC* information of the *OPC Enforcer*. In the process, the device interrupts open communication connections.

Operation

Uncommitted changes present

Displays if the *OPC Enforcer* profiles applied to the data stream differ from the profiles saved in the device. When you click the  button, the device applies the specified profiles.

Possible values:

marked

At least one of the active *OPC Enforcer* profiles saved in the device contains modified settings.

unmarked

The *OPC Enforcer* profiles applied to the data stream match the profiles saved in the device.

Table

For information on how to customize the appearance of the table, see “Working with tables” on page 16.

Buttons



Add

Opens the [Create](#) window to add a table row.

- In the [Index](#) field, you specify the number of the profile.

Possible values:

1 . 32

When you click the [Ok](#) button, the device adds the table row. The device assigns the number specified in the [Index](#) field to the table row.



Remove

Removes the selected table row.

If you mark the [Profile active](#) checkbox for the profile, then the device stops you from removing the profile.



Copy

Opens the [Copy](#) window to copy an existing table row. The prerequisite is that the table row for the profile to be copied is selected.

- In the [Index](#) field, you specify the number of the profile.

Possible values:

1 . 32

The device adds the table row. The device assigns the number specified in the [Index](#) field to the table row.



Commit changes

The device applies the specified profiles to the data stream.

Index

Displays the sequential number of the profile to which the table row relates. You specify the index number when you add a table row.

Description

Specifies a name for the profile.

Possible values:

Alphanumeric ASCII character string with 0..64 characters
(default setting: [opc](#))

Sanity check

Activates/deactivates the plausibility check for the data packets.

Possible values:

`marked` (default setting)

The plausibility check is active.

The device checks the plausibility of the data packets regarding format and specification.

The device blocks the data packets that violate the specified profiles.

`unmarked`

The plausibility check is inactive.

Fragment check

Activates/deactivates the fragment check for the data packets.

Possible values:

`marked` (default setting)

The fragment check is active.

The device checks the data packets for fragment characteristics.

`unmarked`

The fragment check is inactive.

Timeout at connect

Specifies the time in seconds after which the device removes the dynamic TCP ports, if there is no longer an active *OPC* data connection on the dynamic TCP ports.

Possible values:

`1..300` (default setting: 5)

`0`

The value `0` deactivates the function. The *OPC* data connection remains set up without a time limit.

Profile active

Activates/deactivates the profile.

Possible values:

`marked`

The profile is active.

The device applies the *OPC Enforcer* profiles specified in this table row to the data packets.

`unmarked`

The profile is inactive.

4.6.3 Deep Packet Inspection - DNP3 Enforcer

[Network Security > DPI > DNP3 Enforcer]

This dialog lets you specify the *DNP3 Enforcer* (*Distributed Network Protocol v3 Enforcer*) settings and define the *DNP3 Enforcer* specific profiles.

The *DNP3* protocol is designed to help ensure reliable communication between components in process automation systems. The protocol provides multiplexing, error checking, link control, prioritization, and layer 2 addressing services for user data. The *DNP3 Enforcer* function activates the Deep Packet Inspection (DPI) firewall capabilities for the *DNP3* data stream. The device blocks the data packets that violate the specified profiles. Upon user request, the device verifies the data packets for their plausibility and their fragment characteristics. The device verifies and monitors *DNP3* data connections and helps protect against invalid or falsified data packets.

When the *DNP3 Enforcer* profile is active (checkbox in the *Profile active* column is marked), the device applies the profiles to the data stream.

- The device permits data packets containing only the *function codes* specified in the *Function code list* column.
- The device rejects the data packets containing any other *function codes* that are not specified in the *Function code list* column.

The menu contains the following dialogs:

[DNP3 Profile](#)
[DNP3 Object](#)

4.6.3.1 DNP3 Profile

[Network Security > DPI > DNP3 Enforcer > Profile]

This dialog lets you set up profiles for the *DNP3 Enforcer* function. The profile lets you forward or discard data packets based on the specified values.

Information


Uncommitted changes present

Displays if the *DNP3 Enforcer* profiles applied to the data stream differ from the profiles saved in the device.

Possible values:

marked

At least one of the active *DNP3 Enforcer* profiles saved in the device contains modified settings.

To apply the pending profiles to the data stream, click the  button.

unmarked

The *DNP3 Enforcer* profiles applied to the data stream match the profiles saved in the device.

Table

For information on how to customize the appearance of the table, see “[Working with tables](#)” on [page 16](#).

Buttons



Add

Opens the *Create* window to add a table row.

- In the *Index* field, you specify the number of the profile.

Possible values:

1 . 32

When you click the *Ok* button, the device adds the table row. The device assigns the number specified in the *Index* field to the table row.



Remove

Removes the selected table row.



Copy

Opens the [Copy](#) window to copy an existing table row. The prerequisite is that the table row for the profile to be copied is selected.

- In the [Index](#) field, you specify a new number which identifies the copied profile.

Possible values:

[1 . 32](#)

The device adds the table row. The device assigns the number specified in the [Index](#) field to the table row.



Commit changes

The device applies the specified profiles to the data stream.

Index

Displays the sequential number of the profile to which the table row relates. You specify the index number when you add a table row.

Description

Specifies a name for the profile.

Possible values:

Alphanumeric ASCII character string with 0..32 characters
(default setting: [dnp3](#))

Function code list

Displays the *function codes* for the [DNP3 Enforcer](#) profile. The device permits data packets with the specified properties.

The device lets you specify multiple *function codes*. To do this, perform the following steps:

For the relevant profile, click into the [Function code list](#) column.

The dialog displays the [Function code list](#) window. See “[[Function code list](#)]” on [page 163](#).

From the [Function code list](#) drop-down list, select the desired *function code* item.

Click the [Add](#) button.

To add multiple *function codes*, repeat the previously described steps.

Click the [Ok](#) button.

Possible values:

[0 . 255](#)

You find the meaning of the numbers in section “[[Meaning of the Function code list values](#)]” on [page 163](#).

Index of Default Object List

Specifies the *index numbers* used in the *default object list*.

Possible values:

[all](#) (default setting)

The device applies the [DNP3 Enforcer](#) profile to every data packet regardless of the *index number*.

1. . 317

The device applies the *DNP3 Enforcer* profile only to data packets containing the specified *index number*.

The field lets you specify the following options:

- A single *index number* with a single numerical value, for example 1.
- Multiple *index numbers* with numerical values separated by a comma, for example 1, 2, 3.
- A range with numerical values joined by a dash, for example 7-25.
- You can also combine single numerical values and ranges, for example 2, 7-25, 56.

none

The device does not apply the *index number* to the *DNP3 Enforcer* profile.

CRC check

Activates/deactivates the CRC check for the data packets to validate the checksum contained in the *DNP3* data packets.

Possible values:

marked (default setting)

The CRC check is active.

The device calculates the checksum and compares it with the checksum field in the *DNP3* data packets.

unmarked

The CRC check is inactive.

Sanity check

Activates/deactivates the plausibility check for the data packets.

Possible values:

marked (default setting)

The plausibility check is active.

The device checks the plausibility of the data packets regarding format and specification. The device blocks the data packets that violate the specified profiles.

unmarked

The plausibility check is inactive.

Check outstation traffic

Activates/deactivates the checking of the data packets that originate at an *outstation*.

Possible values:

marked

The checking of data packets from an *outstation* is active.

unmarked

The checking of data packets from an *outstation* is inactive.

TCP reset

Activates/deactivates the resetting of the TCP connection in case of a protocol violation or if the plausibility check detects an error.

Possible values:

[marked](#) (default setting)

The resetting of the TCP connection is active.

If the device identifies a protocol violation or detects a plausibility check error, then the device terminates the TCP connection.

[unmarked](#)

The resetting of the TCP connection is inactive. The TCP connection remains established.

Profile active

Activates/deactivates the profile.

Possible values:

[marked](#)

The profile is active.

The device applies the *DNP3 Enforcer* profiles specified in this table row to the data packets.

[unmarked](#)

The profile is inactive.

[Function code list]

Function code list

Specifies the *function codes* for the relevant *DNP3 Enforcer* profile.

You find the meaning of the numbers in section “[Meaning of the Function code list values](#)” on [page 163](#).

Add

Adds the items you selected from the drop-down list to the *Function code list* field.



Removes the item from the *Function code list* field.

Meaning of the Function code list values

#	Meaning
0	Confir m
1	Read
2	Write
3	Sel ect
4	Operate
5	Di rect Operate

#	Meaning
6	Direct Operate-No Response Required
7	Freeze
8	Freeze-No Response Required
9	Freeze Clear
10	Freeze Clear-No Response Required
11	Freeze at Time
12	Freeze at Time-No Response Required
13	Cold Restart
14	Warm Restart
15	Initialize Data
16	Initialize Application
17	Start Application
18	Stop Application
19	Save Configuration
20	Enable Unsolicited Messages
21	Disable Unsolicited Messages
22	Assign Class
23	Delay Measurement
24	Record Current Time
25	Open File
26	Close File
27	Delete File
28	Get File Information
29	Authenticate File
30	Abort File Transfer
31	Active Configuration
32	Authentication Request
33	Authenticate Request-No Acknowledgment
129	Response
130	Unsolicited Response
131	Authentication Response

4.6.3.2 DNP3 Object

[Network Security > DPI > DNP3 Enforcer > Object]

The *DNP3* function uses objects to transmit values and information between devices. The *DNP3* function uses group numbers to categorize the data type and variation numbers to specify how the data within the group is encoded. Each instance of an encoded information element that defines a unique group and variation in the message, is a *DNP3* object.

This window lets you add custom *DNP3* objects and also lets you view the previously added custom *DNP3* objects. To verify that an added *DNP3* object is valid in a particular *request message/response message*, check the following parameters:

- *Type*
- *Group no.*
- *Variation*
- *Function*
- *Qualifier*
- *Length*
- *Function name*

Based on the IEEE 1815-2012 standard, the *DNP3 Enforcer* function permits by default the data stream containing *DNP3* objects which are available in the *default object list*.

Table

For information on how to customize the appearance of the table, see [“Working with tables” on page 16](#).

Buttons



Add

Opens the *Create* window to add a table row.

- From the *Index* drop-down list, you select the profile *index number*.
- In the *Object index* field, you specify the *index number* of the object.
Possible values:
1 . 256
- From the *Type* drop-down list, you select the type of the message.
Possible values:
request
response
- In the *Group no.* field, you specify a means of classifying the type or the types of data packets in a message. The prerequisite is that in the *Type* field a valid value is specified.
Possible values:
0 . 255
- In the *Variation* field, you specify the *variation number*. The prerequisite is that in the *Group no.* field a valid value is specified.
Possible values:
0 . 255

- In the *Function* field, you specify the *function code*. The *function code* identifies the purpose of the message. The prerequisite is that in the *Variation* field a valid value is specified.

Possible values:

0 . 128

Request messages from masters. Specify a single numerical value, for example 1.

129 . 255

Response messages from outstations. Specify a single numerical value, for example 254.

- In the *Qualifier* field, you specify the *qualifier code* for a pair of each *Group no.*, *Variation*, and *Function* fields. The *qualifier code* is an 8-bit value that defines the *prefix code* and the *range specifier code* for the object in a *DNP3* message. The prerequisite is that in the *Function* field a valid value is specified.

Possible values:

0x00 . 0xf f

You specify multiple individual *qualifier codes* using hexadecimal values separated by a comma for a set of each *Group no.*, *Variation*, and *Function* fields.

When you click the *Ok* button, the device adds the table row. The device assigns the values specified in the *Index*, *Object index*, *Type*, *Group no.*, *Variation*, *Function* and *Qualifier* fields to this table row.



Remove

Removes the selected table row.

Index

Displays the number of the profile to which the table row relates. You specify the index number when you add a table row.

Object index

Displays the number of the object to which the table row relates. You specify the index number when you add a table row.

Type

Specifies the type of the message.

Possible values:

r *request*

Creates a *request message* object in the object list.

r *response*

Creates a *response message* object in the object list.

Group no.

Specifies a means of classifying the type or the types of data packets in a message. The prerequisite is that in the *Type* field a valid value is specified.

Possible values:

0 . 255

Each group number shares a common *point type* and *method of data packet creation*. The *point type* defines the machine in an *outstation*.

Variation

Specifies the *variation number*. The prerequisite is that in the *Group no.* field a valid value is specified. The device applies the *DNP3 Enforcer* profile only to data packets containing the specified value.

The *DNP3* function provides the choice of encoding formats for the type of data packets known as *variation number*. Every value in the *Group no.* field has a set of *variation numbers*.

Possible values:

0 . 255

The field lets you specify the following options:

- You specify a single *variation number* with a single numerical value, for example 1.
- You specify a range with numerical values connected by a dash, for example 0-55.

Function

The *function code* identifies the purpose of the message. The prerequisite is that in the *Variation* field a valid value is specified. The device applies the *DNP3 Enforcer* profile only to data packets containing the specified value.

Possible values:

0 . 128

Request messages from *masters*. Specify a single numerical value, for example 1.

129 . 255

Response messages from *outstations*. Specify a single numerical value, for example 254.

Qualifier

Specifies the *qualifier code* for a pair of each *Group no.*, *Variation*, and *Function* fields. The *qualifier code* is an 8-bit value that defines the *prefix code* and the *range specifier code* for the object in a *DNP3* message. The prerequisite is that in the *Function* field a valid value is specified. The device applies the *DNP3 Enforcer* profile only to data packets containing the specified value.

Possible values:

0x00 . 0xff

You specify multiple individual *qualifier codes* using hexadecimal values separated by a comma for a set of each *Group no.*, *Variation*, and *Function* fields.

Length

Specifies the optional length for the object. The prerequisite is that in the *Function* field a valid value is specified. The device applies the *DNP3 Enforcer* profile only to data packets containing the specified value.

Possible values:

0 . 255

Specify a single numerical value, for example 1.

byte_2

The second byte of the object data contains the length of the remaining portion of the data.

single_bit_packed

If the count of bit values is not a multiple of 8, then the device pads the packed single-bit values up to the next byte boundary.

double_bit_packed

If the count of double bit values is not a multiple of 4, then the device pads the packed double-bit values up to the next byte boundary.

variation

Encodes the length of the object.

Function name

Specifies the optional name for the *function code*. The prerequisite is that in the *Function* field a valid value is specified.

Possible values:

Alphanumeric ASCII character string with 0..32 characters

For example, the device permits data packets with the following *function names*:

- READ
- WRITE
- SELECT

[Index of Default Object List]

Table 1: Request messages

Index	Group no.	Variation	Function	Function name	Length	Qualifier
1	0	209-239	1	READ	-	0x00
2	0	240	1	READ	-	0x00
3	0	240	2	WRITE	byte_2	0x00
4	0	241-243	1	READ	-	0x00
5	0	245-247	1	READ	-	0x00
6	0	245-247	2	WRITE	byte_2	0x00
7	0	248-250	1	READ	-	0x00
8	0	252	1	READ	-	0x00
9	0	254	1	READ	-	0x00 0x06
10	0	255	1	READ	-	0x00 0x06
11	1	0-2	1	READ	-	0x00 0x01 0x06 0x17 0x28
12	1	0	22	ASSIGN CLASS	-	0x00 0x01 0x06 0x17 0x28
13	2	0-3	1	READ	-	0x06 0x07 0x08

Table 1: Request messages (cont.)

Index	Group no.	Variation	Function	Function name	Length	Qualifier
14	3	0-2	1	READ	-	0x00 0x01 0x06 0x17 0x28
15	3	0	22	ASSIGN_CLASS	-	0x00 0x01 0x06 0x17 0x28
16	4	0-3	1	READ	-	0x06 0x07 0x08
17	10	0	1	READ	-	0x00 0x01 0x06 0x17 0x28
18	10	0	22	ASSIGN_CLASS	-	0x00 0x01 0x06 0x17 0x28
19	10	1	2	WRITE	single_bit_packed	0x00 0x01
20	10	2	1	READ	-	0x00 0x01 0x06 0x17 0x28
21	11	0-2	1	READ	-	0x06 0x07 0x08
22	12	0	22	ASSIGN_CLASS	-	0x00 0x01 0x06 0x17 0x28
23	12	1	3	SELECT	11	0x00 0x01 0x17 0x28
24	12	1	4	OPERATE	11	0x00 0x01 0x17 0x28
25	12	1	5	DI RECT_OPERATE	11	0x00 0x01 0x17 0x28

Table 1: Request messages (cont.)

Index	Group no.	Variation	Function	Function name	Length	Qualifier
26	12	1	6	DI RECT_OPERATE_NR	11	0x00 0x01 0x17 0x28
27	12	2	3	SELECT	11	0x07 0x08
28	12	2	4	OPERATE	11	0x07 0x08
29	12	2	5	DI RECT_OPERATE	11	0x07 0x08
30	12	2	6	DI RECT_OPERATE_NR	11	0x07 0x08
31	12	3	3	SELECT	si ngl e_bi t_packed	0x00 0x01
32	12	3	4	OPERATE	si ngl e_bi t_packed	0x00 0x01
33	12	3	5	DI RECT_OPERATE	si ngl e_bi t_packed	0x00 0x01
34	12	3	6	DI RECT_OPERATE_NR	si ngl e_bi t_packed	0x00 0x01
35	13	0-2	1	READ	-	0x06 0x07 0x08
36	20	0-2	1	READ	-	0x00 0x01 0x06 0x17 0x28
37	20	5-6	1	READ	-	0x00 0x01 0x06 0x17 0x28
38	20	0	7	I MMEDI ATE_FREEZE	-	0x00 0x01 0x06 0x17 0x28
39	20	0	8	I MMEDI ATE_FREEZE_NR	-	0x00 0x01 0x06 0x17 0x28
40	20	0	9	FREEZE_CLEAR	-	0x00 0x01 0x06 0x17 0x28

Table 1: Request messages (cont.)

Index	Group no.	Variation	Function	Function name	Length	Qualifier
41	20	0	10	FREEZE_CLEAR_NR	-	0x00 0x01 0x06 0x17 0x28
42	20	0	11	FREEZE_AT_TIME	-	0x00 0x01 0x06 0x17 0x28
43	20	0	12	FREEZE_AT_TIME_NR	-	0x00 0x01 0x06 0x17 0x28
44	20	0	22	ASSIGN_CLASS	-	0x00 0x01 0x06 0x17 0x28
45	21	0-2	1	READ	-	0x00 0x01 0x06 0x17 0x28
46	21	5-6	1	READ	-	0x00 0x01 0x06 0x17 0x28
47	21	9-10	1	READ	-	0x00 0x01 0x06 0x17 0x28
48	21	0	22	ASSIGN_CLASS	-	0x00 0x01 0x06 0x17 0x28
49	22	0-2	1	READ	-	0x06 0x07 0x08
50	22	5-6	1	READ	-	0x06 0x07 0x08
51	23	0-2	1	READ	-	0x06 0x07 0x08

Table 1: Request messages (cont.)

Index	Group no.	Variation	Function	Function name	Length	Qualifier
52	23	5-6	1	READ	-	0x06 0x07 0x08
53	30	0-6	1	READ	-	0x00 0x01 0x06 0x17 0x28
54	30	0	7	IMMEDIATE_FREEZE	-	0x00 0x01 0x06 0x17 0x28
55	30	0	8	IMMEDIATE_FREEZE_NR	-	0x00 0x01 0x06 0x17 0x28
56	30	0	11	FREEZE_AT_TIME	-	0x00 0x01 0x06 0x17 0x28
57	30	0	12	FREEZE_AT_TIME_NR	-	0x00 0x01 0x06 0x17 0x28
58	30	0	22	ASSIGN_CLASS	-	0x00 0x01 0x06 0x17 0x28
59	31	0-8	1	READ	-	0x00 0x01 0x06 0x17 0x28
60	31	0	22	ASSIGN_CLASS	-	0x00 0x01 0x06 0x17 0x28
61	32	0-8	1	READ	-	0x06 0x07 0x08
62	33	0-8	1	READ	-	0x06 0x07 0x08

Table 1: Request messages (cont.)

Index	Group no.	Variation	Function	Function name	Length	Qualifier
63	34	0-3	1	READ	-	0x00 0x01 0x06
64	34	1	2	VRI TE	2	0x00 0x01 0x17 0x28
65	34	2	2	VRI TE	4	0x00 0x01 0x17 0x28
66	34	3	2	VRI TE	4	0x00 0x01 0x17 0x28
67	40	0	1	READ	-	0x00 0x01 0x06
68	40	0	22	ASSI GN_CLASS	-	0x00 0x01 0x06 0x17 0x28
69	40	1-4	1	READ	-	0x00 0x01 0x06 0x17 0x28
70	41	0	22	ASSI GN_CLASS	-	0x00 0x01 0x06 0x17 0x28
71	41	1	3	SELECT	5	0x00 0x01 0x17 0x28
72	41	2	3	SELECT	3	0x00 0x01 0x17 0x28
73	41	3	3	SELECT	5	0x00 0x01 0x17 0x28
74	41	1	4	OPERATE	5	0x00 0x01 0x17 0x28

Table 1: Request messages (cont.)

Index	Group no.	Variation	Function	Function name	Length	Qualifier
75	41	2	4	OPERATE	3	0x00 0x01 0x17 0x28
76	41	3	4	OPERATE	5	0x00 0x01 0x17 0x28
77	41	1	5	DI RECT_OPERATE	5	0x00 0x01 0x17 0x28
78	41	2	5	DI RECT_OPERATE	3	0x00 0x01 0x17 0x28
79	41	3	5	DI RECT_OPERATE	5	0x00 0x01 0x17 0x28
80	41	1	6	DI RECT_OPERATE_NR	5	0x00 0x01 0x17 0x28
81	41	2	6	DI RECT_OPERATE_NR	3	0x00 0x01 0x17 0x28
82	41	3	6	DI RECT_OPERATE_NR	5	0x00 0x01 0x17 0x28
83	42	0-8	1	READ	-	0x06 0x07 0x08
84	43	0-8	1	READ	-	0x06 0x07 0x08
85	50	1	1	READ	-	0x07
86	50	1	2	WRITE	6	0x07
87	50	2	11	FREEZE_AT_TIME	10	0x07
88	50	2	12	FREEZE_AT_TIME_NR	10	0x07
89	50	3	2	WRITE	10	0x07
90	50	4	1	READ	-	0x00 0x01 0x06 0x17 0x28

Table 1: Request messages (cont.)

Index	Group no.	Variation	Function	Function name	Length	Qualifier
91	50	4	2	WRITE	11	0x00 0x01 0x17 0x28
92	60	1	1	READ	-	0x06
93	60	2-4	1	READ	-	0x06 0x07 0x08
94	60	1-4	22	ASSIGN_CLASS	-	0x06
95	60	2-4	20	ENABLE_UNSQLICITED	-	0x06
96	60	2-4	21	DISABLE_UNSQLICITED	-	0x06
97	70	2	29	FILE_AUTHENTICATE	QC_5B_count_1	0x5B
98	70	3	25	OPEN_FILE	QC_5B_count_1	0x5B
99	70	3	27	DELETE_FILE	QC_5B_count_1	0x5B
100	70	4	26	CLOSE_FILE	QC_5B_count_1	0x5B
101	70	4	30	FILE_ABORT	QC_5B_count_1	0x5B
102	70	5-6	1	READ	QC_5B_count_1	0x5B
103	70	5	2	WRITE	QC_5B_count_1	0x5B
104	70	7	28	GET_FILE_INFORMATION	QC_5B_count_1	0x5B
105	70	8	31	ACTIVATE_CONFIGURATION	QC_5B_count_1	0x5B
106	80	1	1	READ	-	0x00 0x01
107	80	1	2	WRITE	single_bit_packed	0x00 0x01
108	81	1	1	READ	-	0x00 0x01
109	82	1	1	READ	-	0x00 0x01
110	83	1	1	READ	-	0x00 0x01
111	85	0	1	READ	-	0x06
112	85	1	1	READ	-	0x00 0x01 0x06 0x17 0x28
113	85	1	2	WRITE	QC_5B	0x5B
114	86	0	22	ASSIGN_CLASS	-	0x00 0x01 0x06 0x17 0x28
115	86	1-3	1	READ	-	0x00 0x01 0x06 0x17 0x28

Table 1: Request messages (cont.)

Index	Group no.	Variation	Function	Function name	Length	Qualifier
116	86	1	2	WRITE	QC_5B	0x5B
117	86	3	2	WRITE	QC_5B	0x5B
118	87	0	1	READ	-	0x06
119	87	1	1	READ	-	0x00 0x01 0x06 0x17 0x28
120	87	1	2	WRITE	QC_5B	0x5B
121	87	1	3	SELECT	QC_5B	0x5B
122	87	1	4	OPERATE	QC_5B	0x5B
123	87	1	5	DIRECT_OPERATE	QC_5B	0x5B
124	87	1	6	DIRECT_OPERATE_NR	QC_5B	0x5B
125	88	0-1	1	READ	-	0x06 0x07 0x08
126	90	1	16	INITIALIZE_APPLICATION	QC_5B	0x5B
127	90	1	17	START_APPLICATION	QC_5B	0x5B
128	90	1	18	STOP_APPLICATION	QC_5B	0x5B
129	101	1-3	1	READ	-	0x00 0x01 0x06 0x17 0x28
130	102	1	1	READ	-	0x00 0x01 0x03 0x04 0x05 0x06 0x17 0x28
131	102	1	2	WRITE	1	0x00 0x01 0x03 0x04 0x05 0x17 0x28
132	110	128	1	READ	-	0x00 0x01 0x03 0x04 0x05 0x06 0x17 0x28

Table 1: Request messages (cont.)

Index	Group no.	Variation	Function	Function name	Length	Qualifier
133	110	128	2	VRI TE	vari ati on	0x00 0x01 0x03 0x04 0x05 0x17 0x28
134	110	128	31	ACTI VATE_CONFI GURATI ON	vari ati on	0x5B
135	111	128	1	READ	-	0x06
136	112	128	2	VRI TE	vari ati on	0x00 0x01 0x17 0x28
137	113	0	1	READ	-	0x00 0x01 0x17 0x28
138	113	0	22	ASSI GN_CLASS	-	0x00 0x01 0x06 0x17 0x28

Table 2: Response messages

Index	Group no.	Variation	Function	Function name	Length	Qualifier
139	0	209- 239	129	RESPONSE	byte_2	0x00 0x17
140	0	240	129	RESPONSE	byte_2	0x00 0x17
141	0	241- 243	129	RESPONSE	byte_2	0x00 0x17
142	0	245- 247	129	RESPONSE	byte_2	0x00 0x17
143	0	248- 250	129	RESPONSE	byte_2	0x00 0x17
144	0	252	129	RESPONSE	byte_2	0x00 0x17
145	0	255	129	RESPONSE	byte_2	0x00 0x17
146	1	1	129	RESPONSE	si ngl e_bi t_packed	0x00 0x01 0x17 0x28
147	1	2	129	RESPONSE	1	0x00 0x01 0x17 0x28

Table 2: Response messages (cont.)

Index	Group no.	Variation	Function	Function name	Length	Qualifier
148	2	1	129	RESPONSE	1	0x17 0x28
149	2	2	129	RESPONSE	7	0x17 0x28
150	2	3	129	RESPONSE	3	0x17 0x28
151	2	1	130	UNSOLI CI TED_RESPONSE	1	0x17 0x28
152	2	2	130	UNSOLI CI TED_RESPONSE	7	0x17 0x28
153	2	3	130	UNSOLI CI TED_RESPONSE	3	0x17 0x28
154	3	1	129	RESPONSE	double_bit_packed	0x00 0x01 0x17 0x28
155	3	2	129	RESPONSE	1	0x00 0x01 0x17 0x28
156	4	1	129	RESPONSE	1	0x17 0x28
157	4	2	129	RESPONSE	7	0x17 0x28
158	4	3	129	RESPONSE	3	0x17 0x28
159	4	1	130	UNSOLI CI TED_RESPONSE	1	0x17 0x28
160	4	2	130	UNSOLI CI TED_RESPONSE	7	0x17 0x28
161	4	3	130	UNSOLI CI TED_RESPONSE	3	0x17 0x28
162	10	2	129	RESPONSE	1	0x00 0x01 0x17 0x28
163	11	1	129	RESPONSE	1	0x17 0x28
164	11	2	129	RESPONSE	7	0x17 0x28
165	11	1	130	UNSOLI CI TED_RESPONSE	1	0x17 0x28
166	11	2	130	UNSOLI CI TED_RESPONSE	7	0x17 0x28
167	12	1	129	RESPONSE	11	0x00 0x01 0x17 0x28

Table 2: Response messages (cont.)

Index	Group no.	Variation	Function	Function name	Length	Qualifier
168	12	2	129	RESPONSE	11	0x07 0x08
169	12	3	129	RESPONSE	single_bit_packed	0x00 0x01
170	13	1	129	RESPONSE	1	0x17 0x28
171	13	2	129	RESPONSE	7	0x17 0x28
172	13	1	130	UNSOLICITED_RESPONSE	1	0x17 0x28
173	13	2	130	UNSOLICITED_RESPONSE	7	0x17 0x28
174	20	1	129	RESPONSE	5	0x00 0x01 0x17 0x28
175	20	2	129	RESPONSE	3	0x00 0x01 0x17 0x28
176	20	5	129	RESPONSE	4	0x00 0x01 0x17 0x28
177	20	6	129	RESPONSE	2	0x00 0x01 0x17 0x28
178	21	1	129	RESPONSE	5	0x00 0x01 0x17 0x28
179	21	2	129	RESPONSE	3	0x00 0x01 0x17 0x28
180	21	5	129	RESPONSE	4	0x00 0x01 0x17 0x28
181	21	6	129	RESPONSE	2	0x00 0x01 0x17 0x28
182	21	9	129	RESPONSE	4	0x00 0x01 0x17 0x28

Table 2: Response messages (cont.)

Index	Group no.	Variation	Function	Function name	Length	Qualifier
183	21	10	129	RESPONSE	2	0x00 0x01 0x17 0x28
184	22	1	129	RESPONSE	5	0x17 0x28
185	22	2	129	RESPONSE	3	0x17 0x28
186	22	1	130	UNSOLI CI TED_RESPONSE	5	0x17 0x28
187	22	2	130	UNSOLI CI TED_RESPONSE	3	0x17 0x28
188	22	5	129	RESPONSE	11	0x17 0x28
189	22	6	129	RESPONSE	9	0x17 0x28
190	22	5	130	UNSOLI CI TED_RESPONSE	11	0x17 0x28
191	22	6	130	UNSOLI CI TED_RESPONSE	9	0x17 0x28
192	23	1	129	RESPONSE	5	0x17 0x28
193	23	2	129	RESPONSE	3	0x17 0x28
194	23	1	130	UNSOLI CI TED_RESPONSE	5	0x17 0x28
195	23	2	130	UNSOLI CI TED_RESPONSE	3	0x17 0x28
196	23	5	129	RESPONSE	11	0x17 0x28
197	23	6	129	RESPONSE	9	0x17 0x28
198	23	5	130	UNSOLI CI TED_RESPONSE	11	0x17 0x28
199	23	6	130	UNSOLI CI TED_RESPONSE	9	0x17 0x28
200	30	1	129	RESPONSE	5	0x00 0x01 0x17 0x28
201	30	2	129	RESPONSE	3	0x00 0x01 0x17 0x28
202	30	3	129	RESPONSE	4	0x00 0x01 0x17 0x28

Table 2: Response messages (cont.)

Index	Group no.	Variation	Function	Function name	Length	Qualifier
203	30	4	129	RESPONSE	2	0x00 0x01 0x17 0x28
204	30	5	129	RESPONSE	5	0x00 0x01 0x17 0x28
205	30	6	129	RESPONSE	9	0x00 0x01 0x17 0x28
206	31	1	129	RESPONSE	5	0x00 0x01 0x17 0x28
207	31	2	129	RESPONSE	3	0x00 0x01 0x17 0x28
208	31	3	129	RESPONSE	11	0x00 0x01 0x17 0x28
209	31	4	129	RESPONSE	9	0x00 0x01 0x17 0x28
210	31	5	129	RESPONSE	4	0x00 0x01 0x17 0x28
211	31	6	129	RESPONSE	2	0x00 0x01 0x17 0x28
212	31	7	129	RESPONSE	5	0x00 0x01 0x17 0x28
213	31	8	129	RESPONSE	9	0x00 0x01 0x17 0x28
214	32	1	129	RESPONSE	5	0x17 0x28
215	32	2	129	RESPONSE	3	0x17 0x28
216	32	3	129	RESPONSE	11	0x17 0x28

Table 2: Response messages (cont.)

Index	Group no.	Variation	Function	Function name	Length	Qualifier
217	32	4	129	RESPONSE	9	0x17 0x28
218	32	5	129	RESPONSE	5	0x17 0x28
219	32	6	129	RESPONSE	9	0x17 0x28
220	32	7	129	RESPONSE	11	0x17 0x28
221	32	8	129	RESPONSE	15	0x17 0x28
222	32	1	130	UNSOLI CI TED_RESPONSE	5	0x17 0x28
223	32	2	130	UNSOLI CI TED_RESPONSE	3	0x17 0x28
224	32	3	130	UNSOLI CI TED_RESPONSE	11	0x17 0x28
225	32	4	130	UNSOLI CI TED_RESPONSE	9	0x17 0x28
226	32	5	130	UNSOLI CI TED_RESPONSE	5	0x17 0x28
227	32	6	130	UNSOLI CI TED_RESPONSE	9	0x17 0x28
228	32	7	130	UNSOLI CI TED_RESPONSE	11	0x17 0x28
229	32	8	130	UNSOLI CI TED_RESPONSE	15	0x17 0x28
230	33	1	129	RESPONSE	5	0x17 0x18
231	33	2	129	RESPONSE	3	0x17 0x28
232	33	3	129	RESPONSE	11	0x17 0x28
233	33	4	129	RESPONSE	9	0x17 0x28
234	33	5	129	RESPONSE	5	0x17 0x28
235	33	6	129	RESPONSE	9	0x17 0x28
236	33	7	129	RESPONSE	11	0x17 0x28
237	33	8	129	RESPONSE	15	0x17 0x28
238	33	1	130	UNSOLI CI TED_RESPONSE	5	0x17 0x28
239	33	2	130	UNSOLI CI TED_RESPONSE	3	0x17 0x28

Table 2: Response messages (cont.)

Index	Group no.	Variation	Function	Function name	Length	Qualifier
240	33	3	130	UNSOLI CI TED_RESPONSE	11	0x17 0x28
241	33	4	130	UNSOLI CI TED_RESPONSE	9	0x17 0x28
242	33	5	130	UNSOLI CI TED_RESPONSE	5	0x17 0x28
243	33	6	130	UNSOLI CI TED_RESPONSE	9	0x17 0x28
244	33	7	130	UNSOLI CI TED_RESPONSE	11	0x17 0x28
245	33	8	130	UNSOLI CI TED_RESPONSE	15	0x17 0x28
246	34	1	129	RESPONSE	2	0x00 0x01
247	34	2-3	129	RESPONSE	4	0x00 0x01
248	40	1	129	RESPONSE	5	0x00 0x01 0x17 0x28
249	40	2	129	RESPONSE	3	0x00 0x01 0x17 0x28
250	40	3	129	RESPONSE	5	0x00 0x01 0x17 0x28
251	40	4	129	RESPONSE	9	0x00 0x01 0x17 0x28
252	41	1	129	RESPONSE	5	0x00 0x01 0x17 0x28
253	41	2	129	RESPONSE	3	0x00 0x01 0x17 0x28
254	41	3	129	RESPONSE	5	0x00 0x01 0x17 0x28
255	42	1	129	RESPONSE	5	0x17 0x28
256	42	2	129	RESPONSE	3	0x17 0x28

Table 2: Response messages (cont.)

Index	Group no.	Variation	Function	Function name	Length	Qualifier
257	42	3	129	RESPONSE	11	0x17 0x28
258	42	4	129	RESPONSE	9	0x17 0x28
259	42	5	129	RESPONSE	5	0x17 0x28
260	42	6	129	RESPONSE	9	0x17 0x28
261	42	7	129	RESPONSE	11	0x17 0x28
262	42	8	129	RESPONSE	15	0x17 0x28
263	42	1	130	UNSOLI CI TED_RESPONSE	5	0x17 0x28
264	42	2	130	UNSOLI CI TED_RESPONSE	3	0x17 0x28
265	42	3	130	UNSOLI CI TED_RESPONSE	11	0x17 0x28
266	42	4	130	UNSOLI CI TED_RESPONSE	9	0x17 0x28
267	42	5	130	UNSOLI CI TED_RESPONSE	5	0x17 0x28
268	42	6	130	UNSOLI CI TED_RESPONSE	9	0x17 0x28
269	42	7	130	UNSOLI CI TED_RESPONSE	11	0x17 0x28
270	42	8	130	UNSOLI CI TED_RESPONSE	15	0x17 0x28
271	43	1	129	RESPONSE	5	0x17 0x28
272	43	2	129	RESPONSE	3	0x17 0x28
273	43	3	129	RESPONSE	11	0x17 0x28
274	43	4	129	RESPONSE	9	0x17 0x28
275	43	5	129	RESPONSE	5	0x17 0x28
276	43	6	129	RESPONSE	9	0x17 0x28
277	43	7	129	RESPONSE	11	0x17 0x28
278	43	8	129	RESPONSE	15	0x17 0x28
279	43	1	130	UNSOLI CI TED_RESPONSE	5	0x17 0x28

Table 2: Response messages (cont.)

Index	Group no.	Variation	Function	Function name	Length	Qualifier
280	43	2	130	UNSOLI CI TED_RESPONSE	3	0x17 0x28
281	43	3	130	UNSOLI CI TED_RESPONSE	11	0x17 0x28
282	43	4	130	UNSOLI CI TED_RESPONSE	9	0x17 0x28
283	43	5	130	UNSOLI CI TED_RESPONSE	5	0x17 0x28
284	43	6	130	UNSOLI CI TED_RESPONSE	9	0x17 0x28
285	43	7	130	UNSOLI CI TED_RESPONSE	11	0x17 0x28
286	43	8	130	UNSOLI CI TED_RESPONSE	15	0x17 0x28
287	50	1	129	RESPONSE	6	0x07
288	50	4	129	RESPONSE	11	0x00 0x01 0x17 0x28
289	51	1-2	129	RESPONSE	6	0x07
290	51	1-2	130	UNSOLI CI TED_RESPONSE	6	0x07
291	52	1-2	129	RESPONSE	2	0x07
292	70	2	129	RESPONSE	QC_5B_count_1	0x5B
293	70	4-7	129	RESPONSE	QC_5B_count_1	0x5B
294	70	4-7	130	UNSOLI CI TED_RESPONSE	QC_5B_count_1	0x5B
295	80	1	129	RESPONSE	2	0x00 0x01
296	81	1	129	RESPONSE	3	0x07
297	82	1	129	RESPONSE	QC_5B_count_1	0x5B
298	82	1	130	RESPONSE	QC_5B_count_1	0x5B
299	83	1-2	129	RESPONSE	QC_5B	0x5B
300	83	1	130	UNSOLI CI TED_RESPONSE	QC_5B	0x5B
301	85	1	129	RESPONSE	QC_5B	0x5B
302	86	1	129	RESPONSE	QC_5B	0x5B
303	86	2	129	RESPONSE	1	0x00 0x01 0x17 0x28
304	86	3	129	RESPONSE	QC_5B	0x5B
305	87	1	129	RESPONSE	QC_5B	0x5B
306	88	1	129	RESPONSE	QC_5B	0x5B
307	88	1	130	UNSOLI CI TED_RESPONSE	QC_5B	0x5B
308	91	1	129	RESPONSE	QC_5B	0x5B

Table 2: Response messages (cont.)

Index	Group no.	Variation	Function	Function name	Length	Qualifier
309	101	1	129	RESPONSE	2	0x00 0x01 0x17 0x28
310	101	2	129	RESPONSE	4	0x00 0x01 0x17 0x28
311	101	3	129	RESPONSE	8	0x00 0x01 0x17 0x28
312	102	1	129	RESPONSE	1	0x00 0x01 0x03 0x04 0x05 0x17 0x28
313	110	128	129	RESPONSE	variati on	0x00 0x01 0x03 0x04 0x05 0x17 0x28
314	111	128	129	RESPONSE	variati on	0x00 0x01 0x03 0x04 0x05 0x17 0x28
315	111	128	130	UNSOLI CI TED_RESPONSE	variati on	0x00 0x01 0x17 0x28
316	113	128	129	RESPONSE	variati on	0x00 0x01 0x17 0x28
317	113	128	130	UNSOLI CI TED_RESPONSE	variati on	0x00 0x01 0x17 0x28

4.6.4 Deep Packet Inspection - IEC104 Enforcer

[Network Security > DPI > IEC104 Enforcer]

This dialog lets you specify the *IEC104 Enforcer* settings and define the *IEC104 Enforcer* specific profiles.

The *IEC104* protocol is a communication protocol used in the automation sector. The *IEC104* protocol helps to transfer the *IEC104* data packets between a *control station* (client) and a *substation* (server) using a TCP/IP network. The *IEC104 Enforcer* function activates the Deep Packet Inspection (DPI) firewall capabilities for the *IEC104* data stream. The *type IDs* in the *IEC104* protocol specify the purpose of the data transfer. The device blocks the data packets that violate the specified profiles.

When the *IEC104 Enforcer* profile is active, the device applies the profile to the data stream.


The device permits only data packets containing the values specified in the following columns:

- *Function type*
- *Advanced type ID list*
- *Originator address list*
- *Common address list*

Operation

Uncommitted changes present

Displays if the *IEC104 Enforcer* profiles applied to the data stream differ from the profiles saved in the device.

When you click the  button, the device applies the specified profiles.

Possible values:

marked

At least one of the active *IEC104 Enforcer* profiles saved in the device contains modified settings.

unmarked

The *IEC104 Enforcer* profiles applied to the data stream match the profiles saved in the device.

Table

For information on how to customize the appearance of the table, see “Working with tables” on page 16.

Buttons



Add

Opens the *Create* window to add a table row.

- In the *Index* field, you specify the number of the profile.

Possible values:

1 . 32

When you click the *Ok* button, the device adds the table row. The device assigns the number specified in the *Index* field to the table row.



Remove

Removes the selected table row.

If you mark the *Profile active* checkbox for the profile, then the device stops you from removing the profile.



Copy

Opens the *Copy* window to copy an existing table row. The prerequisite is that the table row for the profile to be copied is selected.

- In the *Index* field, you specify the new number of the copied profile.

Possible values:

1 . 32

The device adds the table row. The device assigns the number specified in the *Index* field to the table row.



Commit changes

The device applies the specified profiles to the data stream.

If you changed the values in the *Function type* field, then the device assigns the specific values to the related profile.

Index

Displays the sequential number of the profile to which the table row relates. You specify the index number when you add a table row.

Description

Specifies a name for the profile.

Possible values:

Alphanumeric ASCII character string with 0..128 characters
(default setting: *iec104*)

Function type

Specifies the function type for the *IEC104 Enforcer* profile. After clicking the ✓ button, the device assigns the corresponding *type IDs*.

Possible values:

[readOnly](#)

Assigns the *type IDs* for the *read* function.

1, 3, 5, 7, 9, 11, 13, 15, 20, 21, 30-40, 70, 100-102

[readWrite](#)

Assigns the *type IDs* for the *read/write* functions.

1, 3, 5, 7, 9, 11, 13, 15, 20, 21, 30-40, 45-51, 58-64, 70, 100-102

[common](#)

Assigns the *type IDs* for the *common* functions.

1, 3, 5, 7, 9, 11, 13, 15, 20, 21, 30-40, 45-51, 58-64, 70, 100-102, 110-113, 120-127

[any](#) (default setting)

Assigns the *type IDs* for every function.

1, 2, . . . , 254, 255

The device does not permit any subsequent changes in the *Advanced type ID list* column.

[advanced](#)

Lets you specify user-defined values in the *Advanced type ID list* column.

Advanced type ID list

Displays the *advanced type IDs* for the *IEC104 Enforcer* profile. The device permits data packets with the specified properties. The prerequisite is that in the *Function type* column a value other than *any* is specified.

The device lets you specify multiple *Advanced type IDs*. To do this, perform the following steps:

For the relevant profile, click into the *Advanced type ID list* column.

The dialog displays the *Advanced type ID list* window.

From the *Advanced type ID list* drop-down list, select the desired *type ID* item.

Click the *Add* button.

To add multiple *type IDs*, repeat the previously described steps.

Click the *Ok* button.

Possible values:

0 . 255

You find the meaning of the numbers in section “[Meaning of the Advanced type ID list values](#)” on page 192.

Originator address list

Specifies the addresses from which data packets originated. The prerequisite is that in the *Cause of transmission size* column the value 2 is specified.

Possible values:

<empty> (default setting)

The device permits data packets from any *originator* address.

0 . 255

The device permits data packets with the specified *originator* address.

Common address list

Specifies the addresses to which the device forwards the *IEC104* data packets.

Possible values:

0 . 255

The device permits data packets with the specified *common* address. The prerequisite is that in the *Common address size* column the value 1 is specified.

0 . 65535 (2¹⁶ - 1)

The device permits data packets with the specified *common* address. The prerequisite is that in the *Common address size* column the value 2 is specified.

Cause of transmission size

Specifies the size in octets that defines the variation of the respective fields in the data packets. The device performs the *DPI* function based on these settings.

Possible values:

1

The data packets do not contain an *originator* address.

2 (default setting)

The data packets contain an *originator* address.

Common address size

Specifies the size in octets of the *common* address to which the device forwards the *IEC104* data packets. This setting affects the setting in the *Common address list* column.

Possible values:

1

2 (default setting)

IO address size

Specifies the size in octets of the *information object address*.

Possible values:

1

2

3 (default setting)

Allow IEC_60870_5_101

Activates/deactivates the *type IDs* defined in the *IEC101* specification.

Possible values:

marked

The *type IDs* defined in the *IEC101* specification are active.

The device permits the *type ID* values 2, 4, 6, 8, 10, 12, 14, 16, 17, 18, 19, 103, 104, 105, 106 along with the *type IDs* based on the values specified in the *Function type* column or *Advanced type ID list* column.

unmarked (default setting)

The *type IDs* defined in the *IEC101* specification are inactive.

The device permits only the *type ID* values based on the values specified in the *Function type* or *Advanced type ID list* column.

Sanity check

Activates/deactivates the plausibility check for the data packets.

Possible values:

`marked` (default setting)

The plausibility check is active.

The device checks the plausibility of the data packets regarding format and specification. The device blocks the data packets that violate the specified profiles.

`unmarked`

The plausibility check is inactive.

TCP reset

Activates/deactivates the resetting of the TCP connection in case of a protocol violation or if the plausibility check detects an error.

Possible values:

`marked` (default setting)

The resetting of the TCP connection is active.

If the device identifies a protocol violation or detects a plausibility check error, then the device terminates the TCP connection. The device establishes the TCP connection again on receiving a new request.

`unmarked`

The resetting of the TCP connection is inactive.

Debug

Activates/deactivates the debugging of the profiles.

Possible values:

`marked`

Debugging is active.

The device sends the reset packet along with the information related to the termination of TCP connection. The prerequisite is that in the *TCP reset* column the checkbox is marked.

`unmarked` (default setting)

Debugging is inactive.

Profile active

Activates/deactivates the profile.

Possible values:

`marked`

The profile is active.

The device applies the *IEC104 Enforcer* profiles specified in this table row to the data packets.

`unmarked`

The profile is inactive.

[Advanced type ID list]

Advanced type ID list

Specifies the *Advanced type IDs* for the relevant *IEC104 Enforcer* profile.

You find the meaning of the numbers in section “Meaning of the Advanced type ID list values” on page 192.

Add

Adds the items you selected from the drop-down list to the *Advanced type ID list* field.



Removes the item from the *Advanced type ID list* field.

Meaning of the Advanced type ID list values

#	Meaning
1	Single point information M _{SP} _NA_1
2	Single point information with time tag M _{SP} _TA_1
3	Double point information M _{DP} _NA_1
4	Double point information with time tag M _{DP} _TA_1
5	Step position information M _{ST} _NA_1
6	Step position information with time tag M _{ST} _TA_1
7	Bit string of 32 bit M _{BO} _NA_1
8	Bit string of 32 bit with time tag M _{BO} _TA_1
9	Measured value, normalized value M _{ME} _NA_1
10	Measured value, normalized value with time tag M _{ME} _TA_1
11	Measured value, scaled value M _{ME} _NB_1
12	Measured value, scaled value with time tag M _{ME} _TB_1
13	Measured value, short floating point value M _{ME} _NC_1
14	Measured value, short floating point value with time tag M _{ME} _TC_1
15	Integrated totals M _{IT} _NA_1
16	Integrated totals with time tag M _{IT} _TA_1
17	Event of protection equipment with time tag M _{EP} _TA_1
18	Packed start events of protection equipment with time tag M _{EP} _TB_1
19	Packed output circuit information of protection equipment with time tag M _{EP} _TC_1
20	Packed single-point information with status change detection M _{PS} _NA_1
21	Measured value, normalized value without quality descriptor M _{ME} _ND_1
30	Single point information with time tag CP56Time2a M _{SP} _TB_1
31	Double point information with time tag CP56Time2a M _{DP} _TB_1
32	Step position information with time tag CP56Time2a M _{ST} _TB_1
33	Bit string of 32 bit with time tag CP56Time2a M _{BO} _TB_1
34	Measured value, normalized value with time tag CP56Time2a M _{ME} _TD_1
35	Measured value, scaled value with time tag CP56Time2a M _{ME} _TE_1
36	Measured value, short floating point value with time tag CP56Time2a M _{ME} _TF_1

#	Meaning
37	Integrated totals with time tag CP56Time2a MIT_TB_1
38	Event of protection equipment with time tag CP56Time2a MEP_TD_1
39	Packed start events of protection equipment with time tag CP56time2a MEP_TE_1
40	Packed output circuit information of protection equipment with time tag CP56Time2a MEP_TF_1
45	Single command C_SC_NA_1
46	Double command C_DC_NA_1
47	Regulating step command C_RC_NA_1
48	Setpoint command, normalized value C_SE_NA_1
49	Setpoint command, scaled value C_SE_NB_1
50	Setpoint command, short floating point value C_SE_NC_1e
51	Bit string 32 bit C_BO_NA_1
58	Single command with time tag CP56Time2a C_SC_TA_1
59	Double command with time tag CP56Time2a C_DC_TA_1
60	Regulating step command with time tag CP56Time2a C_RC_TA_1
61	Setpoint command, normalized value with time tag CP56Time2a C_SE_TA_1
62	Setpoint command, scaled value with time tag CP56Time2a C_SE_TB_1
63	Setpoint command, short floating point value with time tag CP56Time2a C_SE_TC_1
64	Bit string 32 bit with time tag CP56Time2a C_BO_TA_1
70	End of initialization MEI_NA_1
100	(General -) Interrogation command C_IC_NA_1
101	Counter interrogation command C_CI_NA_1
102	Read command C_RD_NA_1
103	Clock synchronization command C_CS_NA_1
104	(IEC 101) Test command C_TS_NB_1
105	Reset process command C_RP_NC_1
106	(IEC 101) Delay acquisition command C_CD_NA_1
107	Test command with time tag CP56Time2a C_TS_TA_1
110	Parameter of measured value, normalized value P_ME_NA_1
111	Parameter of measured value, scaled value P_ME_NB_1
112	Parameter of measured value, short floating point value P_ME_NC_1
113	Parameter activation P_AC_NA_1
120	File ready F_FR_NA_1
121	Section ready F_SR_NA_1
122	Call directory, select file, call file, call section F_SC_NA_1
123	Last section, last segment F_LS_NA_1
124	Ack file, Ack section F_AF_NA_1
125	Segment F_SG_NA_1
126	F_DR_TA_1
127	QueryLog - Request archive file F_SC_NB_1

4.6.5 Deep Packet Inspection - AMP Enforcer

[Network Security > DPI > AMP Enforcer]

This dialog lets you specify the *AMP Enforcer* (*ASCII Message Protocol Enforcer*) settings and define the *AMP Enforcer* specific profiles.

The ASCII Message Protocol (AMP) is a communication protocol widely used in the automation industry for *Supervisory Control and Data Acquisition* (SCADA) and system integration. The ASCII Message Protocol (AMP) is designed to help ensure reliable communication between industrial equipment. The ASCII Message Protocol (AMP) is used to monitor and control industrial automation equipment such as Programmable Logic Controllers (PLCs), sensors, and meters.

The device uses the Deep Packet Inspection (DPI) function to discard data packets that violate one of the specified profiles. The *AMP Enforcer* function supports *Common ASCII Message Protocol* (CAMP) and *Non-Intelligent Terminal Protocol* (NITP) using *TCP*. The device uses the *AMP Enforcer* function to perform the *DPI* function on the *CAMP* and *NITP* data stream. The device performs the *DPI* function based on the *Program and mode protect* function and the specified profiles.

When the *AMP Enforcer* profile is active, the device applies the profiles to the data stream. The device permits only data packets that contain the values specified in the following columns depending on the status of the *Program and mode protect* function:

- *Protocol*
- *Message type*
- *Address class*
- *Device class*
- *Memory address*
- *Data word*
- *Task code*
- *Task code data*
- *Block check characters*
- *Error check characters*
- *Sanity check*

The menu contains the following dialogs:

- *AMP Global*
- *AMP Profile*

4.6.5.1 AMP Global

[Network Security > DPI > AMP Enforcer > Global]

In this dialog, you specify the global settings for the *AMP Enforcer* profile.

Protect mode

Program and mode protect

Activates/deactivates the inspection of the data packets that contain the *Task codes* with the value `config` in the *Mode* column.

Possible values:

`marked` (default setting)

The inspection is active. The device forwards only the data packets that match the parameters specified in the profiles. The device discards data packets that contain the value `config` in the *Mode* column for the *Task codes* specified in the profiles.

`unmarked`

The inspection is inactive. The device forwards the data packets that match the parameters specified in the profiles, including the data packets that contain *Task codes* with the value `config` in the *Mode* column.

Operation


Uncommitted changes present

Displays if the *AMP Enforcer* profiles applied to the data stream differ from the profiles saved in the device.

Possible values:

`marked`

At least one of the active *AMP Enforcer* profiles saved in the device contains modified settings.

When you click the  button, the device applies the specified profiles.

`unmarked`

The *AMP Enforcer* profiles applied to the data stream match the profiles saved in the device.

Table

For information on how to customize the appearance of the table, see [“Working with tables” on page 16](#).

Buttons



Add

Opens the [Create](#) window to add a table row.

- In the [Task code](#) field, you specify the number of the profile.

Possible values:

00 . FF

When you click the [Ok](#) button, the device adds the table row. The device assigns the [Task code](#) specified in the [Task code](#) field to the table row.



Remove

Removes the selected table row.



Commit changes

The device applies the specified profiles to the data stream.

If you changed the values in the field, then the device assigns the specific values to the related profile.

Task code

Specifies the user-defined [Task code](#) for the [AMP Enforcer](#) profile represented by 2 ASCII characters. The [Task codes](#) are the command or response messages associated with:

- modification of the configuration, application program, or operational mode of the equipment.
- read or write the equipment data.

Possible values:

00 . FF

You find the meaning of the default [Task codes](#) in section [“Meaning of the Task code values” on page 204](#).

Description

Specifies a name for the *Task code*.

Possible values:

Alphanumeric ASCII character string with 0..64 characters

Mode

Specifies the mode applicable for the *Task code*.

Possible values:

`conf i g`

Specifies commands associated with the modification of the controller settings, the application program or the operational mode.

`non-conf i g`

Specifies read/write commands, excluding the commands associated with modification of the controller settings, application program or operational mode.

4.6.5.2 AMP Profile

[Network Security > DPI > AMP Enforcer > Profile]

This dialog lets you set up profiles for the *AMP Enforcer* function. The profile lets you forward or discard data packets based on the specified values.

Table

For information on how to customize the appearance of the table, see [“Working with tables” on page 16](#).

Buttons



Add

Opens the *Create* window to add a table row.

- In the *Index* field, you specify the number of the profile.
Possible values:

1 . 32

When you click the *Ok* button, the device adds the table row. The device assigns the number specified in the *Index* field to the table row.



Remove

Removes the selected table row.

If you mark the *Profile active* checkbox for the profile, then the device stops you from removing the profile.



Copy

Opens the *Copy* window to copy an existing table row. The prerequisite is that the table row for the profile to be copied is selected.

- In the *Index* field, you specify the new number of the copied profile.
Possible values:

1 . 32

The device adds the table row. The device assigns the number specified in the *Index* field to the table row.

Index

Displays the sequential number of the profile to which the table row relates. You specify the index number when you add a table row.

Description

Specifies a name for the profile.

Possible values:

Alphanumeric ASCII character string with 0..32 characters
(default setting: [amp](#))

Protocol

Specifies the TCP payload protocol type of the data packets to which the device applies the profile. The device applies the profile only to data packets that contain the specified value in the *Protocol* field.

Possible values:

[camp](#)
Common ASCII Message Protocol

[ni tp](#)
Non-Intelligent Terminal Protocol

[any](#) (default setting)
The device applies the profile to every data packet without evaluating the protocol.

Message type

Specifies if the message is of the type *command* or *response*. The prerequisite is that in the *Protocol* column the value [camp](#) is specified.

Possible values:

[any](#) (default setting)
The device applies the profile to every data packet without evaluating the message type.

[00 . 03](#) and [FF](#)

The device applies the profile only to data packets that contain the specified message type. The field lets you specify the following options:

- You specify a message type with a single hexadecimal value.
Example: [02](#)
- You specify multiple individual message types with comma-separated hexadecimal values.
Example: [02, 03, FF](#)

[00 . 01, 04 . 09](#) and [FF](#)

The device applies the profile only to data packets that contain the specified message type. The field lets you specify the following options:

- You specify a message type with a single hexadecimal value.
Example: [04](#)
- You specify multiple individual message types with comma-separated hexadecimal values.
Example: [04, 05, 06, FF](#)

You find the meaning of the hexadecimal values in section “[Meaning of the Message type values](#)” on page 205.

Address class

Specifies the particular type of the memory to be accessed on the equipment.

Prerequisites:

- In the *Protocol* column, the value [camp](#) is specified.
- In the *Message type* column, a hexadecimal value in the range [00 . 01](#) or [04 . 09](#) or the hexadecimal value [FF](#) is specified.

Possible values:

[any](#) (default setting)

The device applies the profile to every data packet without evaluating the address class.

[0000 . FFFF](#)

The device applies the profile only to data packets that contain the specified address class.

The field lets you specify the following options:

- You specify an address class with a single hexadecimal value.
Example: [0000](#)
- You specify multiple individual address classes with the hexadecimal values separated by a comma.
Example: [0000, 0003, FFFF](#)
- You specify an address class range with hexadecimal values connected by a dash.
Example: [0004-000A](#)
- You can also combine address classes and address class ranges.
Example: [0000, 0003, 0004-000A](#)

The field lets you specify up to 205 hexadecimal values. When you enter [0000, 0003, 0004-000A](#), for example, you use 4 of 205 hexadecimal values.

Device class

Specifies the type of device class (vendor specific device) to be accessed.

Prerequisites:

- In the *Protocol* column, the value [camp](#) is specified.
- In the *Message type* column, a hexadecimal value in the range [00 . 03](#) or the hexadecimal value [FF](#) is specified.

Possible values:

[any](#) (default setting)

The device applies the profile to every data packet without evaluating the device class.

[0000 . FFFF](#)

The device applies the profile only to data packets that contain the specified device class.

The field lets you specify the following options:

- You specify a device class with a single hexadecimal value.
Example: [0000](#)
- You specify multiple individual device classes with hexadecimal values separated by a comma.
Example: [0000, 0003, FFFF](#)
- You specify a device class range with hexadecimal values connected by a dash.
Example: [0004-000A](#)
- You can also combine device classes and device class ranges.
Example: [0000, 0003, 0004-000A](#)

The field lets you specify up to 205 hexadecimal values. When you enter [0000, 0003, 0004-000A](#), for example, you use 4 of 205 hexadecimal values.

Memory address

Specifies the starting address of the memory to be read or written.

Prerequisites:

- In the *Protocol* column, the value [camp](#) is specified.
- In the *Message type* column, a hexadecimal value in the range [00 . 01](#) or [04 . 09](#) or the hexadecimal value [FF](#) is specified.

Possible values:

[any](#) (default setting)

The device applies the profile to every data packet without evaluating the memory address.

[0000 . FFFF](#)

The device applies the profile only to data packets that contain the specified memory address.

The field lets you specify the following options:

- You specify a memory address with a single hexadecimal value.

Example: [0000](#)

- You specify multiple individual memory addresses with hexadecimal values separated by a comma.

Example: [0000, 0003, FFFF](#)

- You specify a memory address range with hexadecimal values connected by a dash.

Example: [0004-000A](#)

- You can also combine memory addresses and memory address ranges.

Example: [0000, 0003, 0004-000A](#)

The field lets you specify up to 205 hexadecimal values. When you enter [0000, 0003, 0004-000A](#), for example, you use 4 of 205 hexadecimal values.

Data word

Specifies the starting address that the equipment uses to read data from the packet.

Prerequisites:

- In the *Protocol* column, the value [camp](#) is specified.
- In the *Message type* column, a hexadecimal value in the range [00 . 01](#) or [08 . 09](#) or the hexadecimal value [FF](#) is specified.

Possible values:

[any](#) (default setting)

The device applies the profile to every data packet without evaluating the data word.

[0000 . FFFF](#)

The device applies the profile only to data packets that contain the specified data word.

The field lets you specify the following options:

- You specify a data word with a single hexadecimal value.

Example: [0000](#)

- You specify multiple individual data words with hexadecimal values separated by a comma.

Example: [0000, 0003, FFFF](#)

- You specify a data word range with hexadecimal values connected by a dash.

Example: [0004-000A](#)

- You can also combine data words and data word ranges.

Example: [0000, 0003, 0004-000A](#)

The field lets you specify up to 205 hexadecimal values. When you enter [0000, 0003, 0004-000A](#), for example, you use 4 of 205 hexadecimal values.

Task code

Displays the *Task codes* of the *AMP Enforcer* profile. You can add user-specific *Task codes* in the *Network Security > DPI > AMP Enforcer > Global* dialog.

The prerequisite is that in the *Protocol* column one of the following values is specified:

- [ni tp](#)
- [camp](#)
Additionally, in the *Message type* column, a hexadecimal value in the range [00 . 03](#) or the hexadecimal value [FF](#) is specified.
- [any](#)
Additionally, in the *Message type* column, the value [any](#) is specified.

The device lets you specify multiple *Task codes*. To do this, perform the following steps:

- Click in the *Task code* column of the relevant profile.
- The dialog displays the *Task code* window.
- Select the desired *Task code* from the *Task code* drop-down list.
- Click the *Add* button.
- To add multiple *Task codes*, repeat the previously described steps.
- Click the *Ok* button.

Possible values:

any (default setting)

The device applies every *Task code* available in the *Available task codes* field.

00 . FF

The device permits data packets with the specified codes.

The field lets you specify the following options:

- A single *Task code* with a single hexadecimal value.
Example: *00*
- Multiple *Task codes* with hexadecimal values separated by a comma.
Example: *00, 01, 02*

You find the meaning of the hexadecimal values in section “[Meaning of the Task code values](#)” on page 204.

Task code data

Specifies the task code data for the *Task code*.

The prerequisite is that in the *Protocol* column one of the following values is specified:

- *camp*
Additionally, in the *Message type* column, a hexadecimal value in the range *00 . 03* or the hexadecimal value *FF*, and in the *Task code* column a single hexadecimal value are specified.
- *ni tp*
Additionally, in the *Task code* column, a single hexadecimal value is specified.

Possible values:

0 . F

The device applies the profile only to data packet that contains the specified task code data. The maximum length is 72 bytes.

Error check characters

Activates/deactivates the error checking of the characters contained in the *CAMP* and *NITP* data packets.

Prerequisite:

- In the *Protocol* column, the value *camp* and in the *Message type* column, a hexadecimal value in the range *00 . 03* or the hexadecimal value *FF* is specified.
or
- In the *Protocol* column, the value *ni tp* is specified.

Possible values:

marked (default setting)

The checking is active.

unmarked

The checking is inactive.

Block check characters

Activates/deactivates the checking of *block check characters* to validate the checksum contained in the *CAMP* data packets.

Prerequisites:

- In the *Protocol* column, the value *camp* is specified.
- In the *Message type* column, a hexadecimal value in the range *00 . 09* or the hexadecimal value *FF* is specified.

Possible values:

marked (default setting)

The checking is active.

unmarked

The checking is inactive.

Debug

Activates/deactivates the debugging of the profiles.

Possible values:

marked

Debugging is active.

The device sends the reset packet along with the information related to the termination of TCP connection. The prerequisite is that in the *TCP reset* column the checkbox is marked.

unmarked (default setting)

Debugging is inactive.

TCP reset

Activates/deactivates the resetting of the TCP connection in case of a protocol violation or if the plausibility check detects an error.

Possible values:

marked (default setting)

The resetting of the TCP connection is active.

If the device identifies a protocol violation or detects a plausibility check error, then the device terminates the TCP connection. The device establishes the TCP connection again on receiving a new connection request.

unmarked

The resetting of the TCP connection is inactive.

Sanity check

Activates/deactivates the plausibility check for the data packets.

Possible values:

marked (default setting)

The plausibility check is active.

The device checks the plausibility of the data packets regarding format and specification. The device blocks the data packets that violate the specified profiles.

unmarked

The plausibility check is inactive.

Profile active

Activates/deactivates the profile.

Possible values:

marked

The profile is active.

The device applies the *AMP Enforcer* profiles specified in this table row to the data packets.

unmarked

The profile is inactive.

[Task code]

Task code

Specifies the *Task codes* for the relevant *AMP Enforcer* profile.

You find the meaning of the hexadecimal values in section “[Meaning of the Task code values](#)” on [page 204](#).

Add

Adds the items you selected from the drop-down list to the *AMP Enforcer* field.



Removes the item from the *AMP Enforcer* field.

Meaning of the Task code values

#	Meaning
01	Read Word Memory Random
02	Write Word Memory Area Random
30	Read Operational Status
32	Program to Run Mode
33	Go to Program Mode
34	Execute Power-up
35	Execute Complete (Warm) Start
36	Execute Partial (Hot) Start
50	Read User Word Area Block
51	Write User Word Area Starting at Address
58	Set Controller Time of Day Clock
59	Write Discrete I/O Status or Force via Data Element Type
5A	Write Block
6B	Read Discrete I/O Status or Force via Data Element Type
71	Read Controller Time of Day Clock
7D	Read SF/Loop Processor Mode
7E	Read Random

#	Meaning
7F	Read Block
88	Select Number of SF Module Task Codes Per Scan
89	Read Number of SF Module Task Codes Per Scan
99	Write VME Memory Area Block/Random
9A	Read VME Memory Area Block/Random

Meaning of the Message type values

#	Meaning
00	Module General Query Command
01	Module General Response Command
02	Packet T/C Command
03	Packed T/C Response
04	Read data Command
05	Read data Response
06	Write data Command
07	Write data Response
08	Mem Exch Command
09	Mem Exch Response
FF	Protocol Error

4.6.6 Deep Packet Inspection - ENIP Enforcer

[Network Security > DPI > ENIP Enforcer]

This dialog lets you specify the *ENIP Enforcer* (*Ethernet Industrial Protocol Enforcer*) settings and define the *ENIP Enforcer* specific profiles.

The Ethernet Industrial Protocol (ENIP) is part of the Common Industrial Protocol (CIP). The Common Industrial Protocol (CIP) defines the object structure and specifies the message transfer. The *ENIP Enforcer* function applies the Deep Packet Inspection (DPI) function to the ENIP and CIP data stream. The Ethernet Industrial Protocol (ENIP) is used to monitor and control industrial automation equipment such as PLCs (Programmable Logic Controllers), sensors, and meters.

The device uses the *ENIP Enforcer* function to perform the DPI function on the data stream. The device performs the DPI function based on the values defined in the specified profiles. The device blocks the data packets that violate the specified profiles.

Note: The *ENIP Enforcer* function performs the DPI function only on packets that contain an *explicit request*, and drops packets that contain an *implicit request*. An *explicit request* contains CIP message over TCP. An *implicit request* contains CIP message over UDP.

When the *ENIP Enforcer* profile is active, the device applies the profile to the data stream. The device permits only data packets containing the values specified in the following columns:

- *Function type*
- *Sanity check*
- *Default object list*

- [Wildcard service codes](#)
- [Allow embedded PCCC \(Programmable Controller Communication Commands\)](#)

The menu contains the following dialogs:

- [ENIP Profile](#)
- [ENIP Object](#)

4.6.6.1 ENIP Profile

[Network Security > DPI > ENIP Enforcer > Profile]

In this dialog, you specify the global settings for the *ENIP Enforcer* profile.

Operation


Uncommitted changes present

Displays if the *ENIP Enforcer* profiles applied to the data stream differ from the profiles saved in the device.

Possible values:

marked

At least one of the active *ENIP Enforcer* profiles saved in the device contains modified settings.

When you click the  button, the device applies the specified profiles.

unmarked

The *ENIP Enforcer* profiles applied to the data stream match the profiles saved in the device.

Table

For information on how to customize the appearance of the table, see “[Working with tables](#)” on [page 16](#).

Buttons



Add

Opens the *Create* window to add a table row.

- In the *Index* field, you specify the number of the profile.

Possible values:

1 . 32

When you click the *Ok* button, the device adds the table row. The device assigns the number specified in the *Index* field to the table row.



Remove

Removes the selected table row.

If you mark the *Profile active* checkbox for the profile, then the device stops you from removing the profile.



Copy

Opens the [Copy](#) window to copy an existing table row. The prerequisite is that the table row for the profile to be copied is selected.

- In the [Index](#) field, you specify the new number of the copied profile.

Possible values:

[1 . 32](#)

The device adds the table row. The device assigns the number specified in the [Index](#) field to the table row.



Commit changes

The device applies the specified profiles to the data stream.

If you changed the values in the [Function type](#) field, then the device assigns the specific values to the related profile.

Index

Displays the sequential number of the profile to which the table row relates. You specify the index number when you add a table row.

Description

Specifies a name for the profile.

Possible values:

Alphanumeric ASCII character string with 0..32 characters
(default setting: [eni p](#))

Function type

Specifies the function type for the [ENIP Enforcer](#) profile. After clicking the button, the device assigns the corresponding *class IDs* and *service codes*.

Possible values:

[readonly](#)

Assigns the *class IDs* for the *read* function.

You find the list of the readonly *class IDs* in [table 4 on page 221](#).

[readwrite](#)

Assigns the *class IDs* for the *read/write* functions.

You find the list of the read/write *class IDs* in [table 5 on page 226](#).

[any](#) (default setting)

Assigns the *class IDs* for every function. You cannot specify user-defined *class IDs* through the [Object](#) value if the function type is *any*.

[advanced](#)

Lets you specify user-defined *class IDs*.

Allow embedded PCCC

Activates/deactivates DPI for *PCCC messages* encapsulated in data packets. *PCCC messages* are embedded within the Ethernet Industrial Protocol (ENIP). Activating this setting is useful when securing network traffic to and from PLC-5 and MicroLogix controllers.

Possible values:

marked

DPI for *PCCC messages* is active. The device assigns the *command codes* and *function codes*, corresponding to the value you specify in the *Function type* column.

You find the lists of the *command codes* and *function codes* in following tables:

- See table 6 on page 236.
- See table 7 on page 236.
- See table 8 on page 238.

unmarked (default setting)

DPI for *PCCC messages* is inactive.

Sanity check

Activates/deactivates the plausibility check for the data packets.

Possible values:

marked (default setting)

The plausibility check is active.

The device checks the plausibility of the data packets regarding format and specification. The device blocks the data packets that violate the specified profiles.

unmarked

The plausibility check is inactive.

TCP reset

Activates/deactivates the resetting of the TCP connection in case of a protocol violation or if the plausibility check detects an error.

Possible values:

marked (default setting)

The resetting of the TCP connection is active.

If the device identifies a protocol violation or detects a plausibility check error, then the device terminates the TCP connection. The device establishes the TCP connection again on receiving a new connection request.

unmarked

The resetting of the TCP connection is inactive.

Debug

Activates/deactivates the debugging of the profiles.

Possible values:

marked

Debugging is active.

The device sends the reset packet along with the information related to the termination of TCP connection. The prerequisite is that in the *TCP reset* column the checkbox is marked.

unmarked (default setting)

Debugging is inactive.

Default object list

Specifies the *index numbers* used in the *default object list*.

Possible values:

`all`

The device applies the *ENIP Enforcer* profile to every data packet regardless of the *index number*.

`1..347`

The device applies the *ENIP Enforcer* profile only to data packets containing *class IDs* and *service codes* in the specified *index number*.

The field lets you specify the following options:

- You specify a single *index number* with a single numerical value.

Example: `1`

- You specify multiple *index numbers* with numerical values separated by a comma.

Example: `1, 2, 3`

- You specify an *index number* range with numerical values connected by a dash.

Example: `7-25`

- You can also combine *index numbers* and *index number* ranges.

Example: `2, 7-25, 56`

The field lets you specify up to 347 numerical values. When you enter `2, 7-25, 56`, for example, you use 4 of 347 numerical values.

You find the list of the *class IDs* and corresponding *service codes* in [table 3 on page 212](#).

`none` (default setting)

The device does not apply the *index number* to the *ENIP Enforcer* profile.

Wildcard service codes

Specifies the *service codes* which device permits with any valid *class IDs*.

Possible values:

`0x00..0x7F`

The device applies the profile only to data packets that contain the specified *service codes*.

The field lets you specify the following options:

- You specify a service list with a single hexadecimal value.

Example: `0x00`

- You specify multiple individual *service codes* with comma-separated hexadecimal values.

Example: `0x02, 0x03, 0x04, 0x05`

The field lets you specify up to 128 hexadecimal values. When you enter `0x02, 0x03, 0x04, 0x05`, for example, you use 4 of 128 hexadecimal values.

Profile active

Activates/deactivates the profile.

Possible values:

`marked`

The profile is active.

The device applies the *ENIP Enforcer* profiles specified in this table row to the data packets.

`unmarked` (default setting)

The profile is inactive.

4.6.6.2 ENIP Object

[Network Security > DPI > ENIP Enforcer > Object]

The ENIP function uses objects to transmit values and information between devices. The ENIP function uses *class IDs* and *service codes* to specify how the data within the object is encoded. Each instance of an encoded information element that defines a unique *class ID* and a unique *service code* in a message, is an ENIP object.

This window lets you add custom ENIP objects and also lets you view the previously added custom ENIP objects. To verify that an added ENIP object is valid, check the following parameters:

- [Class ID](#)
- [Service codes](#)

Table

For information on how to customize the appearance of the table, see [“Working with tables” on page 16](#).

Buttons



Add

Opens the [Create](#) window to add a table row.

- From the [Index](#) drop-down list, you select the profile *index number*.
- In the [Class ID](#) field, you specify the user-defined *class IDs*.

Possible values:

[0x00](#) . [0xFFFFFFFF](#)

- In the [Service codes](#) field, you specify the *service codes*.

Possible values:

[0x00](#) . [0x7F](#)

When you click the [Ok](#) button, the device adds the table row. The device assigns the values specified in the [Index](#), [Class ID](#) and [Service codes](#) fields to this table row.



Remove

Removes the selected table row.

Index

Displays the number of the profile to which the table row relates. You specify the index number when you add a table row.

Class ID

Specifies the user-defined *class IDs* for the *ENIP Enforcer* profile.

Possible values:

0x00 . 0xFFFFFFFF

Service codes

Specifies the *service codes*.

Possible values:

0x00 . 0x7F

The device applies the profile only to data packets that contain the specified *service codes*.

The field lets you specify the following options:

- You specify a service list with a single hexadecimal value.
 Example: 0x00
- You specify multiple individual *service codes* with comma-separated hexadecimal values.
 Example: 0x02, 0x03, 0x04, 0x05

The field lets you specify up to 128 hexadecimal values. When you enter 0x02, 0x03, 0x04, 0x05, for example, you use 4 of 128 hexadecimal values.

Description

Displays the name of the object.

[Default object list]

Table 3: Default object list

Index	Class ID	Service codes
1	0x01 = Identity	0x01=Get Attributes All
2		0x05= Reset
3		0x0E= Get Attribute Signal
4		0x10= Set Attribute Signal
5		0x11= Find Next Object Instance
6		0x18= Get Member
7	0x02 = Message Router	0x01= Get Attributes All
8		0x0E = Get Attribute Single
9		0x4B = Write Data Table (Rockwell)
10	0x04 = Assembly	0x08 = Create
11		0x09 = Delete
12		0x0E = Get Attribute Single
13		0x10 = Set Attribute Single
14		0x18 = Get Member
15		0x19 = Set Member
16		0x1A = Insert Member
17		0x1B = Remove Member

Table 3: Default object list (cont.)

Index	Class ID	Service codes
18	Ox05 = Connecti on	Ox05 = Reset
19		Ox08 = Create
20		Ox09 = Del ete
21		Ox0D = Appl y Attri butes
22		Ox0E = Get Attri bute Si ngl e
23		Ox10 = Set Attri bute Si ngl e
24		Ox11 = Fi nd Next Obj ect Instance
25		Ox4B = Connecti on Bi nd
26		Ox4C = Producti on Appli cati on Lookup
27		Ox4E = Safety Cl ose
28		Ox54 = Safety Open
29		Ox06 = Off-Li nk Connecti on Manager ¹
30	Ox02 = Set Attri butes Al l	
31	Ox0E = Get Attri bute Si ngl e	
32	Ox10 = Set Attri bute Si ngl e	
33	Ox4E = Forward Cl ose	
34	Ox52 = Unconnected Send	
35	Ox54 = Forward Open	
36	Ox56 = Get Connecti on Data	
37	Ox57 = Search Connecti on Data	
38	Ox5A = Get Connecti on Owner	
39	Ox5B = Large Forward Open	
40	Ox07 = Regi ster	
41		Ox10 = Set Attri bute Si ngl e
42	Ox08 = Di screte I nput Poi nt	Ox01 = Get Attri butes Al l
43		Ox02 = Set Attri butes Al l
44		Ox0E = Get Attri bute Si ngl e
45		Ox10 = Set Attri bute Si ngl e
46	Ox09 = Di screte Output Poi nt	Ox01 = Get Attri butes Al l
47		Ox02 = Set Attri butes Al l
48		Ox0E = Get Attri bute Si ngl e
49		Ox10 = Set Attri bute Si ngl e
50	Ox0A = Anal og I nput Poi nt	Ox01 = Get Attri butes Al l
51		Ox02 = Set Attri butes Al l
52		Ox0E = Get Attri bute Si ngl e
53		Ox10 = Set Attri bute Si ngl e
54	Ox0B = Anal og Output Poi nt	Ox01 = Get Attri butes Al l
55		Ox02 = Set Attri butes Al l
56		Ox0E = Get Attri bute Si ngl e
57		Ox10 = Set Attri bute Si ngl e
58	Ox0E = Presence Sensi ng	Ox0E = Get Attri bute Si ngl e
59		Ox10 = Set Attri bute Si ngl e

Table 3: Default object list (cont.)

Index	Class ID	Service codes
60	0x0F = Parameter	0x01 = Get Attributes All
61		0x05 = Reset
62		0x0D = Apply Attributes
63		0x0E = Get Attribute Single
64		0x10 = Set Attribute Single
65		0x15 = Restore
66		0x16 = Save
67		0x18 = Get Member
68		0x4B = Get EnumString
69	0x10 = Parameter Group	0x01 = Get Attributes All
70		0x0E = Get Attribute Single
71		0x10 = Set Attribute Single
72	0x12 = Group	0x01 = Get Attributes All
73		0x0E = Get Attribute Single
74	0x1D = Discrete Input Group	0x01 = Get Attributes All
75		0x02 = Set Attributes All
76		0x0E = Get Attribute Single
77		0x10 = Set Attribute Single
78	0x1E = Discrete Output Group	0x01 = Get Attributes All
79		0x02 = Set Attributes All
80		0x0E = Get Attribute Single
81		0x10 = Set Attribute Single
82	0x1F = Discrete Group	0x01 = Get Attributes All
83		0x0E = Get Attribute Single
84	0x20 = Analog Input Group	0x01 = Get Attributes All
85		0x02 = Set Attributes All
86		0x0E = Get Attribute Single
87		0x10 = Set Attribute Single
88	0x21 = Analog Output Group	0x01 = Get Attributes All
89		0x02 = Set Attributes All
90		0x0E = Get Attribute Single
91		0x10 = Set Attribute Single
92	0x22 = Analog Group	0x01 = Get Attributes All
93		0x0E = Get Attribute Single
94		0x10 = Set Attribute Single

Table 3: Default object list (cont.)

Index	Class ID	Service codes
95	Ox23 = Position Sensor Object	Ox05 = Reset
96		Ox0D = Apply Attributes
97		Ox0E = Get Attribute Single
98		Ox10 = Set Attribute Single
99		Ox15 = Restore
100		Ox16 = Save
101		Ox18 = Get Member
102	Ox19 = Set Member	
103	Ox24 = Position Controller Supervisor Object	Ox0E = Get Attribute Single
104		Ox10 = Set Attribute Single
105	Ox25 = Position Controller Object	Ox0E = Get Attribute Single
106		Ox10 = Set Attribute Single
107	Ox26 = Block Sequencer Object	Ox0E = Get Attribute Single
108		Ox10 = Set Attribute Single
109	Ox27 = Command Block Object	Ox0E = Get Attribute Single
110		Ox10 = Set Attribute Single
111	Ox28 = Motor Data Object	Ox0E = Get Attribute Single
112		Ox10 = Set Attribute Single
113		Ox15 = Restore
114		Ox16 = Save
115	Ox29 = Control Supervisor Object	Ox0E = Get Attribute Single
116		Ox10 = Set Attribute Single
117		Ox05 = Reset
118	Ox2A = AC/DC Drive Object	Ox0E = Get Attribute Single
119		Ox10 = Set Attribute Single
120		Ox15 = Restore
121		Ox16 = Save
122	Ox2B = Acknowledge Handler Object	Ox08 = Create
123		Ox09 = Delete
124		Ox0E = Get Attribute Single
125		Ox10 = Set Attribute Single
126		Ox4B = Add AckData Path
127	Ox4C = Remove AckData Path	
128	Ox2C = Overload Object	Ox0E = Get Attribute Single
129		Ox10 = Set Attribute Single
130		Ox15 = Restore
131		Ox16 = Save
132	Ox2D = Softstart Object	Ox0E = Get Attribute Single
133		Ox10 = Set Attribute Single
134		Ox15 = Restore
135		Ox16 = Save

Table 3: Default object list (cont.)

Index	Class ID	Service codes
136	Ox2E = Selection Object	Ox05 = Reset
137		Ox06 = Start
138		Ox07 = Stop
139		Ox08 = Create
140		Ox09 = Delete
141		Ox0E = Get Attribute Single
142		Ox10 = Set Attribute Single
143		Ox18 = Get Member
144		Ox19 = Set Member
145		Ox1A = Insert Member
146	Ox1B = Remove Member	
147	Ox30 = S-Device Supervisor Object	Ox05 = Reset
148		Ox06 = Start
149		Ox07 = Stop
150		Ox0E = Get Attribute Single
151		Ox10 = Set Attribute Single
152		Ox4B = Abort
153		Ox4C = Recover
154		Ox4E = Perform Diagnostics
155	Ox31 = S-Analog Sensor Object	Ox01 = Get Attributes All
156		Ox0E = Get Attribute Single
157		Ox4B = Zero Adjust
158		Ox4C = Gain Adjust
159	Ox32 = S-Analog Actuator Object	Ox0E = Get Attribute Single
160		Ox10 = Set Attribute Single
161	Ox33 = S-Single Stage Controller Object	Ox0E = Get Attribute Single
162		Ox10 = Set Attribute Single
163		Ox63 = Calibrate
164	Ox34 = S-Gas Calibration Object	Ox0E = Get Attribute Single
165		Ox10 = Set Attribute Single
166		Ox4B = Get All Instances
167	Ox35 = Trip Point Object	Ox0E = Get Attribute Single
168		Ox10 = Set Attribute Single

Table 3: Default object list (cont.)

Index	Class ID	Service codes
169	Ox37 = File Object	Ox06 = Start
170		Ox07 = Stop
171		Ox08 = Create
172		Ox09 = Delete
173		Ox0E = Get Attribute Single
174		Ox10 = Set Attribute Single
175		Ox15 = Restore
176		Ox16 = Save
177		Ox18 = Get Member
178		Ox4B = Initiate Upload
179		Ox4C = Initiate Download
180		Ox4D = Initiate Partial Read
181		Ox4E = Initiate Partial Write
182		Ox4F = Upload Transfer
183		Ox50 = Download Transfer
184		Ox51 = Clear File
185	Ox38 = S-Partial Pressure Object	Ox01 = Get Attributes All
186		Ox08 = Create
187		Ox09 = Delete
188		Ox0E = Get Attribute Single
189		Ox10 = Set Attribute Single
190		Ox4B = Create Range
191		Ox4C = Get Instance List
192		Ox4D = Get Pressures
193		Ox4E = Get All Pressures
194		Ox4F = Group Enable
195	Ox40 = S-Sensor Calibration Object	Ox0E = Get Attribute Single
196		Ox10 = Set Attribute Single
197		Ox4B = Get all Instances
198	Ox41 = Event Log Object	Ox05 = Reset
199		Ox06 = Start
200		Ox07 = Stop
201		Ox0E = Get Attribute Single
202		Ox10 = Set Attribute Single
203		Ox18 = Get Member
204		Ox19 = Set Member
205		Ox1A = Insert Member
206		Ox1B = Remove Member

Table 3: Default object list (cont.)

Index	Class ID	Service codes
207	0x42 = Motion Device Axis Object	0x03 = Get Attribute List
208		0x04 = Set Attribute List
209		0x0E = Get Attribute Single
210		0x10 = Set Attribute Single
211		0x1C = GroupSync
212		0x4B = Get Axis Attributes List
213		0x4C = Set Axis Attributes List
214		0x4D = Set Cyclic Write List
215		0x4E = Set Cyclic Read List
216		0x4F = Run Motor Test
217		0x50 = Get Motor Test Data
218		0x51 = Run Inertia Test
219		0x52 = Get Inertia Test Data
220	0x53 = Run Hookup Test	
221	0x54 = Get Hookup Test Data	
222	0x43 = Time Sync Object	0x01 = Get Attributes All
223		0x03 = Get Attribute List
224		0x04 = Set Attribute List
225		0x0E = Get Attribute Single
226		0x10 = Set Attribute Single
227	0x44 = Modbus Object	0x0E = Get Attribute Single
228		0x4B = Read Discrete Inputs
229		0x4C = Read Coils
230		0x4D = Read Input Registers
231		0x4E = Read Holding Registers
232		0x4F = Write Coils
233		0x50 = Write Holding Registers
234		0x51 = Modbus Passthrough
235	0x45 = Originator Connection List Object	0x08 = Create
236		0x09 = Delete
237		0x4C = Connection Read
238	0x46 = Modbus Serial Link Object	0x01 = Get Attributes All
239		0x05 = Reset
240		0x0E = Get Attribute Single
241		0x10 = Set Attribute Single
242		0x4B = Get And Clear

Table 3: Default object list (cont.)

Index	Class ID	Service codes	
243	Ox47 = Device Level Ring (DLR) Object	Ox01 = Get Attributes All	
244		Ox0E = Get Attribute Single	
245		Ox10 = Set Attribute Single	
246		Ox18 = Get Member	
247		Ox4B = Verify Fault Location	
248		Ox4C = Clear Rapid Faults	
249		Ox4D = Restart Sign On	
250		Ox4E = Clear Gateway Partial Fault	
251		Ox48 = QoS Object	Ox01 = Get Attributes All
252			Ox0E = Get Attribute Single
253	Ox10 = Set Attribute Single		
254	Ox4D = Target Connection List Object	Ox01 = Get Attributes All	
255		Ox0E = Get Attribute Single	
256		Ox4C = Connection Read	
257	Ox4E = Base Energy Object	Ox01 = Get Attributes All	
258		Ox03 = Get Attribute List	
259		Ox04 = Set Attribute List	
260		Ox05 = Reset	
261		Ox08 = Create	
262		Ox09 = Delete	
263		Ox0E = Get Attribute Single	
264		Ox10 = Set Attribute Single	
265		Ox18 = Get Member	
266		Ox19 = Set Member	
267		Ox1A = Insert Member	
268		Ox1B = Remove Member	
269		Ox4B = Start Metering	
270		Ox4C = Stop Metering	
271		Ox4F = Electrical Energy Object	Ox01 = Get Attributes All
272	Ox03 = Get Attribute List		
273	Ox0E = Get Attribute Single		
274	Ox50 = Non-Electrical Energy Object	Ox01 = Get Attributes All	
275		Ox03 = Get Attribute List	
276		Ox0E = Get Attribute Single	
277	Ox51 = Base Switch Object	Ox01 = Get Attributes All	
278		Ox0E = Get Attribute Single	
279		Ox10 = Set Attribute Single	
280	Ox52 = SNMP Object	Ox01 = Get Attributes All	
281		Ox0E = Get Attribute Single	
282		Ox10 = Set Attribute Single	

Table 3: Default object list (cont.)

Index	Class ID	Service codes
283	Ox53 = Power Management Object	Ox01 = Get Attributes All
284		Ox03 = Get Attribute List
285		Ox04 = Set Attribute List
286		Ox0E = Get Attribute Single
287		Ox10 = Set Attribute Single
288		Ox18 = Get Member
289		Ox19 = Set Member
290		Ox4D = Power Management
291		Ox4E = Set Pass Code
292		Ox4F = Clear Pass Code
293	Ox54 = RSTP Bridge Object	Ox01 = Get Attributes All
294		Ox0E = Get Attribute Single
295		Ox10 = Set Attribute Single
296	Ox55 = RSTP Port Object	Ox01 = Get Attributes All
297		Ox0E = Get Attribute Single
298		Ox10 = Set Attribute Single
299	OxF3 = Connection Configuration Object	Ox01 = Get Attributes All
300		Ox02 = Set Attributes All
301		Ox08 = Create
302		Ox09 = Delete
303		Ox0E = Get Attribute Single
304		Ox10 = Set Attribute Single
305		Ox15 = Restore
306		Ox4B = Kick Timer
307		Ox4C = Open Connection
308		Ox4D = Close Connection
309		Ox4E = Stop Connection
310		Ox4F = Change Start
311		Ox50 = Get Status
312		Ox51 = Change Complete
313	Ox52 = Audit Changes	
314	OxF4 = Port Object	Ox01 = Get Attributes All
315		Ox05 = Reset
316		Ox0E = Get Attribute Single
317		Ox10 = Set Attribute Single
318	OxF5 = TCP/IP Interface Object	Ox01 = Get Attributes All
319		Ox02 = Set Attributes All
320		Ox0E = Get Attribute Single
321		Ox10 = Set Attribute Single

Table 3: Default object list (cont.)

Index	Class ID	Service codes
322	0xF6 = EtherNet Link Object	0x01 = Get Attributes All
323		0x0E = Get Attribute Single
324		0x10 = Set Attribute Single
325		0x4C = Get And Clear
326	0x300 = Module Diagnostics	0x01 = Get Attributes All
327		0x0E = Get Attribute Single
328	0x301 = Input/Output	0x01 = Get Attributes All
329		0x0E = Get Attribute Single
330	0x302 = Local Slaves	0x01 = Get Attributes All
331		0x0E = Get Attribute Single
332	0x400 = Service Port Control Object	0x01 = Get Attributes All
333		0x0E = Get Attribute Single
334	0x401 = Dynamic I/O Control Object	0x01 = Get Attributes All
335		0x0E = Get Attribute Single
336	0x402 = Router Diagnostics Object	0x01 = Get Attributes All
337		0x0E = Get Attribute Single
338	0x403 = Router Routing Table Object	0x01 = Get Attributes All
339		0x0E = Get Attribute Single
340	0x404 = SMTP	0x01 = Get Attributes All
341		0x0E = Get Attribute Single
342		0x32 = Clear All
343	0x405 = SNMP	0x01 = Get Attributes All
344		0x0E = Get Attribute Single
345		0x32 = Clear All
346	0x406 = HSBY	0x01 = Get Attributes All
347		0x0E = Get Attribute Single

1. A packet with *Class ID*=0x06 contains embedded CIP messages. In this case, the device performs an additional level of DPI on the data packets that contain the *service codes* 0x4E, 0x52, 0x54 and 0x5B. The device blocks a data packet if it contains other than the preceding *service codes* for this *Class ID*.

[List of the class IDs for different function types]

Table 4: Class IDs for function type *readonly*

Class ID	Service codes
0x01 = Identity	0x01=Get Attributes All
	0x0E= Get Attribute Signal
	0x11= Find Next Object Instance
	0x18= Get Member

Table 4: Class IDs for function type *readonly* (cont.)

Class ID	Service codes
0x02 = Message Router	0x01 = Get Attributes All
	0x0E = Get Attribute Single
	0x4B = Write Data Table (Rockwell)
	0x54
0x04 = Assembly	0x0E = Get Attribute Single
	0x18 = Get Member
0x05 = Connection	0x08 = Create
	0x0E = Get Attribute Single
	0x11 = Find Next Object Instance
	0x4C = Production Application Lookup
0x06 = Off-Link Connection Manager ¹	0x01 = Get Attributes All
	0x0E = Get Attribute Single
	0x4C
	0x4E = Forward Close
	0x52 = Unconnected Send
	0x54 = Forward Open
	0x56 = Get Connection Data
	0x57 = Search Connection Data
	0x59
	0x5A = Get Connection Owner
	0x5B = Large Forward Open
0x07 = Register	0x0E = Get Attribute Single
0x08 = Discrete Input Point	0x01 = Get Attributes All
	0x0E = Get Attribute Single
0x09 = Discrete Output Point	0x01 = Get Attributes All
	0x0E = Get Attribute Single
0x0A = Analog Input Point	0x01 = Get Attributes All
	0x0E = Get Attribute Single
0x0B = Analog Output Point	0x01 = Get Attributes All
	0x0E = Get Attribute Single
0x0E = Presence Sensing	0x0E = Get Attribute Single
0x0F = Parameter	0x01 = Get Attributes All
	0x0E = Get Attribute Single
	0x18 = Get Member
	0x4B = Get Enum String
0x10 = Parameter Group	0x01 = Get Attributes All
	0x0E = Get Attribute Single
0x12 = Group	0x01 = Get Attributes All
	0x0E = Get Attribute Single
0x1D = Discrete Input Group	0x01 = Get Attributes All
	0x0E = Get Attribute Single

Table 4: Class IDs for function type *readonly* (cont.)

Class ID	Service codes
0x1E = Discrete Output Group	0x01 = Get Attributes All
	0x0E = Get Attribute Single
0x1F = Discrete Group	0x01 = Get Attributes All
	0x0E = Get Attribute Single
0x20 = Analog Input Group	0x01 = Get Attributes All
	0x0E = Get Attribute Single
0x21 = Analog Output Group	0x01 = Get Attributes All
	0x0E = Get Attribute Single
0x22 = Analog Group	0x01 = Get Attributes All
	0x0E = Get Attribute Single
0x23 = Position Sensor Object	0x0E = Get Attribute Single
	0x18 = Get Member
0x24 = Position Controller Supervisor Object	0x0E = Get Attribute Single
	0x0E = Get Attribute Single
0x25 = Position Controller Object	0x0E = Get Attribute Single
0x26 = Block Sequencer Object	0x0E = Get Attribute Single
0x27 = Command Block Object	0x0E = Get Attribute Single
0x28 = Motor Data Object	0x0E = Get Attribute Single
0x29 = Control Supervisor Object	0x0E = Get Attribute Single
0x2A = AC/DC Drive Object	0x0E = Get Attribute Single
0x2B = Acknowledge Handler Object	0x0E = Get Attribute Single
0x2C = Overload Object	0x0E = Get Attribute Single
0x2D = Softstart Object	0x0E = Get Attribute Single
0x2E = Selection Object	0x0E = Get Attribute Single
	0x18 = Get Member
0x30 = S-Device Supervisor Object	0x0E = Get Attribute Single
0x31 = S-Analog Sensor Object	0x01 = Get Attributes All
	0x0E = Get Attribute Single
0x32 = S-Analog Actuator Object	0x0E = Get Attribute Single
0x33 = S-Single Stage Controller Object	0x0E = Get Attribute Single
0x34 = S-Gas Calibration Object	0x0E = Get Attribute Single
	0x4B = Get All Instances
0x35 = Trip Point Object	0x0E = Get Attribute Single
0x37 = File Object	0x0E = Get Attribute Single
	0x18 = Get Member
	0x4B = Initiate Upload
	0x4D = Initiate Partial Read
	0x4F = Upload Transfer

Table 4: Class IDs for function type *readonly* (cont.)

Class ID	Service codes
0x38 = S-Partial Pressure Object	0x01 = Get Attributes All
	0x0E = Get Attribute Single
	0x4C = Get Instance List
	0x4D = Get Pressures
	0x4E = Get All Pressures
0x40 = S-Sensor Calibration Object	0x0E = Get Attribute Single
	0x4B = Get all Instances
0x41 = Event Log Object	0x0E = Get Attribute Single
	0x18 = Get Member
0x42 = Motion Device Axis Object	0x03 = Get Attribute List
	0x0E = Get Attribute Single
	0x4B = Get Axis Attributes List
	0x50 = Get Motor Test Data
	0x52 = Get Inertia Test Data
0x43 = Time Sync Object	0x01 = Get Attributes All
	0x03 = Get Attribute List
	0x0E = Get Attribute Single
0x44 = Modbus Object	0x0E = Get Attribute Single
	0x4B = Read Discrete Inputs
	0x4C = Read Coils
	0x4D = Read Input Registers
0x45 = Modbus Object	0x4E = Read Holding Registers
	0x4C = Connection Read
0x46 = Modbus Serial Link Object	0x01 = Get Attributes All
	0x0E = Get Attribute Single
0x47 = Device Level Ring (DLR) Object	0x01 = Get Attributes All
	0x0E = Get Attribute Single
	0x18 = Get Member
0x48 = CoS Object	0x01 = Get Attributes All
	0x0E = Get Attribute Single
0x4D = Target Connection List Object	0x01 = Get Attributes All
	0x0E = Get Attribute Single
0x4E = Base Energy Object	0x01 = Get Attributes All
	0x03 = Get Attribute List
	0x0E = Get Attribute Single
	0x18 = Get Member
0x4F = Electrical Energy Object	0x01 = Get Attributes All
	0x03 = Get Attribute List
	0x0E = Get Attribute Single

Table 4: Class IDs for function type *readonly* (cont.)

Class ID	Service codes
0x50 = Non-Electrical Energy Object	0x01 = Get Attributes All
	0x03 = Get Attribute List
	0x0E = Get Attribute Single
0x51 = Base Switch Object	0x01 = Get Attributes All
	0x0E = Get Attribute Single
0x52 = SNMP Object	0x01 = Get Attributes All
	0x0E = Get Attribute Single
0x53 = Power Management Object	0x01 = Get Attributes All
	0x03 = Get Attribute List
	0x0E = Get Attribute Single
	0x18 = Get Member
0x54 = RSTP Bridge Object	0x01 = Get Attributes All
	0x0E = Get Attribute Single
0x55 = RSTP Port Object	0x01 = Get Attributes All
	0x0E = Get Attribute Single
0x91 = ANSI Extended Symbol Segment	0x03
	0x55
0x6B	0x55
0x6C	0x01
0xAC	0x01
	0x4C
0xB2	0x08
	0x4E
	0x4F
0xF3 = Connection Configuration Object	0x01 = Get Attributes All
	0x0E = Get Attribute Single
	0x4C = Open Connection
	0x4D = Close Connection
	0x4E = Stop Connection
0x50 = Get Status	0x50 = Get Status
0xF4 = Port Object	0x01 = Get Attributes All
	0x0E = Get Attribute Single
0xF5 = TCP/IP Interface Object	0x01 = Get Attributes All
	0x0E = Get Attribute Single
0xF6 = EtherNet Link Object	0x01 = Get Attributes All
	0x0E = Get Attribute Single
0x300 = Module Diagnostics	0x01 = Get Attributes All
	0x0E = Get Attribute Single
0x301 = Input/OCnx	0x01 = Get Attributes All
	0x0E = Get Attribute Single
0x302 = Local Slaves	0x01 = Get Attributes All
	0x0E = Get Attribute Single

Table 4: Class IDs for function type *readonly* (cont.)

Class ID	Service codes
0x400 = Service Port Control Object	0x01 = Get Attributes All
	0x0E = Get Attribute Single
0x401 = Dynamic IO Control Object	0x01 = Get Attributes All
	0x0E = Get Attribute Single
0x402 = Router Diagnostics Object	0x01 = Get Attributes All
	0x0E = Get Attribute Single
0x403 = Router Routing Table Object	0x01 = Get Attributes All
	0x0E = Get Attribute Single
0x404 = SMTP	0x01 = Get Attributes All
	0x0E = Get Attribute Single
0x405 = SNTP	0x01 = Get Attributes All
	0x0E = Get Attribute Single
0x406 = HSBY	0x01 = Get Attributes All
	0x0E = Get Attribute Single

1. A packet with *Class ID=0x06* contains embedded CIP messages. In this case, the device performs an additional level of DPI on the data packets that contain the *service codes 0x4E, 0x52, 0x54 and 0x5B*. The device blocks a data packet if it contains other than the preceding *service codes* for this *Class ID*.

Table 5: Class IDs for function type *readwrite*

Class ID	Service codes
0x01 = Identity	0x01=Get Attributes All
	0x0E= Get Attribute Signal
	0x10= Set Attribute Signal
	0x11= Find Next Object Instance
	0x18= Get Member
0x02 = Message Router	0x01= Get Attributes All
	0x0E = Get Attribute Single
	0x4B = Write Data Table (Rockwell)
	0x54
0x04 = Assembly	0x08 = Create
	0x09 = Delete
	0x0E = Get Attribute Single
	0x10 = Set Attribute Single
	0x18 = Get Member
	0x19 = Set Member
	0x1A = Insert Member
	0x1B = Remove Member
	0x4B
	0x4C

Table 5: Class IDs for function type *readwrite* (cont.)

Class ID	Service codes
0x05 = Connecti on	0x05 = Reset
	0x08 = Create
	0x09 = Delete
	0x0D = Apply Attri butes
	0x0E = Get Attri bute Singl e
	0x10 = Set Attri bute Singl e
	0x11 = Fi nd Next Object Instance
	0x4B = Connecti on Bi nd
	0x4C = Producti on Appli cati on Lookup
	0x4E = Safety Cl ose
	0x54 = Safety Open
0x06 = Off-Li nk Connecti on Manager ¹	0x01 = Get Attri butes Al l
	0x02 = Set Attri butes Al l
	0x0E = Get Attri bute Singl e
	0x10 = Set Attri bute Singl e
	0x4C
	0x4E = Forward Cl ose
	0x52 = Unconnected Send
	0x54 = Forward Open
	0x56 = Get Connecti on Data
	0x57 = Search Connecti on Data
	0x59
	0x5A = Get Connecti on Owner
	0x5B = Large Forward Open
0x07 = Regi ster	0x0E = Get Attri bute Singl e
	0x10 = Set Attri bute Singl e
0x08 = Di screte Input Poi nt	0x01 = Get Attri butes Al l
	0x02 = Set Attri butes Al l
	0x0E = Get Attri bute Singl e
	0x10 = Set Attri bute Singl e
0x09 = Di screte Output Poi nt	0x01 = Get Attri butes Al l
	0x02 = Set Attri butes Al l
	0x0E = Get Attri bute Singl e
	0x10 = Set Attri bute Singl e
0x0A = Anal og Input Poi nt	0x01 = Get Attri butes Al l
	0x02 = Set Attri butes Al l
	0x0E = Get Attri bute Singl e
	0x10 = Set Attri bute Singl e
0x0B = Anal og Output Poi nt	0x01 = Get Attri butes Al l
	0x02 = Set Attri butes Al l
	0x0E = Get Attri bute Singl e
	0x10 = Set Attri bute Singl e

Table 5: Class IDs for function type *readwrite* (cont.)

Class ID	Service codes
0x0E = Presence Sensing	0x0E = Get Attribute Single
	0x10 = Set Attribute Single
0x0F = Parameter	0x01 = Get Attributes All
	0x05 = Reset
	0x0D = Apply Attributes
	0x0E = Get Attribute Single
	0x10 = Set Attribute Single
	0x15 = Restore
	0x16 = Save
	0x18 = Get Member
0x10 = Parameter Group	0x4B = Get Enum String
	0x01 = Get Attributes All
	0x0E = Get Attribute Single
	0x10 = Set Attribute Single
0x12 = Group	0x01 = Get Attributes All
	0x0E = Get Attribute Single
0x1D = Discrete Input Group	0x01 = Get Attributes All
	0x02 = Set Attributes All
	0x0E = Get Attribute Single
0x1E = Discrete Output Group	0x10 = Set Attribute Single
	0x01 = Get Attributes All
	0x02 = Set Attributes All
0x1F = Discrete Group	0x0E = Get Attribute Single
	0x01 = Get Attributes All
	0x02 = Set Attributes All
0x20 = Analog Input Group	0x0E = Get Attribute Single
	0x01 = Get Attributes All
	0x02 = Set Attributes All
	0x10 = Set Attribute Single
0x21 = Analog Output Group	0x0E = Get Attribute Single
	0x01 = Get Attributes All
	0x02 = Set Attributes All
	0x10 = Set Attribute Single
0x22 = Analog Group	0x0E = Get Attribute Single
	0x01 = Get Attributes All
	0x10 = Set Attribute Single

Table 5: Class IDs for function type *readwrite* (cont.)

Class ID	Service codes
0x23 = Position Sensor Object	0x05 = Reset
	0x0D = Apply Attributes
	0x0E = Get Attribute Single
	0x10 = Set Attribute Single
	0x15 = Restore
	0x16 = Save
	0x18 = Get Member
	0x19 = Set Member
0x24 = Position Controller Supervisor Object	0x0E = Get Attribute Single
	0x10 = Set Attribute Single
0x25 = Position Controller Object	0x0E = Get Attribute Single
	0x10 = Set Attribute Single
0x26 = Block Sequencer Object	0x0E = Get Attribute Single
	0x10 = Set Attribute Single
0x27 = Command Block Object	0x0E = Get Attribute Single
	0x10 = Set Attribute Single
0x28 = Motor Data Object	0x0E = Get Attribute Single
	0x10 = Set Attribute Single
	0x15 = Restore
0x29 = Control Supervisor Object	0x0E = Get Attribute Single
	0x10 = Set Attribute Single
0x2A = AC/DC Drive Object	0x05 = Reset
	0x0E = Get Attribute Single
	0x10 = Set Attribute Single
	0x15 = Restore
0x2B = Acknowledge Handler Object	0x16 = Save
	0x08 = Create
	0x09 = Delete
	0x0E = Get Attribute Single
	0x10 = Set Attribute Single
	0x4B = Add AckData Path
0x2C = Overload Object	0x4C = Remove AckData Path
	0x0E = Get Attribute Single
	0x10 = Set Attribute Single
	0x15 = Restore
0x2D = Softstart Object	0x16 = Save
	0x0E = Get Attribute Single
	0x10 = Set Attribute Single
	0x15 = Restore
	0x16 = Save

Table 5: Class IDs for function type *readwrite* (cont.)

Class ID	Service codes
0x2E = Selection Object	0x05 = Reset
	0x06 = Start
	0x07 = Stop
	0x08 = Create
	0x09 = Delete
	0x0E = Get Attribute Single
	0x10 = Set Attribute Single
	0x18 = Get Member
	0x19 = Set Member
	0x1A = Insert Member
	0x1B = Remove Member
0x30 = S-Device Supervisor Object	0x05 = Reset
	0x06 = Start
	0x07 = Stop
	0x0E = Get Attribute Single
	0x10 = Set Attribute Single
	0x4B = Abort
	0x4C = Recover
0x4E = Perform Diagnostics	
0x31 = S-Analog Sensor Object	0x01 = Get Attributes All
	0x0E = Get Attribute Single
	0x4B = Zero Adjust
	0x4C = Gain Adjust
0x32 = S-Analog Actuator Object	0x0E = Get Attribute Single
	0x10 = Set Attribute Single
0x33 = S-Single Stage Controller Object	0x0E = Get Attribute Single
	0x10 = Set Attribute Single
	0x63 = Calibrate
0x34 = S-Gas Calibration Object	0x0E = Get Attribute Single
	0x10 = Set Attribute Single
	0x4B = Get All Instances
0x35 = Trip Point Object	0x0E = Get Attribute Single
	0x10 = Set Attribute Single

Table 5: Class IDs for function type *readwrite* (cont.)

Class ID	Service codes
0x37 = File Object	0x06 = Start
	0x07 = Stop
	0x08 = Create
	0x09 = Delete
	0x0E = Get Attribute Single
	0x10 = Set Attribute Single
	0x15 = Restore
	0x16 = Save
	0x18 = Get Member
	0x4B = Initiate Upload
	0x4C = Initiate Download
	0x4D = Initiate Partial Read
	0x4E = Initiate Partial Write
	0x4F = Upload Transfer
	0x50 = Download Transfer
	0x51 = Clear File
0x38 = S-Partial Pressure Object	0x01 = Get Attributes All
	0x08 = Create
	0x09 = Delete
	0x0E = Get Attribute Single
	0x10 = Set Attribute Single
	0x4B = Create Range
	0x4C = Get Instance List
	0x4D = Get Pressures
	0x4E = Get All Pressures
0x4F = Group Enable	
0x40 = S-Sensor Calibration Object	0x0E = Get Attribute Single
	0x10 = Set Attribute Single
	0x4B = Get all Instances
0x41 = Event Log Object	0x05 = Reset
	0x06 = Start
	0x07 = Stop
	0x0E = Get Attribute Single
	0x10 = Set Attribute Single
	0x18 = Get Member
	0x19 = Set Member
	0x1A = Insert Member
0x1B = Remove Member	

Table 5: Class IDs for function type *readwrite* (cont.)

Class ID	Service codes
0x42 = Motion Device Axis Object	0x03 = Get Attribute List
	0x04 = Set Attribute List
	0x0E = Get Attribute Single
	0x10 = Set Attribute Single
	0x1C = GroupSync
	0x4B = Get Axis Attributes List
	0x4C = Set Axis Attributes List
	0x4D = Set Cyclic Write List
	0x4E = Set Cyclic Read List
	0x4F = Run Motor Test
	0x50 = Get Motor Test Data
	0x51 = Run Inertia Test
	0x52 = Get Inertia Test Data
	0x53 = Run Hookup Test
	0x54 = Get Hookup Test Data
0x43 = Time Sync Object	0x01 = Get Attributes All
	0x03 = Get Attribute List
	0x04 = Set Attribute List
	0x0E = Get Attribute Single
	0x10 = Set Attribute Single
0x44 = Modbus Object	0x0E = Get Attribute Single
	0x4B = Read Discrete Inputs
	0x4C = Read Coils
	0x4D = Read Input Registers
	0x4E = Read Holding Registers
	0x4F = Write Coils
	0x50 = Write Holding Registers
0x51 = Modbus Passthrough	
0x45 = Originator Connection List Object	0x08 = Create
	0x09 = Delete
	0x4C = Connection Read
0x46 = Modbus Serial Link Object	0x01 = Get Attributes All
	0x05 = Reset
	0x0E = Get Attribute Single
	0x10 = Set Attribute Single
	0x4B = Get And Clear

Table 5: Class IDs for function type readwrite (cont.)

Class ID	Service codes
0x47 = Device Level Ring (DLR) Object	0x01 = Get Attributes All
	0x0E = Get Attribute Single
	0x10 = Set Attribute Single
	0x18 = Get Member
	0x4B = Verify Fault Location
	0x4C = Clear Rapid Faults
	0x4D = Restart Sign On
	0x4E = Clear Gateway Partial Fault
0x48 = CoS Object	0x01 = Get Attributes All
	0x0E = Get Attribute Single
	0x10 = Set Attribute Single
0x4D = Target Connection List Object	0x01 = Get Attributes All
	0x0E = Get Attribute Single
	0x4C = Connection Read
0x4E = Base Energy Object	0x01 = Get Attributes All
	0x03 = Get Attribute List
	0x04 = Set Attribute List
	0x05 = Reset
	0x08 = Create
	0x09 = Delete
	0x0E = Get Attribute Single
	0x10 = Set Attribute Single
	0x18 = Get Member
	0x19 = Set Member
	0x1A = Insert Member
	0x1B = Remove Member
	0x4B = Start Metering
	0x4C = Stop Metering
0x4F = Electrical Energy Object	0x01 = Get Attributes All
	0x03 = Get Attribute List
	0x0E = Get Attribute Single
0x50 = Non-Electrical Energy Object	0x01 = Get Attributes All
	0x03 = Get Attribute List
	0x0E = Get Attribute Single
0x51 = Base Switch Object	0x01 = Get Attributes All
	0x0E = Get Attribute Single
	0x10 = Set Attribute Single
0x52 = SNMP Object	0x01 = Get Attributes All
	0x0E = Get Attribute Single
	0x10 = Set Attribute Single

Table 5: Class IDs for function type *readwrite* (cont.)

Class ID	Service codes
0x53 = Power Management Object	0x01 = Get Attributes All
	0x03 = Get Attribute List
	0x04 = Set Attribute List
	0x0E = Get Attribute Single
	0x10 = Set Attribute Single
	0x18 = Get Member
	0x19 = Set Member
	0x4D = Power Management
	0x4E = Set Pass Code
	0x4F = Clear Pass Code
0x54 = RSTP Bridge Object	0x01 = Get Attributes All
	0x0E = Get Attribute Single
	0x10 = Set Attribute Single
0x55 = RSTP Port Object	0x01 = Get Attributes All
	0x0E = Get Attribute Single
	0x10 = Set Attribute Single
0x91 = ANSI Extended Symbol Segment	0x03
	0x55
0x6B	0x55
0x6C	0x01
0xAC	0x01
	0x4C
0xB2	0x08
	0x4E
	0x4F
0xF3 = Connection Configuration Object	0x01 = Get Attributes All
	0x02 = Set Attributes All
	0x08 = Create
	0x09 = Delete
	0x0E = Get Attribute Single
	0x10 = Set Attribute Single
	0x15 = Restore
	0x4B = Kick Timer
	0x4C = Open Connection
	0x4D = Close Connection
	0x4E = Stop Connection
	0x4F = Change Start
	0x50 = Get Status
	0x51 = Change Complete
0x52 = Audit Changes	

Table 5: Class IDs for function type *readwrite* (cont.)

Class ID	Service codes
0xF4 = Port Object	0x01 = Get Attributes All
	0x05 = Reset
	0x0E = Get Attribute Single
	0x10 = Set Attribute Single
0xF5 = TCP/IP Interface Object	0x01 = Get Attributes All
	0x02 = Set Attributes All
	0x0E = Get Attribute Single
	0x10 = Set Attribute Single
0xF6 = EtherNet Link Object	0x01 = Get Attributes All
	0x0E = Get Attribute Single
	0x10 = Set Attribute Single
	0x4C = Get And Clear
0x300 = Module Diagnostics	0x01 = Get Attributes All
	0x0E = Get Attribute Single
0x301 = Input/Output	0x01 = Get Attributes All
	0x0E = Get Attribute Single
0x302 = Local Slaves	0x01 = Get Attributes All
	0x0E = Get Attribute Single
0x400 = Service Port Control Object	0x01 = Get Attributes All
	0x0E = Get Attribute Single
0x401 = Dynamic I/O Control Object	0x01 = Get Attributes All
	0x0E = Get Attribute Single
0x402 = Router Diagnostics Object	0x01 = Get Attributes All
	0x0E = Get Attribute Single
0x403 = Router Routing Table Object	0x01 = Get Attributes All
	0x0E = Get Attribute Single
0x404 = SMP	0x01 = Get Attributes All
	0x0E = Get Attribute Single
	0x32 = Clear All
0x405 = SNMP	0x01 = Get Attributes All
	0x0E = Get Attribute Single
	0x32 = Clear All
0x406 = HSBY	0x01 = Get Attributes All
	0x0E = Get Attribute Single

1. A packet with *Class ID=0x06* contains embedded CIP messages. In this case, the device performs an additional level of DPI on the data packets that contain the *service codes 0x4E, 0x52, 0x54 and 0x5B*. The device blocks a data packet if it contains other than the preceding *service codes* for this *Class ID*.

[List of the PCCC command codes for different function types]

Table 6: PCCC command codes for function type *readonly*

Command codes	Function codes
0x0F	0x04
	0x09
	0xA7
	0xA2
	0x17
	0x29
	0x68
	0x01
0x01	None
0x04	None
0x06	0x00
	0x01
	0x03
	0x09

Table 7: PCCC command codes for function type *readwrite*

Command codes	Function codes
0x00	None

Table 7: PCCC command codes for function type *readwrite* (cont.)

Command codes	Function codes	
0x0F	0x02	
	0x04	
	0x03	
	0x5E	
	0x09	
	0x08	
	0xA7	
	0xAF	
	0xA2	
	0xAA	
	0x17	
	0x26	
	0x79	
	0x29	
	0x0A	
	0x12	
	0x68	
	0x67	
	0x53	
	0x55	
	0x06	
	0x01	
	0x00	
	0x18	
	0x01	None
	0x02	None
0x03	None	
0x04	None	
0x05	None	
0x06	0x03	
	0x00	
	0x01	
	0x09	
	0x07	
	0x08	
	0x06	
	0x0A	
	0x05	
	0x04	
0x02		

Table 7: PCCC command codes for function type *readwrite* (cont.)

Command codes	Function codes
0x07	0x00
	0x01
	0x03
0x08	None

Table 8: PCCC command codes for function types *any* and *advanced*

Command codes	Function codes
0x00	None

Table 8: PCCC command codes for function types *any* and *advanced* (cont.)

Command codes	Function codes
0x0F	0x8F
	0x02
	0x3A
	0x82
	0x41
	0x50
	0x52
	0x05
	0x04
	0x03
	0x11
	0x57
	0x5E
	0x81
	0x09
	0x08
	0xA7
	0xAF
	0xA2
	0xAA
	0x17
	0x26
	0x79
	0x29
	0x0A
	0x12
	0x3A
	0x80
	0x07
	0x68
	0x67
	0x53
	0x55
	0x06
0x01	
0x00	
0x18	
0x01	None
0x02	None
0x03	None
0x04	None
0x05	None

Table 8: PCCC command codes for function types *any* and *advanced* (cont.)

Command codes	Function codes	
0x06	0x03	
	0x00	
	0x01	
	0x09	
	0x07	
	0x08	
	0x06	
	0x0A	
	0x05	
	0x04	
	0x02	
	0x07	0x00
		0x01
0x03		
0x04		
0x05		
0x06		
0x08	None	

4.7 DoS

[Network Security > DoS]

Denial of Service (DoS) is a cyberattack that aims to make certain services or devices unusable. In this dialog, you can set up several filters to help protect the device itself and other devices in the network from DoS attacks.

The menu contains the following dialogs:

[DoS Global](#)

4.7.1 DoS Global

[Network Security > DoS > Global]

In this dialog, you specify the DoS settings for the TCP/UDP, IP and ICMP protocols.

Note: We recommend activating the filters to increase the level of security of the device.

TCP/UDP

A scanner uses port scans to prepare network attacks. The scanner uses different techniques to determine running devices and open ports. This frame lets you activate filters for specific scanning techniques.

The device supports the detection of the following scan types:

- Null scans
- Xmas scans
- SYN/FIN scans
- TCP Offset attacks
- TCP SYN attacks
- L4 Port attacks
- Minimal Header scans

Null Scan filter

Activates/deactivates the Null Scan filter.

The device detects and discards incoming TCP packets with the following properties:

- No TCP flags are set.
- The TCP sequence number is 0.

Possible values:

`marked`

The filter is active.

`unmarked` (default setting)

The filter is inactive.

Xmas filter

Activates/deactivates the Xmas filter.

The device detects and discards incoming TCP packets with the following properties:

- The TCP flags *FIN*, *URG* and *PSH* are simultaneously set.
- The TCP sequence number is 0.

Possible values:

`marked`

The filter is active.

`unmarked` (default setting)

The filter is inactive.

SYN/FIN filter

Activates/deactivates the SYN/FIN filter.

The device detects incoming data packets with the TCP flags *SYN* and *FIN* set simultaneously and discards them.

Possible values:

`marked`

The filter is active.

`unmarked` (default setting)

The filter is inactive.

TCP Offset protection

Activates/deactivates the TCP Offset protection.

The TCP Offset protection detects incoming TCP data packets whose fragment offset field of the IP header is equal to 1 and discards them.

The TCP Offset protection accepts UDP and ICMP packets whose fragment offset field of the IP header is equal to 1.

Possible values:

`marked`

The protection is active.

`unmarked` (default setting)

The protection is inactive.

TCP SYN protection

Activates/deactivates the TCP SYN protection.

The TCP SYN protection detects incoming data packets with the TCP flag *SYN* set and a L4 source port <1024 and discards them.

Possible values:

`marked`

The protection is active.

`unmarked` (default setting)

The protection is inactive.

L4 Port protection

Activates/deactivates the L4 Port protection.

The L4 Port protection detects incoming TCP and UDP data packets whose source port number and destination port number are identical and discards them.

Possible values:

`marked`

The protection is active.

`unmarked` (default setting)

The protection is inactive.

Min. Header Size filter

Activates/deactivates the Minimal Header filter.

The Minimal Header filter compares the TCP header of incoming data packets. If the data offset value multiplied by 4 is smaller than the minimum TCP header size, then the filter discards the data packet.

Possible values:

[marked](#)

The filter is active.

[unmarked](#) (default setting)

The filter is inactive.

Min. TCP header size

Displays the minimum size of a valid TCP header.

IP

Land Attack filter

Activates/deactivates the *Land Attack* filter. With the *Land Attack* method, the attacking station sends data packets whose source and destination addresses are identical to the IP address of the recipient.

Possible values:

[marked](#)

The filter is active. The device discards data packets whose source and destination addresses are identical.

[unmarked](#) (default setting)

The filter is inactive.

Drop IP Source Route

Activates/deactivates filtering of the received IP data packets with *Strict Source Routing* or *Loose Source Routing*. The *Strict Source Routing* or *Loose Source Routing* is an option in the IP header where the sender specifies the routing path. The data packets follow this routing path to reach the destination.

Possible values:

[marked](#) (default setting)

The filter is active. The device discards IP data packets with a specified routing path in the IP header.

[unmarked](#)

The filter is inactive.

ICMP

This dialog provides you with filter options for the following ICMP parameters:

- Fragmented data packets
- ICMP packets from a specific size upwards

Fragmented packets filter

Activates/deactivates the filter for fragmented ICMP packets.

The filter detects fragmented ICMP packets and discards them.

Possible values:

`marked`

The filter is active.

`unmarked` (default setting)

The filter is inactive.

Packet size filter

Activates/deactivates the filter for incoming ICMP packets.

The filter detects ICMP packets whose payload size exceeds the size specified in the *Allowed payload size [byte]* field and discards them.

Possible values:

`marked`

The filter is active.

`unmarked` (default setting)

The filter is inactive.

Allowed payload size [byte]

Specifies the maximum allowed payload size of ICMP packets in bytes.

Possible values:

`0 . 1472` (default setting: `512`)

5 Virtual Private Network

The menu contains the following dialogs:

- [VPN Overview](#)
- [VPN Certificates](#)
- [VPN Connections](#)

5.1 VPN Overview

[Virtual Private Network > Overview]

Virtual Private Networks (VPN) provide secure communications for remote users or branch offices, allowing them to connect to servers within other branch offices, or even other companies using public networks. Even though the VPN tunnel uses a public network, it has the same behavior as a private network.

VPN tunnels provide secure communications to support the current trend of increased telecommuting and global business operations. In such cases, remote users or branch offices are able to connect to each other and central resources.

To provide secure communications, VPNs use IP Security (IPSec). IPSec has 2 functions for providing confidentiality namely, data encryption and data integrity. To provide authentication and integrity of the source with encryption, the device uses the IPSec Encapsulating Security Payload (ESP). Only the sender and receiver know the security key.

The device also uses the Negotiated Security Association method. The first packet received initiates a negotiation, between the sender and receiver, for which Security Association (SA) parameters the devices are going to use. The devices use the Internet Key Exchange (IKE) for the negotiation process. When negotiating the parameters, the sending and receiving devices agree on the authentication and data-security methods. The devices also perform mutual authentication, and then generate a shared key. The devices use the shared key to encrypt the data contained in each packet.

The dialog contains tabs which display the current VPN tunnels and statuses.

The [Connection errors](#) tab displays detected errors that are helpful when troubleshooting a VPN tunnel.

The dialog contains the following tabs:

- [\[Overview\]](#)
- [\[Diagnostics\]](#)
- [\[Connection errors\]](#)

Connection

Connections (max.)

Displays the maximum number of VPN tunnels supported. The device limits maximum number of active VPN tunnels to the amount set in [Max. active connections](#).

Max. active connections

Displays the maximum number of active VPN tunnels supported.

[Overview]

Table

For information on how to customize the appearance of the table, see [“Working with tables” on page 16](#).

VPN index

Displays the table row index for unique identification of a VPN tunnel.

VPN description

Displays the user-defined name for the VPN tunnel.

VPN active

Displays if the VPN tunnel is active/inactive.

The device limits the maximum number of set-up VPN tunnels to the value displayed in the [Connections \(max.\)](#) field. The device also limits the maximum number of active VPN tunnels to the value specified in the [Max. active connections](#) column.

Possible values:

[marked](#)

The VPN tunnel is active.

[unmarked](#)

The VPN tunnel is inactive.

Used IKE version

Displays the version of the IKE protocol that the VPN tunnel uses.

Possible values:

[i ke v1](#)

The device uses the IKE version 1 (ISAKMP) protocol.

[i ke v2](#)

The device uses the IKE version 2 protocol.

Startup

Displays the starting role for mediating the key exchange for VPN tunnel.

Possible values:

[i n i t i a t o r](#)

If you specify the role of the device as an *Initiator* for the VPN tunnel, then the device actively initiates the Internet Key Exchange (IKE) and parameter negotiation.

[r e s p o n d e r](#)

If you specify the role of the device as a *Responder* for the VPN tunnel, then the device waits for the *Initiator* to begin a key exchange (IKE) and connection parameter negotiation.

Operational status

Displays the current status of the VPN tunnel.

Possible values:

[u p](#)

VPN tunnel is established.

[d o w n](#)

VPN tunnel is not established.

[n e g o t i a t i o n](#)

If you specify the VPN tunnel for this device as the *Initiator*, then the value indicates that the key exchange and negotiation algorithm is in progress. If the VPN tunnel for this device is the *Responder*, then the value indicates that the VPN tunnel is waiting for the process to begin.

[c o n s t r u c t i n g](#)

The IKE-SA is up. However, the device has detected at least one unestablished IPsec-SA for this instance.

[d o r m a n t](#)

The device is waiting for you to complete the configuration before starting the VPN tunnel setup. For example, the device has an unsuccessful hostname resolution.

[r e - k e y i n g](#)

The key exchange is in progress. The device displays the value after the expiration of either the IKE or the IPSEC lifetime timer.

Connection established [s]

Displays the time, in seconds, since the device established the VPN tunnel for this device. The device updates the value after every IKE re-authentication.

Local host

Displays the name and/or IP address of the local host that the device detected using IKE.

Possible values:

Alphanumeric ASCII character string with 1..128 characters

Remote host

Displays the name and/or IP address of the remote host that the device detected using IKE.

Possible values:

Alphanumeric ASCII character string with 1..128 characters

IKE proposal

Displays the algorithms that IKE uses for the key exchange.

The device displays a combination of the *IKE key agreement*, *IKE integrity (MAC)* and *IKE encryption* parameters.

If you set up an IKE algorithm for the device in the *Virtual Private Network > Connections* dialog, and the remote endpoint has a more secure algorithm set up, then it is possible that both the local and remote devices use the remote algorithm.

The device displays the current cipher suite used for the connection.

IPsec proposal

Displays the algorithms that IPsec uses for data communication.

The device displays a combination of the *IPsec key agreement*, *IPsec integrity (MAC)* and *IPsec encryption* parameters.

If you select an IPsec algorithm for the instance in the *Virtual Private Network > Connections* dialog, and the remote endpoint has a better, more secure algorithm set up, then it is possible that both the local and remote devices use the better algorithm.

The device displays the current cipher suite used for the connection.

Tunnels

Displays the number of IPsec tunnels within the VPN network.

[Diagnostics]

Table

For information on how to customize the appearance of the table, see [“Working with tables” on page 16](#).

VPN index

Displays the table row index for unique identification of a VPN tunnel.

VPN description

Displays the user-defined name for the VPN tunnel.

VPN active

Displays if the VPN tunnel is active/inactive.

The device limits the maximum number of set-up VPN tunnels to the value displayed in the [Connections \(max.\)](#) field. The device also limits the maximum number of active VPN tunnels to the value specified in the [Max. active connections](#) column.

Possible values:

[marked](#)

The VPN tunnel is active.

[unmarked](#)

The VPN tunnel is inactive.

Tunnel index

Displays the index value that, together with the value in the [VPN index](#) column, identifies the entry in the connection tunnel info table.

Traffic selector index

Displays the index value that, together with the value in the [VPN index](#) column, identifies the entry in the traffic selector table which is mapped into the IPsec tunnel.

Possible values:

[0](#)

The traffic selector index is unknown.

[1..16](#)

Operational status

Displays the current status of the VPN tunnel.

Possible values:

[up](#)

The Internet Key Exchange-Security Association (IKE-SA) and every Internet Protocol Security-Security Association (IPsec-SA) is up.

[down](#)

The IKE-SA and IPsec-SAs are inactive.

[negotiation](#)

If you specify the VPN tunnel for this instance as the *Initiator*, then the value indicates that the key exchange and negotiation algorithm is in progress. If the VPN tunnel for this instance is the *Responder*, then the value indicates that the VPN tunnel is waiting for the process to begin.

[constructing](#)

The IKE-SA is up. However, the device has detected at least one unestablished IPsec-SA for this instance.

[dormant](#)

The device is waiting for you to complete the configuration before starting the VPN tunnel setup. For example, the device has an unsuccessful hostname resolution.

[re-keying](#)

The key exchange is in progress. The device displays the value after the expiration of either the IKE or the IPSEC lifetime timer.

IKE re-authentication [s]

Displays the remaining time, in seconds, before the next IKE re-authentication. The value 0 indicates that re-authentication is not set up.

Next IKE re-keying [s]

Displays the remaining time, in seconds, before the next IKE re-key. The value 0 indicates that re-keying is not set up.

IKE initiator SPI

Displays the Security Parameter Index (SPI) of the *Initiator*, depending which device you specify as the *Initiator*. For example, when you specify this device as the *Initiator*, then this value is the SPI of the local device.

IKE responder SPI

Displays the SPI of the *Responder*, depending which device you specify as the *Initiator*. For example, when you specify this device as the *Initiator*, then this value is the SPI of the remote device.

Local traffic selector

Displays the local traffic selector for this IPsec tunnel. As a result of the negotiation process between the peers, the local traffic selector can be different from the set-up traffic selector.

Remote traffic selector

Displays the remote traffic selector for this IPsec tunnel. As a result of the negotiation process between the peers, the traffic selector can be different from the set-up traffic selector.

Tunnel status

Displays the current operational status of the IPsec tunnel.

Possible values:

[unknown](#)

The IPsec proposal is in progress. No traffic selectors or security parameters have been negotiated for this IPsec-SA.

created

The key exchange and the negotiation algorithm is finished for this IPsec-SA, but the tunnel is inactive.

routed

The encryption policies for the data stream are established, but the negotiation process has not started.

installing

The peer authentication is established, but the IPsec proposal for this tunnel is still in progress.

installed

The IPsec-SA is installed.

updating

The device updates the security associations.

re-keying

The key exchange is in progress for this IPsec-SA. The device displays the value after the expiration of the IPsec lifetime timer.

re-keyed

The key exchange for this IPsec-SA is finished and the device sets up a new tunnel. The tunnel will become active after the expiration of the previous IPsec proposal.

re-trying

The key exchange for this IPsec-SA failed. The device will automatically try to initiate a new key exchange.

deleting

The device replaces the IPsec tunnel during re-keying. The device keeps the tunnel up for delayed packets. The old and the new tunnel are open simultaneously for 5 seconds in the default setting. After the IPsec lifetime timer has expired, the device deletes the tunnel.

destroying

The IPsec lifetime timer has expired. The device deletes the tunnel.

IPsec input SPI

Displays IPsec Security Parameter Index (SPI) that the device applies to the data it receives from the VPN tunnel. The SPI lets the device select the Security Association (SA) under which it processes a received packet.

IPsec output SPI

Displays IPsec Security Parameter Index (SPI) that the device applies to the data it transmits to the VPN tunnel.

Next IPsec re-keying [s]

Displays the remaining time, in seconds, before the next re-keying starts for this IPsec tunnel.

IPsec tunnel input [byte]

Displays the number of bytes received into this VPN tunnel.

IPsec-tunnel input [packets]

Displays the number of packets received into this VPN tunnel.

Last IPsec data received [s]

Displays the time, in seconds, since the VPN tunnel has received the last time data.

IPsec tunnel output [byte]

Displays the number of bytes sent into this VPN tunnel.

IPsec tunnel output [packets]

Displays the number of packets sent into this VPN tunnel.

Last IPsec data transmitted [s]

Displays the time, in seconds, since the VPN tunnel has sent the last time data.

[Connection errors]

Table

For information on how to customize the appearance of the table, see [“Working with tables” on page 16](#).

VPN index

Displays the table row index for unique identification of a VPN tunnel.

VPN description

Displays the user-defined name for the VPN tunnel.

VPN active

Displays if the VPN tunnel is active/inactive.

The device limits the maximum number of set-up VPN tunnels to the value displayed in the [Connections \(max.\)](#) field. The device also limits the maximum number of active VPN tunnels to the value specified in the [Max. active connections](#) column.

Possible values:

[marked](#)

The VPN tunnel is active.

[unmarked](#)

The VPN tunnel is inactive.

Last connection error

Displays the last error notification that occurred for this VPN tunnel.

When the connection remains inactive, this value is useful to help you isolate detected errors. This value helps you determine if a detected error occurred in the proposal exchange or during tunnel establishment.

Possible values:

Alphanumeric ASCII character string with 1..512 characters

5.2 VPN Certificates

[Virtual Private Network > Certificates]

A Certification Authority (CA) issues digital certificates to authenticate the identity of devices requesting a VPN tunnel. You set up the devices that form a VPN tunnel to trust the Certification Authority (CA) that signed the digital certificate. When a trusted Certification Authority (CA) signs a digital certificate, the device considers it to be valid. Using a trusted Certification Authority (CA), lets you renew and change the digital certificates transferred onto the device without affecting the VPN. The prerequisite is, that the actual identity information is correct.

Using digital certificates also lets you reduce the required maintenance work. The reason for this is because you change digital certificates less often as you change pre-shared keys. The Certification Authority (CA) generates digital certificates with commence and expiration date. The digital certificate is only valid during this time. When a digital certificate expires, the device requires a new digital certificate.

You generate a self signed certificate using the strongSwan application in conjunction with the Linux Operating System.

Note: RC2 certificate encryption algorithms are unsupported, for example PKCS12 containers with RC2 encryption or passphrase protection.

Table

For information on how to customize the appearance of the table, see [“Working with tables” on page 16](#).

Buttons

 Remove

Removes the selected table row.


 Upload

Opens the [Upload certificate](#) window to add a digital certificate to the table.

- In the [Passphrase \(private key\)](#) field, you enter the passphrase used with this digital certificate. Possible values:
 - Alphanumeric ASCII character string with 0..128 characters
- In the [URL](#) field, you specify the path and file name of the digital certificate.

The device gives you the following options for transferring the file onto the device:

- Import from the PC

When the file is located on your PC or on a network drive, drag and drop it onto the  area. As an alternative, click in the area to select the file.

You can also use SFTP or SCP to transfer the file from your PC to the device. Perform the following steps:

On your PC, open an SFTP or SCP client, for example WinSCP.

Use the SFTP or SCP client to open a connection to the device.

Transfer the file onto the device, into the directory /upload/vpn-cert.

When the file transfer is complete, the device starts installing the digital certificate. If the installation was successful, then the device generates an ok file in the directory /upload/vpn-cert and deletes the transferred file.

Index

Displays the table row index of the digital certificate entry.

Possible values:

1..100

File name

Displays the name of the file uploaded to the device.

Possible values:

Alphanumeric ASCII character string with 1..64 characters

Subject

Displays the subject field of digital certificate.

The subject field of the digital certificate is a combination of the following items the country (C), state (ST), organization (O), organizational unit (OU), common name (CN), and email address of the recipient (emailAddress).

Possible values:

Alphanumeric ASCII character string with 0..64 characters

Issuer

Displays the issuer of the digital certificate.

Possible values:

Alphanumeric ASCII character string with 0..64 characters

Valid from

Displays the date and time when the digital certificate became effective.

Possible values:

Date and time stamp

Valid until

Displays the digital certificate expiration time and date.

Possible values:

Date and time stamp

Type

Displays the type of the container file used.

Possible values:

[ca](#)

The transferred file is a digital certificate signed by a Certification Authority (CA).

[peer](#)

The transferred file is a peer certificate.

[pkcs12](#)

The transferred file is a p12 bundle.

[encryptedkey](#)

The transferred file is a key file with password encryption.

[encryptedpkcs12](#)

The transferred file is a p12 bundle with password encryption.

Upload date

Displays the date and time when the digital certificate was last transferred onto the device.

Possible values:

Date and time stamp

Private key status

Displays the status of the private key in the peer certificate. Use a peer certificate with a private key.

Possible values:

[none](#)

The peer certificate does not contain a private key.

present

The device has located and extracted the private key from the peer certificate.

not Found

The device has located a private key. However, the key is missing the passphrase and the device has suspended the transfer.

Private key file

Displays the name of the private key file.

The device lets you enter alphanumeric characters plus hyphens, underscores and dots.

Possible values:

Alphanumeric ASCII character string with 0..64 characters

Active connections

Displays the number of active connections that are using this digital certificate.

The device lets you delete the digital certificate only when the value is 0.

Possible values:

0 . 256

5.3 VPN Connections

[Virtual Private Network > Connections]

This dialog lets you set up VPN tunnels.

Note: The device uses software for DES and AES-Galois/Counter Mode (GCM) encryption.

Table

For information on how to customize the appearance of the table, see [“Working with tables” on page 16](#).

Buttons



Add

Opens the [Create](#) window to add a table row.

- From the [VPN description](#) drop-down list, you select an existing description or specify a new description. To enter a new description, click the **+** icon.
Possible values:
 - Alphanumeric ASCII character string with 0..128 characters
- In the [Traffic selector index](#) field, you specify the index of the VPN tunnel traffic selector.
Possible values:
 - [1..16](#)



Remove

Removes the selected table row.



Wizard

Opens the [Wizard](#) window that helps you associate the ports with the address of one or more desired senders. See [“\[Wizard: VPN configuration\]” on page 268](#).

VPN description

Specifies the user-defined name for the VPN tunnel.

Possible values:

Alphanumeric ASCII character string with 1..128 characters

Traffic selector index

Displays the index value that, together with the value in the [VPN index](#) column, identifies the entry in the traffic selector table.

Possible values:

[1..16](#)

The device lets you specify any available value within the given range.

Status

Displays if the VPN tunnel is active/inactive.

The device limits the maximum number of set-up VPN tunnels to the value displayed in the [Connections \(max.\)](#) field. The device also limits the maximum number of active VPN tunnels to the value displayed in [Max. active connections](#).

Possible values:

[marked](#)

The VPN tunnel is active.

[unmarked](#) (default setting)

The VPN tunnel is inactive.

Traffic selector description

Specifies the name of the traffic selector.

Possible values:

Alphanumeric ASCII character string with 1..128 characters

Source address (CIDR)

Specifies the IP address and netmask of the source host. When the device forwards packets containing this source IP address over a VPN tunnel, the device applies the settings specified in this table row. Furthermore, the device applies the associated IPsec and IKE-SA settings, to every IP packet it forwards containing this address.

Possible values:

Valid IPv4 address and netmask in CIDR notation

[any](#) (default setting)

The device applies the settings in this table row to every packet it forwards.

Source restrictions

Specifies the optional source restrictions using names or numbers entered as [<protocol /port>](#). The device sends only the type of data specified through the VPN tunnel.

Examples:

- [tcp/http](#) is equal to [6/80](#)
- [udp](#) is equal to [udp/any](#)
- [/53](#) is equal to [any/53](#)

Possible values:

Alphanumeric ASCII character string with 1..32 characters

[<empty>](#) (default setting)

The device uses [any/any](#) as the restriction.

Destination address (CIDR)

Specifies the IP address and netmask of the destination. When the device forwards packets containing this destination IP address over a VPN tunnel, the device applies the settings specified in this table row. Furthermore, for every IP packet the device forwards containing this address, it applies the associated IPsec and IKE-SA settings.

Possible values:

Valid IPv4 address and netmask in CIDR notation

`any` (default setting)

The device applies the settings in this table row to every packet it forwards.

Destination restrictions

Specifies the optional destination restrictions using names or numbers entered as `<protocol / port>`. The device accepts only the type of data specified from the VPN tunnel.

Examples:

- `tcp/http` is equal to `6/80`
- `udp` is equal to `udp/any`
- `/53` is equal to `any/53`

Possible values:

Alphanumeric ASCII character string with 1..32 characters

`<empty>` (default setting)

The device uses `any/any` as the restriction.

Version

Specifies the version of the IKE protocol for the VPN connection.

Possible values:

`auto` (default setting)

The VPN starts with protocol IKEv2 as the *Initiator* and accepts IKEv1/v2 as the *Responder*.

`ikev1`

The VPN starts with the IKEv1 protocol.

`ikev2`

The VPN starts with the IKEv2 protocol.

Startup

Specifies if the device starts this instance as a *Responder* or *Initiator*.

If you specify the local peer as the *Responder*, and the remote peer sends data packets to a specific selector, then the device attempts to establish the connection as the *Responder*. Establishing a connection as a *Responder* depends upon other settings for this connection. For example, if you specify in the *Remote endpoint* field the value `any`, then the device cannot initiate the connection.

Possible values:

`initiator`

If you specify that the device starts as an *Initiator*, then the device starts an key exchange with the *Responder*.

`responder` (default setting)

If you specify that the device starts as a *Responder*, then the device waits for the *Initiator* to start the key exchange and parameter negotiation.

IKEv1 DPD timeout [s]

Specifies the timeout, in seconds, before the local peer declares the remote peer dead, if the remote peer is unresponsive.

The device supports the *IKEv1 DPD timeout [s]* function using IKEv1.

Possible values:

- 0
Deactivates the function.
- 1.. 86400 (24 h) (default setting: 120)

IKE lifetime [s]

Specifies the lifetime, in seconds, of the IKE security association between two network devices to support secure communication. The devices establish a security association after exchanging a set of pre-defined keys.

Possible values:

- 300.. 86400 (default setting: 28800)
The default setting is 8 hours. The maximum setting is 24 hours.

IKE exchange mode

Specifies the use of the phase 1 exchange mode for IKEv1.

The purpose of IKE phase 1 is to establish a secure authenticated communication channel. The device uses the Diffie-Hellman key exchange algorithm to generate a shared secret key. The device then uses the shared secret key to further encrypt IKE communications.

Possible values:

- main (default setting)
The main mode for phase 1 provides identity protection.
- aggressive
You use the aggressive mode to reduce round trips.

Authentication

Specifies the type of authentication that the device uses.

Possible values:

- psk (default setting)
Select this value for the device to use a key that was previously generated and saved on both the remote and local devices.
- individual x509
Select this value for the device to use a digital certificate in X.509 format.
Use a separate digital certificate for Certification Authority (CA) and local identification.
- pkcs12
Select this value for the device to use a PKCS12 container with the needed digital certificates, which also includes the Certification Authority (CA).

Pre-shared key

Specifies the pre-shared key. The prerequisite is that in the *Authentication* column the value *psk* is specified.

Possible values:

Alphanumeric ASCII character string with 0..128 characters excluding double-quote and new line characters

The device also lets you generate pre-shared secrets as hexadecimal or Base64 encoded binary values. The device interprets a character sequence starting with *0x* as a sequence with hexadecimal digits. Similarly, the device also interprets a character sequence starting with multiple zeros as Base64 encoded binary data.

IKE auth. cert. CA

Specifies the name of the Certification Authority (CA) which issued the digital certificate. The device uses this digital certificate for signature verification of the local and remote certificates. The prerequisite is that in the *Authentication* column the value *i ndi vi dual x509* is specified.

Possible values:

Alphanumeric ASCII character string with 1..128 characters

IKE auth. cert. local

Specifies the file name of the digital certificate the local device uses. The device uses this digital certificate for authentication of the local peer on the remote side.

Possible values:

Alphanumeric ASCII character string with 1..128 characters

The behavior depends on the value you specify in the *Authentication* column:

- *i ndi vi dual x509*

The digital certificate binds the identity of the local peer to the specified public key signed by the Certification Authority (CA) specified in the *IKE auth. cert. CA* column.

- *pkcs12*

The digital certificate in the PKCS bundle binds the identity of the local peer to the specified public key. The device performs this check independently of the digital certificate displayed in the *IKE auth. cert. CA* column.

IKE auth. cert. remote

Specifies the file name of the digital certificate the remote device uses. The device uses this digital certificate for authentication of the remote peer on the local side. This digital certificate binds the identity of the remote peer to the specified public key. The prerequisite is that in the *Authentication* column the value *i ndi vi dual x509* is specified.

Possible values:

Alphanumeric ASCII character string with 0..128 characters

The value is optional, because the remote peer typically sends the digital certificate and the device only checks the validity of the digital certificate.

Encrypted private key

Specifies the file name for the private key.

Prerequisites:

- In the *Authentication* column, the value *i ndi vi dual x509* is specified.
- The key saved in the device is encrypted with a passphrase.

The key requires that, in the *Encrypted key/PKCS12 passphrase* column, you specify the passphrase. The device considers the key and the digital certificate unmatched until the key is decrypted.

Possible values:

Alphanumeric ASCII character string with 0..128 characters

Encrypted key/PKCS12 passphrase

Specifies the passphrase that the device uses for decryption of the private key specified in the *Encrypted private key* column or *pkcs12* certificate container.

Possible values:

Alphanumeric ASCII character string with 0..128 characters

IKE local identifier type

Specifies the type of local peer identifier that the device uses for the *IKE local ID* parameter.

Possible values:

default (default setting)

The behavior depends on the value you specify in the *Authentication* column:

– *psk*

The device uses the IP address specified in the *Local endpoint* column as the local identifier.

– *individual x509* or *pkcs12*

The device uses the distinguished name (DN) contained in the local *IKE auth. cert. local* certificate.

address

In the *IKE local ID* column, the device uses the IP address or the DNS hostname specified in the *Local endpoint* column.

id

The device identifies the value specified in the *IKE local ID* column as one of the following types:

- An IPv4 address or a DNS hostname
- A key identifier which specifies the data that the device uses to pass vendor-specific information. The device uses the information to identify which pre-shared key it uses for aggressive mode authentication during negotiations.
- An FQDN web address, for example, *foo.bar.com*
- An email address
- The *ASN.1 X.500 Distinguished Name (DN)* contained within the *IKE auth. cert. remote* column. The local and remote devices exchange their digital certificates to establish the Security Association (SA).

IKE local ID

Specifies the local peer identifier that the device sends to the remote device in the ID payload during phase 1 negotiations. The device uses the ID payload to identify the *Initiator* of the Security Association (SA). The *Responder* uses the identity to determine the correct host system policy requirement for the Security Association (SA).

The formats for this parameter depend on the type specified in the *IKE local identifier type* column.

Possible values:

`<empty>` (default setting)

If in the *IKE local identifier type* column, you specify the value `i d`, then specify the value using one of the following options:

- An IPv4 address or a DNS hostname
- A previously specified key identifier, specifying data that the device uses to pass vendor-specific information.
- An FQDN web address, for example, `foo.bar.com`
- An email address
- An X.500 distinguished name

Refer to the following syntax as an example when adding the item:

`CN = XY-D, C = DE, L = NT, ST = BW O = COMPANY, OU = DEV, E = testuser@company.com`

Remote identifier type

Specifies the type of remote peer identifier that the device uses for the *Remote ID* parameter.

Possible values:

`any` (default setting)

The device accepts every received remote identifier without further verification.

`address`

In the *Remote ID* column, the device uses the IP address or the DNS hostname specified in the *Remote endpoint* column.

`i d`

The device identifies the value specified in the *Remote ID* column as one of the following types:

- An IPv4 address or a DNS hostname
- A key identifier which specifies the data that the device uses to pass vendor-specific information. The device uses the information to identify which pre-shared key it uses for aggressive mode authentication during phase 1 negotiations.
- An FQDN web address, for example, `foo.bar.com`
- An email address
- The *ASN.1 X.500 Distinguished Name (DN)* contained within the *IKE auth. cert. remote* column. The local and remote devices exchange their digital certificates to establish the Security Association (SA).

Remote ID

Specifies the remote peer identifier which the device compares with the value in the ID payload during phase 1 negotiations. The device uses the ID payload to identify the *Initiator* of the Security Association (SA). The *Responder* uses the identity to determine the correct host system policy requirement for the Security Association (SA).

The formats for this parameter depend on the type specified in the *Remote identifier type* column.

Possible values:

`<empty>` (default setting)

If in the *Remote identifier type* column, you specify the value `i d`, then specify the value using one of the following options:

- An IPv4 address or a DNS hostname
- A previously specified key identifier, specifying data that the device uses to pass vendor-specific information.
- An FQDN web address, for example, `foo.bar.com`
- An email address
- An X.500 distinguished name

Refer to the following syntax as an example when adding the item:

`CN = XY-D, C = DE, L = NT, ST = BW O = COMPANY, OU = DEV, E = testuser@company.com`

IKE key agreement

Specifies which Diffie-Hellman (DH) key agreement algorithm the device uses for establishing the IKE-SA session key.

Possible values:

[any](#)

The device accepts every algorithm when specified as the *Responder*.

[modp1024](#) (default setting)

The value represents an RSA algorithm with 1024 bits modulus which is DH Group 2.

[modp1536](#)

The value represents an RSA algorithm with 1536 bits modulus which is DH Group 5.

[modp2048](#)

The value represents an RSA algorithm with 2048 bits modulus which is DH Group 14.

[modp3072](#)

The value represents an RSA algorithm with 3072 bits modulus which is DH Group 15.

[modp4096](#)

The value represents an RSA algorithm with 4096 bits modulus which is DH Group 16.

IKE integrity (MAC)

Specifies which IKE Integrity Message Authentication Code (MAC) algorithm the device uses. To help keep the information on the VPN secure, the Hash-based Message Authentication Code (HMAC) process in the sending device mixes (hashes) the message data with a shared secret key. The receiving device mixes the results (hash value) with the secret key again, and then applies the hash function a second time.

Possible values:

[any](#)

When you specify the device as the *Responder*, the device accepts any algorithm. When you specify the device as the *Initiator*, the device uses various pre-defined algorithms.

[hmacmd5](#)

The device uses the Message Digest Algorithm 5 (MD5) for the hash function calculation.

[hmacsha1](#) (default setting)

The device uses the Secure Hash Algorithm version 1 (SHA-1) for the hash function calculation.

[hmacsha256](#)

The device uses SHA-256 (part of the version 2 family) for the hash function calculation which the device computes with 32 bit words.

[hmacsha384](#)

The device uses SHA-384 (part of the version 2 family) for hash function calculation which the device computes using a shorter version of SHA-512.

[hmacsha512](#)

The device uses SHA-512 (part of the version 2 family) for hash function calculation which the device computes with 64 bit words.

Note: We recommend to use the setting [hmacsha256](#) or higher.

IKE encryption

Specifies the IKE encryption algorithm that the device uses.

Possible values:

[any](#)

When you specify the device as the *Responder*, the device accepts any algorithm. When you specify the device as the *Initiator*, the device uses various pre-defined algorithms.

des

The device uses the Data Encryption Standard (DES) block cipher for encryption of message data with a 56 bit key.

des3

The device uses the Triple DES block cipher for encryption of message data which applies the 56 bit key, from DES, 3 times to each block.

aes128 (default setting)

The device uses the Advanced Encryption Standard (AES) with a block size of 128 bits, and a key length of 128 key bits.

aes192

The device uses the AES with a block size of 128 bits, and a key length of 192 key bits.

aes256

The device uses the AES with a block size of 128 bits, and a key length of 256 key bits.

Note: We recommend to use the setting [aes128](#) or higher.

Local endpoint

Specifies the hostname or IP address of the local IPsec VPN tunnel endpoint.

Possible values:

any (default setting)

The device uses the IP address of the interface the device uses to forward data to the remote endpoint.

Valid IPv4 address and netmask in CIDR notation

hostname

Alphanumeric ASCII character string with 1..128 characters

Remote endpoint

Specifies the hostname or IP address of the remote IPsec VPN tunnel endpoint.

Possible values:

any (default setting)

The device accepts any IP address when establishing an IKE-SA as a VPN *Responder*.

Valid IPv4 address and netmask in CIDR notation

If you specify that the device is a *Responder* for this VPN tunnel, then the device accepts a network in CIDR notation, during IKE-SA establishment.

hostname

Alphanumeric ASCII character string with 1..128 characters

Re-authentication

Activates/deactivates peer re-authentication after an IKE-SA re-key. If in the [Version](#) column, you specify the value [ikev1](#), then the device constantly re-authenticates the VPN tunnel, even when you unmark the checkbox.

Possible values:

marked

The device generates a new IKE-SA and attempts to regenerate the IPsec SAs.

unmarked (default setting)

When you use the IKEv2 protocol, the device re-keys the VPN tunnel and retains the IPsec SAs.

IPsec key agreement

Specifies which Diffie-Hellman key agreement algorithm the device uses for establishing the IPsec-SA session key. If the *Perfect Forward Secrecy (PFS)* function is enabled and a compromise of a single key occurs, then the integrity remains for subsequently generated keys.

Possible values:

[any](#)

When you specify the device as the *Responder*, the device accepts any algorithm. When you specify the device as the *Initiator*, the device uses various pre-defined algorithms.

[modp1024](#) (default setting)

The value represents an Rivest, Shamir, and Adleman (RSA) algorithm with 1024 bits modulus. This value is Diffie-Hellman (DH) Group 2.

[modp1536](#)

The value represents an RSA algorithm with 1536 bits modulus which is DH Group 5.

[modp2048](#)

The value represents an RSA algorithm with 2048 bits modulus which is DH Group 14.

[modp3072](#)

The value represents an RSA algorithm with 3072 bits modulus which is DH Group 15.

[modp4096](#)

The value represents an RSA algorithm with 4096 bits modulus which is DH Group 16.

[none](#)

The device disables the *PFS* function. Disabling the *PFS* function is considered a confidentiality violation and therefore a security risk.

IPsec integrity (MAC)

Specifies which IPsec Integrity MAC algorithm the device uses for the instance. To help keep the information on the VPN secure, the Hash-based Message Authentication Code (HMAC) process in the sending device mixes (hashes) the message data with a shared secret key. The receiving device mixes the results (hash value) with the secret key again, and then applies the hash function a second time.

Possible values:

[any](#)

When you specify the device as the *Responder*, the device accepts any algorithm. When you specify the device as the *Initiator*, the device uses various pre-defined algorithms.

[hmacmd5](#)

The device uses the Message Digest Algorithm 5 (MD5) for the hash function calculation.

[hmacsha1](#) (default setting)

The device uses the Secure Hash Algorithm version 1 (SHA-1) for the hash function calculation.

[hmacsha256](#)

The device uses SHA-256 (part of the version 2 family) for the hash function calculation which the device computes with 32 bit words.

[hmacsha384](#)

The device uses SHA-384 (part of the version 2 family) for hash function calculation which the device computes using a shorter version of SHA-512.

[hmacsha512](#)

The device uses SHA-512 (part of the version 2 family) for hash function calculation which the device computes with 64 bit words.

Note: We recommend to use the setting [hmacsha256](#) or higher.

IPsec encryption

Specifies the IPsec encryption algorithm that the device uses.

Possible values:

[any](#)

When you specify the device as the *Responder*, the device accepts any algorithm. When you specify the device as the *Initiator*, the device uses various pre-defined algorithms.

[des](#)

The device uses the Data Encryption Standard (DES) block cipher for encryption of message data with a 56 bit key.

[des3](#)

The device uses the Triple DES block cipher for encryption of message data which applies the 56 bit key, from DES, 3 times to each block.

[aes128](#) (default setting)

The device uses the Advanced Encryption Standard (AES) with a block size of 128 bits, and a key length of 128 key bits.

[aes192](#)

The device uses the AES with a block size of 128 bits, and a key length of 192 key bits.

[aes256](#)

The device uses the AES with a block size of 128 bits, and a key length of 256 key bits.

[aes128ctr](#)

AES-CTR with 128 key bits.

[aes192ctr](#)

AES-CTR with 192 key bits.

[aes256ctr](#)

AES-CTR with 256 key bits.

[aes128gcm64](#)

The device uses the AES-Galois/Counter Mode (GCM) with a 64 bit Integrity Check Value (ICV) and 128 key bits.

[aes128gcm96](#)

AES-GCM with a 96 bit ICV and 128 key bits.

[aes128gcm128](#)

AES-GCM with a 128 bit ICV and 128 key bits.

[aes192gcm64](#)

AES-GCM with a 64 bit ICV and 192 key bits.

[aes192gcm96](#)

AES-GCM with a 96 bit ICV and 192 key bits.

[aes192gcm128](#)

AES-GCM with a 128 bit ICV and 192 key bits.

[aes256gcm64](#)

AES-GCM with a 64 bit ICV and 256 key bits.

[aes256gcm96](#)

AES-GCM with a 96 bit ICV and 256 key bits.

[aes256gcm128](#)

AES-GCM with a 128 bit ICV and 256 key bits.

Note: We recommend to use the setting [aes128](#) or higher.

IPsec lifetime [s]

Specifies the lifetime, in seconds, of the IPsec security association between 2 network devices to support secure communication. The devices establish a security association after exchanging a set of pre-defined keys.

Possible values:

[300](#) . [28800](#) (default setting: [3600](#))

The default setting is one hour. The maximum setting is 8 hours.

Margin time [s]

Specifies the period in seconds, before [IKE lifetime \[s\]](#) and [IPsec lifetime \[s\]](#) expire, after which the device attempts to negotiate a new key.

Possible values:

[1](#) . [1800](#) (default setting: [150](#))

The default setting is equal to 2.5 minutes. The maximum value is half an hour.

Log informational entries

Activates/deactivates event log entries for debugging proposes only.

Possible values:

[marked](#)

The device receives and processes the informational messages for this VPN tunnel, and enters the message in the event log.

[unmarked](#) (default setting)

The device receives and processes the informational messages for this connection, without an event log entry.

Log unhandled messages

Activates/deactivates message handling for messages unknown to strongSwan for debugging proposes only.

Possible values:

[marked](#)

The device enters the non-strongSwan messages received for this connection, in the event log.

[unmarked](#) (default setting)

The device ignores the non-strongSwan messages received for this connection.

[Wizard: VPN configuration]

The [Wizard](#) window lets you set up a VPN tunnel. The device also lets you add or change a VPN tunnel directly in the dialog.

The [Wizard](#) window guides you through the following steps:

- [Create or select entry](#)
- [Authentication](#)
- [Endpoint and traffic selectors](#)
- [Advanced configuration](#)

Create or select entry

VPN

Displays the existing VPN tunnels setup in the device. Select an item to continue. As an alternative, specify a VPN tunnel in the [VPN index](#) and [VPN description](#) fields.

VPN index

Specifies the index number of the VPN tunnel.

Possible values:

[1.. 256](#)

VPN description

Specifies the user-defined description for the VPN tunnel.

Possible values:

Alphanumeric ASCII character string with 1..128 characters


Authentication

For each VPN tunnel you can specify the authentication methods using the following tabs:

- [Authentication - Pre-shared key](#)

Authentication - Pre-shared key

Pre-shared key

Specifies the pre-shared key. You can view the specified values by clicking the  icon.

Possible values:

Alphanumeric ASCII character string with 0..128 characters excluding double-quote and new line characters

The device also lets you generate pre-shared secrets as hexadecimal or Base64 encoded binary values. The device interprets a character sequence starting with [0x](#) as sequence with hexadecimal digits. Similarly, the device also interprets a character sequence starting with multiple zeros as Base64 encoded binary data.

Authentication - X.509

IKE auth. cert. local

Specifies the name of the local peer identified in the digital certificate. The device uses this digital certificate for authentication of the local peer on the remote side. The digital certificate binds the identity of the local peer to the specified public key signed by the certification authority (CA) specified in the *IKE auth. cert. CA* field.

Possible values:

Alphanumeric ASCII character string with 1..128 characters

IKE auth. cert. CA

Specifies the name of the Certification Authority (CA) which signed the digital certificate. The device uses this digital certificate for signature verification of the local and remote certificates.

Possible values:

Alphanumeric ASCII character string with 1..128 characters


Encrypted private key

Specifies the file name for the private key. The prerequisite is that the key saved in the device is encrypted with a passphrase. The key requires that, in the *Encrypted key/PKCS12 passphrase* field, you specify the passphrase. The device considers the key and the digital certificate unmatched until the key is decrypted.

Possible values:

Alphanumeric ASCII character string with 0..128 characters

Encrypted key/PKCS12 passphrase

Specifies the passphrase that the device uses for decryption of the private key specified in the *Encrypted private key* field. You can view the passphrase by clicking the  icon.

Possible values:

Alphanumeric ASCII character string with 0..128 characters

Authentication - PKCS 12


IKE auth. cert. local

Specifies the name of the local peer identified in the digital certificate. The device uses this digital certificate for authentication of the local peer on the remote side.

Possible values:

Alphanumeric ASCII character string with 1..128 characters

Encrypted key/PKCS12 passphrase

Specifies the passphrase that the device uses for decryption of the private key specified in the *Encrypted private key* field. You can view the passphrase by clicking the  icon.

Possible values:

Alphanumeric ASCII character string with 0..128 characters

Endpoint and traffic selectors

Local endpoint

Specifies the hostname or IP address of the local IPsec VPN tunnel endpoint.

Possible values:

[any](#) (default setting)

The device uses the IP address of the interface the device uses to forward data to the remote endpoint.

Valid IPv4 address and netmask in CIDR notation

hostname

Alphanumeric ASCII character string with 1..128 characters

Remote endpoint

Specifies the hostname or IP address of the remote IPsec VPN tunnel endpoint.

Possible values:

[any](#) (default setting)

The device accepts any IP address when establishing an IKE-SA as a VPN *Responder*.

Valid IPv4 address and netmask in CIDR notation

If you specify that the device is a *Responder* for this VPN tunnel, then the device accepts a network in CIDR notation, during IKE-SA establishment.

hostname

Alphanumeric ASCII character string with 1..128 characters

Add traffic selector

Traffic selector description

Specifies the user-defined description for the traffic selector.

Possible values:

Alphanumeric ASCII character string with 1..128 characters

Source address (CIDR)

Specifies the IP address and netmask of the source host. When the device forwards packets containing this source IP address over a VPN tunnel, the device applies the settings specified in this field. Furthermore, the device applies the associated IPsec and IKE-SA settings, to every IP packet that the device forwards containing the source IP address in the range specified by the source IP and netmask.

Possible values:

Valid IPv4 address and netmask in CIDR notation

[any](#) (default setting)

The device applies the settings to every packet that the device forwards.

Source restrictions

Specifies the optional source restrictions using names or numbers entered as `<protocol /port>`. The device sends only the type of data specified through the VPN tunnel.

Examples:

- `tcp/http` is equal to `6/80`
- `udp` is equal to `udp/any`
- `/53` is equal to `any/53`

Possible values:

Alphanumeric ASCII character string with 1..32 characters

`<empty>` (default setting)

The device uses `any/any` as the restriction.

Destination address (CIDR)

Specifies the IP address and netmask of the destination. When the device forwards packets containing this destination IP address over a VPN tunnel, the device applies the settings specified in this field. Furthermore, the device applies the associated IPsec and IKE-SA settings to every IP packet that the device forwards containing the destination IP address in the range specified by the destination IP and netmask.

Possible values:

Valid IPv4 address and netmask in CIDR notation

`any` (default setting)

The device applies the settings to every packet that the device forwards.

Destination restrictions

Specifies the optional destination restrictions using names or numbers entered as `<protocol /port>`. The device accepts only the type of data specified from the VPN tunnel.

Examples:

- `tcp/http` is equal to `6/80`
- `udp` is equal to `udp/any`
- `/53` is equal to `any/53`

Possible values:

Alphanumeric ASCII character string with 1..32 characters

`<empty>` (default setting)

The device uses `any/any` as the restriction.



Removes the corresponding table row.

Add

Adds a table row to the *Add traffic selector* table.

Advanced configuration

For each VPN tunnel you can specify the parameters using the following tabs:

- [Advanced configuration - General](#)

Advanced configuration - General

Margin time [s]

Specifies the time in seconds before the connection or the keying channel expires. Afterwards, the device attempts to negotiate a replacement.

Possible values:

[1..1800](#) (default setting: [150](#))

The default setting is equal to 2.5 minutes. The maximum value is half an hour.

Advanced configuration - IKE/Key-exchange

Version

Specifies the version of the IKE protocol for the VPN connection.

Possible values:

[auto](#) (default setting)

The VPN starts with protocol IKEv2 as the *Initiator* and accepts IKEv1/v2 as the *Responder*.

[i ke v1](#)

The VPN starts with the IKEv1 (ISAKMP) protocol.

[i ke v2](#)

The VPN starts with the IKEv2 protocol.

Startup

Specifies if the device starts this instance as a *Responder* or *Initiator*.

Possible values:

[i n i t i a t o r](#)

The device starts a key exchange with the *Responder*.

[r e s p o n d e r](#) (default setting)

The device waits for the *Initiator* to start the key exchange and parameter negotiation.

If the remote peer sends data packets to a specific selector, then the device attempts to establish the connection as the *Responder*. Establishing a connection as a *Responder* depends upon other settings for this connection. For example, if you specify in the [Remote endpoint](#) field the value [any](#), then the device prevents the remote device from initiating the connection.

IKEv1 DPD timeout [s]

Specifies the timeout, in seconds, before the local peer declares the remote peer dead, if the remote peer is unresponsive.

The device supports the [IKEv1 DPD timeout \[s\]](#) function using IKEv1.

Possible values:

- 0
Deactivates the function.
- 1 . . 86400 (24 h) (default setting: 120)

IKE lifetime [s]

Specifies the lifetime, in seconds, of the IKE security association between two network devices to support secure communication. The devices establish a security association after exchanging a set of pre-defined keys.

Possible values:

- 300 . . 86400 (default setting: 28800)
The default setting is 8 hours. The maximum setting is 24 hours.

IKE local identifier type

Specifies the type of local peer identifier that the device uses for the *IKE local ID* parameter.

Possible values:

default (default setting)

The behavior depends on the value you specify in the following authentication methods:

- *Pre-shared key*
The device uses the IP address specified in the *Local endpoint* field as the local identifier. You find the *Local endpoint* field in section “Endpoint and traffic selectors” on page 271.
- *X.509* or *PKCS 12*
The device uses the distinguished name (DN) contained in the local *IKE auth. cert. local* certificate.

address

In the *IKE local ID* field, the device uses the IP address or the DNS hostname specified in the *Local endpoint* field. You find the *Local endpoint* field in section “Endpoint and traffic selectors” on page 271.

id

The device identifies the value specified in the *IKE local ID* field as one of the following types:

- An IPv4 address or a DNS hostname
- A key identifier which specifies the data that the device uses to pass vendor-specific information. The device uses the information to identify which pre-shared key it uses for aggressive mode authentication during negotiations.
- An FQDN web address, for example, *foo.bar.com*
- An email address
- The *ASN.1 X.500 Distinguished Name (DN)* contained within the *IKE auth. cert. remote* field. The local and remote devices exchange their digital certificates to establish the Security Association (SA).

IKE local ID

Specifies the local peer identifier that the device sends to the remote device in the ID payload during phase 1 negotiations. The device uses the ID payload to identify the *Initiator* of the Security Association (SA). The *Responder* uses the identity to determine the correct host system policy requirement for the Security Association (SA).

The formats for this parameter depend on the type specified in the *IKE local identifier type* field.

Possible values:

<empty> (default setting)

If in the *IKE local identifier type* field, you specify the value *i d*, then specify the value using one of the following options:

- An IPv4 address or a DNS hostname
- A previously specified key identifier, specifying data that the device uses to pass vendor-specific information.
- An FQDN web address, for example, `foo.bar.com`
- An email address
- An X.500 distinguished name

Refer to the following syntax as an example when adding the item:

`CN = XY-D, C = DE, L = NT, ST = BW O = COMPANY, OU = DEV, E = testuser@example.com`

Remote identifier type

Specifies the type of remote peer identifier that the device uses for the *Remote ID* parameter.

Possible values:

any (default setting)

The device accepts every received remote identifier without further verification.

address

In the *Remote ID* field, the device uses the IP address or the DNS hostname specified in the *Remote endpoint* field. You find the *Remote endpoint* field in section “Endpoint and traffic selectors” on page 271.

i d

The device identifies the value specified in the *Remote ID* field as one of the following types:

- An IPv4 address or a DNS hostname
- A key identifier which specifies the data that the device uses to pass vendor-specific information. The device uses the information to identify which pre-shared key it uses for aggressive mode authentication during negotiations.
- An FQDN web address, for example, `foo.bar.com`
- An email address
- The *ASN.1 X.500 Distinguished Name (DN)* contained within the *IKE auth. cert. remote* field. The local and remote devices exchange their digital certificates to establish the Security Association (SA).

Remote ID

Specifies the remote peer identifier which the device compares with the value in the ID payload during phase 1 negotiations. The device uses the ID payload to identify the *Initiator* of the Security Association (SA). The *Responder* uses the identity to determine the correct host system policy requirement for the Security Association (SA).

The formats for this parameter depend on the type specified in the *Remote identifier type* field.

Possible values:

<empty> (default setting)

If in the *Remote identifier type* field, you specify the value *i d*, then specify the value using one of the following options:

- An IPv4 address or a DNS hostname
- A previously specified key identifier, specifying data that the device uses to pass vendor-specific information.
- An FQDN web address, for example, `foo.bar.com`

- An email address
 - An X.500 distinguished name
- Refer to the following syntax as an example when adding the item:
CN = XY-D, C = DE, L = NT, ST = BW, O = COMPANY, OU = DEV, E = testuser@example.com

IKE exchange mode

Specifies the use of the phase 1 exchange mode for IKEv1.

The purpose of IKE phase 1 is to establish a secure authenticated communication channel. The device uses the Diffie-Hellman (DH) key exchange algorithm to generate a shared secret key. The device then uses the shared secret key to further encrypt IKE communications.

Possible values:

[main](#) (default setting)

The main mode for phase 1 provides identity protection.

[aggressive](#)

You use the aggressive mode to reduce round trips.

IKE key agreement

Specifies which Diffie-Hellman (DH) key agreement algorithm the device uses for establishing the IKE-SA session key.

Possible values:

[any](#)

The device accepts every algorithm when specified as the *Responder*.

[modp1024](#) (default setting)

The value represents an RSA algorithm with 1024 bits modulus which is DH Group 2.

[modp1536](#)

The value represents an RSA algorithm with 1536 bits modulus which is DH Group 5.

[modp2048](#)

The value represents an RSA algorithm with 2048 bits modulus which is DH Group 14.

[modp3072](#)

The value represents an RSA algorithm with 3072 bits modulus which is DH Group 15.

[modp4096](#)

The value represents an RSA algorithm with 4096 bits modulus which is DH Group 16.

IKE integrity (MAC)

Specifies which IKE Integrity Message Authentication Code (MAC) algorithm the device uses. To help keep the information on the VPN secure, the Hash-based Message Authentication Code (HMAC) process in the sending device mixes (hashes) the message data with a shared secret key. The receiving device mixes the results (hash value) with the secret key again, and then applies the hash function a second time.

Possible values:

[any](#)

When you specify the device as the *Responder*, the device accepts any algorithm. When you specify the device as the *Initiator*, the device uses various pre-defined algorithms.

[hmacmd5](#)

The device uses the Message Digest Algorithm 5 (MD5) for the hash function calculation.

[hmacsha1](#) (default setting)

The device uses the Secure Hash Algorithm version 1 (SHA-1) for the hash function calculation.

[hmacsha256](#)

The device uses SHA-256 (part of the version 2 family) for the hash function calculation which the device computes with 32 bit words.

[hmacsha384](#)

The device uses SHA-384 (part of the version 2 family) for hash function calculation which the device computes using a shorter version of SHA-512.

[hmacsha512](#)

The device uses SHA-512 (part of the version 2 family) for hash function calculation which the device computes with 64 bit words.

Note: We recommend to use the setting [hmacsha256](#) or higher.

IKE encryption

Specifies the IKE encryption algorithm that the device uses.

Possible values:

[any](#)

When you specify the device as the *Responder*, the device accepts any algorithm. When you specify the device as the *Initiator*, the device uses various pre-defined algorithms.

[des](#)

The device uses the Data Encryption Standard (DES) block cipher for encryption of message data with a 56 bit key.

[des3](#)

The device uses the Triple DES block cipher for encryption of message data which applies the 56 bit key, from DES, 3 times to each block.

[aes128](#) (default setting)

The device uses the Advanced Encryption Standard (AES) with a block size of 128 bits, and a key length of 128 key bits.

[aes192](#)

The device uses the AES with a block size of 128 bits, and a key length of 192 key bits.

[aes256](#)

The device uses the AES with a block size of 128 bits, and a key length of 256 key bits.

Note: We recommend to use the setting [aes128](#) or higher.

Advanced configuration - IPsec/Data-exchange

IPsec lifetime [s]

Specifies the lifetime, in seconds, of the IPsec security association between 2 network devices to support secure communication. The devices establish a security association after exchanging a set of pre-defined keys.

Possible values:

[300](#) . [28800](#) (default setting: [3600](#))

The default setting is one hour. The maximum setting is 8 hours.

IPsec integrity (MAC)

Specifies which IPsec Integrity MAC algorithm the device uses for the instance. To help keep the information on the VPN secure, the Hash-based Message Authentication Code (HMAC) process in the sending device mixes (hashes) the message data with a shared secret key. The receiving device mixes the results (hash value) with the secret key again, and then applies the hash function a second time.

Possible values:

[any](#)

When you specify the device as the *Responder*, the device accepts any algorithm. When you specify the device as the *Initiator*, the device uses various pre-defined algorithms.

[hmacmd5](#)

The device uses the Message Digest Algorithm 5 (MD5) for the hash function calculation.

[hmacsha1](#) (default setting)

The device uses the Secure Hash Algorithm version 1 (SHA-1) for the hash function calculation.

[hmacsha256](#)

The device uses SHA-256 (part of the version 2 family) for the hash function calculation which the device computes with 32 bit words.

[hmacsha384](#)

The device uses SHA-384 (part of the version 2 family) for hash function calculation which the device computes using a shorter version of SHA-512.

[hmacsha512](#)

The device uses SHA-512 (part of the version 2 family) for hash function calculation which the device computes with 64 bit words.

Note: We recommend to use the setting [hmacsha256](#) or higher.

IPsec encryption

Specifies the IPsec encryption algorithm that the device uses.

Possible values:

[any](#)

When you specify the device as the *Responder*, the device accepts any algorithm. When you specify the device as the *Initiator*, the device uses various pre-defined algorithms.

[des](#)

The device uses the Data Encryption Standard (DES) block cipher for encryption of message data with a 56 bit key.

[des3](#)

The device uses the Triple DES block cipher for encryption of message data which applies the 56 bit key, from DES, 3 times to each block.

[aes128](#) (default setting)

The device uses the Advanced Encryption Standard (AES) with a block size of 128 bits, and a key length of 128 key bits.

[aes192](#)

The device uses the AES with a block size of 128 bits, and a key length of 192 key bits.

[aes256](#)

The device uses the AES with a block size of 128 bits, and a key length of 256 key bits.

[aes128ctr](#)

AES-CTR with 128 key bits.

[aes192ctr](#)

AES-CTR with 192 key bits.

[aes256ctr](#)

AES-CTR with 256 key bits.

[aes128gcm64](#)

AES-GCM with a 64 bit ICV and 128 key bits.

[aes128gcm96](#)

AES-GCM with a 96 bit ICV and 128 key bits.

[aes128gcm128](#)

AES-GCM with a 128 bit ICV and 128 key bits.

[aes192gcm64](#)

AES-GCM with a 64 bit ICV and 192 key bits.

[aes192gcm96](#)

AES-GCM with a 96 bit ICV and 192 key bits.

[aes192gcm128](#)

AES-GCM with a 128 bit ICV and 192 key bits.

[aes256gcm64](#)

AES-GCM with a 64 bit ICV and 256 key bits.

[aes256gcm96](#)

AES-GCM with a 96 bit ICV and 256 key bits.

[aes256gcm128](#)

AES-GCM with a 128 bit ICV and 256 key bits.

Note: We recommend to use the setting [aes128](#) or higher.

IPsec key agreement

Specifies which Diffie-Hellman key agreement algorithm the device uses for establishing the IPsec-SA session key. If the *Perfect Forward Secrecy (PFS)* function is enabled and a compromise of a single key occurs, then the integrity remains for subsequently generated keys.

Possible values:

[any](#)

When you specify the device as the *Responder*, the device accepts any algorithm. When you specify the device as the *Initiator*, the device uses various pre-defined algorithms.

[modp1024](#) (default setting)

The value represents an Rivest-Shamir-Adleman (RSA) algorithm with 1024 bits modulus. This value is Diffie Hellman (DH) Group 2.

[modp1536](#)

The value represents an RSA algorithm with 1536 bits modulus which is DH Group 5.

[modp2048](#)

The value represents an RSA algorithm with 2048 bits modulus which is DH Group 14.

[modp3072](#)

The value represents an RSA algorithm with 3072 bits modulus which is DH Group 15.

[modp4096](#)

The value represents an RSA algorithm with 4096 bits modulus which is DH Group 16.

[none](#)

The device disables the *PFS* function. Disabling the *PFS* function is considered a confidentiality violation and therefore a security risk.

6 Switching

The menu contains the following dialogs:

- Switching Global
- Rate Limiter
- Filter for MAC Addresses
- QoS/Priority
- VLAN

6.1 Switching Global

[Switching > Global]

This dialog lets you specify the following settings:

- Change the Aging time of the MAC address table (forwarding database) entries
- Enable the flow control in the device

If a large number of data packets are received in the priority queue of a port at the same time, then this can cause the port memory to overflow. This happens, for example, when the device receives data on a Gigabit port and forwards it to a port with a lower bandwidth. The device discards superfluous data packets.

The flow control mechanism defined in IEEE 802.3 helps ensure that no data packets are lost due to a buffer overflow on a port. Shortly before the buffer memory of a port is completely full, the device signals to the connected devices that it is not accepting any more data packets from them.

- In full-duplex mode, the device sends a pause data packet.
- In half-duplex mode, the device simulates a collision.

The connected devices then stop sending data packets for the duration of the signaling. On an uplink port, this can possibly cause undesired sending interruptions in the higher-level network segment (“wandering backpressure”). The flow control mechanism thus lowers the network to the bandwidth that the slowest device in the network can process.

Configuration

MAC address

Displays the MAC address of the device.

Aging time [s]

Specifies the aging time in seconds.

Possible values:

10 . 500000 (default setting: 30)

The device monitors the age of the learned unicast MAC addresses. The device deletes address entries that exceed a particular age (aging time) from its MAC address table (forwarding database).

You find the MAC address table (forwarding database) in the [Switching > Filter for MAC Addresses](#) dialog.

In connection with the router redundancy, specify a time 30 s.

Flow control

Activates/deactivates the flow control in the device.

Possible values:

[marked](#)

The flow control is active in the device.

Additionally activate the flow control on the required ports. See the [Basic Settings > Port](#) dialog, [Configuration](#) tab, checkbox in the [Flow control](#) column.

[unmarked](#) (default setting)

The flow control is inactive in the device.

6.2 Rate Limiter

[Switching > Rate Limiter]

The device lets you limit the amount of data packets on the ports to help provide stable operation even with a large data volume. If the amount of data packets on a port exceed the threshold value, then the device discards the excess data packets on this port.

The rate limiter function operates only on Layer 2, and is used to limit the effects of storms of data packets that flood the device (typically Broadcasts).

The rate limiter function ignores protocol information on higher layers, such as IP or TCP.

The dialog contains the following tabs:

[\[Ingress\]](#)

[Ingress]

In this tab you enable the [Rate Limiter](#) function. The threshold value specifies the maximum amount of data packets the port receives. If the amount of data packets on a port exceed the specified threshold value, then the device discards the excess data packets on this port.

Table

For information on how to customize the appearance of the table, see [“Working with tables” on page 16](#).

Port

Displays the port number.

Unit

Specifies the unit for the threshold value:

Possible values:

[per cent](#) (default setting)

Specifies the threshold value as a percentage of the data rate of the port.

[pps](#)

Specifies the threshold value in data packets per second.

Broadcast mode

Activates/deactivates the rate limiter function for received broadcast data packets.

Possible values:

[marked](#)

[unmarked](#) (default setting)

If the threshold value is exceeded, then the device discards the excess broadcast data packets on this port.

Broadcast threshold

Specifies the threshold value for received broadcasts on this port.

Possible values:

0 . 14880000 (default setting: 0)

The value 0 deactivates the rate limiter function on this port.

If you select the value `percent` in the `Unit` column, then enter a percentage value from 1 to 100.

If you select the value `pps` in the `Unit` column, then enter an absolute value for the data rate.

Multicast mode

Activates/deactivates the rate limiter function for received multicast data packets.

Possible values:

`marked`

`unmarked` (default setting)

If the threshold value is exceeded, then the device discards the excess multicast data packets on this port.

Multicast threshold

Specifies the threshold value for received multicasts on this port.

Possible values:

0 . 14880000 (default setting: 0)

The value 0 deactivates the rate limiter function on this port.

If you select the value `percent` in the `Unit` column, then enter a percentage value from 0 to 100.

If you select the value `pps` in the `Unit` column, then enter an absolute value for the data rate.

Unknown unicast mode

Activates/deactivates the rate limiter function for received unicast data packets with an unknown destination address.

Possible values:

`marked`

`unmarked` (default setting)

If the threshold value is exceeded, then the device discards the excess unicast data packets on this port.

Unicast threshold

Specifies the threshold value for received unicasts with an unknown destination address on this port.

Possible values:

0 . 14880000 (default setting: 0)

The value 0 deactivates the rate limiter function on this port.

If you select the value `percent` in the `Unit` column, then enter a percentage value from 0 to 100.

If you select the value `pps` in the `Unit` column, then enter an absolute value for the data rate.

6.3 Filter for MAC Addresses

[Switching > Filter for MAC Addresses]

This dialog lets you display and edit address filters for the MAC address table (forwarding database). Address filters specify the way the data packets are forwarded in the device based on the destination MAC address.

Each table row represents one filter. The device automatically sets up the filters. The device lets you set up additional filters manually.

The device forwards the data packets as follows:

- When the table contains the destination address of a data packet, the device forwards the data packet from the receiving port to the port specified in the table row.
- When there is no table row for the destination address, the device forwards the data packet from the receiving port to every other port.

Table

To delete the learned MAC addresses from the MAC address table (forwarding database), click in the [Basic Settings > Restart](#) dialog the [Clear FDB](#) button.

For information on how to customize the appearance of the table, see [“Working with tables” on page 16](#).

Buttons



Add

Opens the [Create](#) window to add a table row.

- In the [MAC address](#) field, you specify the destination MAC address.
- In the [VLAN ID](#) field, you specify the VLAN ID.
- In the list field, you select the ports.
 - If the destination MAC address is a unicast address, select exactly one port.
 - If the destination MAC address is a multicast or broadcast address, select one or more ports.
 - Do not select a port to add a [Discard](#) filter. The device discards data packets with the destination MAC address specified in the table row.



Remove

Removes the selected table row.



Clear FDB

Removes the MAC addresses from the forwarding table that have the value [Learned](#) in the [Status](#) column.

Address

Displays the destination MAC address to which the table row relates.

VLAN ID

Displays the ID of the VLAN to which the table row relates.

The device learns the MAC addresses for every VLAN separately (independent VLAN learning).

Status

Displays how the device has set up the address filter.

Possible values:

[Learned](#)

Address filter set up automatically by the device based on received data packets.

[Mgmt](#)

MAC address of the device. The address filter is protected against changes.

[Permanent](#)

Address filter set up manually. The address filter stays set up permanently.

<Port number>

Displays how the corresponding port transmits data packets which it directs to the adjacent destination address.

Possible values:

-

The port does not transmit any data packets to the destination address.

[learned](#)

The port transmits data packets to the destination address. The device has automatically set up the filter based on received data packets.

[unicast static](#)

The port transmits data packets to the destination address. A user has set up the filter.

[multicast static](#)

The port transmits data packets to the destination address. A user has set up the filter.

6.4 QoS/Priority

[Switching > QoS/Priority]

Communication networks transmit a number of applications at the same time that have different requirements as regards availability, bandwidth and latency periods.

QoS (Quality of Service) is a procedure defined in IEEE 802.1D. It is used to distribute resources in the network. You therefore have the possibility of providing minimum bandwidth for necessary applications. The prerequisite is that the end devices and the devices in the network support prioritized data transmission. Data packets with high priority are given preference when transmitted by devices in the network. You transfer data packets with lower priority when there are no data packets with a higher priority to be transmitted.

The device provides the following setting options:

- You specify how the device evaluates QoS/prioritization information for inbound data packets.
- For outbound packets, you specify which QoS/prioritization information the device writes in the data packet (for example priority for management packets, *Port priority*).

Note: If you use the functions in this menu, then disable the flow control. The flow control is inactive if in the *Switching > Global* dialog, *Configuration* frame the *Flow control* checkbox is unmarked.

The menu contains the following dialogs:

- QoS/Priority Global
- QoS/Priority Port Configuration
- 802.1D/p Mapping

6.4.1 QoS/Priority Global

[Switching > QoS/Priority > Global]

The device lets you maintain access to the device management, even in situations with heavy utilization. In this dialog, you specify the required QoS/priority settings.

Configuration

VLAN priority for management packets

Specifies the VLAN priority for sending management data packets. Depending on the VLAN priority, the device assigns the data packet to a specific *traffic class* and thus to a specific priority queue of the port.

Possible values:

0 . 7 (default setting: 0)

In the [Switching > QoS/Priority > 802.1D/p Mapping](#) dialog, you assign a *traffic class* to every VLAN priority.

IP DSCP value for management packets

Specifies the IP DSCP value for sending management data packets. Depending on the IP DSCP value, the device assigns the data packet to a specific *traffic class* and thus to a specific priority queue of the port.

Possible values:

0 (be/cs0) . . 63 (default setting: 0 (be/cs0))

Some values in the list also have a DSCP keyword, for example 0 (be/cs0), 10 (af 11) and 46 (ef). These values are compatible with the *IP Precedence* model.

Queues per port

Displays the number of priority queues per port.

The device has 8 priority queues per port. You assign every priority queue to a specific *traffic class* (*traffic class* according to IEEE 802.1D).

6.4.2 QoS/Priority Port Configuration

[Switching > QoS/Priority > Port Configuration]

In this dialog, you specify for every port how the device processes received data packets based on their QoS/priority information.

Table

For information on how to customize the appearance of the table, see [“Working with tables” on page 16](#).

Port

Displays the port number.

Port priority

Specifies what VLAN priority information the device writes into a data packet if the data packet contains no priority information. After this, the device forwards the data packet depending on the value specified in the *Trust mode* column.

Possible values:

0 . 7 (default setting: 0)

6.4.3 802.1D/p Mapping

[Switching > QoS/Priority > 802.1D/p Mapping]

The device forwards data packets with a VLAN tag according to the contained QoS/priority information with a higher or lower priority.

In this dialog, you see which VLAN priority is assigned to which *traffic class*. You assign the *traffic classes* to the priority queues of the ports.

Table

For information on how to customize the appearance of the table, see [“Working with tables” on page 16](#).

VLAN priority

Displays the VLAN priority.

Traffic class

Specifies the *traffic class* assigned to the VLAN priority.

Possible values:

0 . 7

0 assigned to the priority queue with the lowest priority.

7 assigned to the priority queue with the highest priority.

Note: Among other things redundancy mechanisms use the highest *traffic class*. Therefore, select another *traffic class* for application data.

Default assignment of the VLAN priority to traffic classes

VLAN Priority	Traffic class	Content description according to IEEE 802.1D
0	2	Best Effort Normal data without prioritizing
1	0	Background Non-time-sensitive data and background services
2	1	Standard Normal data
3	3	Excellent Effort Crucial data
4	4	Controlled Load Time-sensitive data with a high priority
5	5	Video Video transmission with delays and jitter <100 ms
6	6	Voice Voice transmission with delays and jitter <10 ms
7	7	Network Control Data for network management and redundancy mechanisms

6.5 VLAN

[Switching > VLAN]

With VLAN (Virtual Local Area Network) you distribute the data packets in the physical network to logical subnets. This provides you with the following advantages:

- High flexibility
 - With VLAN you distribute the data packets to logical networks in the existing infrastructure. Without VLAN, it would be necessary to have additional devices and complicated cabling.
 - With VLAN you specify network segments independently of the location of the individual end devices.
- Improved throughput
 - In VLANs data packets can be transferred by priority. When the priority is high, the device transfers the data of a VLAN preferentially, for example for time-sensitive applications such as VoIP phone calls.
 - When the data packets and Broadcasts are distributed in small network segments instead of in the entire network, the network load is considerably reduced.
- Increased security
 - The distribution of the data packets among individual logical networks makes unwanted accessing more difficult and strengthens the system against attacks such as MAC Flooding or MAC Spoofing.

The device supports packet-based “tagged” VLANs according to IEEE 802.1Q. The VLAN tagging in the data packet indicates the VLAN to which the data packet belongs.

The device forwards the tagged data packets of a VLAN only on ports that are assigned to the same VLAN. This reduces the network load.

The device learns the MAC addresses for every VLAN separately (independent VLAN learning).

The menu contains the following dialogs:

- [VLAN Global](#)
- [VLAN Configuration](#)
- [VLAN Port](#)

6.5.1 VLAN Global

[Switching > VLAN > Global]

This dialog lets you view general VLAN parameters for the device.

Configuration

Buttons

 Reset VLAN settings

Resets the VLAN settings of the device to the default setting.

Note that you lose your connection to the device if you have changed the VLAN for the device management in the [Basic Settings > Network > Global](#) dialog.

Max. VLAN ID

Highest ID assignable to a VLAN.

See the [Switching > VLAN > Configuration](#) dialog.

VLANs (max.)

Displays the maximum number of VLANs possible.

See the [Switching > VLAN > Configuration](#) dialog.

VLANs

Number of VLANs currently set up in the device.

See the [Switching > VLAN > Configuration](#) dialog.

The VLAN 1 is permanently set up in the device.

6.5.2 VLAN Configuration

[Switching > VLAN > Configuration]

In this dialog, you manage the VLANs. To set up a VLAN, add a further table row. There you specify for each port if it transmits data packets of the respective VLAN and if the data packets contain a VLAN tag.

You distinguish between the following VLANs:

- The user sets up static VLANs.
- The device sets up dynamic VLANs automatically and removes them if the prerequisites cease to apply.

For the following functions the device sets up dynamic VLANs:

- *Routing*: The device sets up a VLAN for every router interface.

Table

For information on how to customize the appearance of the table, see [“Working with tables” on page 16](#).

Buttons



Add

Opens the *Create* window to add a table row.

In the *VLAN ID* field, you specify the VLAN ID.



Remove

Removes the selected table row.

VLAN ID

ID of the VLAN.

The device supports up to 64 VLANs simultaneously set up.

Possible values:

1..4042

Status

Displays how the VLAN is set up.

Possible values:

other

VLAN 1

permanent

VLAN set up by the user.

If you save the settings in the non-volatile memory, then the VLANs with this setting remain set up after a restart.

Name

Specifies the name of the VLAN.

Possible values:

Alphanumeric ASCII character string with 1..32 characters

<Port number>

Specifies if the respective port transmits data packets of the VLAN and if the data packets contain a VLAN tag.

Possible values:

- (default setting)

The port is not a member of the VLAN and does not transmit data packets of the VLAN.

T = Tagged

The port is a member of the VLAN and transmits the data packets with a VLAN tag. You use this setting for uplink ports, for example.

LT = Tagged Learned

The port is a member of the VLAN and transmits the data packets with a VLAN tag. The device has automatically set up the entry based on the *GVRP* or *MVRP* function.

F = Forbidden

The port is not a member of the VLAN and does not transmit data packets of this VLAN.

U = Untagged (default setting for VLAN 1)

The port is a member of the VLAN and transmits the data packets without a VLAN tag. Use this setting if the connected device does not evaluate any VLAN tags, for example on end ports.

LU = Untagged Learned

The port is a member of the VLAN and transmits the data packets without a VLAN tag.

The device has automatically set up the entry based on the *GVRP* or *MVRP* function.

Note: Verify that the port on which the network management station is connected is a member of the VLAN in which the device transmits the management data. In the default setting, the device transmits the management data on VLAN 1. Otherwise, the connection to the device terminates when you transfer the changes to the device. The access to the device management is possible only using the Command Line Interface through the serial interface.

6.5.3 VLAN Port

[Switching > VLAN > Port]

In this dialog, you specify how the device handles received data packets that have no VLAN tag, or whose VLAN tag differs from the VLAN ID of the port.

This dialog lets you assign a VLAN to the ports and thus specify the port VLAN ID.

Additionally, you also specify for each port how the device forwards data packets and one of the following situations occurs:

- The port receives data packets without a VLAN tagging.
- The port receives data packets with VLAN priority information (VLAN ID 0, priority tagged).
- The VLAN ID in the tag of the data packet differs from the VLAN ID of the port.

Table

For information on how to customize the appearance of the table, see [“Working with tables” on page 16](#).

Port

Displays the port number.

Port-VLAN ID

Specifies the VLAN ID which the device assigns to data packets received without a VLAN tag.

Prerequisites:

- In the *Acceptable packet types* column, the value `admi tAl l` is specified.

Possible values:

- 1. `4042` (default setting: 1)
A VLAN you set up.

Acceptable packet types

Specifies if the port transmits or discards received data packets without a VLAN tag.

Possible values:

- `admi tAl l` (default setting)
The port accepts data packets both with and without a VLAN tag.
- `admi tOnl yVl anTagged`
The port accepts only data packets tagged with a VLAN ID 1.

Ingress filtering

Activates/deactivates the ingress filtering.

Possible values:

marked (default setting)

The ingress filtering is active.

The device compares the VLAN ID in the data packet with the VLANs of which the port is a member. See the [Switching > VLAN > Configuration](#) dialog. If the VLAN ID in the data packet matches one of these VLANs, then the device forwards the data packet. Otherwise, the device discards the data packet.

unmarked

The ingress filtering is inactive.

The device forwards received data packets without comparing the VLAN ID. Thus, the device also forwards data packets in VLANs in which the port is not a member.

7 Routing

The menu contains the following dialogs:

- [Routing Global](#)
- [Routing Interfaces](#)
- [ARP](#)
- [Open Shortest Path First](#)
- [Routing Table](#)
- [L3 Relay](#)
- [Loopback Interface](#)
- [L3-Redundancy](#)
- [NAT](#)

7.1 Routing Global

[Routing > Global]

The [Routing](#) menu lets you specify the Routing functions settings for transmitting data on Layer 3 of the ISO/OSI layer model.

For security reasons, the following functions are permanently disabled in the device:

- [Source Routing](#)
With source routing, the data packet contains the routing information and overwrites the settings in the router with it.
- [ICMP Redirects](#)
ICMP redirect data packets are able to modify the routing table. The device generally ignores received ICMP redirect data packets. The settings in the [Routing > Interfaces > Configuration](#) dialog, column [ICMP redirects](#), have an effect only on the sending of ICMP redirect data packets.

In accordance with RFC 2644, the device does not exchange any broadcast data packets from external networks in a local network. This behavior supports you in protecting the devices in the local network against overloading, for example due to so-called smurf attacks.

This dialog lets you enable the routing function in the device and to specify further settings.

Operation

Operation

Enables/disables the [Routing](#) function in the device.

Possible values:

[On](#)

The [Routing](#) function is enabled.

Also activate the routing function on the router interfaces. See the [Routing > Interfaces > Configuration](#) dialog.

[Off](#) (default setting)

The [Routing](#) function is disabled.

ICMP filter

In the *ICMP filter* frame, you have the option of limiting the transmission of ICMP messages on the set up router interfaces. A limitation is meaningful for several reasons:

- A large number of *ICMP Error* messages influences the router performance and reduces the available network bandwidth.
- Malicious senders use *ICMP Redirect* messages to perform man-in-the-middle attacks or to divert data packets through “black hole” for the purpose of supervision or denial-of-service (DoS).
- An *ICMP echo reply* packet is the response to an *ICMP echo request* packet which can be misused to discover vulnerable devices and routers in the network.

Send echo reply

Activates/deactivates the responding to pings on the router interfaces.

Possible values:

marked (default setting)

Responding to pings is active.

The device responds to a received *ICMP echo request* packet with an *ICMP echo reply* packet.

unmarked

Responding to pings is inactive.

Send redirects

Activates/deactivates the sending of *ICMP Redirect* messages on the router interfaces.

Possible values:

marked (default setting)

The sending of *ICMP Redirect* messages is active.

In the [Routing > Interfaces > Configuration](#) dialog, you have the option of individually activating the sending on every router interface. See the *ICMP redirects* function.

unmarked

The sending of *ICMP Redirect* messages is inactive.

This setting helps prevent the multiplication of data packets, if both hardware and software functions of the device forward a copy of the same data packet.

Rate limit interval [ms]

Specifies the average minimum period in milliseconds between each *ICMP echo request* packet sent by the device. The device limits its *ICMP echo reply* packets to a number determined by a *Token bucket* algorithm.

Possible values:

0 . 2147483647 ($2^{31} - 1$) (default setting: 1000)

The *Rate limit* is disabled.

10 . 2147483647 ($2^{31} - 1$) (default setting: 1000)

- In periods without sending an ICMP packet, the device accumulates tokens to send bursts when necessary.
- In the case of a burst, the interval is shorter than specified here.
- The maximum allowed value of the transmission *Rate limit* is 100 data packets per 1000 ms.

Rate limit burst size

Displays the maximum number of ICMP packets, the device sends during a burst to each receiver.

Possible values:

6

Information

Default TTL

Displays the fixed TTL value 64 which the device adds to IP packets that the device management sends.

TTL (Time To Live, also known as “Hop Count”) identifies the maximum number of routing steps, which the sent *ICMP echo request* packet may traverse on its way from the sender to the receiver. Every router on the transmission path reduces the value in the IP packet by 1. If a router receives a data packet with the TTL value 1, then the router discards the IP packet. The router reports to the source that it has discarded the IP packet.

7.2 Routing Interfaces

[Routing > Interfaces]

This menu lets you specify the settings for the router interfaces.

The menu contains the following dialogs:

[Routing Interfaces Configuration](#)

[Routing Interfaces Secondary Interface Addresses](#)

7.21 Routing Interfaces Configuration

[Routing > Interfaces > Configuration]

This dialog lets you specify the settings for the router interfaces.

To set up a port-based router interface, edit the table rows. To set up a VLAN-based router interface, use the [Wizard](#) window.

Note: To help prevent losing data packets, we recommend connecting the device to a device that supports Shared VLAN Learning (SVL) through a single port rather than through one port for each VLAN interface.

Table

For information on how to customize the appearance of the table, see [“Working with tables” on page 16](#).

Buttons



Add

Opens the [Create](#) window to add a table row. In the [VLAN ID](#) field, you specify the VLAN ID.



Remove

Removes the selected table row.



Wizard

Opens the [Wizard](#) window that helps you associate the ports with the address of one or more desired senders. See [“\[Wizard: Configure VLAN router interface\]” on page 303](#).

Port

Displays the number of the port or VLAN belonging to the router interface.

Name

Name of the port.

Possible values:

Alphanumeric ASCII character string with 0..64 characters

The device accepts the following characters:

- <space>
- 0 . 9
- a . z
- A . Z
- ! # \$ % & ' () * + , - . / : ; <=> ? @ [\ \] ^ _ ` { } ~

Port on

Activates/deactivates the port.

Possible values:

`marked` (default setting)

The port is active.

`unmarked`

The port is inactive. The port does not send or receive any data.

Port status

Displays the operating state of the port.

Possible values:

`up`

The port is enabled.

`down`

The port is disabled.

IP address

Specifies the IP address for the router interface.

Possible values:

Valid IPv4 address (default setting: `0.0.0.0`)

Verify that the IP subnet of the router interface does not overlap with any subnet connected to another interface of the device:

- management port
- router interface
- loopback interface

Netmask

Specifies the netmask for the router interface.

Possible values:

Valid IPv4 netmask (default setting: `0.0.0.0`)

Routing

Activates/deactivates the *Routing* function on the router interface.

In the process, the device removes the state information from the packet filter. This includes potential DCE RPC information of the OPC enforcer. In the process, the device interrupts open communication connections.

Possible values:

`marked`

The *Routing* function is active.

- With port-based routing, the device transforms the port into a router interface. Enabling the *Routing* function removes the port from the VLANs in which it was previously a member. Disabling the *Routing* function does not re-establish the assignment; the port is not a member of any VLAN.
- With VLAN-based routing, the device forwards the data packets in the related VLAN.

`unmarked` (default setting)

The *Routing* function is inactive.

With VLAN-based routing, the device is still reachable through the router interface if the IP address and netmask are specified for the router interface.

Proxy ARP

Activates/deactivates the *Proxy ARP* function on the router interface. This feature lets you connect devices from other networks as if these devices could be reached in the same network.

Possible values:

`marked`

The *Proxy ARP* function is active.

The device responds to ARP requests from end devices that are located in other networks.

`unmarked` (default setting)

The *Proxy ARP* function is inactive.

MTU value

Specifies the maximum allowed size of IP packets on the router interface in bytes.

Possible values:

`0`

Restores the default value (`1500`).

`68 . 1500` (default setting: `1500`)

ICMP unreachable

Displays if the sending of *ICMP Destination Unreachable* messages is active on the router interface.

Possible values:

`marked`

The router interface sends *ICMP Destination Unreachable* messages.

ICMP redirects

Displays if the sending of *ICMP Redirect* messages is active on the router interface.

Possible values:

`marked`

The router interface sends *ICMP Redirect* messages.

`unmarked` (default setting)

The router interface does not send *ICMP Redirect* messages.

[Wizard: Configure VLAN router interface]

This *Wizard* window lets you set up VLAN-based router interfaces.

The *Wizard* window guides you through the following steps:

- [Create or select VLAN](#)
- [Setup VLAN](#)

Create or select VLAN

VLAN ID

Displays the VLANs set up in the device. To continue, select an item from the list. As an alternative, specify a value in the [VLAN ID](#) field below.

VLAN ID

Specifies the ID of a VLAN. As an alternative, select an item in the [VLAN ID](#) overview above. You can also set up a VLAN in the [Switching > VLAN > Configuration](#) dialog.

Possible values:

1..4042

Setup VLAN

VLAN ID

Displays the ID of the VLAN that you have specified in the preceding *Wizard* step.

Name

Specifies the name of the VLAN. This setting overwrites the setting specified for the port in the [Switching > VLAN > Configuration](#) dialog.

Possible values:

Alphanumeric ASCII character string with 1..32 characters
(hexadecimal ASCII code [0x20](#)..[0x7E](#)) including space characters

<Port number>

Displays the port number.

Member

Activates/deactivates the VLAN membership of the port. As a VLAN member, the port belongs to the router interface to be set up. This setting overwrites the setting for the port specified in the [Switching > VLAN > Configuration](#) dialog.

Possible values:

[marked](#)

The port is a member of the VLAN.

[unmarked](#)

The port is not a member of the VLAN.

Untagged

Activates/deactivates sending the data packets with a VLAN tag on the port. This setting overwrites the setting for the port specified in the [Switching > VLAN > Configuration](#) dialog.

Possible values:

[marked](#)

The port sends the data packets without a VLAN tag.

Use this setting if the connected device does not evaluate any VLAN tags, for example on ports to which an end device is directly connected.

[unmarked](#)

The port sends the data packets with a VLAN tag.

Port-VLAN ID

Specifies the VLAN ID which the device assigns to data packets received without a VLAN tag. This setting overwrites the setting for the port specified in the [Switching > VLAN > Port](#) dialog, column [Port-VLAN ID](#).

Possible values:

A VLAN you set up (default setting: 1)

Setup virtual router port

The device lets you specify up to 2 IP addresses (1 primary, 1 secondary) for a router interface and a total of up to 64 IP addresses.

When you assign a port to the router interface that already sends data packets to another VLAN, the device displays a message upon closing the [Wizard](#) window:

- If you click the [Yes](#) button, then the related ports send the data packets from now on only in the router VLAN.
In the [Switching > VLAN > Configuration](#) dialog, the related ports in the table row of the router VLAN have the value [U](#) or [T](#), in the table rows of other VLANs the value [-](#).
- If you click the [No](#) button, then the related ports send the data packets in the router VLAN and other VLANs. This setting possibly causes undesired behavior and may also pose a security risk.

Primary address

Address

Specifies the primary IP address for the router interface.

Possible values:

Valid IPv4 address

Netmask

Specifies the primary netmask for the router interface.

Possible values:

Valid IPv4 netmask

Secondary addresses

Address

Specifies a further IP address for the router interface (Multinetting).

Possible values:

Valid IPv4 address

Note: Specify an IP address which is different from the primary IP address of the router interface.

Netmask

Specifies the netmask for the secondary IP address.

Possible values:

Valid IPv4 netmask

Add

Adds a VLAN-based router interface.

7.2.2 Routing Interfaces Secondary Interface Addresses

[Routing > Interfaces > Secondary Interface Addresses]

This dialog lets you assign further IP addresses to the router interfaces. You use this function to connect a router interface to several subnets.

The device lets you specify up to 2 IP addresses (1 primary, 1 secondary) for a router interface and a total of up to 64 IP addresses.

Table

For information on how to customize the appearance of the table, see [“Working with tables” on page 16](#).

Buttons



Add

Opens the [Create](#) window to add another IP address to the router interface selected in the table.

- From the [Port](#) drop-down list, you select the port or VLAN to be assigned to the router interface.
- In the [Additional IP address](#) field, you specify the IP address.

Possible values:

Valid IPv4 address

- In the [Additional netmask](#) field, you specify the netmask.

Possible values:

Valid IPv4 netmask

Verify that the IP subnet of the router interface does not overlap with any subnet connected to another interface of the device:

- management port
- router interface
- loopback interface



Remove

Removes the selected table row.

Port

Displays the number of the port or VLAN belonging to the router interface.

IP address

Displays the primary IP address of the router interface. See the [Routing > Interfaces > Configuration](#) dialog.

Netmask

Displays the primary netmask of the router interface. See the [Routing > Interfaces > Configuration](#) dialog.

Additional IP address

Displays further IP addresses assigned to the router interface.

Additional netmask

Displays further netmasks assigned to the router interface.

7.3 **ARP**

[Routing > ARP]

Using the Address Resolution Protocol (ARP), the device learns the MAC address that belongs to an IP address.

The menu contains the following dialogs:

- [ARP Global](#)
- [ARP Current](#)
- [ARP Static](#)

7.3.1 ARP Global

[Routing > ARP > Global]

This dialog lets you set the ARP parameters and view statistical values.

Configuration

Aging time [s]

Specifies the average time in seconds, after which the device removes an entry from the ARP table. The device actually removes an entry after a randomly determined time in the range (0.5 to 1.5)x of the value defined here.

When there is data exchange with the associated device within this time period, the time measuring begins from the start again.

Possible values:

15 . 21600 (default setting: 1200)

Response timeout [s]

Specifies the time in seconds, that the device waits for a response before the query is seen as a failure.

Possible values:

1 . 10 (default setting: 1)

Retries

Specifies how many times the device repeats a failed query before it discards the query to this address.

Possible values:

0 . 10 (default setting: 4)

Information

Current entries total

Displays the number of entries that the ARP table currently contains.

This includes:

- Addresses of the devices which are connected to the router interfaces. See the [Routing > ARP > Current](#) dialog.
- Addresses of the devices which are connected to the device management. See the [Diagnostics > System > ARP](#) dialog.

Entries (max.)

Displays how many entries the ARP table can contain at a maximum.

Total entry peaks

Displays how many entries the ARP table has already contained at a maximum.

To reset the counter to the value 0, in the [Routing > ARP > Current](#) dialog, click the  button.

Current static entries

Displays the current number of statically set-up entries in the ARP table. See the [Routing > ARP > Static](#) dialog.

Static entries (max.)

Displays the number of statically set-up entries the ARP table can contain at a maximum.

7.3.2 ARP Current

[Routing > ARP > Current]

This dialog lets you view the ARP table and delete the dynamically set-up entries.

Table

For information on how to customize the appearance of the table, see [“Working with tables” on page 16](#).

Buttons

 Remove

Removes the selected table row.

 Clear ARP table

Removes the dynamically set up addresses from the ARP table.

Port

Displays the router interface on which the device has learned the IP/MAC address assignment.

IP address

Displays the IP address of the device that responded to an ARP query on this router interface.

MAC address

Displays the MAC address of the device that responded to an ARP query on this router interface.

Last updated

Displays the time in seconds since the current settings of the entry were registered in the ARP table.

Type

Displays the way in which the ARP entry was set up.

Possible values:

dynamic


Dynamically set-up entry.

When no data packet was sent to or received from the associated device by the end of the aging time, the device removes this entry from the ARP table.

You specify the aging time in the [Routing > ARP > Global](#) dialog, field [Aging time \[s\]](#).

static

Statically set-up entry.

When you remove the dynamically set-up addresses from the ARP table using the  button, the entry remains.

local

Identifies the IP/MAC address assignment of the router interface.

invalid

Invalid entry.

7.3.3 ARP Static

[Routing > ARP > Static]

This dialog lets you add to the ARP table IP/MAC address assignments that you have specified yourself.

Table

For information on how to customize the appearance of the table, see [“Working with tables” on page 16](#).

Buttons



Removes the selected table row.



Opens the [Wizard](#) window that helps you associate the ports with the address of one or more desired senders. See [“\[Wizard: ARP\]” on page 313](#).

IP address

Displays the IP address of the static ARP entry.

MAC address

Displays the MAC address that the device assigns to the IP address when answering an ARP request.

Port

Displays the router interface to which the device applies the IP/MAC address assignment.

Possible values:

[<Router interface>](#)

The device applies the IP/MAC address assignment to this router interface.

[no port](#)

The IP/MAC address assignment is currently not assigned to a router interface.

Active

Displays if the IP/MAC address assignment is active or inactive.

Possible values:

[marked](#)

The IP/MAC address assignment is active. The ARP table of the device contains the IP/MAC address assignment as a static entry.

[unmarked](#) (default setting)

The IP/MAC address assignment is inactive.

[Wizard: ARP]

The *Wizard* window lets you add the IP/MAC address assignments in the ARP table. The prerequisite is that at least one router interface is set up.

Edit ARP table

Perform the following steps:

Specify the IP address and the associated MAC address.

Note: Verify the MAC address carefully. Doing so can help protect the network against unauthorized devices that might perform a Man-in-the-Middle (MITM) attack.

Insert the IP/MAC address assignment in the *Static entries* field. To do this, click the *Add* button.

Close the *Wizard* window. To do this, click the *Finish* button.

Specify the router interface in the *Port* column.

Enable the IP/MAC address assignment. To do this, mark the checkbox in the *Active* column.

Static entries

Displays the static entries set-up. You can remove a static entry by clicking the **X** icon.

IP address

Specifies the IP address of the static ARP entry.

Possible values:

Valid IPv4 address

MAC address

Specifies the MAC address that the device assigns to the IP address when answering an ARP request.

Possible values:

Valid MAC address

7.4 Open Shortest Path First

[Routing > OSPF]

Open Shortest Path First (OSPF) version 2 is a routing protocol described in RFC 2328, which is applicable to networks with many routers.

In contrast to the hop count based distance-vector routing protocols such as RIP, OSPF provides a link state algorithm. OSPF bases its link state algorithm on link cost meaning that the criteria for the routing decisions are the path costs instead of hop counts. The path cost is calculated as $(100 \text{ Mbit/s}) / (\text{bandwidth in Mbit/s})$. OSPF also supports Variable Length Subnet Masking (VLSM) or Classless Inter-Domain Routing (CIDR) networks.

OSPF convergence of the entire network is slow. However, after initialization the protocol is quick in reacting to topology changes. The convergence time for OSPF is 5 to 15 seconds, depending on the size of the network.

OSPF supports networks grouped to "Areas" and thus reduces the administrative effort when maintaining the overall network (OSPF domain). The routers participating in the network know and only manage their own "Area" by flooding Link State Advertisements (LSAs) into the area. Using the LSAs, each router builds its own topology database.

- The Area Border Routers (ABR) flood LSAs in an "Area" informing the local networks about destinations in other areas within the OSPF domain. The Designated Routers (DR) send LSAs informing about destinations in other areas.
- With *Hello* packets, neighboring routers periodically identify themselves and signal their availability. If a router misses the *Hello* packets of another router, then after the expiration of the dead-interval timer, the router considers this router as unreachable.

The device lets you use the md5 algorithm for data transmission. If you use the md5 mode, then specify the same values in the devices in the same area. Specify the area relevant values connected to the ABRs and ASBRs.

OSPF divides routers into the following roles:

- Designated Router (DR)
- Backup Designated Router (BDR)
- Area Border Router (ABR)
- Autonomous System Boundary Router (ASBR)

The menu contains the following dialogs:

- [OSPF Global](#)
- [OSPF Areas](#)
- [OSPF Stub Areas](#)
- [OSPF Not So Stubby Areas](#)
- [OSPF Interfaces](#)
- [OSPF Virtual Links](#)
- [OSPF Ranges](#)
- [OSPF Diagnostics](#)

7.4.1 OSPF Global

[Routing > OSPF > Global]

This dialog lets you specify the basic *OSPF* settings.

The menu contains the following dialogs:

- [General]
- [Configuration]
- [Redistribution]

[General]

This tab lets you enable the *OSPF* function in the device and to specify network parameters.

Operation

Operation

Enables/disables the *OSPF* function in the device.

Possible values:

- On**
The *OSPF* function is enabled.
- Off** (default setting)
The *OSPF* function is disabled.

Configuration

Router ID

Specifies the unique identifier for the router in the Autonomous System (AS). It influences the election of the *Designated Router (DR)* and the *Backup Designated Router (BDR)*. Ideally, you use the IP address of a router interface in the device.

Possible values:

- <IP address of an interface> (default setting: 0.0.0.0)

External LSDB limit

Specifies the maximum number of entries, non-default AS-external-LSAs, that the device saves in the link state database. When this limit is reached, the router enters the overflow state.

Possible values:

- 1** (default setting)
The router continues to save entries until the memory is full.
- 0..2147483647** ($2^{31} - 1$)
The device saves up to the specified number of entries.
Specify the same value in the routers on the OSPF backbone and in any regular OSPF area.

External LSAs

Displays the current number of entries, non-default AS-external-LSAs, that the device currently holds in the link state database.

Autocost reference bandwidth

Specifies a reference for router interface bandwidth calculations, in Mbps. You use this value for metric calculations.

Possible values:

1.. 4294967 (default setting: 100)

Paths (max.)

Specifies the maximum number of ECMP routes that the *OSPF* function adds to the routing table when multiple routes exist for a subnet with same path costs, but different next hops.

Possible values:

1.. 4 (default setting: 4)

5.. 16

Available when the *ip4DataCenter* routing profile is currently applied. See the *Routing profile* frame in the *Routing > Global* dialog.

Default metric

Specifies the default metric value for the *OSPF* function.

Possible values:

0 (default setting)

The *OSPF* function automatically assigns a cost of 20 for routes learned from external sources (static or directly connected).

1.. 16777214 ($2^24 - 2$)

Send trap

Activates/deactivates the sending of SNMP traps when the device detects a change in an OSPF parameter.

Possible values:

marked

The sending of SNMP traps is active. The prerequisite is that in the *Diagnostics > Status Configuration > Alarms (Traps)* dialog the *Alarms (Traps)* function is enabled and at least one trap destination is specified.

If the device detects changes in the OSPF parameters, then the device sends an SNMP trap.

unmarked (default setting)

The sending of SNMP traps is inactive.

Shortest path first

Delay time [s]

Specifies the delay time, in seconds, between when the router receives a topology change and when it starts an SPF calculation.

Possible values:

0

The router immediately begins the SPF calculation after receiving the *Topology Change* packet.

1.. 65535 ($2^16 - 1$) (default setting: 5)

Hold time [s]

Specifies the minimum time, in seconds, between consecutive SPF calculations.

Possible values:

0.. 65535 ($2^16 - 1$) (default setting: 10)

The value 0 means that after the router completes an SPF calculation it immediately begins the next consecutive SPF calculation.

Exit overflow interval [s]

Specifies the time in seconds after entering the overflow state that a router attempts to leave the overflow state. When the router leaves the overflow state, the router sends new non-default AS-external-LSAs.

Possible values:

0.. 2147483647 ($2^{31} - 1$) (default setting: 0)

The value 0 means that the router remains in the Overflow-State until restarted.

Information

ASBR status

Displays if the device operates as an *Autonomous System Boundary Router (ASBR)*.

Possible values:

marked

The router is an ASBR.

unmarked

The router functions in a role other than the role of an ASBR.

ABR status

Displays if the device operates as an *Area Border Router (ABR)*.

Possible values:

marked

The router is a ABR.

unmarked

The router functions in a role other than the role of an ABR.

External LSA checksum

Displays the link state checksums of the external LSAs contained in the link state database. This value helps to determine when changes occur in a link state database of the router, and to compare the link state database to other routers.

New LSA originated

Displays the number of new link state advertisements originated on this router. The router increments this number each time it originates a new Link State Advertisement (LSA).

LSAs received

Displays the number of LSAs received that the router determined to be new instances. This number also excludes newer instances of self-originated LSAs.

[Configuration]

This dialog lets you specify the following settings:

- the manner in which the device calculates the path costs
- how the *OSPF* function handles *default routes*
- the type of route the *OSPF* function uses for the path-cost calculation

RFC 1583 compatibility

The Network Working Group is continually developing the *OSPF* function improving and adding parameters. This router provides parameters in accordance with RFC 2328. With parameters in this dialog, you make the router compatible with routers developed under RFC 1583. Activating the compatibility function lets you install this device in a network containing routers developed under RFC 1583.

RFC 1583 compatibility

Enables/disabled the device to be compatible with routers developed under RFC 1583.

To minimize the chance of routing loops, set this function to the same value on the OSPF enabled routers in an OSPF domain.

Possible values:

On (default setting)

Enable the function when routers are present in the domain without software containing the external path preference functionality described in RFC 2328.

Off

Disable the function when every router present in the domain has software containing the external path preference functionality described in RFC 2328.

Preferences

The preferences in this dialog are metrics values which the device uses as a tie breaker between identical routes with different distance types. For example, when a route is inside the local area (intra-area) and the other is outside the local area (inter-area or external). If the metric values are the same for intra, inter and external, then the order of preference is intra, inter then external.

The *OSPF* function considers routes specified with a preference value of 255 as unreachable.

Preference (intra)

Specifies the "administrative distance" between routers within the same area (intra-area OSPF routes).

Possible values:

1..255 (default setting: 110)

Preference (inter)

Specifies the "administrative distance" between routers in different areas (inter-area OSPF routes).

Possible values:

1..255 (default setting: 110)

Preference (external)

Specifies the "administrative distance" between routers external to the areas (external OSPF routes).

Possible values:

1..255 (default setting: 110)

Default route

Advertise

Activates/deactivates OSPF advertisements of *default routes* learned from other protocols.

For example, area border routers of stub areas advertise a *default route* into the stub area through summary link advertisements. When you set up the router as an AS boundary router, it advertises the *default route* in AS external link advertisements.

Possible values:

marked

The router advertises *default routes*.

unmarked (default setting)

The router suppresses advertisements of *default routes*.

Advertise always

Displays if the router constantly advertises `0.0.0.0/0` as the *default route*.

When routers forward an IP packet, the router constantly forwards the packet to the best matching destination address. A *default route* with a destination address of `0.0.0.0` and a mask of `0.0.0.0` is a match for every IP destination address. Matching every IP destination address lets an AS boundary router operate as a gateway for destinations outside of the AS.

Possible values:

`marked`

The router constantly advertises `0.0.0.0/0` as the *default route*.

`unmarked` (default setting)

The device uses the settings specified in the *Advertise* parameter.

Metric

Specifies the metric of the *default route*, which the *OSPF* function advertises when learned from other protocols.

Possible values:

`0`

The device uses the value specified in the *Default metric* field.

`1..16777214 (222 - 2)`

Metric type

Displays the metric type of the *default route* which the *OSPF* function advertises when learned from another protocol.

Possible values:

`external Type1`

Includes both the external path cost from the ABR to the ASBR that originated the route plus the internal path cost to the ABR that advertised the route in the local area.

`external Type2` (default setting)

Includes only the external path cost.

[Redistribution]

A router with a disabled *OSPF* function on a routed interface does not propagate the network of this interface on its other interfaces. Thus, the network cannot be reached. To propagate such networks, enable the *Redistribution* for "connected" networks.

Redistribution is helpful in cases where multiple network administrators manage different departments, or in multi-vendor networks with multiple protocols. OSPF redistribution lets you convert route information such as cost and distance to a destination from other protocols into *OSPF*.

The number of routes that the device learns through the *OSPF* function is limited to the size of the routing table.

Table

For information on how to customize the appearance of the table, see [“Working with tables” on page 16](#).

Source

Displays the source protocol, from which the *OSPF* function redistributes routes. This object also acts as the identifier for the table row.

Activating a table row lets the device redistribute routes from the specific source protocol into OSPF.

Possible values:

[connected](#)

The router is directly connected to the route.

[static](#)

A network administrator has specified the route in the router.

Active

Activates/deactivates route redistribution from the source protocol into OSPF.

Possible values:

[marked](#)

Redistribution of routes learned from the source protocol is active.

[unmarked](#) (default setting)

OSPF route redistribution is inactive.

Metric

Specifies the metric value for routes redistributed from this protocol.

Possible values:

[0](#) (default setting)

The device uses the value specified in the *Default metric* field.

[1..16777214 \(2²⁴ - 2\)](#)

Metric type

Specifies the route metric type which the *OSPF* function redistributes from other source protocols.

Possible values:

[external Type1](#)

This metric type includes both the external path cost from the ABR to the ASBR that originated the route plus the internal path cost to the ABR that advertised the route in the local area.

[external Type2](#) (default setting)

This metric type is only that of the external path cost.

Tag

Specifies a tag for routes redistributed into the *OSPF* function.

When you set a route tag, the *OSPF* function assigns the value to every redistributed route from this source protocol. This function is useful when 2 or more border routers connect an autonomous system to an external network. To help prevent double redistribution, specify the same value in every border router when redistributing the same protocol.

Possible values:

0 . 4294967295 ($2^{32} - 1$) (default setting: 0)

Subnets

Activates/deactivates subnet route redistribution into the *OSPF* function.

The *OSPF* function only redistributes classful routes into the OSPF domain. To redistribute subnet routes into OSPF, activate the subnet parameter.

Possible values:

marked (default setting)

The router redistributes classful and subnet routes into OSPF.

unmarked

The router redistributes only classful routes into OSPF.

7.4.2 OSPF Areas

[Routing > OSPF > Areas]

OSPF supports networks divided into "Areas" and thus reduces the administrative effort when maintaining the network. The routers participating in the network know and only manage their own "Area" by flooding Link State Advertisements (LSAs) into the area. Using the LSAs, each router builds its own topology database.

The device lets you specify up to a total of 64 OSPF Areas.

Table

For information on how to customize the appearance of the table, see ["Working with tables" on page 16](#).

Buttons



Add

Opens the [Create](#) window to add a table row.

- In the [Area ID](#) field you specify the area ID for the new table row.
Possible values:
Octet value displayed like an IPv4 address



Remove

Removes the selected table row.

Area ID

Displays the area ID.

Area type

Specifies the import policy of AS external LSAs for the area which determines the Area Type.

OSPF import policies apply to external routes only. An external route is a route that is outside the OSPF autonomous system.

Possible values:

[area](#) (default setting)

The router imports *Type 5 AS external* LSAs into the area.

[stub area](#)

The router ignores *Type 5 AS external* LSAs.

[nssa](#)

The router translates *Type 7 AS external* LSAs into *Type 5 NSSA summary* LSAs and imports them into the area.

SPF runs

Displays the number of times that the router calculated the intra-area routing table using the link state database of this area. The router uses Dijkstra's algorithm for route calculation.

Area border router

Displays the total number of ABRs reachable within this area. The number of reachable routers is initially 0. The *OSPF* function calculates the number in each SPF Pass.

AS boundary router

Displays the total number of ASBRs reachable within this area. The number of reachable ASBRs is initially 0. The *OSPF* function calculates the number in each SPF Pass.

Area LSAs

Displays the total number of link state advertisements in the link state database of this area, excluding AS External LSAs.

Area LSA checksum

Displays the total number of LS checksums contained in the LS database of this area. This sum excludes *Type 5 external* LSAs. You use the sum to determine if there has been a change in an LS database of a router, and to compare the LS database to other routers.

7.4.3 OSPF Stub Areas

[Routing > OSPF > Stub Areas]

OSPF lets you specify certain areas as stub areas. The *Area Border Router (ABR)* of a stub area enters the information learned from AS external LSAs in its database without flooding the AS external LSAs across the stub area. The ABR instead sends a summary LSA into the stub area advertising a *default route*. The *default route* advertised in the summary LSA pertains only to the particular stub area. When forwarding data to AS external destinations, the routers in a stub area use the default ABR only. Sending a summary LSA containing the *default route* instead of AS external LSAs reduces the link state database size, and therefore the memory requirements for an internal router of a stub area.

The device gives you the following options for adding a Stub Area:

- Convert an Area into a Stub Area. To do this, perform the following step:
 In the [Routing > OSPF > Areas](#) dialog, change the value in the *Area type* column to *Stub Area*.
- Create a Stub Area. To do this, perform the following steps:
 In the [Routing > OSPF > Areas](#) dialog, add a table row.
 Change the value in the *Area type* column to *stub area*.

Table

For information on how to customize the appearance of the table, see [“Working with tables” on page 16](#).

Area ID

Displays the area ID for the stub area.

Default cost

Specifies the external metric value for the metric type.

Possible values:

[0 . 16777215 \(2²⁴ - 1\)](#) (default setting: 1)

The router sets the default value to equal the lower cost within the area for the metric type.

Metric type

Specifies the type of metric used for the *default route* advertised into the area.

The border router of a stub area advertises a *default route* as a network summary LSA.

Possible values:

[OSPF metric](#) (default setting)

The ABR advertises the metric as OSPF internal, which is the cost of an intra-area route to the ABR.

[External type 1](#)

The ABR advertises the metric as [External type 1](#), which is the cost of the OSPF internal metric plus external metric to the ASBR.

[External type 2](#)

The ABR advertises the metric as [External type 2](#), which is the cost of the external metric to the ASBR. You use this value for NSSAs.

Totally stub

Activates/deactivates the import of summary LSAs into stub areas.

Possible values:

`marked`

The router does not import area summaries. The stub area relies entirely on the *default route*. This makes the *default route* a Totally Stub Area.

`unmarked` (default setting)

The router both summarizes and propagates summary LSAs into the stub area.

7.4.4 OSPF Not So Stubby Areas

[Routing > OSPF > NSSA]

NSSAs are similar to the OSPF stub area. However, NSSAs have the additional capability of importing limited AS external routes. The ABR sends external routes out of the NSSA by converting *Type 7 AS external* LSAs into *Type 5 AS external* LSAs. The ASBR in an NSSA originates *Type 7* LSAs. The only difference between the *Type 5* and *Type 7* LSAs is that the router sets the *N* bit for NSSAs. Both NSSA neighbors have the "N" bit set. This forms the OSPF neighbor adjacency.

Beside the internal data stream, NSSAs act like transit areas by transport data coming from external sources to other areas within the OSPF domain.

The device gives you the following options for adding an NSSA:

- Convert an Area into an NSSA. To do this, perform the following step:
 In the [Routing > OSPF > Areas](#) dialog, change the value in the *Area type* column to *nssa*.
- Create an NSSA. To do this, perform the following steps:
 In the [Routing > OSPF > Areas](#) dialog, add a table row.
 Change the value in the *Area type* column to *nssa*.

Table

For information on how to customize the appearance of the table, see [“Working with tables” on page 16](#).

Area ID

Displays the area ID to which the table entries apply.

Redistribute

Activates/deactivates external route redistribution into the NSSA.

Possible values:

marked (default setting)

The NSSA ASBRs suppress external route redistribution into the NSSA. Furthermore, the ASBR stops generating *Type 7 external* LSAs for external routes.

unmarked

The NSSA ASBRs redistribute external routes into the NSSA.

Originate default info

Activates/deactivates generating *Type 7 default* LSAs.

The prerequisite is that the router is an NSSA ABR or ASBR.

Possible values:

marked

The router generates *Type 7 default* LSAs and sends then into the NSSA.

unmarked (default setting)

The router suppresses *Type 7 default* LSAs.

Default metric

Specifies the metric value advertised in the *Type 7 default* LSA.

Possible values:

1..16777214 ($2^24 - 2$) (default setting: 10)

Default metric type

Specifies the metric type advertised in the *Type 7 default* LSA.

Possible values:

`ospfMetric`

The router advertises the metric as OSPF internal, which is the cost of an intra-area route to the ABR.

`comparable`

The router advertises the metric as *external Type 1*, which is the cost of the OSPF internal metric plus external metric to the ASBR.

`nonComparable`

The router advertises the metric as *external Type 2*, which is the cost of the external metric to the ASBR.

Translator role

Specifies the ability of an NSSA border router to perform translation of *Type 7* LSAs into *Type 5* LSAs.

NSSA Area Border Routers receive *Type 5* LSAs containing information about external routes. The NSSA border routers block the *Type 5* LSAs from entering into the NSSA. However, using *Type 7* LSAs the border routers inform each other about external routes. The ABRs then translate the *Type 7* LSAs to *Type 5 external* LSAs and flood the information to the rest of the OSPF network.

Possible values:

`always`

The router translates *Type 7* LSAs to *Type 5* LSAs.

When the router receives a *Type 5* LSAs from another router with a router ID higher than its own, it flushes its *Type 5* LSAs.

`candidate` (default setting)

The router translates *Type 7* LSAs to *Type 5* LSAs.

To help prevent routing loops, the *OSPF* function performs a translator election. When multiple candidates exist, the *OSPF* function elects the router with the higher router ID as the translator.

Translator status

Displays if and how the router is translating *Type 7* LSAs into *Type 5* LSAs.

Possible values:

`enabled`

The *Translator role* of the router is set to `always`.

`elected`

As a candidate, the NSSA Border router is translating *Type 7* LSAs into *Type 5* LSAs.

`disabled`

Another NSSA border router is translating *Type 7* LSAs into *Type 5* LSAs.

Translator stability interval [s]

Specifies the time in seconds after the router loses a translation election that it continues to translate *Type 7* LSAs into *Type 5* LSAs.

Possible values:

0 . 65535 (2¹⁶ - 1) (default setting: 40)

Translator events

Displays the number of translator status changes that have occurred since the last system startup.

Discontinuities in the value of this counter occur while the *OSPF* function is disabled and can occur during re-initialization of the management system.

Totally NSSA

Activates/deactivates importation of summary routes into the NSSA as *Type 3 summary* LSAs.

Possible values:

marked

The router suppresses summary route importation making the area a Totally NSSA.

unmarked (default setting)

The router imports summary routes into the NSSA as *Type 3 summary* LSAs.

7.4.5 OSPF Interfaces

[Routing > OSPF > Interfaces]

This dialog lets you specify, activate, and display OSPF parameters on the router interfaces.

The device lets you activate up to 64 OSPF router interfaces.

The device uses the OSPF routing protocol to exchange reachability information between the routers. The device uses routing information learned from peers to determine the next hop towards the destination. To route the data packets correctly, the router authenticates OSPF protocol exchanges to help prevent malicious or incorrect routing information from getting introduced into the routing table.

The *OSPF* function supports multiple types of authentication. You set up the type of authentication in use on a per interface basis. The cryptographic authentication option *md5*, helps protect the network against passive attacks and helps provide significant protection against active attacks. When using the cryptographic authentication option, each router appends a "message digest" to its transmitted OSPF packets. Receivers then use the shared secret key and received digest to verify that each received OSPF packet is authentic.

Table

For information on how to customize the appearance of the table, see ["Working with tables" on page 16](#).

Port

Displays the interface to which the table row relates.

IP address

Displays the IP address of this OSPF interface.

Active

Activates/deactivates the OSPF administrative status of the interface.

Possible values:

marked

The router advertises the values specified on the interface, and the interface as an OSPF internal route.

unmarked (default setting)

The interface is external to the *OSPF* function.

Area ID

Specifies the area ID of the domain to which the interface connects.

Possible values:

<Area ID>

You specify the area IDs in the *Routing > OSPF > Areas* dialog.

Priority

Specifies the priority of this interface.

In multi-access networks, the router uses the value in the *Designated Router (DR)* election algorithm. When a tie occurs, the routers use their router ID as a tie breaker. The highest router ID wins.

Possible values:

0

The router is unable to become the *Designated Router (DR)* on this particular network.

1.. 255 (default setting: 1)

Transmit delay [s]

Specifies the estimated number of seconds it takes to transmit a *Link State update* packet over this interface.

This setting is useful for low speed links. The timer increases the age of the LS updates to compensate for estimated delays on the interface. Increasing the packet age too much results in a reply that is younger than the original packet.

Possible values:

0.. 3600 (default setting: 1)

Retrans interval [s]

Specifies the time in seconds between *Link State Advertisement* retransmissions for adjacencies belonging to this interface.

You also use this value when retransmitting database description and link state request packets.

Possible values:

0.. 3600 (default setting: 5)

Hello interval [s]

Specifies the time in seconds between *Hello* packet transmissions on the interface.

Set this value the same for the routers attached to a common network. Verify that every router in an area has the same value.

Possible values:

1.. 65535 ($2^1 - 1$) (default setting: 10)

Dead interval [s]

Specifies the time in seconds that the device waits for the *Hello* packets before it declares the neighboring router to be unavailable.

Specify the value to a multiple of the *Hello interval [s]*. Specify the same value for the router interfaces within the same area.

Possible values:

1.. 65535 ($2^1 - 1$) (default setting: 40)

Specify a lower value to get a faster detection of a neighbor that is unavailable.

Note: Lower values are prone to interoperability issues.

Status

Displays the OSPF interface state.

Possible values:

[down](#) (default setting)

The interface is in the initial state and is blocking data packets.

[loopback](#)

The interface is a loopback interface of the device. Although packets are not sent out on the loopback interface, the router LSAs continue to advertise the interface address.

[waiting](#)

Applies only to interfaces connected to broadcast and Non-broadcast Multi-access (NBMA) network types. While in this state, the router attempts to identify the state of the network DR and BDR by sending and receiving *Hello* packets. The wait timer causes the interface to exit the [waiting](#) state and select a DR. The period of this timer is the same as the value in the [Dead interval \[s\]](#) field.

[pointToPoint](#)

Applies only to interfaces connected to point-to-point, point-to-multipoint, and virtual link network types. While in this state the interface sends *Hello* packets every [Hello interval \[s\]](#) and establishes an adjacency with its neighbor.

[designatedRouter](#)

The router is the DR for the multi-access network and establishes adjacencies with the other network routers.

[backupDesignatedRouter](#)

The router is the BDR for the multi-access network and establishes adjacencies with the other network routers.

[otherDesignatedRouter](#)

The router is only a network participant. The router establishes adjacencies only with the DR and BDR and tracks its network neighbors.

Designated router

Displays the IP address of the *Designated Router*.

Possible values:

Valid IPv4 address (default setting: [0.0.0.0](#))

Backup designated router

Displays the IP address of the Backup Designated Router.

Possible values:

Valid IPv4 address (default setting: [0.0.0.0](#))

Events

Displays the number of times this OSPF interface changed its state, or the router detected an error.

Network type

Specifies the OSPF network type of the autonomous system.

Possible values:

[broadcast](#)

Use this value for broadcast networks, such as Ethernet and IEEE 802.5. The *OSPF* function performs a DR and BDR election with which the non-designated routers form an adjacency.

[nbra](#)

Use this value for non-broadcast multi-access networks such as X.25 and similar technologies. The *OSPF* function performs a DR and BDR election to limit the number of adjacencies formed.

[poi nt ToPoi nt](#)

Use this value for networks that link only 2 interfaces.

[poi nt ToMil ti poi nt](#)

Use this value when you collect several point-to-point links into a non-broadcast network. Every router in the network sends *Hello* packets to other routers in the network, but without having a DR and BDR election.

Auth type

Specifies the authentication type for an interface.

If you specify [si mpl e](#) or [MD5](#), then this router requires other routers to pass an authentication process before this router accepts the other routers as neighbors.

If you use authentication to help protect the network, then use the same type and key for every router in your autonomous system.

Possible values:

[none](#) (default setting)

Network authentication is inactive.

[si mpl e](#)

The router uses clear text authentication. In this case, the router sends the passwords as clear text.

[MD5](#)

The router uses the message-digest algorithm MD5 authentication. This type of authentication helps make the network more secure.

Auth key

Specifies the authentication key.

After entering the field displays ***** (asterisk) instead of the authentication key.

Possible values:

Alphanumeric ASCII character string with 16 characters

– with 8 characters if from the *Auth type* drop-down list the [si mpl e](#) item is selected

– with 16 characters if from the *Auth type* drop-down list the [MD5](#) item is selected

If you specify a shorter authentication key, then the device fills in the remaining characters with 0.

Auth key ID

Specifies the [MD5](#) authentication key ID value.

The cryptographic authentication option [MD5](#), helps protect the network against passive attacks and helps provide significant protection against active attacks.

The prerequisite is that for changing the value in the *Auth type* column the value *MD5* is specified.

Possible values:

0 . 255 (default setting: 0)

Cost

Specifies the internal metric.

The *OSPF* function uses link cost as the metric. The *OSPF* function also uses the cost of a link to calculate the SPF routes. The *OSPF* function prefers the route with the smaller value.

The formula to calculate cost is reference bandwidth divided by interface bandwidth. Reference bandwidth is specified in the *Autocost reference bandwidth* field and is set to 100 Mbit/s by default. See the *Routing > OSPF > Global* dialog, *General* tab.

Example:

The interface bandwidth is 10 Mbit/s.

The metric is 100 Mbit/s divided by 10 Mbit/s = 10.

Possible values:

auto (default setting)

The device calculates the metric and automatically adjusts the value when the interface bandwidth changes.

1 . 65535 ($2^1 - 1$)

The *OSPF* function uses the value specified here as metric.

Calculated cost

Displays the metric value which the *OSPF* function currently uses for this interface.

MTU ignore

Activates/deactivates the IP MTU (*Maximum Transmission Unit*) mismatch detection on this OSPF interface.

Possible values:

marked

Disables the IP MTU check and makes adjacencies possible when the MTU value differs on the interfaces.

unmarked (default setting)

The router checks if neighbors are using the same MTU value on the interfaces.

7.4.6 OSPF Virtual Links

[Routing > OSPF > Virtual Links]

The *OSPF* function requires that you link every area to the backbone area. The physical location of routers often prohibits a direct link to the backbone. Virtual links allow you to connect physically separated areas to the backbone through a transit area. You specify both routers on the endpoints of a virtual link as ABRs on a point-to-point link.

Table

For information on how to customize the appearance of the table, see [“Working with tables” on page 16](#).

Buttons



Add

Opens the *Create* window to add a table row.

- From the *Area ID* drop-down list you select the area ID for the new table row.
- In the *Neighbor ID* field you specify the router ID of the virtual neighbor.



Remove

Removes the selected table row.

Area ID

Displays the area ID of the transit area through which the virtual link connects the separated areas.

Neighbor ID

Displays the router ID of the virtual neighbor.

The router learns this value from *Hello* packets received from the virtual neighbor. The value is a static value for virtual adjacencies.

Transmit delay [s]

Specifies the estimated number of seconds it takes to transmit an LS update packet over this interface.

This setting is useful for low speed links. The timer increases the age of the LS updates to compensate for estimated delays on the interface. Increasing the packet age too much results in a reply that is younger than the original packet.

Possible values:

0 . 3600 (default setting: 1)

Retrans interval [s]

Specifies the time in seconds between *Link State Advertisement* retransmissions for adjacencies belonging to this interface.

You also use this value when retransmitting Database Description (DD) and LS Request packets.

Possible values:

0 . 3600 (default setting: 5)

Dead interval [s]

Specifies the time in seconds that the device waits for the *Hello* packets before it declares the neighboring router to be unavailable.

Specify the value to a multiple of the *Hello interval [s]*. Specify the same value for the router interfaces within the same area.

Possible values:

1 . 65535 (2¹⁶ - 1) (default setting: 40)

Specify a lower value to get a faster detection of a neighbor that is unavailable.

Note: Lower values are prone to interoperability issues.

Hello interval [s]

Specifies the time in seconds between *Hello* packet transmissions on the interface.

Set this value the same for the routers attached to a common network.

Possible values:

1 . 65535 (2¹⁶ - 1) (default setting: 10)

Status

Displays the OSPF virtual interface state.

Possible values:

down (default setting)

The interface is in the initial state and is blocking data packets.

pointToPoint

Applies only to interfaces connected to point-to-point, point-to-multipoint, and virtual link network types. While in this state the interface sends *Hello* packets every *Hello interval [s]* and establishes an adjacency with its neighbor.

Events

Displays the number of times this interface changed its state due to a received event.

Auth type

Specifies the authentication type for a virtual link.

If you specify [simple](#) or [MD5](#), then this router requires other routers to pass an authentication process before this router accepts the other routers as neighbors.

If you use authentication to help protect the network, then use the same type and key for every router in your autonomous system.

Possible values:

[none](#) (default setting)

Network authentication is inactive.

[simple](#)

The router uses clear text authentication. In this case, the router sends the passwords as clear text.

[MD5](#)

The router uses the message-digest algorithm MD5 authentication. This type of authentication helps make the network more secure.

Auth key

Specifies the authentication key.

After entering the field displays ***** (asterisk) instead of the authentication key.

Possible values:

Alphanumeric ASCII character string with 16 characters

- with 8 characters if from the [Auth type](#) drop-down list the [simple](#) item is selected

- with 16 characters if from the [Auth type](#) drop-down list the [MD5](#) item is selected

If you specify a shorter authentication key, then the device fills in the remaining characters with 0.

Auth key ID

Specifies the [MD5](#) authentication key ID value.

The cryptographic authentication option [md5](#), helps protect the network against passive attacks and helps provide significant protection against active attacks.

The prerequisite is that for specifying this value, in the [Auth type](#) column the value [MD5](#) is specified.

Possible values:

[0 . 255](#) (default setting: 0)

7.4.7 OSPF Ranges

[Routing > OSPF > Ranges]

In large areas, OSPF messages flooded across the network reduce available bandwidth and increase the size of the routing table. A large routing table increases the amount of CPU processing that the router requires to enter the information into the routing table. A large routing table also reduces available memory. To decrease the number of OSPF messages flooded across the network, the *OSPF* function lets you split a large area into smaller subnets.

To summarize routing information into and out of a subnet, the *Area Border Router (ABR)* specifies the subnet as a single address range. The ABR advertises each address range as a single route to the external area. The IP address that the ABR advertises for the subnet is an address and mask pair. Unadvertised ranges allow you to hide the existence of subnets from other areas.

The router specifies cost of the advertised route as the greater cost in the set component subnets.

Table

For information on how to customize the appearance of the table, see [“Working with tables” on page 16](#).

Buttons



Add

Opens the *Create* window to add a table row.

- From the *Area ID* drop-down list you select the area ID of the address range.
- From the *LSDB type* drop-down list you select the route information aggregated by the address range.

Possible values:

[summaryLink](#)

The area range aggregates *Type 5* route information.

[nssaExternalLink](#)

The area range aggregates *Type 7* route information.

- In the *Network* field you specify the IP address for the area subnet.
- In the *Netmask* field you specify the netmask for the area subnet.



Remove

Removes the selected table row.

Area ID

Displays the area ID of the address range.

LSDB type

Displays the route information aggregated by the address range.

Possible values:

[summaryLink](#)

The area range aggregates *Type 5* route information.

[nssaExternalLink](#)

The area range aggregates *Type 7* route information.

Network

Displays the IP address of the subnet of the range.

Netmask

Displays the netmask of the subnet of the range.

Effect

Specifies the external advertisement of the subnet ranges.

Possible values:

[advertiseMatching](#) (default setting)

The router advertises the range in other areas.

[doNotAdvertiseMatching](#)

The router withholds range advertisement to other external areas.

7.4.8 OSPF Diagnostics

[Routing > OSPF > Diagnostics]

To function properly, the *OSPF* function relies on 2 basic processes.

- forming adjacencies
- after forming adjacencies, the neighboring routers exchange information and update their routing table

The statistics displayed in the tabs help you to analyze the OSPF processes.

The dialog contains the following tabs:

[Statistics]
[Link state database]
[Neighbors]
[Virtual neighbors]
[External link state database]
[Route]

[Statistics]

To accomplish the 2 basic processes, OSPF routers send and receive various messages containing information to form adjacencies, and update routing tables. The counters in the tab indicate the amount of message data packets transmitted on the OSPF interfaces.

- Link State Acknowledgments (LSAcks) provide a response to a *Link State update (LS update)* request as part of the link state exchange process.
- The *Hello* packets allow a router to discover other OSPF routers in the area and to establish adjacencies between the neighboring devices. After establishing adjacencies, the routers advertise their credentials for establishing a role as either a *Designated Router (DR)*, a *Backup Designated Router (BDR)*, or only as a participant in the OSPF network. The routers then use the *Hello* packets to exchange information about the OSPF settings in the Autonomous System (AS).
- Database Description (DD) messages contain descriptions of the AS or area topology. The messages also propagate the contents of the link state database for the AS or area from a router to other routers in the area.
- Link State Requests (LS Request) messages provide a means of requesting updated information about a portion of the Link State Database (LSDB). The message specifies the link or links for which the requesting router requires current information.
- LS Update messages contain updated information about the state of certain links on the LSDB. The router sends the updates as a response to an LS Request message. The router also broadcast or multicast messages periodically. The router uses the message contents to update the information in the LSDBs of routers that receive them.
- LSAs contain the local routing information for the OSPF area. The router sends the LSAs to other routers in an OSPF area and only on interfaces connecting the router to the specific OSPF area.
- *Type 1* LSAs are *Router* LSAs. Each router in an area originates a *Router* LSA. A single *Router* LSA describes the state and cost of every link in the area. The router floods *Type 1* LSAs only across its own area.
- *Type 2* LSAs are *Network* LSAs. The DR generates a *Network* LSA from information received in the *Type 1* LSAs. The DR originates in its own area a *Network* LSA for each broadcast and NBMA network it is connected to. The LSA describes every router attached to the network, including the DR itself. The router floods *Type 2* LSAs only across its own area.

- *Type 3 LSAs are Network Summary LSAs.* An *Area Border Router (ABR)* generates a single network summary LSA from the information contained in the *Type 1* and *Type 2* LSAs received from the DRs. The ABR sends network summary LSAs describing inter-area destinations. The router floods *Type 3* LSAs across every area connected to it, except that this is the area for which it generated the *Type 3* LSA.
- *Type 4 LSAs are Autonomous System Boundary Router (ASBR) summary LSAs.* An ABR generates a single ASBR summary LSA from the information contained in the *Type 1* and *Type 2* LSAs received from the DRs. The ABR sends *Type 4* LSAs to areas different from the area it resides in to describe the ASBRs from which the ABR received *Type 5* LSAs. The router floods *Type 4* LSAs across every area connected to it, except that this is the area for which it generated the *Type 4* LSA.
- *Type 5 LSAs are AS external LSAs.* The AS boundary routers generate the *AS external* LSAs describing destinations external to the AS. The *Type 5* LSAs contain information redistributed into the *OSPF* function from other routing processes. The router floods *Type 5* LSAs to every area except stub and NSSA areas.

Function

LSA retransmitted

Displays the total number of LSAs retransmitted since resetting the counters. When the router sends the same LSA to multiple neighbors, the router increments the count for each neighbor.

Hello received

Displays the total number of OSPFv2 *Hello* packets received since resetting the counters.

Hello transmitted

Displays the total number of OSPFv2 *Hello* packets transmitted since resetting the counters.

DB descriptions received

Displays the total number of OSPFv2 Database Description packets received since resetting the counters.

DB descriptions transmitted

Displays the total number of OSPFv2 Database Description packets transmitted since resetting the counters.

LS requests received

Displays the total number of OSPFv2 Link State Request packets received since resetting the counters.

LS requests transmitted

Displays the total number of OSPFv2 Link State Request packets transmitted since resetting the counters.

LS updates received

Displays the total number of OSPFv2 LS Update packets received since resetting the counters.

LS updates transmitted

Displays the total number of OSPFv2 LS Update packets transmitted since resetting the counters.

LS ACK updates received

Displays the total number of OSPFv2 LS Acknowledgement packets received since resetting the counters.

LS ACK updates transmitted

Displays the total number of OSPFv2 LS Acknowledgement packets transmitted since resetting the counters.

Max. rate of LSU received in any 5sec

Displays the maximum rate of OSPFv2 LS Update packets received over any 5-second interval since resetting the counters. The field displays the rate in packets per second. For example, the number of packets received during the 5-second interval, divided by 5.

Max. rate of LSU transmitted in any 5sec

Displays the maximum rate of OSPFv2 LS Update packets transmitted over any 5-second interval since resetting the counters. The field displays the rate in packets per second. For example, the number of packets transmitted during the 5-second interval, divided by 5.

Type-1 (Router) LSAs received

Displays the number of *Type 1 router* LSAs received since resetting the counters.

Type-2 (Network) LSAs received

Displays the number of *Type 2 network* LSAs received since resetting the counters.

Type-3 (Summary) LSAs received

Displays the number of *Type 3 network summary* LSAs received since resetting the counters.

Type-4 (ASBR) LSAs received

Displays the number of *Type 4 ASBR summary* LSAs received since resetting the counters.

Type-5 (External) LSAs received

Displays the number of *Type 5 external* LSAs received since resetting the counters.

[Link state database]

A router maintains a separate link state database for every area to which it belongs.

The router adds LSAs to the database in the following cases:

- When the router receives an LSA, for example during the flooding process.
- When the router originates the LSA.

When a router deletes an LSA from the database, it also removes the LSA from the link state retransmission lists of the other routers in the network. A router deletes an LSA from its database in the following cases:

- A newer instance overwrites the LSA during the flooding process.
- The router originates a newer instance of a self-originated LSA.
- The LSA ages out and the router flushes the LSA from the routing domain.

Table

For information on how to customize the appearance of the table, see [“Working with tables” on page 16](#).

Area ID

Displays the area ID from which router received the LSA.

Type

Displays the type of the LSAs received.

Each LSA type has a separate advertisement format.

Possible values:

[routerLink](#)

The router received the information from another router in the same area. Routers announce their existence and list the links to other routers within the same area using a *Type 1* LSA. The link state ID is the originating router ID.

[networkLink](#)

The router received the information from a DR on a broadcast segment using a *Type 2* LSA. The DR compiles the information received in *Type 1* LSAs and lists the routers linked together by the segment. The link state ID is the IP interface address of the DR.

[summaryLink](#)

The router received the information from an ABR using a *Type 3* LSA describing routes to networks. ABRs compile information learned from *Type 1* and *Type 2* LSAs received from the attached areas before sending the routing information to the other areas. The link state ID is the destination network number which is the results of the summarization process.

[asSummaryLink](#)

The router received the information from an ABR using a *Type 4* LSA describing routes to ASBRs. ABRs compile information learned from *Type 1* and *Type 2* LSAs received from the attached areas before sending the routing information to the other areas. The link state ID is the destination network number.

[asExternalLink](#)

The router received the information from an ASBR using a *Type 5* LSA describing routes to another AS. The link state ID is the router id of the ASBR.

[nssaExternalLink](#)

The router received the information from a router in a NSSA using a *Type 7* LSA.

LSID

Displays the Link State ID (LSID) value received in the LSA.

The LSID is a field located in the LSA header. The field contains either a router ID or an IP address according to the LSA type.

Possible values:

<Router ID>

Valid IPv4 address

Router ID

Displays the router ID uniquely identifying the originating router.

Sequence

Displays the value of the sequence field in an LSA.

The router examines the contents or the LS checksum field whenever the LS sequence number field indicates that 2 instances of an LSA are the same. When there is a difference, the router considers the instance with the larger LS checksum to be most recent.

Age

Displays the age of the link state advertisement in seconds.

When the router generates the LSA, the router sets the LS age to the value 0. As the routers transmit the LSA across the network, they increment the value by the value specified in the *Transmit delay [s]* column.

If a router receives 2 LSAs for the same segment having identical LS sequence numbers and LS checksums, then the router examines the age of the LSAs.

- The router immediately accepts LSA with MaxAge.
- Otherwise, the router accepts the LSA with the smaller age.

Checksum

Displays the contents of the checksum.

The field is a checksum of the complete contents of the LSA, except for the age field. The age field value of the advertisement increases with each router that transmits the message. Excluding the age field lets the router send the message without updating the checksum field.

[Neighbors]

The *Hello* packet is responsible for neighbor acquisition, maintenance, and bidirectional communication between neighbors.

During the acquisition process, the routers on a segment compare their settings for compatibility. If the routers are compatible, then the routers form adjacencies. The routers discover their master or slave status using information provided in the *Hello* packets.

After the routers discover their roles, they exchange routing information to synchronize their routing databases. When the routers finish updating their databases, the neighbors are fully adjacent and the LSA lists the adjacency.

Table

For information on how to customize the appearance of the table, see [“Working with tables” on page 16](#).

Neighbor ID

Displays the router ID of the neighboring router.

The router learns this value from *Hello* packets received from the neighbor. The value is a static value for virtual adjacencies.

IP address

Displays the IP address of the neighboring router interface attached to the port.

When sending unicast protocol packets on this adjacency, the router uses the value as the destination IP address. When the neighboring router is the DR, the router is also used in router LSAs as the link ID for the attached network. The router learns the neighbor IP address when it receives *Hello* packets from the neighbor. For virtual links, the router learns the neighbor IP address while building the routing table.

Interface

Displays the interface to which the table row relates.

Status

Displays the state of the relationship with the neighbor listed in this instance.

An event invokes each state change, such as a received *Hello* packet. This event produces different effects, depending on the current state of the neighbor. Also, depending on the state of neighbor change, the routers initiate a DR election.

Possible values:

down (default setting)

The initial state of a neighbor conversation or a router terminated the conversation due to expiration of the *Dead interval [s]* timer.

attempt

The state is only valid for neighbors attached to NBMA networks. The information from the neighbor remains unresolved. The router actively attempts to contact the neighbor by sending the neighbor *Hello* packets in the interval specified in the *Hello interval [s]* column.

init

The router has recently received a *Hello* packet from the neighbor. However, the router has only established uni-directional communication with the neighbor. For example, the router ID of this router is missing from the *Hello* packet of the neighbor. When sending *Hello* packets, the associated interface lists neighbors in this state or higher.

twoWay

Communication between the 2 routers is bidirectional. The router verifies the operation by examining the contents of the *Hello* packet. The routers elect a DR and BDR from the set of neighbors while in or after the bidirectional state.

exchangeStart

The first step in setting up an adjacency between the 2 neighboring routers. The goal of this step is to decide which router is the master and to decide upon the initial *Sequence* number.

exchange

The router is announcing its entire link state database by sending DD packets (Database Description) to the neighbor. The router explicitly acknowledges each DD packet. Each packet has a sequence number. The adjacencies only allow one DD packet to be outstanding at any time. In this state, the router sends LS Request packets asking for up-to-date database information. The adjacencies are fully capable of transmitting OSPF routing protocol packets.

loading

The router sends LS Request packets to the neighbor inquiring about the outstanding database updates sent in the exchange state.

full

The neighboring routers are fully adjacent. The adjacencies now appear in router LSAs and network LSAs.

Dead time

Displays the amount of time remaining before the router declares the neighbor to be unavailable. The timer initiates the count down after the router receives a *Hello* packet.

[Virtual neighbors]

The *OSPF* function requires a continuous connection of the Autonomous System backbone area. The *OSPF* function also requires that every area has a connection to the backbone area. The physical location of routers often prohibits an area from directly connecting to the backbone area. Virtual links allow you to connect physically separated areas to the backbone area.

The ABRs of the backbone area and the physically separated area form a point-to-point link through a transit area. When the ABRs establish an adjacency, the backbone router LSAs include the link and OSPF packets flow over the virtual link. Furthermore, the routing database of each endpoint router includes the link state information of the other endpoint router.

Note: The *OSPF* function lets you specify virtual links through every type of area except for stub areas.

Table

For information on how to customize the appearance of the table, see [“Working with tables” on page 16](#).

Area ID

Displays the transit area ID of the virtual link.

Router ID

Displays the router ID of the other virtual endpoint ABR.

After virtual adjacencies form, the virtual data link carries OSPF packets such as *Hello* packets and LS update packets containing database information. The prerequisite is that the LSAs of the neighbor router contain the router ID of the local router.

IP address

Displays the IP address of the virtual neighbor.

The router uses the IP address to send OSPF packets across the transit network to the virtual neighbor.

Options

Displays the information contained in the *Options* field of the LSA. This value indicates the capabilities of virtual neighbor.

The *Options* field used in the *Hello* packets lets a router identify and share its optional capabilities with other routers. This mechanism lets you mix routers of different capabilities within a routing domain.

The router supports 4 options by setting the following bits in the *Options* field either high or low depending on the capabilities of the router. The field displays the value by adding the following option bits together. You read the fields from least significant bit to most significant bit.

- The routers advertise the ability to process TOS 0 in AS external routes when it sets the E bit high. The E bit is the second bit in the *Options* field and represents the value 2^1 or 2.
- The routers advertise the ability to process multicast routes when it sets the MC bit high. The MC bit is the third bit in the *Options* field and represents the value 2^2 or 4.
- The routers advertise the ability to process AS external routes in an NSSA summary with *Type 7* LSAs when it sets the N/P bit high. The N/P bit is the fourth bit in the *Options* field and represents the value 2^3 or 8.
- The routers advertise the ability to process demand circuits when it sets the DC bit high. The DC bit is the sixth bit in the *Options* field and represents the value 2^5 or 32.

In a special case, the router sets the E bit low.

- The routers advertise the ability to process TOS metrics other than TOS 0 when it sets the E bit low. The E bit is the second bit in the *Options* field and when set low, the bit represents the value 0.

Possible values:

[2](#), [6](#), [10](#), [14](#), [34](#), [38](#), [42](#), [46](#)

The values indicate that the virtual neighbor supports Type of Service metric (TOS) 0 in AS external LSAs.

[0](#), [4](#), [8](#), [12](#), [32](#), [36](#), [40](#), [44](#)

The values indicate that the virtual neighbor supports TOS metrics other than TOS 0.

[4](#), [6](#), [12](#), [14](#), [36](#), [38](#), [44](#), [46](#)

The values indicate that the virtual neighbor supports multicast routing.

[8](#), [10](#), [12](#), [14](#), [40](#), [42](#), [44](#), [46](#)

The values indicate that the virtual neighbor supports *Type 7* LSAs.

[32](#), [34](#), [36](#), [38](#), [40](#), [42](#), [44](#), [46](#)

The values indicate that the virtual neighbor supports demand circuits.

Status

Displays the state of the relationship with the neighbor listed in this instance.

An event invokes each state change, such as a received *Hello* packet. This event produces different effects, depending on the current state of the neighbor. Also, depending on the state of neighbor change, the routers initiate a DR election.

Possible values:

down (default setting)

The initial state of a neighbor conversation or a router terminated the conversation due to expiration of the *Dead interval [s]* timer.

attempt

The state is only valid for neighbors attached to NBMA networks. Information from the neighbor remains unresolved. The router actively attempts to contact the neighbor by sending the neighbor *Hello* packets in the interval specified in the *Hello interval [s]* column.

init

The router has recently received a *Hello* packet from the neighbor. However, the router has only established uni-directional communication with the neighbor. For example, the router ID of this router is missing from the *Hello* packet of the neighbor. When sending *Hello* packets, the associated interface lists neighbors in this state or higher.

twoWay

Communication between the 2 routers is bidirectional. The router verifies the operation by examining the contents of the *Hello* packet. The routers elect a DR and BDR from the set of neighbors while in or after the bidirectional state.

exchangeStart

The first step in setting up an adjacency between the 2 neighboring routers. The goal of this step is to decide which router is the master and to decide upon the initial *Sequence* number.

exchange

The router is announcing its entire link state database by sending DD packets (Database Description) to the neighbor. The router explicitly acknowledges each DD packet. Each packet has a sequence number. The adjacencies only allow one DD packet to be outstanding at any time. In this state, the router sends LS Request packets asking for up-to-date database information. The adjacencies are fully capable of transmitting OSPF routing protocol packets.

loading

The router sends LS Request packets to the neighbor inquiring about the outstanding database updates sent in the exchange state.

full

The neighboring routers are fully adjacent. The adjacencies now appear in router LSAs and network LSAs.

Events

Displays the number of times this interface changed its state due to a received event. For example, if the device has received a *Hello* packet or the device has established bidirectional communication.

Length of retransmission queue

Displays the length of the retransmission list.

To flood LSAs out of an interface to the neighbor, the router places the LSAs on the link state retransmission list of the adjacency. To validate LSA flooding, the router retransmits the LSAs until the neighbor acknowledges the LSA reception. You specify the length of time between retransmissions in the *Routing > OSPF > Interfaces* dialog in the *Retrans interval [s]* column.

Suppressed Hellos

Displays if the router is suppressing *Hello* packets to the neighbor.

Suppressing *Hello* packet transmission to the neighbor lets demand circuits close, on point-to-point links, during periods of inactivity. In NBMA networks, the periodic transmission of LSAs causes the circuit to remain open.

Possible values:

[marked](#)

The router suppresses *Hello* packets.

[unmarked](#)

The router transmits *Hello* packets.

[External link state database]

The table displays the contents of the external link state database, with an entry for each unique link state ID. External links allow the area to connect to destinations outside of the autonomous system. Routers pass information about the external links throughout the network as *Link State updates*.

Table

For information on how to customize the appearance of the table, see [“Working with tables” on page 16](#).

Type

Displays the type of the link state advertisement. When the router detects an external link state advertisement, the router enters the information in the table.

Possible values:

[asExternal Link](#)

LSID

Displays the Link State ID is an LS type-specific field containing either a router ID or an IP address. The value identifies the routing domain described in the advertisement.

Router ID

Displays the router ID uniquely identifying the originating router.

Sequence

Displays the value of the sequence field in an LSA.

The router examines the contents or the LS checksum field whenever the LS sequence number field indicates that 2 instances of an LSA are the same. When there is a difference, the router considers the instance with the larger LS checksum to be most recent.

Age

Displays the age of the link state advertisement in seconds.

When the router generates the LSA, the router sets the LS age to the value [0](#). As the routers transmit the LSA across the network, they increment the value by the value specified in the [Transmit delay \[s\]](#) column.

If a router receives 2 LSAs for the same segment having identical LS sequence numbers and LS checksums, then the router examines the age of the LSAs.

- The router immediately discards LSA with MaxAge.
- Otherwise, the router discards the LSA with the smaller age.

Checksum

Displays the contents of the checksum.

The field is a checksum of the complete contents of the LSA, except for the age field. The age field of the advertisement increases as the router transmits the message across the network. Excluding the age field lets the router send the message without updating the checksum field.

[Route]

The dialog displays the OSPF route information learned from the Link State Advertisements (LSA).

Table

For information on how to customize the appearance of the table, see [“Working with tables” on page 16](#).

IP address

Displays the IP address of the network or subnet for the route.

Netmask

Displays the netmask for the network or subnet.

Metric

Displays the route cost, calculated in the SPF algorithm, to reach the network.

Type

Displays the type of route that was learned from OSPF.

Possible values:

[i n t r a](#)

Entry for routes from OSPF within an area.

[i n t e r](#)

Entry for routes from OSPF between areas.

[e x t - t y p e 1](#)

These routes were imported from an Autonomous System Boundary Router (ASBR) into the OSPF area. These routes use the costs relating to the connection between the ASBR and the route costs includes this device.

[e x t - t y p e 2](#)

These routes were imported from an Autonomous System Boundary Router (ASBR) into the OSPF area. These routes do not use the costs relating to the connection between the ASBR and the route costs includes this device.

[nssa-type1](#)

These routes were imported from an Autonomous System Boundary Router (ASBR) into the Not-So-Stub Area. These routes use the costs relating to the connection between the ASBR and the route costs includes this device.

[nssa-type2](#)

These routes were imported from an Autonomous System Boundary Router (ASBR) into the Not-So-Stub Area. These routes do not use the costs relating to the connection between the ASBR and the route costs includes this device.

7.5 Routing Table

[Routing > Routing Table]

This dialog displays the routing table with the routes set up in the device. Using the routing table, the device learns the router interface through which it transfers IP packets that are addressed to recipients in a different network.

Configuration

Preference

Specifies the preference number that the device assigns by default to the newly set-up static routes.

Possible values:

1.. 255 (default setting: 1)

Routes with a value of 255 will be ignored by the device in the routing decision.

Table

For information on how to customize the appearance of the table, see [“Working with tables” on page 16](#).

Buttons



Add

Opens the [Create](#) window to add a static route.

- In the [Network address](#) field, you specify the address of the destination network.
 Possible values:
 Valid IPv4 address
 If you specify a *default route* (0.0.0.0), then you specify a *default gateway* in the [Next hop IP address](#) field. This setting takes precedence over the setting in the following dialog:
 – [Basic Settings > Network > IPv4](#) dialog, [Gateway address](#) field
- In the [Netmask](#) field, you specify the netmask that identifies the network prefix in the address of the destination network.
 Possible values:
 Valid IPv4 netmask

- In the *Next hop IP address* field, you specify the IP address of the next router on the path to the destination network.
Possible values:
Valid IPv4 address
To make a *reject* type route, specify the value `0.0.0.0` in this field. With this route, the device discards IP packets addressed to the destination network and informs the sender.
- In the *Preference* field, you specify the preference number that the device uses to decide which of several existing routes to the destination network it will use.
Possible values:
`1..255`
In routing decisions, the device gives preference to the route with the numerically lowest value. The default setting is the value specified in the *Configuration* frame, field *Preference*.
- From the *Track name* drop-down list, you select the tracking object with which the device links the route.
Possible values:
-
No tracking object selected.
Name of the tracking object, made up of *Type* and *Track ID*.



Remove

Removes the selected table row.

Port

Displays the router interface through which the device currently sends IP packets addressed to the destination network.

Possible values:

`<Router interface>`

The device uses this router interface to transfer IP packets addressed to the destination network.

`no port`

The static route is currently not assigned to a router interface.

Network address

Displays the address of the destination network.

Netmask

Displays the netmask.

Next hop IP address

Displays the IP address of the next router on the path to the destination network.

Type

Displays the type of the route.

Possible values:

`local`

The router interface is directly connected to the destination network.

[remote](#)

The router interface is connected to the destination network through a router (*Next hop IP address*).

[reject](#)

The device discards IP packets addressed to the destination network and informs the sender.

[other](#)

The route is inactive. See the [Active](#) checkbox.

Protocol


Displays the origin of this route.

Possible values:

[local](#)

The device added this route when setting up the router interface. See the [Routing > Interfaces > Configuration](#) dialog.

[netmngt](#)

A user added this static route with the  button.

Note: You can make static routes with the same destination and preference, but with different next hops. The device uses Equal Cost Multi Path (ECMP) forwarding mechanism to help ensure load sharing and redundancy over the network. Depending on the selected routing profile in the [Routing > Global](#) dialog, ECMP can use up to 4 routes. If you select the [IPv4 Data Center](#) routing profile, then ECMP can use up to 16 routes.

[ospf](#)

The *OSPF* function added this route. See the [Routing > OSPF](#) dialog.

Preference

Specifies the "administrative distance" of the route.

The device uses this value instead of the metric, when the metric of the routes is incomparable.

Possible values:

[0](#)


Reserved for routes that the device added when setting up the router interfaces. These routes have the value [local](#) in the *Protocol* column.

[1..254](#)

In routing decisions, the device gives preference to the route with the numerically lowest value.

[255](#)

In routing decisions, the device ignores the route.

The *Administrative Distance* can be set for static routes added using the  button.

Metric

Displays the metric of the route.

The device sends the data packets using the route with the numerically lowest value.

Last update [s]

Displays the time in seconds, since the current settings of the route were entered in the routing table.

Track name

Specifies the tracking object with which the device links the route.

The device automatically activates or deactivates static routes – depending on the link status of an interface or the reachability of a remote router or end device.

You set up tracking objects in the [Advanced > Tracking > Configuration](#) dialog.

Possible values:

Name of the tracking object, made up of [Type](#) and [Track ID](#).

–

No tracking object selected.

This function is used only for static routes. (Column [Protocol](#) = [net mgnt](#))

Active

Displays if the route is active or inactive.

Possible values:

[marked](#)

The route is active; the device uses the route.

[unmarked](#)

The route is inactive.

7.6 L3 Relay

[Routing > L3 Relay]

In a Layer 3 subnet, clients send Bootstrap Protocol (BOOTP)/Dynamic Host Configuration Protocol (DHCP) broadcast messages to the DHCP server to request information for the network settings, such as IP addresses. Routers help provide a boundary for broadcast messages so that BOOTP/DHCP requests are confined within the local subnet. The *L3 Relay* function acts as a proxy for clients that require information from a BOOTP/DHCP server located in a different Layer 3 network segment.

When you set up the client device to retrieve its network settings from a Dynamic Host Configuration Protocol (DHCP) server located in a different subnet, the *L3 Relay* function lets the network device relay requests to a BOOTP/DHCP server located in a different network.

Using *IP helper addresses* and *UDP helper ports*, the L3 Relay function relays Dynamic Host Configuration Protocol (DHCP) packets between the clients and the servers. The *IP helper address* is the IP address of the DHCP server.

Clients use the *UDP helper port* to send broadcast requests to DHCP servers on UDP port *67*.

Operation

Operation

Enables/disables the *L3 Relay* function.

Possible values:

On

The *L3 Relay* function is globally enabled.

Off (default setting)

The *L3 Relay* function is globally disabled.

Configuration

Circuit ID

Activates/deactivates the BOOTP/DHCP circuit ID option mode.

The network device sends circuit ID suboption information, which identifies the local agent, to the DHCP server. When the DHCP server responds, the network device then recognizes its role as the L3 Relay agent. With the help of the suboption information, the network device helps ensure that the responses are directed back to the appropriate agent.

Possible values:

marked

The device adds the circuit ID of the DHCP L3 Relay agent to the suboptions for client requests.

unmarked (default setting)

The device does not add the circuit ID of its L3 Relay agent to the suboptions for client requests.

BOOTP/DHCP wait time (min.)

Specifies the minimum amount of time in seconds that the device waits before relaying the BOOTP/DHCP request.

The end devices send broadcast requests on the local network. This setting lets a local BOOTP/DHCP server respond to the client request before the router relays the client request.

Possible values:

0 . 100 (default setting: 0)

If there is no local BOOTP/DHCP server in the network, then set the value to 0.

BOOTP/DHCP hops (max.)

Specifies the maximum number of cascaded relay agent devices allowed to relay the BOOTP/DHCP request. Each relay agent device that relays a message, increments the hop count value by 1.

If the hop count of a received BOOTP/DHCP packet exceeds the maximum number of hops specified here, then the device drops the BOOTP/DHCP request. This keeps the message from repeating indefinitely within the network.

Possible values:

1 . 16 (default setting: 4)

Information

The following fields display the values since the last device restart. The device resets the values to 0 after a restart.

DHCP client messages received

Displays the number of DHCP requests received from the clients.

DHCP client messages relayed

Displays the number of DHCP requests relayed to the servers specified in the table.

DHCP server messages received

Displays the number of DHCP offers received from the servers specified in the table.

DHCP server messages relayed

Displays the number of DHCP offers relayed to the clients from the servers specified in the table.

UDP messages received

Displays the number of UDP requests received from the clients.

UDP messages relayed

Displays the number of UDP requests relayed to the servers specified in the table.

Packets with expired TTL

Displays the number of UDP packets received with an expired TTL value.

Discarded packets

Displays the number of UDP packets that the device discarded.

Table

For information on how to customize the appearance of the table, see [“Working with tables” on page 16](#).

Buttons



Add

Opens the [Create](#) window to add a table row.

- In the [Port](#) field, you specify the port-based router interface.

Note: The device does not support the [L3 Relay](#) function on VLAN-based router interfaces.

Possible values:

[All](#) (default setting)

The device processes the data packets received on all the interfaces. Relay entries with this value specify a global setting.

[<available interfaces>](#)

The device processes the data packets received on the specified interfaces.

Interface configurations take priority over global configurations. If the destination UDP port for a data packet matches an entry on an ingress interface, then the device processes the data packet according to the interface configuration. If none of the interface entries match the data packet, then the device processes the data packet according to the global configuration.

- In the [UDP port](#) field, you specify the [UDP helper port](#) values for data packets received on this interface. When active, the device relays data packets received with this destination [UDP port](#) value to the IP address specified in the [IP address](#) field.

Possible values:

[dhcp](#)

Equal to UDP port [67](#).

The device relays Dynamic Host Configuration Protocol (DHCP) requests for IP address assignment and networking parameters.

- In the [IP address](#) field, you specify the [IP helper address](#) for the data packets received on the interface.

Possible values:

Valid IP address

The IP address [0.0.0.0](#) specifies the entry as a discard entry. The device drops data packets that match a discard entry. You specify discard entries only on the interfaces.

Prerequisites:

- To enter the IP address [0.0.0.0](#), verify that in the [Port](#) field, a value other than [All](#) is specified.
- To enter an IP address other than [0.0.0.0](#), verify that in the [Port](#) field, the value [All](#) is specified.



Removes the selected table row.



Resets the table statistics.

Port

Displays the port-based router interface to which the table row relates.

Note: The device does not support the [L3 Relay](#) function on VLAN-based router interfaces.

UDP port

Displays the destination UDP port for client messages received on the interface. The device relays DHCP requests that match the UDP port criteria to the specified *IP helper address*.

IP address

Displays the *IP helper address* for the data packets received on the interface.

Status

Displays if the *IP helper address* and the *UDP port* items added to the respective port are active.

7.7 Loopback Interface

[Routing > Loopback Interface]

A loopback interface is a virtual network interface without reference to a physical port. Loopback interfaces are constantly available while the device is in operation.

The device lets you set up router interfaces on the basis of loopback interfaces. Using such a router interface, the device is constantly available, even during periods of inactivity of individual router interfaces.

Up to 8 loopback interfaces can be set up in the device.

Table

For information on how to customize the appearance of the table, see [“Working with tables” on page 16](#).

Buttons



Add

Opens the [Create](#) window to add a loopback interface.

- In the [Index](#) field, you specify the number that uniquely identifies the loopback interface.
Possible values:
1..8



Remove

Removes the selected table row.

Index

Displays the number that uniquely identifies the loopback interface. You specify the index number when you add a table row.

Port

Displays the name of the loopback interface.

IP address

Specifies the IP address for the loopback interface.

Possible values:

Valid IPv4 address (default setting: 0.0.0.0)

Subnet mask

Specifies the netmask for the loopback interface.

Possible values:

Valid IPv4 netmask (default setting: 0.0.0.0)

Example: 255.255.255.255

Active

Displays if the loopback interface is active or inactive.

Possible values:

`marked` (default setting)

The loopback interface is active.

When sending SNMP traps, the device uses the IP address of the first loopback interface as the sender.

`unmarked`

The loopback interface is inactive.

7.8 L3-Redundancy

[Routing > L3-Redundancy]

The menu contains the following dialogs:

[VRRP](#)

7.8.1 VRRP

[Routing > L3-Redundancy > VRRP]

The Virtual Router Redundancy Protocol (VRRP) is a procedure that lets the system react to the failure of a router.

You use VRRP in networks with end devices that support one entry for the *default gateway*. If the *default gateway* fails, then VRRP helps ensure that the end devices find a redundant gateway.

Note: For further information on the [VRRP](#) function, see the “Configuration” user manual.

The menu contains the following dialogs:

[VRRP Configuration](#)

[VRRP Statistics](#)

[VRRP Tracking](#)

7.8.1.1 VRRP Configuration

[Routing > L3-Redundancy > VRRP > Configuration]

This dialog lets you specify the following settings:

- up to 8 virtual routers per router interface
- up to 2 addresses per virtual router

Operation

Operation

Enables/disables the [VRRP](#) redundancy in the device.

Possible values:

[On](#)

The [VRRP](#) function is enabled.

[Off](#) (default setting)

The [VRRP](#) function is disabled.

Configuration

Send trap (VRRP master)

Activates/deactivates the sending of SNMP traps when the device is the VRRP master.

Possible values:

[marked](#)

The sending of SNMP traps is active. The prerequisite is that in the [Diagnostics > Status Configuration > Alarms \(Traps\)](#) dialog the [Alarms \(Traps\)](#) function is enabled and at least one trap destination is specified.

If the device is the VRRP master, then the device sends an SNMP trap.

[unmarked](#) (default setting)

The sending of SNMP traps is inactive.

Send trap (VRRP authentication failure)

Activates/deactivates the sending of SNMP traps when the device receives a VRRP packet including authentication information.

Note: The device supports only VRRP packets without authentication information. To operate the device in conjunction with other devices that support VRRP authentication, verify that on those devices the VRRP authentication is not applied.

Possible values:

marked

The sending of SNMP traps is active. The prerequisite is that in the [Diagnostics > Status Configuration > Alarms \(Traps\)](#) dialog the [Alarms \(Traps\)](#) function is enabled and at least one trap destination is specified.

If the device receives a VRRP packet including authentication information, then the device sends an SNMP trap.

unmarked (default setting)

The sending of SNMP traps is inactive.

Information

Version

Specifies the VRRP version.

Table

For information on how to customize the appearance of the table, see [“Working with tables” on page 16](#).

Buttons



Add

Opens the [Create](#) window to add a table row.

- From the [Port](#) drop-down list, you select the port number.
- In the [VRID](#) field, you specify the Virtual Router Identifier (VRID).



Remove

Removes the selected table row.



Wizard

Opens the [Wizard](#) window that helps you associate the ports with the address of one or more desired senders. See [“\[Wizard: VRRP configuration\]” on page 367](#).

Port

Displays the port number to which the table row relates.

VRID

Displays the Virtual Router Identifier.

Active

Activates/deactivates the VRRP instance specified in this table row.

Possible values:

`marked`

The *VRRP* instance is active.

`unmarked` (default setting)

The *VRRP* instance is inactive.

Oper status

Displays the table row status. The operational state of the related virtual router controls the row status of a currently active table row.

Possible values:

`active`

The instance is available for use.

`notInService`

The instance exists in the device, but necessary information is missing and it is unavailable for use.

`notReady`

The instance exists in the device, but necessary information is missing and it is unavailable for use.

State

Displays the VRRP state.

Possible values:

`initialize`

VRRP is in the initialization phase, the function is inactive, or the master router is still unnamed.

`backup`

The router sees the possibility of becoming the master router.

`master`

The router is the master router.

Base priority

Specifies the priority of the virtual router. If the value differs from the value in the *Priority* field, then the tracked object is unavailable or the virtual router is the IP address owner.

Possible values:

`1..254` (default setting: `100`)

The higher the number, the higher the priority. When you set up multiple VRRP routers in a single instance, distribute the priority values uniformly on the routers. For example, assign the priority value of `50` to the primary router, the value of `100` to the next router. Repeat the steps with the value `150`, and so on. This distribution simplifies adding another router later with a priority between the existing values, for example with the value `75`.

Priority

Displays the *VRRP* priority value. You specify the priority in the *Routing > OSPF > Interfaces* dialog. The router with the higher priority value takes over the master router role. If the virtual router IP address is the same as the IP address of a router interface, then the router is the IP address owner. If an IP address owner exists, then the *VRRP* function lets the device assign the IP address owner the priority value *255* and declares the router as the master router.

Possible values:

0

The higher the number, the higher the priority. When you disable or remove a *VRRP* router, which is in the master role, you force the instance to send an advertisement with priority value *0*. This lets the other backup routers know that the master does not participate. Sending a priority value *0* forces a new election.

1..255

The value *255* means that the virtual router is the IP address owner.

Virtual IP address

Displays the virtual IP address in the subnet of the primary IP address on the interface. If no match is found, then the device returns an unspecified virtual address. If no virtual address is set up, then the device returns *0.0.0.0*.

Possible values:

Valid IPv4 address

Preempt mode

Activates/deactivates the preempt mode. This setting specifies if this router, as a backup router, takes over the master router role when the master router has a lower VRRP priority.

Possible values:

marked (default setting)

The *Preempt mode* is active. The router takes the master router role from a router with a lower VRRP priority without an election.

unmarked

The *Preempt mode* is inactive. The router assumes the role of a backup router and listens for master router advertisements. After the *Master Down* interval expires, and no advertisements received from the master router, the router participates in the master router election process.

Proxy ARP

Activates/deactivates the *Proxy ARP* function on the virtual router interface. The function lets you reach devices in other networks as if these devices were located in the local network. The *Proxy ARP* function is required when the device uses the VRRP instance with *1:1 NAT* rules. The prerequisite is that in the *Routing > Interfaces > Configuration* dialog for the relevant interface in use by the VRRP instance, the *Proxy ARP* checkbox is unmarked.

Possible values:

marked

The *Proxy ARP* function is active.

The device responds to ARP requests received from end devices that are located in other networks.

unmarked (default setting)

The *Proxy ARP* function is inactive.

VRRP master candidate

Specifies the IP address for the primary virtual router. Physical routers within a virtual router instance use the VRRP IP address for the communication. If the virtual router IP address is the same as the IP address of a router interface, then the router is the IP address owner and the master router.

Possible values:

Valid IPv4 address (default setting: 0.0.0.0)

The default setting 0.0.0.0 indicates that the router is using the lower IP address as the *Master IP address*.

You can select the IP address of a router interface set up in the *Routing > Interfaces > Configuration* dialog.

Master IP address

Displays the current master router interface IP address.

Possible values:

Valid IPv4 address (default setting: 0.0.0.0)

Setting up the VRRP router instance

The device lets you set up to 8 virtual routers per router interface.

Before you set up a VRRP instance, verify that network routing functions properly and set the IP addresses on the router interfaces used for the VRRP instances.

Perform the following steps:

In the *Routing > L3-Redundancy > VRRP > Configuration* dialog, open the *Wizard* window.

In the *Wizard* window, open the *Create or select entry* page.

- Select a router interface from the *Port* drop-down list.
- Specify the Virtual Router Identifier in the *VRID* column.

In the *Wizard* window, open the *Edit entry* page.

- In *VRRP* tab in the *Configuration* frame, specify the values for the following parameters:

Priority


Preempt mode

Advertisement interval [s]

Ping answer

Select the *VRRP master candidate* IP address from the drop-down list.

To transfer the settings to the VRRP router interface table, click the *Finish* button.

In the *Routing > L3-Redundancy > VRRP > Configuration* dialog, select the *On* radio button in the *Operation* frame. Then click the  button.

Editing an existing VRRP router instance

Perform one of the following steps:

In the *Routing > L3-Redundancy > VRRP > Configuration* dialog, select a table row and click the  button to edit it.

Or

Double-click a field in the table and edit the value directly.
Or
Right-click a field and select a value.

Deleting a VRRP router instance

Perform the following step:

In the [Routing > L3-Redundancy > VRRP > Configuration](#) dialog, select a table row and click the  button.

[Wizard: VRRP configuration]

The *Wizard* window helps you set up a VRRP router instance.

Prerequisites:

- Network routing is functioning correctly.
- On the router interfaces used in the VRRP instance the IP addresses are specified.

The *Wizard* window guides you through the following steps:

- [Create or select entry](#)
- [Edit entry](#)
- [Tracking](#)
- [Virtual IP addresses](#)

Create or select entry

VRRP instances

Displays the existing instances available in the device. Select an item to continue. As an alternative, select a port and specify a value in the *VRID* field below.

Port

Specifies the port-based or VLAN-based router interface. You verify in the [Routing > Interfaces > Configuration](#) dialog if a router interface is set up on the port.

Possible values:

- <Port number >
Port-based router interface
- VLAN/ <VLAN ID>
VLAN-based router interface

VRID

Specifies the Virtual Router Identifier.

Possible values:

1..255

A virtual router uses `00-00-5E-00-01-XX` as its MAC address. The value specified here replaces the last octet (`XX`) in the MAC address. Assign a unique value to every physical router within a virtual router instance. The device changes the effective priority value to 255 for a physical router with the same IP address as the virtual router.

Edit entry

For each instance you can specify the parameters using the following tabs:

- [Edit entry - VRRP](#)

Edit entry - VRRP

Operation

Enables/disables the [VRRP](#) redundancy for the current instance.

Possible values:

[On](#)

The [VRRP](#) function is enabled for the current instance.

[Off](#) (default setting)

The [VRRP](#) function is disabled for the current instance.

Configuration

Base priority

Specifies the priority of the virtual router. If the value differs from the value in the [Priority](#) field, then the tracked object is unavailable or the virtual router is the IP address owner.

Possible values:

1..254 (default setting: 100)

The higher the number, the higher the priority. When you set up multiple VRRP routers in a single instance, distribute the priority values uniformly on the routers. For example, assign the priority value of 50 to the primary router, the value of 100 to the next router. Repeat the steps with the value 150, and so on. This distribution simplifies adding another router later with a priority between the existing values, for example with the value 75.

Priority

Displays the *VRRP* priority value. You specify the priority in the *Routing > OSPF > Interfaces* dialog. The router with the higher priority value takes over the master router role. If the virtual router IP address is the same as the IP address of a router interface, then the router is the IP address owner. If an IP address owner exists, then the *VRRP* function lets the device assign the IP address owner the priority value *255* and declares the router as the master router.

Possible values:

0

The higher the number, the higher the priority. When you disable or remove a *VRRP* router, which is in the master role, you force the instance to send an advertisement with priority value *0*. This lets the other backup routers know that the master does not participate. Sending a priority value *0* forces a new election.

1..255

The value *255* means that the virtual router is the IP address owner.

Preempt mode

Activates/deactivates the preempt mode. This setting specifies if this router, as a backup router, takes over the master router role when the master router has a lower VRRP priority.

Possible values:

marked (default setting)

The *Preempt mode* is active. The router takes the master router role from a router with a lower VRRP priority without an election.

unmarked

The *Preempt mode* is inactive. The router assumes the role of a backup router and listens for master router advertisements. After the *Master Down* interval expires, and no advertisements received from the master router, the router participates in the master router election process.

Advertisement interval [s]

Specifies the interval between master router advertisements in seconds.

Possible values:

1..255 (default setting: *1*)

Note: The longer the advertisement interval, the longer the time for which backup routers wait for a message from the master router before starting a new election process (*Master Down* interval). Also, specify the same value on every participant in a given virtual router instance.

Proxy ARP

Activates/deactivates the *Proxy ARP* function on the virtual router interface. The function lets you reach devices in other networks as if these devices were located in the local network. The *Proxy ARP* function is required when the device uses the VRRP instance with *1:1 NAT* rules. The prerequisite is that in the [Routing > Interfaces > Configuration](#) dialog for the relevant interface in use by the VRRP instance, the *Proxy ARP* checkbox is unmarked.

Possible values:

marked

The *Proxy ARP* function is active.

The device responds to ARP requests received from end devices that are located in other networks.

unmarked (default setting)

The *Proxy ARP* function is inactive.

VRRP master candidate

Specifies the IP address for the primary virtual router. Physical routers within a virtual router instance use the VRRP IP address for the communication. If the virtual router IP address is the same as the IP address of a router interface, then the router is the IP address owner and the master router.

Possible values:

Valid IP address (default setting: 0.0.0.0)

You can select the IP address of a router interface set up in the [Routing > Interfaces > Configuration](#) dialog.

Tracking

Current track entries

Displays the existing tracking objects available in the device. You set up tracking objects in the [Advanced > Tracking > Configuration](#) dialog. Select an item to continue. As an alternative, select a tracking object in the *Track name* field below.

Each tracking object contains the following parameters separated by a dash:

- Type of the tracking object
- Identification number of the tracking object
- Name of the tracking object

There are the following types of tracking objects:

- *Interface*

The device monitors the link status of its physical ports or of its link aggregation, LRE or VLAN router interface.

- *Ping*
The device monitors the route to a remote router or end device by sending periodic *ICMP echo request* packets.
- *Logical*
The device monitors tracking objects logically linked to each other and thus enables complex monitoring tasks.

Assigned track entries

Displays the tracking objects assigned with a *Decrement* value. You can remove an item clicking the **X** icon.

Track name

Specifies the name of the tracking object to which the virtual router is linked. Select an item from the drop-down list to continue. You set up tracking objects in the *Advanced > Tracking > Configuration* dialog.

If the result for a tracking object is negative, then the *VRRP* instance reduces the priority of the virtual router. The tracking object is negative for example, if the monitored interface is inactive or the monitored router cannot be reached.

Possible values:

Name of the tracking object, made up of *Type* and *Track ID*.

Decrement

Specifies the value by which the VRRP instance reduces the priority of the virtual router when the monitoring result is negative.

Possible values:

1..253

Note: If in the *Routing > L3-Redundancy > VRRP > Configuration* dialog the value in the *Priority* column is 255, then the virtual router is the IP address owner. In this case, the priority of the virtual router remains unchanged.

Add

Adds an item in the *Assigned track entries* field based on the values specified in the *Track name* and *Decrement* fields.

Virtual IP addresses

IP address

Displays the primary IP address of the router interface.

Possible values:

Valid IPv4 address (default setting: 0.0.0.0)

Multinetting

Displays the secondary IP address for the router interface and the subnet mask of the secondary IP addresses. You specify the secondary IP address and subnet mask in the [Routing > Interfaces > Configuration](#) dialog.

Virtual IP addresses

Displays the virtual IP address that you specified in the [IP address](#) field. You can remove an item clicking the **X** icon.

IP address

Specifies the assigned IP address of the master router within a virtual router.

Possible values:

Valid IPv4 address

Add

Adds an item in the [Virtual IP addresses](#) field based on the value specified in the [IP address](#) field.

7.8.1.2 VRRP Statistics

[Routing > L3-Redundancy > VRRP > Statistics]

This dialog displays the number of counters that count events relevant to the [VRRP](#) function.

Information

Checksum errors

Displays the number of VRRP messages received with the wrong checksum.

Version errors

Displays the number of VRRP messages received with an unknown or unsupported version number.

VRID errors

Displays the number of VRRP messages received with an invalid Virtual Router Identifier for this virtual router.

Table

For information on how to customize the appearance of the table, see [“Working with tables” on page 16](#).

Port

Displays the router interface number to which the table row relates.

VRID

Displays the Virtual Router Identifier.

Become master

Displays the number of times that the device has taken the master role. A high number can be an indication of an unstable network.

Advertise received

Displays the number of VRRP advertisements received.

Advertise interval errors

Displays the number of VRRP advertisements received by the router outside the advertisement interval. The value lets you determine if the routers have the same advertise interval specified across the virtual router instance.

Authentication failures

Displays the number of VRRP advertisements received with authentication errors.

IP TTL errors

Displays the number of VRRP advertisements received with an IP TTL not equal to 255.

Priority zero packets received

Displays the number of VRRP advertisements received with priority 0.

Priority zero packets sent

Displays the number of VRRP advertisements that the device sent with priority 0.

Invalid type packets received

Displays the number of VRRP advertisements received with an invalid type.

Address list errors

Displays the number of VRRP advertisements received for which the address list does not match the address list set up locally for the virtual router.

Invalid authentication type

Displays the number of VRRP advertisements received with an invalid authentication type.

Authentication type mismatch

Displays the number of VRRP advertisements received with an incorrect authentication type.

Packet length errors

Displays the number of VRRP advertisements received with an incorrect packet length.

7.8.1.3 VRRP Tracking

[Routing > L3-Redundancy > VRRP > Tracking]

VRRP tracking lets you follow the operation of specific object and react to a change in the object status. The function is periodically notified about the tracked object and displays the changes in the table. The table displays the object statuses as either [up](#), [down](#) or [not Ready](#).

Table

For information on how to customize the appearance of the table, see [“Working with tables” on page 16](#).

Buttons



Add

Opens the [Create](#) window to add a table row.

- From the [Port VRID](#) drop-down list, you select the interface and router ID of a virtual router that has been set up.
- From the [Track name](#) drop-down list, you select the tracking object with which the device links the virtual router.



Remove

Removes the selected table row.

Port

Displays the router interface number of the virtual router.

VRID

Displays the virtual router ID for this virtual router.

Track name

Displays the name of the tracking object to which the virtual router is linked.

If the result for a tracking object is negative, then the [VRRP](#) instance reduces the priority of the virtual router. The tracking object is negative for example, if the monitored interface is inactive or the monitored router cannot be reached.

Possible values:

- Name of the tracking object, made up of [Type](#) and [Track ID](#).
- Logical trackers, which combine multiple trackers
-
- No tracking object selected.

You set up tracking objects in the [Advanced > Tracking > Configuration](#) dialog.

Decrement

Specifies the value by which the VRRP instance reduces the priority of the virtual router when the monitoring result is negative.

Possible values:

- (default setting)
- 1.. 253

Note: If in the [Routing > L3-Redundancy > VRRP > Configuration](#) dialog the value in the *Priority* column is 255, then the virtual router is the IP address owner. In this case, the priority of the virtual router remains unchanged.

Status

Displays the monitoring result of the tracking object.

Possible values:

[not Ready](#)

The tracking object is not operating.

[up](#)

The monitoring result is positive:

- The link status is active.
- or
- The remote router or end device is reachable.

[down](#)

The monitoring result is negative:

- The link status is inactive.
- or
- The remote router or end device is not reachable.

A combination of the [up](#) and [down](#) trackers.

Active

Displays if the monitoring of the tracking object is active or inactive.

Possible values:

[marked](#)

The monitoring of the tracking object is active.

[unmarked](#)

The monitoring of the tracking object is inactive. You activate the monitoring in the [Advanced > Tracking > Configuration](#) dialog, *Active* column.

7.9 NAT

[Routing > NAT]

The menu contains the following dialogs:

- [NAT Global](#)
- [1:1 NAT](#)
- [Destination NAT](#)
- [Masquerading NAT](#)
- [Double NAT](#)


7.9.1 NAT Global

[Routing > NAT > NAT Global]

Network Address Translation (*NAT*) contains several procedures which automatically change the IP address information in the data packet. When set up in the device, the *NAT* function enables communication links between devices in different networks.

This dialog displays how many *NAT* rules can be set up for the individual *NAT* processes and indicates changes to the active *NAT* rules.

The device provides a multi-step approach to set up and apply the *NAT* rules:

- You add a rule.
- You assign the rule to a router interface.
Up to this step, changes have no effect on the behavior of the device and the data stream.
- You apply the rule to the data stream. To do this, click the  button in the respective frame.

1:1 NAT

Buttons

 Commit

Applies the *1:1 NAT* rules saved in the device to the data stream.

In the process, the device also removes the state information from the packet filter. This includes potential *DCE RPC* information of the *OPC Enforcer* function. Consequently, the device interrupts open communication connections.

Note: While the device is activating the saved rules, you cannot establish any new communication connections.

1:1 NAT rules (max.)

Displays the maximum number of *1:1 NAT* rules that the device lets you set up.

Configured 1:1 NAT rules


Displays the number of *1:1 NAT* rules set up in the device.

1:1 NAT pending actions

Displays if the *1:1 NAT* rules applied to the data stream differ from the saved *1:1 NAT* rules.

Possible values:

marked

At least one saved *1:1 NAT* rule contains modified settings. To apply the pending rules to the data stream, click the  button.

unmarked

The device applies the saved *1:1 NAT* rules to the data stream.

Destination NAT

Buttons

 Commit

Applies the *Destination NAT* rules saved in the device to the data stream.

In the process, the device also removes the state information from the packet filter. This includes potential *DCE RPC* information of the *OPC Enforcer* function. Consequently, the device interrupts open communication connections.

Note: While the device is activating the saved rules, you cannot establish any new communication connections.

Destination NAT rules (max.)

Displays the maximum number of *Destination NAT* rules that the device lets you set up.

Configured Destination NAT rules

Displays the number of *Destination NAT* rules set up in the device.

Configured Destination NAT interfaces


Displays the number of *Destination NAT* router interfaces set up in the device.

Destination NAT pending actions

Displays if the *Destination NAT* rules applied to the data stream differ from the saved *Destination NAT* rules.

Possible values:

marked

At least one saved *Destination NAT* rule contains modified settings. To apply the pending rules to the data stream, click the  button.

unmarked

The device applies the saved *Destination NAT* rules to the data stream.

Masquerading NAT

Buttons

 Commit

Applies the *Masquerading NAT* rules saved in the device to the data stream.

In the process, the device also removes the state information from the packet filter. This includes potential *DCE RPC* information of the *OPC Enforcer* function. Consequently, the device interrupts open communication connections.

Note: While the device is activating the saved rules, you cannot establish any new communication connections.

Masquerading NAT rules (max.)

Displays the maximum number of *Masquerading NAT* rules that the device lets you set up.

Configured Masquerading NAT rules

Displays the number of *Masquerading NAT* rules set up in the device.

Configured Masquerading NAT interfaces


Displays the number of *Masquerading NAT* router interfaces set up in the device.

Masquerading NAT pending actions

Displays if the *Masquerading NAT* rules applied to the data stream differ from the saved *Masquerading NAT* rules.

Possible values:

marked

At least one saved *Masquerading NAT* rule contains modified settings. To apply the pending rules to the data stream, click the  button.

unmarked

The device applies the saved *Masquerading NAT* rules to the data stream.

Double NAT

Buttons

 Commit

Applies the *Double NAT* rules saved in the device to the data stream.

In the process, the device also removes the state information from the packet filter. This includes potential *DCE RPC* information of the *OPC Enforcer* function. Consequently, the device interrupts open communication connections.

Note: While the device is activating the saved rules, you cannot establish any new communication connections.

Double NAT rules (max.)

Displays the maximum number of *Double NAT* rules that the device lets you set up.

Configured Double NAT rules

Displays the number of *Double NAT* rules set up in the device.

Configured Double NAT interfaces


Displays the number of *Double NAT* router interfaces set up in the device.

Double NAT pending actions

Displays if the *Double NAT* rules applied to the data stream differ from the saved *Double NAT* rules.

Possible values:

marked

At least one saved *Double NAT* rule contains modified settings. To apply the pending rules to the data stream, click the  button.

unmarked

The device applies the saved *Double NAT* rules to the data stream.

7.9.2 1:1 NAT

[Routing > NAT > 1:1 NAT]

The *1:1 NAT* function lets you establish communication links within a local network to devices that are located in other networks. The *NAT* router virtually “shifts” the devices into the public network. To do this, the *NAT* router replaces the virtual with the actual IP address in the data packet while sending it. A typical application is connecting some identically structured production cells with the same IP address to a server farm.

The prerequisite for the *1:1 NAT* process is that the *NAT* router itself responds to ARP requests. To do this, activate the *Proxy ARP* function for the relevant interface in the *Routing > Interfaces > Configuration* dialog or in the *Routing > L3-Redundancy > VRRP > Configuration* dialog.

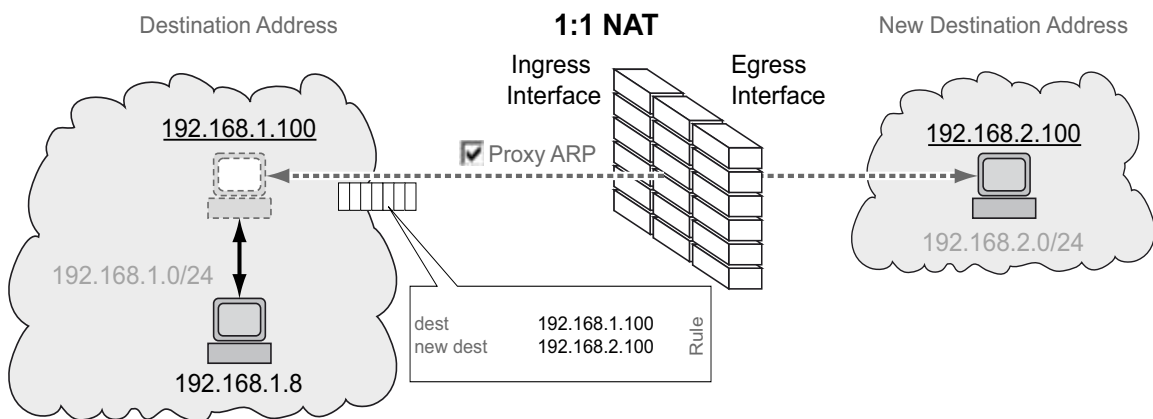


Figure 3: How the *1:1 NAT* function works

To use the *NAT* function, set up a router interface for each network and turn on the routing function in the device.

The data packets go through the filter functions of the device in the following sequence:

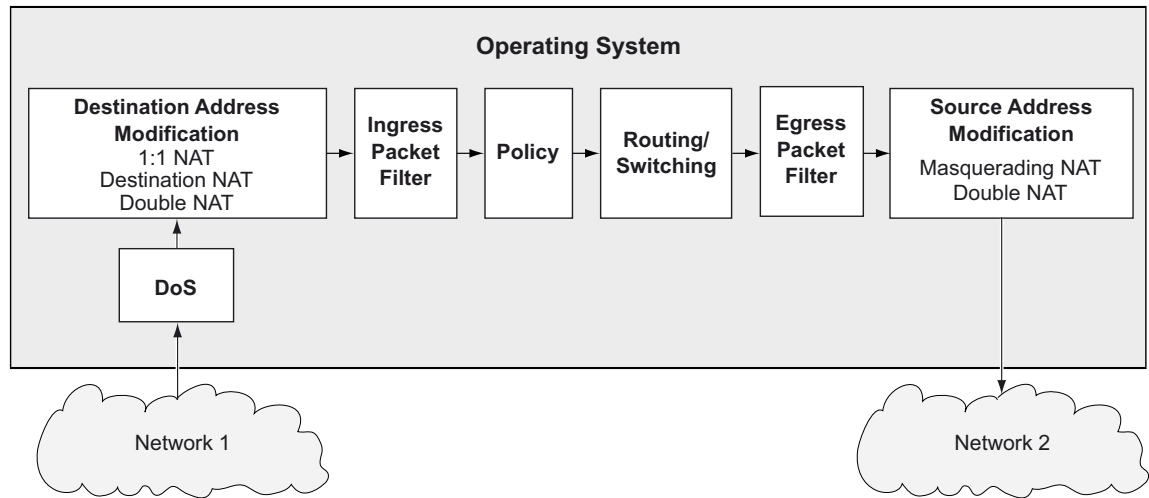


Figure 4: Processing sequence of the data packets in the device

The menu contains the following dialogs:

[1:1 NAT Rule](#)

7.9.21 1:1 NAT Rule

[Routing > NAT > 1:1 NAT > Rule]

In this dialog, you set up the *1:1 NAT* rules and assign router interfaces to which the device applies the *1:1 NAT* rules. The device lets you set up to 255 *1:1 NAT* rules.

Table

For information on how to customize the appearance of the table, see [“Working with tables” on page 16](#).

Buttons



Add

Opens the *Create* window to add a table row.

- In the *Destination address* field, you specify the destination address of the data packets to which the device applies the rule. The device sends data packets with this destination address to the destination address specified in the *New destination address* column.

Possible values:

Valid IPv4 address

The device applies the *1:1 NAT* rule only to data packets which contain the destination address specified here.

Valid IPv4 address and netmask in CIDR notation

The device applies the *1:1 NAT* rule only to data packets which contain a destination address in the subnet specified here.

- In the *New destination address* field, you specify the actual IP address of the destination device. The device sends data packets to the destination address specified here.

Possible values:

Valid IPv4 address

The device replaces the destination address in the data packet with this new destination address.

Valid IPv4 address and netmask in CIDR notation

The device replaces the destination address in the data packet with a destination address in the subnet specified here.

When you click the *Ok* button, the device adds the table row. The device assigns the values specified in the *Destination address* and *New destination address* fields to this table row.



Remove

Removes the selected table row.

Index

Displays the index number to which the table row relates. The device automatically assigns the value when you add a table row.

Rule name

Displays the name of the [1:1 NAT](#) rule. To change the name, click the relevant field.

Possible values:

Alphanumeric ASCII character string with 0..32 characters

Priority

Specifies the priority of the [1:1 NAT](#) rule.

Using the priority, you specify the order in which the device applies several rules to the data stream. The device applies the rules in ascending order starting with priority [0](#).

Possible values:

[0](#) . [6500](#) (default setting: [0](#))

Ingress interface

Assigns the [1:1 NAT](#) rule to the router interface on which the device receives data packets. The [1:1 NAT](#) rule makes the destination device virtually accessible in the network connected here.

Possible values:

[<Interface number >](#)

The device applies the [1:1 NAT](#) rule to this router interface, and only to data packets addressed to the IP address specified in the [Destination address](#) column.

[no Port](#)

No router interface is assigned to the [1:1 NAT](#) rule. Someone removed the router interface after the last edit of the [1:1 NAT](#) rule.

You enable on the ARP proxy function on this router interface in the [Routing > Interfaces > Configuration](#) dialog.

Destination address

Specifies the destination address of the data packets to which the device applies the [1:1 NAT](#) rule. The device sends data packets with this destination address to the destination address specified in the [New destination address](#) column.

Possible values:

Valid IPv4 address

The device applies the [1:1 NAT](#) rule only to data packets which contain the destination address specified here.

Valid IPv4 address and netmask in CIDR notation

The device applies the [1:1 NAT](#) rule only to data packets which contain a destination address in the subnet specified here.

Egress interface

Assigns the *1:1 NAT* rule to the router interface on which the device forwards the modified data packets. The destination device can actually be reached in the network connected here.

Possible values:

`<Interface number>`

The device forwards the modified data packets on this router interface.

`no Port`

No router interface is assigned to the *1:1 NAT* rule. Someone removed the router interface after the last edit of the *1:1 NAT* rule.

New destination address

Specifies the actual IP address of the destination device. The device sends data packets to the destination address specified here.

Possible values:

Valid IPv4 address

The device replaces the destination address in the data packet with this new destination address.

Valid IPv4 address and netmask in CIDR notation

The device replaces the destination address in the data packet with a destination address in the subnet specified here.

Trap

Activates/deactivates the sending of SNMP traps when the device applies a *1:1 NAT* rule to a data packet.

Possible values:

`marked`

The sending of SNMP traps is active. The prerequisite is that in the *Diagnostics > Status Configuration > Alarms (Traps)* dialog the *Alarms (Traps)* function is enabled and at least one trap destination is specified.

If the device applies the *1:1 NAT* rule to a data packet, then the device sends an SNMP trap.

`unmarked` (default setting)

The sending of SNMP traps is inactive.

Log

Activates/deactivates the logging in the log file. See the *Diagnostics > Report > System Log* dialog.

Possible values:

`marked`

Logging is activated.

When the device applies the *1:1 NAT* rule to a data packet, the device places an entry in the log file.

`unmarked` (default setting)

Logging is deactivated.

Active

Activates/deactivates the *1:1 NAT* rule.

Possible values:

marked

The rule is active.

unmarked (default setting)

The rule is inactive.

7.9.3 Destination NAT

[Routing > NAT > Destination NAT]

The *Destination NAT* function lets you divert the data stream of outgoing communication links to or through a server in a local network.

A special form of the *Destination NAT* function is *port forwarding*. You use *port forwarding* to hide the structure of a network from the outside while still allowing communication links from the outside into the network. A typical application is remote control of a PC in a production cell. The maintenance station establishes the communication link to the *NAT* router, and the *Destination NAT* function takes care of the routing to the production cell.

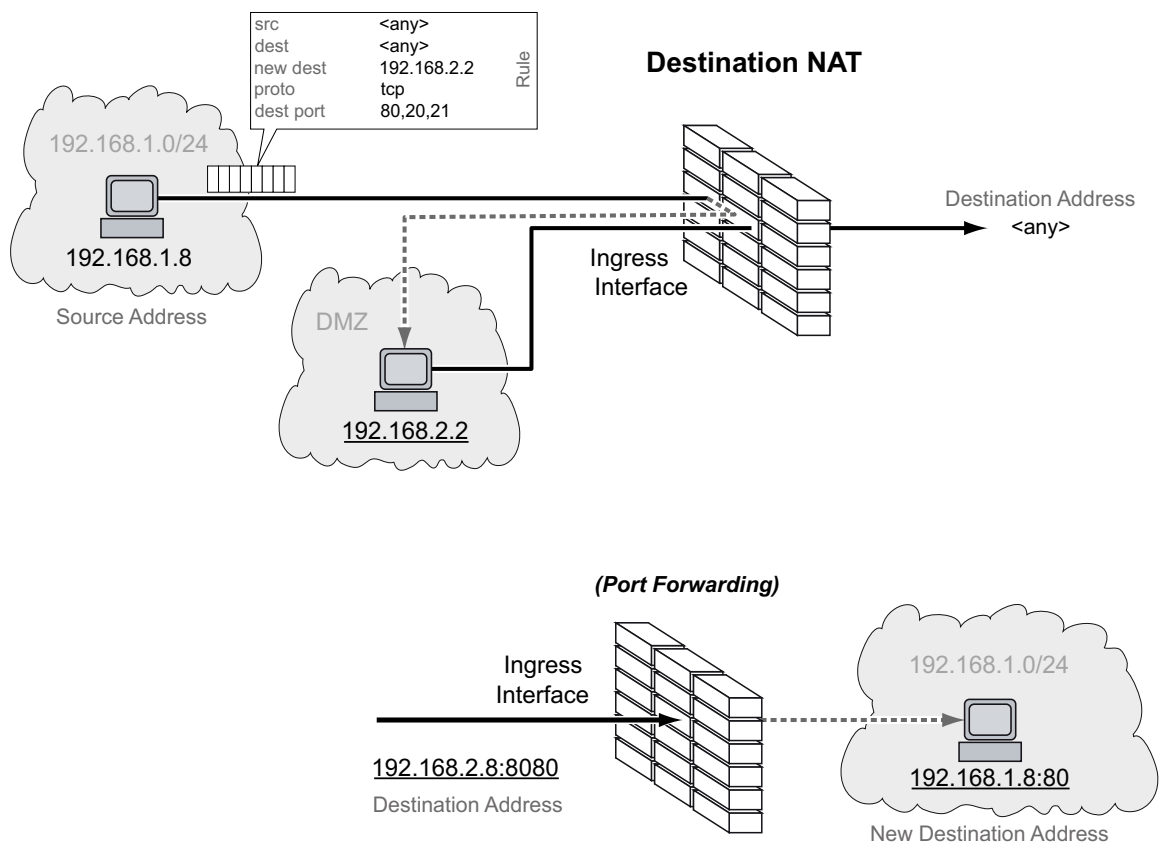


Figure 5: How the *Destination NAT* function works

To use the *NAT* function, set up a router interface for each network and turn on the routing function in the device.

The data packets go through the filter functions of the device in the following sequence:

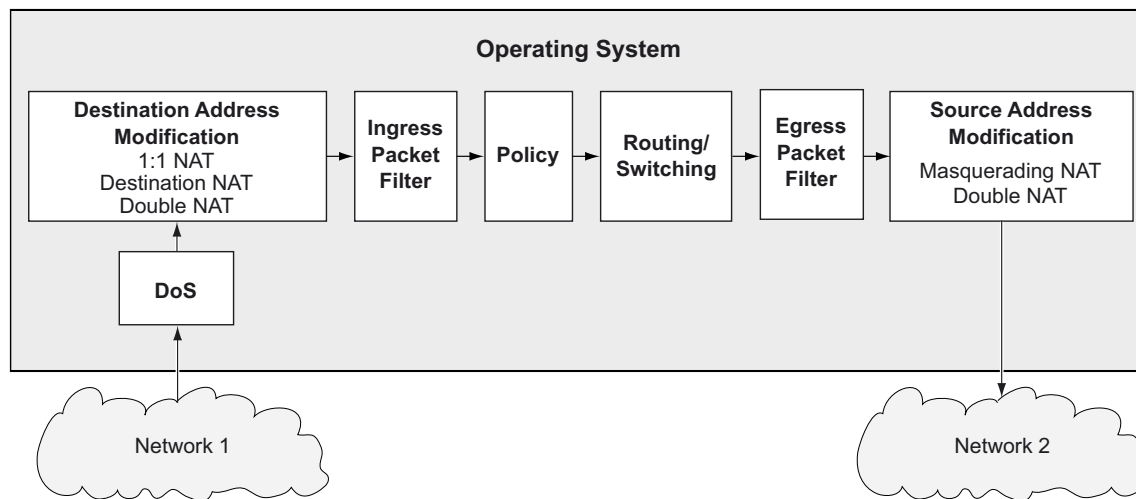


Figure 6: Processing sequence of the data packets in the device

The menu contains the following dialogs:

- [Destination NAT Rule](#)
- [Destination NAT Mapping](#)
- [Destination NAT Overview](#)

7.9.3.1 Destination NAT Rule

[Routing > NAT > Destination NAT > Rule]

In this dialog, you set up the *Destination NAT* rules.

You assign a router interface to the affected *Destination NAT* rule in the *Routing > NAT > Destination NAT > Mapping* dialog.

An overview of which *Destination NAT* rule is to be assigned to which router interface can be found in the *Routing > NAT > Destination NAT > Overview* dialog.

The device lets you set up to 255 *Destination NAT* rules.

Table

For information on how to customize the appearance of the table, see “[Working with tables](#)” on [page 16](#).

Buttons



Add

Opens the *Create* window to add a table row.

- In the *New destination address* field, you specify the actual IP address of the destination device. The device sends data packets to the destination address specified here.

Possible values:

Valid IPv4 address

The device replaces the destination address in the data packet with this new destination address.

When you click the *Ok* button, the device adds the table row. The device assigns the value specified in the *New destination address* field to this table row.



Remove

Removes the selected table row.

Index

Displays the index number to which the table row relates. The device automatically assigns the value when you add a table row.

Rule name

Displays the name of the *Destination NAT* rule. To change the name, click the relevant field.

Possible values:

Alphanumeric ASCII character string with 0..32 characters

Source address

Specifies the source address of the data packets to which the device applies the *Destination NAT* rule.

Possible values:

any (default setting)

The device applies the *Destination NAT* rule to data packets with any source address.

Valid IPv4 address

The device applies the *Destination NAT* rule only to data packets containing the source address specified here.

Valid IPv4 address and netmask in CIDR notation

The device applies the *Destination NAT* rule only to data packets containing a source address in the subnet specified here.

An exclamation mark (!) preceding the IP address reverses the expression into its opposite. The device applies the *Destination NAT* rule to data packets NOT containing the source address specified here.

Source port

Specifies the source port of the data packets to which the device applies the *Destination NAT* rule. The prerequisite is that in the *Protocol* field the value *TCP* or *UDP* is specified.

Possible values:

any (default setting)

The device applies the *Destination NAT* rule to every data packet without evaluating the source port.

1..65535 (2¹⁶ - 1)

The device applies the *Destination NAT* rule only to data packets containing the specified source port.

The field lets you specify the following options:

- You specify a port with a single numerical value, for example 21.
 - You specify multiple individual ports with numerical values separated by commas, for example 21, 80, 110.
 - You specify a port range with numerical values connected by dashes, for example 2000-3000.
 - You can also combine ports and port ranges, for example 21, 2000-3000, 65535.
- The column lets you specify up to 15 numerical values. When you enter 21, 2000-3000, 65535, for example, you use 4 of 15 numerical values.

Destination address

Specifies the destination address of the data packets to which the device applies the *Destination NAT* rule. The device sends data packets with this destination address to the destination address specified in the *New destination address* column.

Possible values:

any

The device applies the *Destination NAT* rule to data packets with any destination address.

Valid IPv4 address

The device applies the *Destination NAT* rule only to data packets which contain the destination address specified here.

Valid IPv4 address and netmask in CIDR notation

The device applies the *Destination NAT* rule only to data packets which contain a destination address in the subnet specified here.

An exclamation mark (!) preceding the IP address reverses the expression into its opposite. The device applies the *Destination NAT* rule to data packets NOT containing the destination address specified here.

Destination port

Specifies the destination port of the data packets to which the device applies the *Destination NAT* rule.

Possible values:

any (default setting)

The device applies the *Destination NAT* rule to every data packet without evaluating the destination port.

1..65535 (2¹⁶ - 1)

The device applies the *Destination NAT* rule only to data packets containing the specified destination port.

The field lets you specify the following options:

- You specify a port with a single numerical value, for example 21.
 - You specify multiple individual ports with numerical values separated by commas, for example 21, 80, 110.
 - You specify a port range with numerical values connected by dashes, for example 2000-3000.
 - You can also combine ports and port ranges, for example 21, 2000-3000, 65535.
- The column lets you specify up to 15 numerical values. When you enter 21, 2000-3000, 65535, for example, you use 4 of 15 numerical values.

New destination address

Specifies the actual IP address of the destination device. The device sends data packets to the destination address specified here.

Possible values:

Valid IPv4 address

The device replaces the destination address in the data packet with this new destination address.

New destination port

Specifies the port of the destination device. The device forwards data packets to the destination port specified here.

Possible values:

any

The device retains the original destination port in the data packet.

1..65535 (2¹⁶ - 1)

The device replaces the destination port in the packet with this new destination port.

Protocol

Restricts the *Destination NAT* rule to an IP protocol. The device applies the *Destination NAT* rule only to data packets of the specified IP protocol.

Possible values:

[i cnp](#)

Internet Control Message Protocol (RFC 792)

[i grp](#)

Internet Group Management Protocol

[i pi p](#)

IP in IP tunneling (RFC 1853)

[t cp](#)

Transmission Control Protocol (RFC 793)

[udp](#)

User Datagram Protocol (RFC 768)

[esp](#)

IPsec Encapsulated Security Payload (RFC 2406)

[ah](#)

IPsec Authentication Header (RFC 2402)

[i cnpv6](#)

Internet Control Message Protocol for IPv6

[any](#) (default setting)

The device applies the *Destination NAT* rule to every data packet without evaluating the IP protocol.

Log

Activates/deactivates the logging in the log file. See the *Diagnostics > Report > System Log* dialog.

Possible values:

[marked](#)

Logging is activated.

When the device applies the *Destination NAT* rule to a data packet, the device places an entry in the log file.

[unmarked](#) (default setting)

Logging is deactivated.

Trap

Activates/deactivates the sending of SNMP traps when the device applies a *Destination NAT* rule to a data packet.

Possible values:

[marked](#)

The sending of SNMP traps is active. The prerequisite is that in the *Diagnostics > Status Configuration > Alarms (Traps)* dialog the *Alarms (Traps)* function is enabled and at least one trap destination is specified.

If the device applies the *Destination NAT* rule to a data packet, then the device sends an SNMP trap.

[unmarked](#) (default setting)

The sending of SNMP traps is inactive.

Active

Activates/deactivates the *Destination NAT* rule.

Possible values:

`marked`


The rule is active.

`unmarked` (default setting)

The rule is inactive.

7.9.3.2 Destination NAT Mapping

[Routing > NAT > Destination NAT > Mapping]

In this dialog, you assign the *Destination NAT* rules to a router interface. To do this, click the  button.

You add and edit the *Destination NAT* rules in the *Routing > NAT > Destination NAT > Rule*.

An overview of which *Destination NAT* rule is to be assigned to which router interface can be found in the *Routing > NAT > Destination NAT > Overview* dialog.

Table

For information on how to customize the appearance of the table, see “Working with tables” on page 16.

Buttons

 Remove

Removes the selected table row.

 Assign

Opens the *Assign* window. In this window, you assign a set-up router interface to an existing *Destination NAT* rule.

Port

Displays the number of the router interface on which the device applies the *Destination NAT* rule.

Rule index

Displays the sequential number of the *Destination NAT* rule. See the *Index* column in the *Routing > NAT > Destination NAT > Rule* dialog. You specify the index number when you add a table row.

Rule name

Displays the name of the *Destination NAT* rule. See the *Rule name* column in the *Routing > NAT > Destination NAT > Rule* dialog.

Direction

Displays if the device applies the *Destination NAT* rule to data packets received or sent.

Possible values:

ingress

The device applies the *Destination NAT* rule to data packets received on the router interface.

Priority

Specifies the priority of the *Destination NAT* rule.

Using the priority, you specify the order in which the device applies several rules to the data stream. The device applies the rules in ascending order starting with priority 1.

Possible values:

1..6500 (default setting: 1)

Active

Activates/deactivates the *Destination NAT* rule.

Possible values:

marked

The rule is active.

unmarked (default setting)

The rule is inactive.

7.9.3.3 Destination NAT Overview

[Routing > NAT > Destination NAT > Overview]

In this dialog, you will find an overview of which *Destination NAT* rule is assigned to which router interface.

You add and edit the *Destination NAT* rules in the *Routing > NAT > Destination NAT > Rule*.

You assign a router interface to the affected *Destination NAT* rule in the *Routing > NAT > Destination NAT > Mapping* dialog.

Table

For information on how to customize the appearance of the table, see “Working with tables” on page 16.

Port

Displays the number of the router interface on which the device applies the *Destination NAT* rule.

Rule index

Displays the sequential number of the *Destination NAT* rule. See the *Index* column in the *Routing > NAT > Destination NAT > Rule* dialog.

Rule name

Displays the name of the *Destination NAT* rule. See the *Rule name* column in the *Routing > NAT > Destination NAT > Rule* dialog.

Destination address

Displays the destination address of the data packets to which the device applies the *Destination NAT* rule. The device sends data packets with this destination address to the destination address specified in the *New destination address* column.

New destination address

Displays the actual IP address of the destination device. The device sends data packets to the destination address specified here.

Trap

Displays if the device sends an SNMP trap when it applies the *Destination NAT* rule to a data packet.

Possible values:

marked

The device sends an SNMP trap. The prerequisite is that in the *Diagnostics > Status Configuration > Alarms (Traps)* dialog the *Alarms (Traps)* function is enabled and at least one trap destination is specified.

unmarked

The device does not send an SNMP trap.

Log

Displays if the device places an entry in the log file when it applies the *Destination NAT* rule to a data packet.

Possible values:

marked

When the device applies the *Destination NAT* rule to a data packet, the device places an entry in the log file. See the *Diagnostics > Report > System Log* dialog.

unmarked

Logging is disabled.

Direction

Displays if the device applies the *Destination NAT* rule to data packets received or sent.

Possible values:

ingress

The device applies the *Destination NAT* rule to data packets received on the router interface.

Priority

Displays the priority of the *Destination NAT* rule.

The device applies rules to the data stream in ascending order starting with priority 1.

7.9.4 Masquerading NAT

[Routing > NAT > Masquerading NAT]

The *Masquerading NAT* function hides any number of devices behind the IP address of the *NAT* router and thus hides the structure of a network from other networks. To do this, the *NAT* router replaces the sender address in the data packet with its own IP address. Also, the *NAT* router replaces the source port in the data packet with its own value to send the response data packets back to the original sender later on.

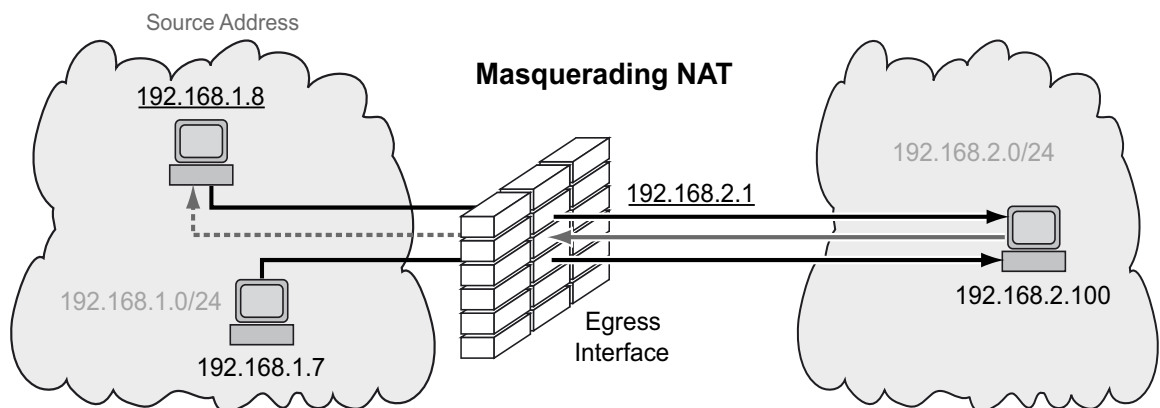


Figure 7: How the *Masquerading NAT* function works

To use the **NAT** function, set up a router interface for each network and turn on the routing function in the device.

Note: If you enable the **VRRP** function on a router interface, then the **Masquerading NAT** function is ineffective on this router interface.

The data packets go through the filter functions of the device in the following sequence:

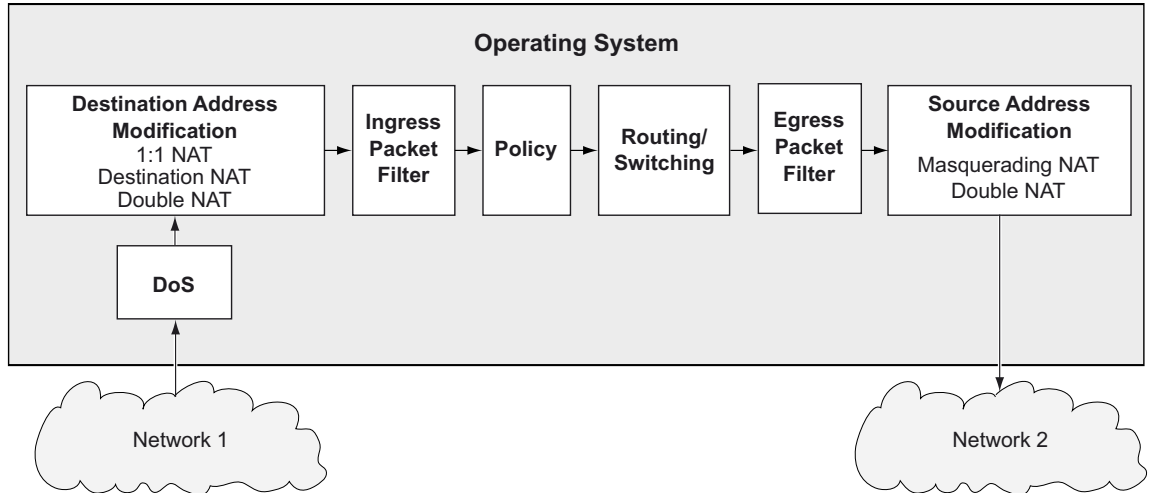


Figure 8: Processing sequence of the data packets in the device

The menu contains the following dialogs:

- [Masquerading NAT Rule](#)
- [Masquerading NAT Mapping](#)
- [Masquerading NAT Overview](#)

7.9.4.1 Masquerading NAT Rule

[Routing > NAT > Masquerading NAT > Rule]

In this dialog, you set up the *Masquerading NAT* rules.

You assign a router interface to the affected *Masquerading NAT* rule in the *Routing > NAT > Masquerading NAT > Mapping* dialog.

An overview of which *Masquerading NAT* rule is to be assigned to which router interface can be found in the *Routing > NAT > Masquerading NAT > Overview* dialog.

The device lets you set up to 128 *Masquerading NAT* rules.

Table

For information on how to customize the appearance of the table, see “Working with tables” on page 16.

Buttons



Add

Adds a table row.



Remove

Removes the selected table row.

Index

Displays the index number to which the table row relates. The device automatically assigns the value when you add a table row.

Rule name

Displays the name of the *Masquerading NAT* rule. To change the name, click the relevant field.

Possible values:

Alphanumeric ASCII character string with 0..32 characters

Source address

Specifies the source address of the data packets to which the device applies the *Masquerading NAT* rule.

Possible values:

any

The device applies the *Masquerading NAT* rule to data packets with any source address.

Valid IPv4 address

The device applies the *Masquerading NAT* rule only to data packets containing the source address specified here.

Valid IPv4 address and netmask in CIDR notation

The device applies the [Masquerading NAT](#) rule only to data packets containing a source address in the subnet specified here.

An exclamation mark (!) preceding the IP address reverses the expression into its opposite. The device applies the [Masquerading NAT](#) rule to data packets NOT containing the source address specified here.

Source port

Specifies the source port of the data packets to which the device applies the [Masquerading NAT](#) rule.

Possible values:

[any](#) (default setting)

The device applies the [Masquerading NAT](#) rule to every data packet without evaluating the source port.

[1..65535](#) (2¹⁶ - 1)

The device applies the [Masquerading NAT](#) rule only to data packets containing the specified source port.

The field lets you specify the following options:

- You specify a port with a single numerical value, for example [21](#).
- You specify multiple individual ports with numerical values separated by commas, for example [21, 80, 110](#).
- You specify a port range with numerical values connected by dashes, for example [2000-3000](#).
- You can also combine ports and port ranges, for example [21, 2000-3000, 65535](#).
The column lets you specify up to 15 numerical values. When you enter [21, 2000-3000, 65535](#), for example, you use 4 of 15 numerical values.

Protocol

Restricts the [Masquerading NAT](#) rule to an IP protocol. The device applies the [Masquerading NAT](#) rule only to data packets of the specified IP protocol.

Possible values:

[tcp](#)

Transmission Control Protocol (RFC 793)

[udp](#)

User Datagram Protocol (RFC 768)

[any](#) (default setting)

The device applies the [Masquerading NAT](#) rule to every data packet without evaluating the IP protocol.

Log

Activates/deactivates the logging in the log file. See the [Diagnostics > Report > System Log](#) dialog.

Possible values:

[marked](#)

Logging is activated.

When the device applies the [Masquerading NAT](#) rule to a data packet, the device places an entry in the log file.

[unmarked](#) (default setting)

Logging is deactivated.

Trap

Activates/deactivates the sending of SNMP traps when the device applies a *Masquerading NAT* rule to a data packet.

Possible values:

marked

The sending of SNMP traps is active. The prerequisite is that in the *Diagnostics > Status Configuration > Alarms (Traps)* dialog the *Alarms (Traps)* function is enabled and at least one trap destination is specified.

If the device applies the *Masquerading NAT* rule to a data packet, then the device sends an SNMP trap.

unmarked (default setting)

The sending of SNMP traps is inactive.

IPsec exempt

Activates/deactivates applying the *Masquerading NAT* rule to IPsec data packets.

Possible values:

marked

The device does not apply the *Masquerading NAT* rule to the IPsec data packets. The device sends IPsec data packets through the VPN tunnel without any modification.

unmarked (default setting)

The device applies the *Masquerading NAT* rule to the IPsec data packets. The device sends IPsec data packets through the VPN tunnel depending on the settings of the Traffic Selector in the *Source address (CIDR)* and *Source restrictions* columns. See the *Virtual Private Network > Connections* dialog.

Active

Activates/deactivates the *Masquerading NAT* rule.

Possible values:

marked


The rule is active.

unmarked (default setting)

The rule is inactive.

7.9.4.2 Masquerading NAT Mapping

[Routing > NAT > Masquerading NAT > Mapping]

In this dialog, you assign the *Masquerading NAT* rules to a router interface. To do this, click the  button.

You add and edit the *Masquerading NAT* rules in the *Routing > NAT > Masquerading NAT > Rule*.

An overview of which *Masquerading NAT* rule is to be assigned to which router interface can be found in the *Routing > NAT > Masquerading NAT > Overview* dialog.

Table

For information on how to customize the appearance of the table, see “Working with tables” on page 16.

Buttons

 Remove

Removes the selected table row.

 Assign

Opens the *Assign* window. In this window, you assign a set-up router interface to an existing *Masquerading NAT* rule.

Port

Displays the number of the router interface on which the device applies the *Masquerading NAT* rule.

Rule index

Displays the sequential number of the *Masquerading NAT* rule. See the *Index* column in the *Routing > NAT > Masquerading NAT > Rule* dialog. You specify the index number when you add a table row.

Rule name

Displays the name of the *Masquerading NAT* rule. See the *Rule name* column in the *Routing > NAT > Masquerading NAT > Rule* dialog.

Direction

Displays if the device applies the *Masquerading NAT* rule to data packets received or sent.

Possible values:

egress

The device applies the *Masquerading NAT* rule to data packets sent on the router interface.

Priority

Specifies the priority of the *Masquerading NAT* rule.

Using the priority, you specify the order in which the device applies several rules to the data stream. The device applies the rules in ascending order starting with priority 1.

Possible values:

1..6500 (default setting: 1)

Active

Activates/deactivates the *Masquerading NAT* rule.

Possible values:

marked

The rule is active.

unmarked (default setting)

The rule is inactive.

7.9.4.3 Masquerading NAT Overview

[Routing > NAT > Masquerading NAT > Overview]

In this dialog, you will find an overview of which *Masquerading NAT* rule is assigned to which router interface.

You add and edit the *Masquerading NAT* rules in the *Routing > NAT > Masquerading NAT > Rule*.

You assign a router interface to the affected *Masquerading NAT* rule in the *Routing > NAT > Masquerading NAT > Mapping* dialog.

Table

For information on how to customize the appearance of the table, see “Working with tables” on page 16.

Port

Displays the number of the router interface on which the device applies the *Masquerading NAT* rule.

Rule index

Displays the sequential number of the *Masquerading NAT* rule. See the *Index* column in the *Routing > NAT > Masquerading NAT > Rule* dialog.

Rule name

Displays the name of the *Masquerading NAT* rule. See the *Rule name* column in the *Routing > NAT > Masquerading NAT > Rule* dialog.

Trap

Displays if the device sends an SNMP trap when it applies the *Masquerading NAT* rule to a data packet.

Possible values:

marked

The device sends an SNMP trap. The prerequisite is that in the *Diagnostics > Status Configuration > Alarms (Traps)* dialog the *Alarms (Traps)* function is enabled and at least one trap destination is specified.

unmarked

The device does not send an SNMP trap.

Log

Displays if the device places an entry in the log file when it applies the *Masquerading NAT* rule to a data packet.

Possible values:

[marked](#)

When the device applies the *Masquerading NAT* rule to a data packet, the device places an entry in the log file. See the *Diagnostics > Report > System Log* dialog.

[unmarked](#)

Logging is disabled.

Direction

Displays if the device applies the *Masquerading NAT* rule to data packets received or sent.

Possible values:

[egress](#)

The device applies the *Masquerading NAT* rule to data packets sent on the router interface.

Priority

Displays the priority of the *Masquerading NAT* rule.

The device applies rules to the data stream in ascending order starting with priority 1.

7.9.5 Double NAT

[Routing > NAT > Double NAT]

The *Double NAT* function lets you establish communication links between end devices located in different IP networks, which have no way to specify a *default gateway* or *default route*. The *NAT* router virtually “shifts” the devices into the other network. To do this, the *NAT* router replaces the source address and the destination address in the data packet during sending. A typical application is the linking of controllers located in different networks.

The prerequisite for the *Double NAT* function is that the *NAT* router itself responds to ARP requests from the respective network. To make this happen, turn on the ARP proxy function on the ingress interface and on the egress interface.

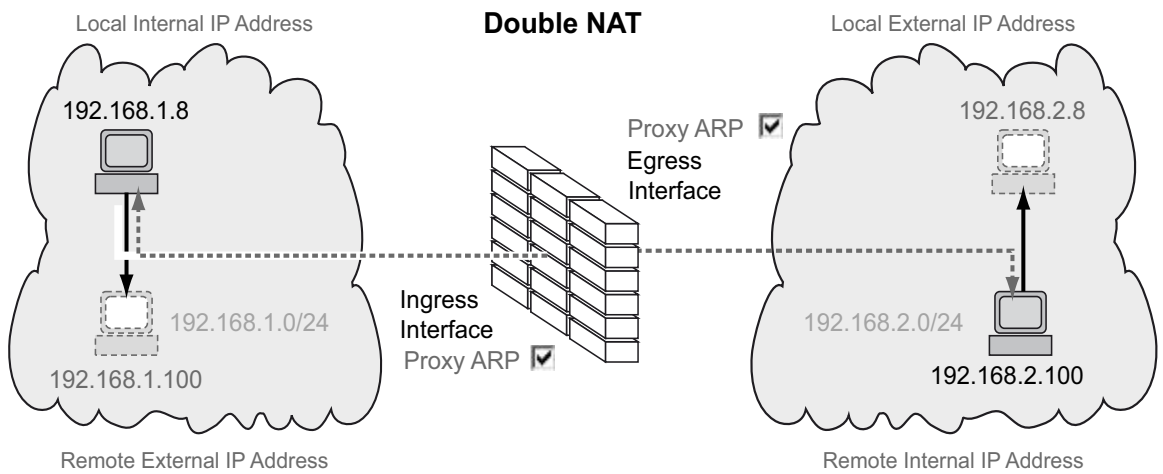


Figure 9: How the *Double NAT* function works

To use the *NAT* function, set up a router interface for each network and turn on the routing function in the device.

The data packets go through the filter functions of the device in the following sequence:

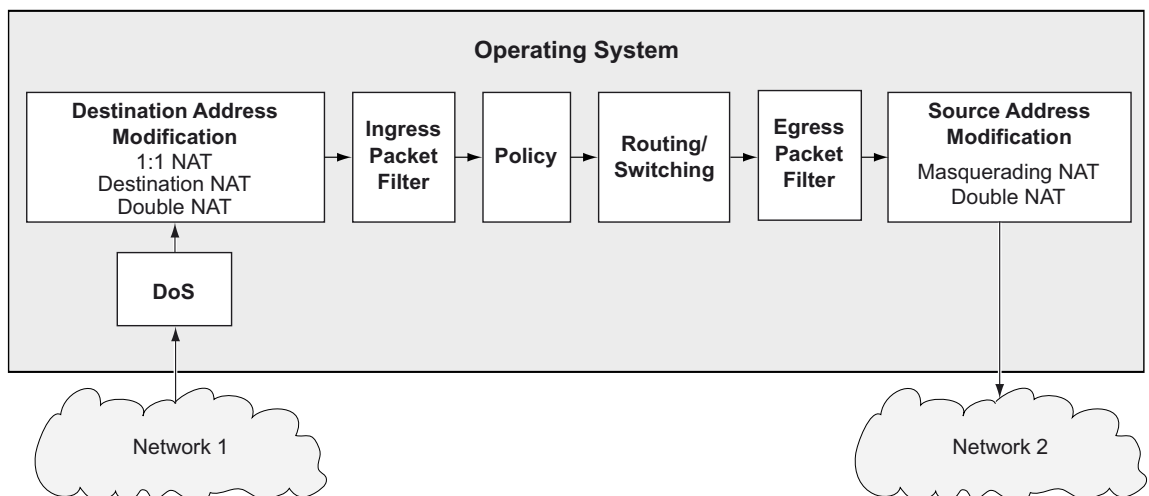


Figure 10: Processing sequence of the data packets in the device

The menu contains the following dialogs:

- Double NAT Rule
- Double NAT Mapping
- Double NAT Overview

7.9.5.1 Double NAT Rule

[Routing > NAT > Double NAT > Rule]

In this dialog, you set up the *Double NAT* rules.

You assign the router interfaces to the related *Double NAT* rule in the *Routing > NAT > Double NAT > Mapping* dialog.

An overview of which *Double NAT* rule is assigned to which router interfaces you find in the *Routing > NAT > Double NAT > Overview* dialog.

The device lets you set up to 255 *Double NAT* rules.

Table

For information on how to customize the appearance of the table, see “[Working with tables](#)” on [page 16](#).

Buttons



Opens the *Create* window to add a table row.

- In the *Local internal IP address* field, you specify the actual IP address for the device placed in the first network.
Possible values:
Valid IPv4 address
The device applies the *Double NAT* rule only to data packets containing the source address specified here.
- In the *Local external IP address* field, you specify the virtual IP address in the second network for the device placed in the first network.
Possible values:
Valid IPv4 address
The device applies the *Double NAT* rule only to data packets containing the source address specified here.

- In the *Remote internal IP address* field, you specify the actual IP address for the device placed in the second network.
Possible values:
Valid IPv4 address
The device applies the *Double NAT* rule only to data packets containing the source address specified here.
 - In the *Remote external IP address* field, you specify the virtual IP address in the first network for the device placed in the second network.
Possible values:
Valid IPv4 address
The device applies the *Double NAT* rule only to data packets containing the source address specified here.
- When you click the *Ok* button, the device adds the table row. The device assigns the values specified in the *Local internal IP address*, *Local external IP address*, *Remote internal IP address* and *Remote external IP address* fields to this table row.



Removes the selected table row.

Index

Displays the index number to which the table row relates. The device automatically assigns the value when you add a table row.

Rule name

Displays the name of the *Double NAT* rule. To change the name, click the relevant field.

Possible values:

Alphanumeric ASCII character string with 0..32 characters

Local internal IP address

Specifies the actual IP address for the device placed in the first network.

Possible values:

Valid IPv4 address

The device applies the *Double NAT* rule only to data packets containing the source address specified here.

Local external IP address

Specifies the virtual IP address in the second network for the device placed in the first network.

Possible values:

Valid IPv4 address

The device applies the *Double NAT* rule only to data packets containing the source address specified here.

Remote internal IP address

Specifies the actual IP address for the device placed in the second network.

Possible values:

Valid IPv4 address

The device applies the *Double NAT* rule only to data packets containing the source address specified here.

Remote external IP address

Specifies the virtual IP address in the first network for the device placed in the second network.

Possible values:

Valid IPv4 address

The device applies the *Double NAT* rule only to data packets containing the source address specified here.

Log

Activates/deactivates the logging in the log file. See the *Diagnostics > Report > System Log* dialog.

Possible values:

marked

Logging is activated.

The device places an entry in the log file when it applies the *Double NAT* rule to a data packet.

unmarked (default setting)

Logging is deactivated.

Trap

Activates/deactivates the sending of SNMP traps when the device applies a *Double NAT* rule to a data packet.

Possible values:

marked

The sending of SNMP traps is active. The prerequisite is that in the *Diagnostics > Status Configuration > Alarms (Traps)* dialog the *Alarms (Traps)* function is enabled and at least one trap destination is specified.

If the device applies the *Double NAT* rule to a data packet, then the device sends an SNMP trap.

unmarked (default setting)

The sending of SNMP traps is inactive.

Active

Activates/deactivates the *Double NAT* rule.

Possible values:

marked


The rule is active.

unmarked (default setting)

The rule is inactive.

7.9.5.2 Double NAT Mapping

[Routing > NAT > Double NAT > Mapping]

In this dialog, you assign the *Double NAT* rules to a router interface. To do this, click the  button.

You add and edit the *Double NAT* rules in the [Routing > NAT > Double NAT > Rule](#).

An overview of which *Double NAT* rule is assigned to which router interfaces you find in the [Routing > NAT > Double NAT > Overview](#) dialog.

Table

For information on how to customize the appearance of the table, see [“Working with tables” on page 16](#).

Buttons

 Remove

Removes the selected table row.

 Assign

Opens the [Assign](#) window. In this window, you assign a set-up router interface to an existing *Double NAT* rule.

Port

Displays the number of the router interface on which the device applies the *Double NAT* rule.

Rule index

Displays the sequential number of the *Double NAT* rule. See the *Index* column in the [Routing > NAT > Double NAT > Rule](#) dialog. You specify the index number when you add a table row.

Rule name

Displays the name of the *Double NAT* rule. See the *Rule name* column in the [Routing > NAT > Double NAT > Rule](#) dialog.

Direction

Displays if the device applies the *Double NAT* rule to data packets received or sent.

Possible values:

ingress


The device applies the *Double NAT* rule to data packets received on the router interface.

egress

The device applies the *Double NAT* rule to data packets sent on the router interface.

both

The device applies the *Double NAT* rule to data packets received or sent on the router interface.

You can change the value when you click the  button.

Priority

Specifies the priority of the *Double NAT* rule.

Using the priority, you specify the order in which the device applies several rules to the data stream. The device applies the rules in ascending order starting with priority 1.

Possible values:

1..6500 (default setting: 1)

Active

Activates/deactivates the *Double NAT* rule.

Possible values:

marked

The rule is active.

unmarked (default setting)

The rule is inactive.

7.9.5.3 Double NAT Overview

[[Routing](#) > [NAT](#) > [Double NAT](#) > [Overview](#)]

In this dialog, you will find an overview of which *Double NAT* rule is assigned to which router interface.

You add and edit the *Double NAT* rules in the [Routing > NAT > Double NAT > Rule](#).

You assign the router interfaces to the related *Double NAT* rule in the [Routing > NAT > Double NAT > Mapping](#) dialog.

Table

For information on how to customize the appearance of the table, see “[Working with tables](#)” on [page 16](#).

Port

Displays the number of the router interface on which the device applies the *Double NAT* rule.

Rule index

Displays the sequential number of the *Double NAT* rule. See the *Index* column in the [Routing > NAT > Double NAT > Rule](#) dialog.

Rule name

Displays the name of the *Double NAT* rule. See the *Rule name* column in the [Routing > NAT > Double NAT > Rule](#) dialog.

Local internal IP address

Displays the actual IP address for the device placed in the first network.

Local external IP address

Displays the virtual IP address in the second network for the device placed in the first network.

Remote internal IP address

Displays the actual IP address for the device placed in the second network.

Remote external IP address

Displays the virtual IP address in the first network for the device placed in the second network.

Trap

Displays if the device sends an SNMP trap when it applies the *Double NAT* rule to a data packet.

Possible values:

marked

The device sends an SNMP trap. The prerequisite is that in the *Diagnostics > Status Configuration > Alarms (Traps)* dialog the *Alarms (Traps)* function is enabled and at least one trap destination is specified.

unmarked

The device does not send an SNMP trap.

Log

Displays if the device places an entry in the log file when it applies the *Double NAT* rule to a data packet.

Possible values:

marked

When the device applies the *Double NAT* rule to a data packet, the device places an entry in the log file. See the *Diagnostics > Report > System Log* dialog.

unmarked

Logging is disabled.

Direction

Displays if the device applies the *Double NAT* rule to data packets received or sent.

Possible values:

ingress

The device applies the *Double NAT* rule to data packets received on the router interface.

egress

The device applies the *Double NAT* rule to data packets sent on the router interface.

both

The device applies the *Double NAT* rule to data packets received or sent on the router interface.

Priority

Displays the priority of the *Double NAT* rule.

The device applies rules to the data stream in ascending order starting with priority 1.

8 Diagnostics

The menu contains the following dialogs:

- Status Configuration
- System
- Syslog
- Ports
- LLDP
- Report

8.1 Status Configuration

[Diagnostics > Status Configuration]

The menu contains the following dialogs:

- Device Status
- Security Status
- Alarms (Traps)

8.1.1 Device Status

[Diagnostics > Status Configuration > Device Status]

The device status provides an overview of the overall condition of the device. Many process visualization systems record the device status for a device to present its condition in graphic form.

The device displays its current status as `error` or `ok` in the *Device status* frame. The device determines this status from the individual monitoring results.

The device displays detected faults in the *Status* tab and also in the *Basic Settings > System* dialog, *Device status* frame.

The dialog contains the following tabs:

[Global]

[Port]

[Status]

[Global]

Device status

Device status

Displays the current status of the device. The device determines the status from the individual monitored parameters.

Possible values:

`ok`

`error`

The device displays this value to indicate a detected error in one of the monitored parameters.

Traps

Send trap

Activates/deactivates the sending of SNMP traps when the device detects a change in a monitored function.

Possible values:

`marked` (default setting)

The sending of SNMP traps is active. The prerequisite is that in the *Diagnostics > Status Configuration > Alarms (Traps)* dialog the *Alarms (Traps)* function is enabled and at least one trap destination is specified.

If the device detects a change in the monitored functions, then the device sends an SNMP trap.

`unmarked`

The sending of SNMP traps is inactive.

Table

For information on how to customize the appearance of the table, see “Working with tables” on page 16.

Connection errors

Activates/deactivates the monitoring of the link status of the port/interface.

Possible values:

marked

Monitoring is active.

If the link interrupts on a monitored port/interface, then in the *Device status* frame, the value changes to *error*.

In the *Port* tab, you have the option of selecting the ports/interfaces to be monitored individually.

unmarked (default setting)

Monitoring is inactive.

Temperature

Activates/deactivates the monitoring of the temperature in the device.

Possible values:

marked (default setting)

Monitoring is active.

If the temperature exceeds the specified upper threshold value or falls below the specified lower threshold value, then in the *Device status* frame, the value changes to *error*.

unmarked

Monitoring is inactive.

You specify the temperature threshold values in the *Basic Settings > System* dialog, *Upper temp. limit [°C]* field and *Lower temp. limit [°C]* field.

External memory removal

Activates/deactivates the monitoring of the active external memory.

Possible values:

marked

Monitoring is active.

If you remove the active external memory from the device, then in the *Device status* frame, the value changes to *error*.

unmarked (default setting)

Monitoring is inactive.

You specify the active external memory in the *Basic Settings > Load/Save* dialog, *External memory* frame.

External memory not in sync

Activates/deactivates the monitoring of the configuration profile in the device and in the external memory.

Possible values:

`marked`

Monitoring is active.

In the *Device status* frame, the value changes to `error` in the following situations:

- The configuration profile only exists in the device.
- The configuration profile in the device differs from the configuration profile in the external memory.

`unmarked` (default setting)

Monitoring is inactive.

Power supply

Activates/deactivates the monitoring of the power supply unit.

Possible values:

`marked` (default setting)

Monitoring is active.

If the device has a detected power supply fault, then in the *Device status* frame, the value changes to `error`.

`unmarked`

Monitoring is inactive.

[Port]**Table**

For information on how to customize the appearance of the table, see [“Working with tables” on page 16](#).

Port

Displays the port number.

Propagate connection error

Activates/deactivates the monitoring of the link on the port/interface.

Possible values:

`marked`

Monitoring is active.

If the link on the selected port/interface is interrupted, then in the *Device status* frame, the value changes to `error`.

`unmarked` (default setting)

Monitoring is inactive.

This setting takes effect when you mark the *Connection errors* checkbox in the *Global* tab.

[Status]**Table**

For information on how to customize the appearance of the table, see [“Working with tables”](#) on page 16.

Timestamp

Displays the date and time of the event in the format, [Month Day, Year](#) [hh: mm ss AM/PM](#)

Cause

Displays the event which caused the SNMP trap.

8.1.2 Security Status

[Diagnostics > Status Configuration > Security Status]

This dialog gives you an overview of the status of the safety-relevant settings in the device.

The device displays its current status as *error* or *ok* in the *Security status* frame. The device determines this status from the individual monitoring results.

The device displays detected faults in the *Status* tab and also in the *Basic Settings > System* dialog, *Security status* frame.

The dialog contains the following tabs:

[Global]

[Port]

[Status]

[Global]

Security status

Security status

Displays the current status of the security-relevant settings in the device. The device determines the status from the individual monitored parameters.

Possible values:

ok

error

The device displays this value to indicate a detected error in one of the monitored parameters.

Traps

Send trap

Activates/deactivates the sending of SNMP traps when the device detects a change in a monitored function.

Possible values:

marked

The sending of SNMP traps is active. The prerequisite is that in the *Diagnostics > Status Configuration > Alarms (Traps)* dialog the *Alarms (Traps)* function is enabled and at least one trap destination is specified.

If the device detects a change in the monitored functions, then the device sends an SNMP trap.

unmarked (default setting)

The sending of SNMP traps is inactive.

Table

For information on how to customize the appearance of the table, see “Working with tables” on page 16.

Password default settings unchanged

Activates/deactivates the monitoring of the password for the locally set up user account `admin`.

Possible values:

`marked` (default setting)

Monitoring is active.

If the password is set to the default setting for the `admin` user account, then in the *Security status* frame, the value changes to `error`.

`unmarked`

Monitoring is inactive.

You set the password in the *Device Security > User Management* dialog.

Min. password length shorter than 8

Activates/deactivates the monitoring of the *Min. password length* policy.

Possible values:

`marked` (default setting)

Monitoring is active.

If the value for the *Min. password length* policy is less than 8, then in the *Security status* frame, the value changes to `error`.

`unmarked`

Monitoring is inactive.

You specify the *Min. password length* policy in the *Device Security > User Management* dialog in the *Configuration* frame.

Password policy settings deactivated

Activates/deactivates the monitoring of the Password policies settings.

Possible values:

`marked` (default setting)

Monitoring is active.

If the value for at least one of the following policies is less than 1, then in the *Security status* frame, the value changes to `error`.

– *Upper-case characters (min.)*

– *Lower-case characters (min.)*

– *Digits (min.)*

– *Special characters (min.)*

`unmarked`

Monitoring is inactive.

You specify the policy settings in the *Device Security > User Management* dialog in the *Password policy* frame.

User account password policy check deactivated

Activates/deactivates the monitoring of the *Policy check* function.

Possible values:

marked

Monitoring is active.

If the *Policy check* function is inactive for at least one user account, then in the *Security status* frame, the value changes to *error*.

unmarked (default setting)

Monitoring is inactive.

You activate the *Policy check* function in the *Device Security > User Management* dialog.

HTTP server active

Activates/deactivates the monitoring of the HTTP server.

Possible values:

marked (default setting)

Monitoring is active.

If you enable the HTTP server, then in the *Security status* frame, the value changes to *error*.

unmarked

Monitoring is inactive.

You enable/disable the HTTP server in the *Device Security > Management Access > Server* dialog, *HTTP* tab.

SNMP unencrypted

Activates/deactivates the monitoring of the SNMP server.

Possible values:

marked (default setting)

Monitoring is active.

If at least one of the following conditions applies, then in the *Security status* frame, the value changes to *error*:

- The *SNMPv1* function is enabled.
- The *SNMPv2* function is enabled.
- The encryption for *SNMPv3* is disabled.

You enable the encryption in the *Device Security > User Management* dialog, in the *SNMP encryption type* column.

unmarked

Monitoring is inactive.

You specify the settings for the SNMP agent in the *Device Security > Management Access > Server* dialog, *SNMP* tab.

Access to system monitor with serial interface possible

Activates/deactivates the monitoring of the system monitor.

When the system monitor is active, you have the possibility to change to the system monitor using a serial connection during the system startup.

Possible values:

`marked`

Monitoring is active.

If you activate the system monitor, then in the *Security status* frame, the value changes to `error`.

`unmarked` (default setting)

Monitoring is inactive.

You activate/deactivate the system monitor in the *Diagnostics > System > Selftest* dialog.

Saving the configuration profile on the external memory possible

Activates/deactivates the monitoring of the configuration profile in the external memory.

Possible values:

`marked`

Monitoring is active.

If you activate the saving of the configuration profile in the external memory, then in the *Security status* frame, the value changes to `error`.

`unmarked` (default setting)

Monitoring is inactive.

You activate/deactivate the saving of the configuration profile in the external memory in the *Basic Settings > External Memory* dialog.

Link interrupted on enabled device ports

Activates/deactivates the monitoring of the link on the active ports.

Possible values:

`marked`

Monitoring is active.

If the link interrupts on an active port, then in the *Security status* frame, the value changes to `error`. In the *Port* tab, you have the option of selecting the ports to be monitored individually.

`unmarked` (default setting)

Monitoring is inactive.

Access with HiDiscovery possible

Activates/deactivates the monitoring of the HiDiscovery function.

Possible values:

`marked` (default setting)

Monitoring is active.

If you enable the HiDiscovery function, then in the *Security status* frame, the value changes to `error`.

`unmarked`

Monitoring is inactive.

You enable/disable the HiDiscovery function in the *Basic Settings > Network > Global* dialog.

Load unencrypted config from external memory

Activates/deactivates the monitoring of loading unencrypted configuration profiles from the external memory.

Possible values:

`marked` (default setting)

Monitoring is active.

If the settings allow the device to load an unencrypted configuration profile from the external memory, then in the *Security status* frame, the value changes to `error`.

If the following preconditions are fulfilled, then the *Security status* frame in the *Basic Settings > System* dialog, displays an alarm.

- The configuration profile stored in the external memory is unencrypted.
and
- The *Config priority* column in the *Basic Settings > External Memory* dialog has the value `first` or `second`.

`unmarked`

Monitoring is inactive.

Self-signed HTTPS certificate present

Activates/deactivates the monitoring of the digital certificate of the HTTPS server.

Possible values:

`marked` (default setting)

Monitoring is active.

If the HTTPS server uses a self-generated digital certificate, then in the *Security status* frame, the value changes to `error`.

`unmarked`

Monitoring is inactive.

[Port]**Table**

For information on how to customize the appearance of the table, see [“Working with tables” on page 16](#).

Port

Displays the port number.

Link interrupted on enabled device ports

Activates/deactivates the monitoring of the link on the active ports.

Possible values:

`marked`

Monitoring is active.

If the port is enabled (*Basic Settings > Port* dialog, *Configuration* tab, *Port on* checkbox is `marked`) and the link is down on the port, then in the *Security status* frame, the value changes to `error`.

`unmarked` (default setting)

Monitoring is inactive.

This setting takes effect when you mark the *Link interrupted on enabled device ports* checkbox in the *Diagnostics > Status Configuration > Security Status* dialog, *Global* tab.

[Status]

Table

For information on how to customize the appearance of the table, see “Working with tables” on [page 16](#).

Timestamp

Displays the date and time of the event in the format, `Month Day, Year hh:mm:ss AM/PM`

Cause

Displays the event which caused the SNMP trap.

8.1.3 Alarms (Traps)

[Diagnostics > Status Configuration > Alarms (Traps)]

The device lets you send an SNMP trap in response to specific events.

You specify the events for which the device triggers an SNMP trap in the following dialogs:

- [Diagnostics > Status Configuration > Device Status](#)
- [Diagnostics > Status Configuration > Security Status](#)

When setting up loopback interfaces, the device uses the IP address of the first loopback interface as the source for the SNMP traps. Otherwise, the device uses the address of the device management.

The menu contains the following dialogs:

[Trap Destinations](#)

8.1.3.1 Trap Destinations

[Diagnostics > Status Configuration > Alarms (Traps) > Trap Destinations]

In this dialog, you specify the trap destinations to which the device sends SNMP traps.

Operation

Operation

Enables/disables sending SNMP traps.

Possible values:

- On** (default setting)
Sending SNMP traps is enabled.
- Off**
Sending SNMP traps is disabled.

SNMPv1/v2 trap community

Name

Specifies the community string that the device sends in each SNMPv1/v2 trap for authentication to the trap destination.

Possible values:

- Alphanumeric ASCII character string with 0..64 characters
- trap** (default setting)

Table

For information on how to customize the appearance of the table, see “Working with tables” on page 16.

Buttons



Add

Opens the [Create](#) window to add a table row. Thus, you set up a trap destination on the device.

- In the [Name](#) field, you specify a name for the trap destination.
Possible values:
Alphanumeric ASCII character string with 1..32 characters
- In the [Address](#) field, you specify the IP address and the port of the trap destination.
Possible values:
[<I Pv4 address>](#): [<por t >](#)
If you do not specify a port, then the device automatically adds port [162](#) to the trap destination.



Remove

Removes the selected table row.

Name

Displays the name you specified for the trap destination (trap host).

Address

Specifies the IP address and the port of the trap destination (trap host).

Possible values:

[<I Pv4 address>](#): [<por t >](#)

If you do not specify a port, then the device automatically adds port [162](#) to the trap destination.

Active

Activates/deactivates the sending of SNMP traps to the trap destination.

Possible values:

[marked](#) (default setting)

The sending of SNMP traps to this trap destination is active.

[unmarked](#)

The sending of SNMP traps to this trap destination is inactive.

8.2 System

[Diagnostics > System]

The menu contains the following dialogs:

- System Information
- Configuration Check
- ARP
- Selftest

8.21 System Information

[Diagnostics > System > System Information]

This dialog displays the current operating condition of individual components in the device. The displayed values are a snapshot; they represent the operating condition at the time the dialog was loaded to the page.

Buttons

 Save system information

Saves the HTML page on your PC using the web browser dialog.

8.2.2 Configuration Check

[Diagnostics > System > Configuration Check]

The device lets you compare the settings in the device with the settings in its neighboring devices. For this purpose, the device uses the information that it received from its neighboring devices through topology recognition (LLDP).

The dialog lists the detected deviations, which affect the performance of the communication between the device and the recognized neighboring devices.

Note: The dialog displays the devices detected as connected to the neighboring device as if they were directly connected to the device itself.

Configuration

Start configuration check...

Starts the check and updates the content of the table.

When the table remains empty, the configuration check was successful and the settings in the device are compatible with the settings in the detected neighboring devices.

Information



Error

Displays the number of **ERROR** level deviations that the device detected during the configuration check.



Warning

Displays the number of **WARNING** level deviations that the device detected during the configuration check.

If you have set up more than 39 VLANs in the device, then the dialog continuously displays a warning. The reason is the limited number of possible VLAN data sets in LLDP packets with a maximum length. The device compares the first 39 VLANs automatically. If you have set up 40 or more VLANs in the device, then check the congruence of the further VLANs manually, if necessary.




Information

Displays the number of **INFORMATION** level deviations that the device detected during the configuration check.

Table

For information on how to customize the appearance of the table, see [“Working with tables” on page 16](#).



Displays detailed information about the detected deviations in the area below the table row. To hide the detailed information again, click the  button. If you click the icon in the table header, you display or hide the detailed information for each table row.

ID

Displays the rule ID of the deviations having occurred. The dialog combines several deviations with the same rule ID under one rule ID.

Level

Displays the level of deviation between the settings in this device and the settings in the detected neighboring devices.

The device differentiates between the following access statuses:

- **INFORMATION**
The performance of the communication between the two devices is not impaired.
- **WARNING**
The performance of the communication between the two devices is possibly impaired.
- **ERROR**
The communication between the two devices is impaired.

Message

Displays a summary of the detected deviations.

8.23 ARP

[Diagnostics > System > ARP]

This dialog displays the MAC and IP addresses of the neighboring devices connected to the device management.

Table

For information on how to customize the appearance of the table, see [“Working with tables” on page 16](#).

Buttons

 Clear ARP table

Removes the dynamically set up addresses from the ARP table.

Port

Displays the port number.

IP address

Displays the IPv4 address of a neighboring device.

MAC address

Displays the MAC address of a neighboring device.

Last updated

Displays the time in seconds since the current settings of the entry were registered in the ARP table.

Type

Displays the type of the entry.

Possible values:

`static`

Static entry. When the ARP table is deleted, the device keeps the static entry.

`dynamic`

Dynamic entry. When the *Aging time [s]* has been exceeded and the device does not receive any data from this device during this time, the device deletes the dynamic entry.

Active

Displays that the ARP table contains the IP/MAC address assignment as an active entry.

8.24 Selftest

[Diagnostics > System > Selftest]

This dialog lets you do the following:

- Activate/deactivate the option of changing to the system monitor during the system startup.
- Specify how the device behaves in the case of a detected error.

Configuration

If the device does not detect any readable configuration profile when restarting, then the following settings block your access to the device permanently.

- [SysMon1 is available](#) checkbox is [unmarked](#).
- [Load default config on error](#) checkbox is [unmarked](#).

This is the case, for example, if the password of the configuration profile that you are loading differs from the password set in the device. To have the device unlocked again, contact your sales partner.

SysMon1 is available

Activates/deactivates the option of changing to the system monitor during the system startup.

Possible values:

[marked](#) (default setting)

The device lets you change to the system monitor during the system startup.

[unmarked](#)

The device starts without the option of changing to the system monitor.

Among other things, the system monitor lets you update the device software and to delete saved configuration profiles.

Load default config on error

Activates/deactivates the loading of the default settings if the device does not detect any readable configuration profile when restarting.

Possible values:

[marked](#) (default setting)

The device loads the default settings.

[unmarked](#)

The device interrupts the restart and stops. The access to the device management is possible only using the Command Line Interface through the serial interface.

To regain the access to the device through the network, open the system monitor and reset the settings. After the system startup, the device uses the default settings.

Table

In this table you specify how the device behaves in the case of a detected error.

For information on how to customize the appearance of the table, see [“Working with tables” on page 16](#).

Cause

Detected error causes to which the device reacts.

Possible values:

[task](#)

The device detects errors in the applications executed, for example if a task terminates or is not available.

[resource](#)

The device detects errors in the resources available, for example if the memory is becoming scarce.

[software](#)

The device detects software errors, for example error in the consistency check.

[hardware](#)

The device detects hardware errors, for example in the chip set.

Action

Specifies how the device behaves if the adjacent event occurs.

Possible values:

[logOnly](#)

The device registers the detected error in the log file. See the [Diagnostics > Report > System Log](#) dialog.

[sendTrap](#)

The device sends an SNMP trap.

The prerequisite is that in the [Diagnostics > Status Configuration > Alarms \(Traps\)](#) dialog the [Alarms \(Traps\)](#) function is enabled and at least one trap destination is specified.

[reboot](#) (default setting)

The device triggers a restart.

8.3 Syslog

[Diagnostics > Syslog]

The device lets you report selected events, independent of the severity of the event, to different syslog servers.

In this dialog, you specify the settings for this function and manage up to 8 syslog servers.

Operation

Operation

Enables/disables the sending of events to the syslog servers.

Possible values:

On

The sending of events is enabled.

The device sends the events specified in the table to the specified syslog servers.

Off (default setting)

The sending of events is disabled.

Table

For information on how to customize the appearance of the table, see [“Working with tables” on page 16](#).

Buttons



Add

Adds a table row.



Remove

Removes the selected table row.

Index

Displays the index number to which the table row relates. The device automatically assigns the value when you add a table row.

When you delete a table row, this leaves a gap in the numbering. When you add a table row, the device fills the first gap.

Possible values:

1..8

IP address

Specifies the IP address of the syslog server.

Possible values:

Valid IPv4 address (default setting: 0.0.0.0)

DNS name in the format <domain>. <tid> or <host>. <domain>. <tid>

The prerequisite is that you also enable the *Client* function in the *Advanced > DNS > Client > Global* dialog.

To establish an encrypted connection using a digital certificate, verify that the *Common Name* or *Subject Alternative Name* information in the digital certificate that you have transferred onto the device matches the value you specify here. Otherwise, the device will not be able to verify the identity of the server.

Destination UDP port

Specifies the UDP port on which the syslog server expects the log entries.

Possible values:

1..65535 (2¹⁶ - 1) (default setting: 514)

Min. severity

Specifies the minimum severity of the events. The device sends a log entry for events with this severity and with more urgent severities to the syslog server.

Possible values:

emergency

alert

critical

error

warning (default setting)

notice

informational

debug

Type

Specifies the type of the log entry transmitted by the device.

Possible values:

syslog (default setting)

audittrail

Active

Activates/deactivates the transmission of events to the syslog server.

Possible values:

`marked`

The device sends events to the syslog server.

`unmarked` (default setting)

The transmission of events to the syslog server is deactivated.

8.4 Ports

[Diagnostics > Ports]

The menu contains the following dialogs:

8.5 LLDP

[Diagnostics > LLDP]

The device lets you gather information about neighboring devices. For this, the device uses the Link Layer Discovery Protocol (LLDP). This information lets a network management station map the structure of the network.

This menu lets you set up the topology discovery and to display the information received in tabular form.

The menu contains the following dialogs:

[LLDP Configuration](#)

[LLDP Topology Discovery](#)

8.5.1 LLDP Configuration

[Diagnostics > LLDP > Configuration]

This dialog lets you set up the topology discovery for every port.

Operation

Operation

Enables/disables the *LLDP* function.

Possible values:

On (default setting)

The *LLDP* function is enabled.

The topology discovery using LLDP is active in the device.

Off

The *LLDP* function is disabled.

Configuration

Transmit interval [s]

Specifies the interval in seconds at which the device sends LLDP data packets.

Possible values:

5 . 32768 (2¹⁶) (default setting: 30)

Transmit interval multiplier

Specifies the factor for determining the time-to-live value for the LLDP data packets.

Possible values:

2 . 10 (default setting: 4)

The time-to-live value coded in the LLDP header results from multiplying this value with the value in the *Transmit interval [s]* field.

Reinit delay [s]

Displays the delay in seconds for the reinitialization of a port.

If in the *Operation* column the value **Off** is specified, then the device tries to reinitialize the port after the time specified here has elapsed.

Transmit delay [s]

Displays the delay in seconds for transmitting successive LLDP data packets after the device settings change.

Notification interval [s]

Specifies the interval in seconds for transmitting LLDP notifications.

Possible values:

5 . 3600 (default setting: 5)

After transmitting a notification trap, the device waits for a minimum of the time specified here before transmitting the next notification trap.

Table

For information on how to customize the appearance of the table, see [“Working with tables” on page 16](#).

Port

Displays the port number.

Operation

Specifies if the port transmits LLDP data packets.

Possible values:

`transmit`

The port sends LLDP data packets but does not save any information about neighboring devices.

`receive`

The port receives LLDP data packets but does not send any information to neighboring devices.

`receive and transmit` (default setting)

The port transmits LLDP data packets and saves information about neighboring devices.

`disabled`

The port does not send LLDP data packets and does not save information about neighboring devices.

Notification

Activates/deactivates the LLDP notifications on the port.

Possible values:

`marked`

LLDP notifications are active on the port.

`unmarked` (default setting)

LLDP notifications are inactive on the port.

Transmit port description

Activates/deactivates the transmitting of a TLV (Type Length Value) with the port description.

Possible values:

`marked` (default setting)

The transmitting of the TLV is active.

The device sends the TLV with the port description.

`unmarked`

The transmitting of the TLV is inactive.

The device does not send a TLV with the port description.

Transmit system name

Activates/deactivates the transmitting of a TLV (Type Length Value) with the device name.

Possible values:

`marked` (default setting)

The transmitting of the TLV is active.

The device sends the TLV with the device name.

`unmarked`

The transmitting of the TLV is inactive.

The device does not send a TLV with the device name.

Transmit system description

Activates/deactivates the transmitting of the TLV (Type Length Value) with the system description.

Possible values:

`marked` (default setting)

The transmitting of the TLV is active.

The device sends the TLV with the system description.

`unmarked`

The transmitting of the TLV is inactive.

The device does not send a TLV with the system description.

Transmit system capabilities

Activates/deactivates the transmitting of the TLV (Type Length Value) with the system capabilities.

Possible values:

`marked` (default setting)

The transmitting of the TLV is active.

The device sends the TLV with the system capabilities.

`unmarked`

The transmitting of the TLV is inactive.

The device does not send a TLV with the system capabilities.

Neighbors (max.)

Limits the number of neighboring devices to be recorded for this port.

Possible values:

1..50 (default setting: 10)

FDB mode

Specifies which function the device uses to record neighboring devices on this port.

Possible values:

[llDpOnly](#)

The device uses only LLDP data packets to record neighboring devices on this port.

[macOnly](#)

The device uses learned MAC addresses to record neighboring devices on this port. The device uses the MAC address only if there is no other entry in the MAC address table (forwarding database) for this port.

[both](#)

The device uses LLDP data packets and learned MAC addresses to record neighboring devices on this port.

[autoDetect](#) (default setting)

If the device receives LLDP data packets at this port, then the device operates the same as with the [llDpOnly](#) setting. Otherwise, the device operates the same as with the [macOnly](#) setting.

8.5.2 LLDP Topology Discovery

[Diagnostics > LLDP > Topology Discovery]

Devices in networks send notifications in the form of packets which are also known as "LLDPDU" (LLDP data units). The data that is sent and received through LLDPDUs is useful for many reasons. Thus the device detects which devices in the network are neighbors and through which ports they are connected.

The dialog lets you display the network and to detect the connected devices along with their specific features.

This dialog displays the collected LLDP information for the neighboring devices. This information lets a network management station map the structure of the network.

When devices both with and without an active topology discovery function are connected to a port, the topology table hides the devices without active topology discovery.

When only devices without active topology discovery are connected to a port, the table contains one line for this port to represent every device. This line contains the number of connected devices.

The MAC address table (forwarding database) contains MAC addresses of devices that the topology table hides for the sake of clarity.

When you use one port to connect several devices, for example through a hub, the table shows one line for each connected device.

Table

For information on how to customize the appearance of the table, see ["Working with tables" on page 16](#).

Port

Displays the port number.

Neighbor identifier

Displays the chassis ID of the neighboring device. This can be the basis MAC address of the neighboring device, for example.

FDB

Displays if the connected device has active LLDP support.

Possible values:

marked

The connected device does not have active LLDP support.

The device uses information from its MAC address table (forwarding database)

unmarked

The connected device has active LLDP support.

Neighbor address

Displays the IPv4 address or hostname with which the access to the neighboring device management is possible.

Neighbor IPv6 address

Displays the IPv6 address with which the access to the neighboring device management is possible.

Neighbor port description

Displays a description for the port of the neighboring device.

Neighbor system name

Displays the device name of the neighboring device.

Neighbor system description

Displays a description for the neighboring device.

Port ID

Displays the ID of the port through which the neighboring device is connected to the device.

Autonegotiation supported

Displays if the port of the neighboring device supports auto-negotiation.

Autonegotiation

Displays if auto-negotiation is active on the port of the neighboring device.

PoE supported

Displays if the port of the neighboring device supports Power over Ethernet (PoE).

PoE enabled

Displays if Power over Ethernet (PoE) is active on the port of the neighboring device.

8.6 Report

[Diagnostics > Report]

The menu contains the following dialogs:

- [Report Global](#)
- [Persistent Logging](#)
- [System Log](#)
- [Audit Trail](#)

8.6.1 Report Global

[Diagnostics > Report > Global]

The device lets you log specific events using the following outputs:

- on the console
- on one or more syslog servers
- on a connection to the Command Line Interface set up using SSH

In this dialog, you specify the required settings. By assigning the severity you specify which events the device registers.

The dialog lets you save a ZIP archive with detailed device information for support purposes on your PC.

Console logging

Buttons

 Download support information

Generates a ZIP archive which the web browser lets you download from the device.

The ZIP archive contains files with detailed device information for support purposes. For further information, see [“Support Information: Files in ZIP archive” on page 446](#).

Operation

Enables/disables the *Console logging* function.

Possible values:

On

The *Console logging* function is enabled.
The device logs the events on the console.

Off (default setting)

The *Console logging* function is disabled.

Severity

Specifies the minimum severity for the events. The device logs events with this severity and with more urgent severities. For further information, see [“Meaning of the event severities” on page 446](#).

The device outputs the messages on the serial interface.

Possible values:

emergency

alert

critical

error

warning (default setting)

notice

informational
debug

SNMP logging

When you enable the logging of SNMP requests, the device sends these as events with the preset severity [notice](#) to the list of syslog servers. The preset minimum severity for a syslog server entry is [critical](#).

To send SNMP requests to a syslog server, you have a number of options to change the default settings. Select the ones that meet your requirements best.

Set the severity for which the device generates SNMP requests as events to [warning](#) or [error](#). Change the minimum severity for a syslog entry for one or more syslog servers to the same value.

You also have the option of adding a separate syslog server entry for this.

Set only the severity for SNMP requests to [critical](#) or higher. The device then sends SNMP requests as events with the severity [critical](#) or higher to the syslog servers.

Set only the minimum severity for one or more syslog server entries to [notice](#) or lower. Then it is possible that the device sends many events to the syslog servers.

Log SNMP get request

Enables/disables the logging for the reception of *SNMP Get requests*.

Possible values:

[On](#)

The logging is enabled.

The device logs each received *SNMP Get request* as an event in the syslog.

From the [Severity get request](#) drop-down list, you select the severity for this event.

[Off](#) (default setting)

The logging is disabled.

Log SNMP set request

Enables/disables the logging for the reception of *SNMP Set requests*.

Possible values:

[On](#)

The logging is enabled.

The device logs each received *SNMP Set request* as an event in the syslog.

From the [Severity set request](#) drop-down list, you select the severity for this event.

[Off](#) (default setting)

The logging is disabled.

Severity get request

Specifies the severity of the event that the device logs for received *SNMP Get requests*. For further information, see [“Meaning of the event severities” on page 446](#).

Possible values:

[emergency](#)

[alert](#)

[critical](#)

error
warning
notice (default setting)
informational
debug

Severity set request

Specifies the severity of the event that the device logs for received *SNMP Set requests*. For further information, see [“Meaning of the event severities” on page 446](#).

Possible values:

emergency
alert
critical
error
warning
notice (default setting)
informational
debug

Buffered logging

The device buffers logged events in 2 separate storage areas so that the log entries for urgent events are kept.

This dialog lets you specify the minimum severity for events that the device buffers in the storage area with a higher priority.

Severity

Specifies the minimum severity for the events. The device buffers log entries for events with this severity and with more urgent severities in the storage area with a higher priority. For further information, see [“Meaning of the event severities” on page 446](#).

Possible values:

emergency
alert
critical
error
warning (default setting)
notice
informational
debug

CLI logging

Operation

Enables/disables the *CLI logging* function.

Possible values:

On

The *CLI logging* function is enabled.

The device logs every command received using the Command Line Interface.

Off (default setting)

The *CLI logging* function is disabled.

Support Information: Files in ZIP archive

File name	Format	Comments
audittrail.html	HTML	Contains the chronological recording of the system events and saved user changes in the <i>Audit Trail</i> protocol.
config.xml	XML	Contains the settings of the device saved in the "Selected" configuration profile.
defaultconfig.xml	XML	Contains the default settings of the device.
script	TEXT	Contains the output of the command <code>show running-config script</code> .
runningconfig.xml	XML	Contains the current operating settings of the device.
supportinfo.html	HTML	Contains device internal service information.
systeminfo.html	HTML	Contains information about the current settings and operating parameters.
systemlog.html	HTML	Contains the logged events in the Log file. See the Diagnostics > Report > System Log dialog.

Meaning of the event severities

Severity	Meaning
emergency	Device not ready for operation
alert	Immediate user intervention required
critical	Critical status
error	Error status
warning	Warning
notice	Significant, normal status
informational	Information message
debug	Debug message

8.6.2 Persistent Logging

[Diagnostics > Report > Persistent Logging]

The device lets you save log entries permanently in a file in the external memory. Therefore, even after the device is restarted you have access to the log entries.

In this dialog, you limit the size of the log file and specify the minimum severity for the events to be saved. When the log file reaches the specified size, the device archives this file and saves the following log entries in a newly generated file.

In the table the device displays you the log files held in the external memory. As soon as the specified maximum number of files has been attained, the device deletes the oldest file and renames the remaining files. This helps ensure that there is enough memory space in the external memory.

Note: Verify that an external memory is connected. To verify if an external memory is connected, see the *Status* column in the *Basic Settings > External Memory* dialog. We recommend to monitor the external memory connection using the *Device Status* function, see the *External memory removal* parameter in the *Diagnostics > Status Configuration > Device Status* dialog.

Operation

Operation

Enables/disables the *Persistent Logging* function.

Only activate this function if the external memory is available in the device.

Possible values:

On (default setting)

The *Persistent Logging* function is enabled.

The device saves the log entries in a file in the external memory.

Off

The *Persistent Logging* function is disabled.

Configuration

Max. file size [kbyte]

Specifies the maximum size of the log file in KBytes. When the log file reaches the specified size, the device archives this file and saves the following log entries in a newly generated file.

Possible values:

0 . 4096 (default setting: **1024**)

The value **0** deactivates saving of log entries in the log file.

Files (max.)

Specifies the number of log files that the device keeps in the external memory.

As soon as the specified maximum number of files has been attained, the device deletes the oldest file and renames the remaining files.

Possible values:

0 . 25 (default setting: 4)

The value 0 deactivates saving of log entries in the log file.

Severity

Specifies the minimum severity of the events. The device saves the log entry for events with this severity and with more urgent severities in the log file in the external memory.

Possible values:

emergency
alert
critical
error
warning (default setting)
notice
informational
debug

Log file target

Specifies the external memory device for logging.

Possible values:

sd (default setting)
External SD memory (ACA31)
usb
External USB memory (ACA21/ACA22)

Table

For information on how to customize the appearance of the table, see [“Working with tables” on page 16](#).

Buttons



Clear persistent log file

Removes the log files from the external memory.

Index

Displays the index number to which the table row relates.

Possible values:

1..25

The device automatically assigns this number.

File name

Displays the file name of the log file in the external memory.

Possible values:

messages

messages.X

File size [byte]

Displays the size of the log file in the external memory in bytes.

8.6.3 System Log

[Diagnostics > Report > System Log]

This dialog displays the System Log file. The device logs device-internal events in the System Log file. The device keeps the logged events even after a restart.

To search the System Log file, use the search function of your web browser.

The dialog lets you download a copy of the System Log file onto your computer. The device provides the file to be downloaded in HTML or CSV format.

Buttons

 Save log file

Downloads a copy of the System Log file onto your computer, based on the web browser settings.

Possible values:

[CSV](#)

The device provides the file in CSV format.

[HTML](#)

The device provides the file in HTML format.

 Clear log file

Clears the System Log file on the device.

8.6.4 Audit Trail

[Diagnostics > Report > Audit Trail]

This dialog displays the Audit Trail. The dialog lets you save the log file as an HTML file on your PC.

To search the log file for search terms, use the search function of your web browser.

The device logs system events and writing user actions to the device. This lets you keep track of WHO changes WHAT in the device and WHEN. The prerequisite is that the access role [auditor](#) or [administrator](#) is assigned to your user account.

The device logs the following user actions, among others:

- A user logging into the device management with the Command Line Interface (local or remote)
- A user logging off manually
- Automatic logging off of a user in the Command Line Interface after a specified period of inactivity
- Device restart
- Locking of a user account due to too many consecutive unsuccessful login attempts
- Locking of the access to the device management due to unsuccessful login attempts
- Commands executed in the Command Line Interface, apart from `showcommands`
- Changes to configuration variables
- Changes to the system time
- File transfer operations, including device software updates
- Configuration changes using HiDiscovery
- Device software updates and automatic configuration of the device through the external memory
- Opening and closing of SNMP through an HTTPS tunnel

The device does not log passwords. The logged entries are write-protected and remain saved in the device after a restart.

Note: During the system startup, access to the system monitor is possible using the default settings of the device. If an attacker gains physical access to the device, then he is able to reset the device settings to its default values using the system monitor. After this, the device and log file are accessible using the standard password. Take appropriate measures to restrict physical access to the device. Otherwise, deactivate access to the system monitor. See the [Diagnostics > System > Selftest](#) dialog, [SysMon1 is available](#) checkbox.

Buttons

 Save audit trail file

Saves the HTML page on your PC using the web browser dialog.

9 Advanced

The menu contains the following dialogs:

- [DNS](#)
- [Tracking](#)
- [Command Line Interface](#)

9.1 DNS

[Advanced > DNS]

The menu contains the following dialogs:

- [DNS Client](#)
- [DNS Cache](#)

9.1.1 DNS Client

[Advanced > DNS > Client]

DNS (Domain Name System) is a service in the network that translates hostnames into IP addresses. This name resolution lets you contact other devices using their hostnames instead of their IP addresses.

Using the *Client* function the device sends requests for resolving hostnames in IP addresses to a DNS server.

The menu contains the following dialogs:

- [DNS Client Global](#)
- [DNS Client Current](#)
- [DNS Client Static](#)

9.1.1.1 DNS Client Global

[Advanced > DNS > Client > Global]

In this dialog, you enable the *Client* function.

Operation

Operation

Enables/disables the *Client* function.

Possible values:

On

The *Client* function is enabled.

The device sends requests for resolving hostnames in IP addresses to a DNS server.

Off (default setting)

The *Client* function is disabled.

9.1.1.2 DNS Client Current

[Advanced > DNS > Client > Current]

This dialog displays to which DNS servers the device sends requests for resolving hostnames in IP addresses.

Table

For information on how to customize the appearance of the table, see [“Working with tables” on page 16](#).

Index

Displays the sequential number of the DNS server.

Address

Displays the IP address of the DNS server. The device forwards requests for resolving hostnames in IP addresses to the DNS server with this IP address.

9.1.1.3 DNS Client Static

[Advanced > DNS > Client > Static]

In this dialog, you specify the DNS servers to which the device forwards requests for resolving hostnames in IP addresses.

The device lets you specify up to 4 IP addresses.

Configuration

Source

Specifies the source from which the device obtains the IP address of DNS servers to which the device addresses requests.

Possible values:

[user](#)

The device uses the IP addresses specified in the table.

Table

For information on how to customize the appearance of the table, see [“Working with tables” on page 16](#).

Buttons



Add

Opens the [Create](#) window to add a table row.

In the [Index](#) field, you specify the index number.

Possible values:

- 1 . 4

The device lets you specify up to 4 external DNS servers.

In the [IP address](#) field, you specify the IP address of the DNS server.

Possible values:

- Valid IPv4 address



Remove

Removes the selected table row.

Index

Displays the sequential number of the DNS server. You specify the index number when you add a table row.

IP address

Specifies the IP address of the DNS server.

Possible values:

Valid IPv4 address

Active

Activates/deactivates the table row.

Prerequisites:

- In the [Advanced > DNS > Client > Global](#) dialog the *DNS client* function is enabled.
- In the [Configuration](#) frame, the item *user* is selected from the *Source* drop-down list.

Possible values:

marked (default setting)

The table row is active.

The device sends requests to the DNS server specified in the first active table row. When the device does not receive a response from this server, it sends the requests to the DNS server specified in the next active table row. The relevant timeout is specified in the [Configuration](#) frame, *Request timeout [s]* field.

unmarked

The table row is inactive.

The device does not send requests to this DNS server.

9.1.2 DNS Cache

[Advanced > DNS > Cache]

The *Cache* function lets the device respond to requests for resolving hostnames in IP addresses.

The menu contains the following dialogs:

[DNS Cache Global](#)

9.1.2.1 DNS Cache Global

[Advanced > DNS > Cache > Global]

In this dialog, you enable the *Cache* function. When the *Cache* function is enabled, the device operates as a Caching DNS server.

When a downstream device requests the IP address of an unknown hostname and the Caching DNS server finds a matching entry in its cache, the Caching DNS server returns the IP address.

The cache provides memory space for up to 128 hostnames with associated IP address.

Operation

Buttons

 Flush cache

Removes every entry from the DNS cache.

Operation

Enables/disables the *Cache* function.

Possible values:

- On** (default setting)
The *Cache* function is enabled.
- Off**
The *Cache* function is disabled.

9.2 Tracking


[Advanced > Tracking]

The tracking function lets you monitor what are known as tracking objects. Examples of monitored tracking objects are the link status of an interface or the reachability of a remote router or end device.

The device forwards status changes of the tracking objects to the registered applications, for example to the routing table or to a VRRP instance. The applications then react to the status changes:

- In the routing table, the device activates/deactivates the route linked to the tracking object.
- The VRRP instance linked to the tracking object reduces the priority of the virtual router so that a backup router takes over the role of the master.

If you set up the tracking objects in the [Advanced > Tracking > Configuration](#) dialog, then you can link applications with the tracking objects:

- You link static routes with a tracking object in the [Routing > Routing Table](#) dialog, [Track name](#) column.
- You link virtual routers with a tracking object in the [Routing > L3-Redundancy > VRRP > Tracking](#) dialog. Click the  button to open the [Create](#) window and select the tracking object from the [Track name](#) drop-down list.

The menu contains the following dialogs:

[Tracking Configuration](#)
[Tracking Applications](#)

9.21 Tracking Configuration

[Advanced > Tracking > Configuration]

In this dialog, you set up the tracking objects.

Table

For information on how to customize the appearance of the table, see [“Working with tables” on page 16](#).

Buttons



Add

Opens the [Create](#) window to add a table row.

- From the [Type](#) drop-down list, you select the type of the tracking object.
Possible values:
 - [i n t e r f a c e](#)
The device monitors the link status of its physical ports or of its link aggregation, LRE or VLAN router interface.
 - [p i n g](#)
The device monitors the route to a remote router or end device by sending periodic *ICMP echo request* packets.
 - [l o g i c a l](#)
The device monitors tracking objects logically linked to each other and thus enables complex monitoring tasks.
- In the [Track ID](#) field, you specify the identification number of the tracking object.
Possible values:
 - [1 . . 256](#)



Remove

Removes the selected table row.

Type

Specifies the type of the tracking object.

Possible values:

[i n t e r f a c e](#)

The device monitors the link status of its physical ports or of its link aggregation, LRE or VLAN router interface.

[p i n g](#)

The device monitors the route to a remote router or end device by sending periodic *ICMP echo request* packets.

[l o g i c a l](#)

The device monitors tracking objects logically linked to each other and thus enables complex monitoring tasks.

Track ID

Specifies the identification number of the tracking object.

Possible values:

1..256

This range is available to every type (interface, ping and logical).

Track name

Displays the name of the tracking object made up of the values displayed in the *Type* and *Track ID* columns.

Active

Activates/deactivates the monitoring of the tracking object.

Possible values:

marked

Monitoring is active. The device monitors the tracking object.

unmarked (default setting)

Monitoring is inactive.

Description

Specifies the description.

Here you describe what the device uses the tracking object for.

Possible values:

Alphanumeric ASCII character string with 0..255 characters

Status

Displays the monitoring result of the tracking object.

Possible values:

up

The monitoring result is positive:

- The link status is active.
- or
- The remote router or end device is reachable.
- or
- The result of the logical link is *TRUE*.

down

The monitoring result is negative:

- The link status is inactive.
- or
- The remote router or end device is not reachable.
- or
- The result of the logical link is *FALSE*.

not Ready

The monitoring of the tracking object is inactive. You activate the monitoring in the *Active* column.

Changes

Displays the number of status changes since the tracking object has been activated.

Last changed

Displays the time of the last status change.

Send trap

Activates/deactivates the sending of an SNMP trap when someone activates or deactivates the tracking object.

Possible values:

`marked`

If someone activates or deactivates the tracking object in the *Active* column, then the device sends an SNMP trap.

`unmarked` (default setting)

The device does not send an SNMP trap.

Port

Specifies the interface to be monitored for tracking objects of the *interface* type.

Possible values:

`<interface number>`

Number of the physical ports or of the link aggregation, LRE or VLAN router interface.

`no Port`

No tracking object of the *interface* type.

Link up delay [s]

Specifies the period in seconds after which the device evaluates the monitoring result as positive. If the link has been active on the interface for longer than the period specified here, then the *Status* column displays the value *up*.

Possible values:

`0 . 255`

`-`

No tracking object of the *logical* type.

Link down delay [s]

Specifies the period in seconds after which the device evaluates the monitoring result as negative. If the link has been inactive on the interface for longer than the period specified here, then the *Status* column displays the value *down*.

Possible values:

`0 . 255`

`-`

No tracking object of the *interface* type.

If the link to every aggregated port is interrupted, then Link aggregation, LRE and VLAN router interfaces have a negative monitoring result.

If the link to every physical port and link-aggregation interface which is a member of the VLAN is interrupted, then a VLAN router interface has a negative monitoring result.

Ping port

Specifies the router interface for tracking objects of the [ping](#) type through which the device sends the *ICMP echo request* packets.

Possible values:

- [<Interface number>](#)
Number of the router interface.
- [noName](#)
No router interface assigned.
- No tracking object of the [ping](#) type.

IP address

Specifies the IP address of the remote router or end device to be monitored.

Possible values:

- Valid IPv4 address
- No tracking object of the [ping](#) type.

Ping interval [ms]

Specifies the interval in milliseconds at which the device periodically sends *ICMP echo request* packets.

Possible values:

- [100 . 20000](#) (default setting: [1000](#))
If you specify a value [<1000](#), then you can set up a maximum of 16 tracking objects of the [ping](#) type.
- No tracking object of the [ping](#) type.

Ping replies to lose

Specifies the number of missed responses from the device after which the device evaluates the monitoring result as negative. If the device does not receive a response to its sent *ICMP echo request* packets for the number of times specified here, then the [Status](#) column displays the value [down](#).

Possible values:

- [1 . 10](#) (default setting: [3](#))
- No tracking object of the [ping](#) type.

Ping replies to receive

Specifies the number of received responses from the device after which the device evaluates the monitoring result as positive. If the device receives a response to its sent *ICMP echo request* packets for the number of times specified here, then the *Status* column displays the value *up*.

Possible values:

1 . 10 (default setting: 2)

-

No tracking object of the *ping* type.

Ping timeout [ms]

Specifies the period in milliseconds for which the device waits for a response. If the device does not receive a response within this period, then the device evaluates this as a missed response. See the *Ping replies to lose* column.

Possible values:

10 . 10000 (default setting: 100)

If a large number of ping tracking objects is set up in the device, then specify a sufficiently large value. If more than 100 instances are present, then specify at least 200 ms.

-

No tracking object of the *ping* type.

Ping TTL

Specifies the TTL value in the IP header with which the device sends the *ICMP echo request* packets.

TTL (Time To Live, also known as "Hop Count") identifies the maximum number of routing steps, which the sent *ICMP echo request* packet may traverse on its way from the sender to the receiver.

Possible values:

-

No tracking object of the *ping* type.

1 . 255 (default setting: 128)

Best route

Displays the number of the router interface through which the best route leads to the monitoring router or end device.

Possible values:

<Port number >

Number of the router interface.

no Port

No route exists.

-

No tracking object of the *ping* type.

Logical operand A

Specifies the first operand of the logical link for tracking objects of the **Logical** type.

Possible values:

Tracking objects set up

–

No tracking object of the **Logical** type.

Logical operand B

Specifies the second operand of the logical link for tracking objects of the **Logical** type.

Possible values:

Tracking objects set up

–

No tracking object of the **Logical** type.

Operator

Links the tracking objects specified in the *Logical operand A* and *Logical operand B* fields.

Possible values:

and

Logical AND link

or

Logical OR link

–


No tracking object of the **Logical** type.

9.2.2 Tracking Applications

[Advanced > Tracking > Applications]

In this dialog, you see which applications are linked with the tracking objects.

The following applications can be linked with tracking objects:

- You link static routes with a tracking object in the [Routing > Routing Table](#) dialog, *Track name* column.
- You link virtual routers with a tracking object in the [Routing > L3-Redundancy > VRRP > Tracking](#) dialog. Click the  button to open the *Create* window and select the tracking object from the *Track name* drop-down list.

Table

For information on how to customize the appearance of the table, see [“Working with tables” on page 16](#).

Type

Displays the type of the tracking object.

Track ID

Displays the identification number of the tracking object.

Application

Displays the name of the application that is linked with the tracking object.

Possible values:

- Tracking objects of the *logical* type
- Static routes
- Virtual router of a VRRP instance

Track name

Displays the name of the tracking object made up of the values displayed in the *Type* and *Track ID* columns.

9.3 Command Line Interface

[Advanced > CLI]

This dialog lets you access the device using the Command Line Interface.

Prerequisites:

- In the [Device Security > Management Access > Server](#) dialog, *SSH* tab the SSH server is enabled.
- On your workstation, install a SSH-capable client application which registers a handler for URLs starting with `ssh://` in your operating system.

Buttons

Open SSH connection

Opens the SSH-capable client application.

When you click the button, the web application passes the URL of the device starting with ssh: // and the user name of the currently logged in user.

If the web browser finds an SSH-capable client application, then the SSH-capable client establishes a connection to the device management using the SSH protocol.

A Index

0-9	
1to1 NAT	380
802.1D/p mapping	290
A	
Access restriction	90
Aging time	281
Alarm	423
ARP	302, 307
ARP table	49, 307, 430
Audit trail	451
Authentication list	66
C	
Certificate	20, 34, 71, 87, 88, 253, 422
CLI	94
Command line interface	94
Community names	96
Configuration check	428
Configuration profile	16, 31
Counter reset	49
D	
Daylight saving time	52
Deep Packet Inspection (DPI)	149
Default gateway	351, 361, 404
Default route	318, 319, 325, 404
Destination NAT	385
Device software	27
Device software backup	27
Device status	19, 414
DHCP L3 Relay	355
Digital certificate	20, 34, 71, 88, 253, 422
DNP3 enforcer	159
DNS	453
DNS cache	457
DNS client	454
Domain name system	453
DoS	240
Double NAT	404
DPI	149
DPI DNP3 enforcer	159
DPI Modbus enforcer	150
DPI OPC enforcer	156
E	
Egress rate limiter	283
Encryption	31
ENVM	29, 35, 41, 415, 421, 448
Event severity	446
External memory	23, 29, 35, 41, 448

F	
FAQ	473
FDB (MAC address table)	49, 285
Filter MAC addresses	285
Fingerprint	83, 87
Firewall learning mode	116
Firewall table	50
Flash memory	29
Flow control	281
H	
HiDiscovery	24, 421, 451
Host key	84
HTML	427, 450
HTTP	84
HTTP server	420
HTTPS	85
I	
ICMP redirect	297, 302
Industrial HiVision	9, 80
Ingress filtering	296
Ingress rate limiter	283
IP access restriction	90
L	
L3 Relay (DHCP)	355
LDAP	66
LLDP	436
Load/save	31
Log file	49, 50, 450
Login banner	95, 97
Loopback interface	359
M	
MAC address table (forwarding database)	49, 285
Management access	24, 90
Management VLAN	24
Modbus enforcer	150
N	
NAT	380, 404
NAT (Network Address Translation)	377
Network Address Translation (NAT)	377
Network time protocol	55
NTP	55
NVM	16, 29, 35
O	
OPC enforcer	156
OSPF	313

P	
Password	62, 419
Password length	62, 419
Persistent log file	50
Persistent logging	447
Port configuration	289
Port forwarding	385
Port priority	289
Port statistics	49
Port VLAN	295
Power supply	21, 416
Pre-Login banner	97
Priority queue	288
Proxy ARP	302
Q	
Queues	288
R	
RADIUS	66, 100
RAM	34
RAM self-test	431
Rate limiter	283
Reboot	49
Relay (DHCP)	355
Router interface	293, 300
Routing table	351
S	
Secure Shell (SSH)	81
Security status	20, 418
Self-test	431
Serial interface	420
Settings	31
Severity	446
SNMP server	79, 420
SNMP traps	47, 316, 362, 414, 418, 423, 462
SNMPv1/v2	96
Software backup	27
Software update	27
Source routing	297
SSH server	81
Stratum	55, 57
Support information	443
Support information (ZIP archive)	446
Syslog	433
System information	427
System log	450
System monitor	431
System time	51

T	
Technical questions	473
Temperature	21, 415
Threshold values network load	283
Time To Live (TTL)	299
Topology discovery	441
Tracking	375, 458
Training courses	473
Trap destination	424
Traps	47, 316, 362, 414, 418, 423, 462
Trust mode	289
TTL (Time To Live)	299
U	
Uptime	21
User administration	61
V	
Virtual local area network	291
Virtual router redundancy protocol	361
VLAN	24, 291
VLAN configuration	293
VLAN ports	295
VRRP	361
VRRP statistics	373
VRRP tracking	375
W	
Watchdog	31, 39
Web server	84, 85
Z	
ZIP archive with support information	446

B Technical support

Technical questions

For technical questions, please contact any Hirschmann dealer in your area or Hirschmann directly. You find the addresses of our partners on the Internet at www.belden.com.

For technical support, visit hirschmann-support.belden.com. This site also includes a free of charge knowledge base and a software download section.

Technical Documents

The current manuals and operating instructions for Hirschmann products are available at doc.hirschmann.com.

Customer Innovation Center

The Customer Innovation Center is ahead of its competitors on three counts with its complete range of innovative services:

Consulting incorporates comprehensive technical advice, from system evaluation through network planning to project planning.

Training offers you an introduction to the basics, product briefing and user training with certification. You find the training courses on technology and products currently available at www.belden.com/solutions/customer-innovation-center.

Support ranges from the first installation through the standby service to maintenance concepts.

With the Customer Innovation Center, you decide against any compromise in any case. Our client-customized package leaves you free to choose the service components you want to use.

C Readers' Comments

What is your opinion of this manual? We are constantly striving to provide as comprehensive a description of our product as possible, as well as important information to assist you in the operation of this product. Your comments and suggestions help us to further improve the quality of our documentation.

Your assessment of this manual:

	Very Good	Good	Satisfactory	Mediocre	Poor
Precise description	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Readability	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Understandability	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Examples	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Structure	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Comprehensive	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Graphics	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Drawings	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Tables	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Did you discover any errors in this manual?
If so, on what page?

Suggestions for improvement and additional information:

General comments:

Sender:

Company / Department:

Name / Telephone number:

Street:

Zip code / City:

E-mail:

Date / Signature:

Dear User,

Please fill out and return this page
as a fax to the number +49 (0)7127/14-1600 or
per mail to
Hirschmann Automation and Control GmbH
Department IRD-NT
Stuttgarter Str. 45-51
72654 Neckartenzlingen
Germany



HIRSCHMANN

A **BELDEN** BRAND







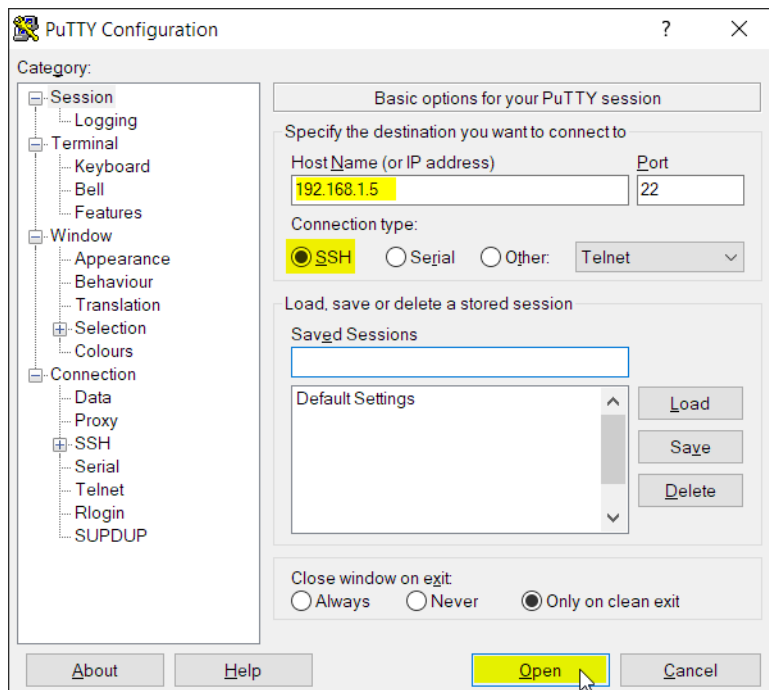


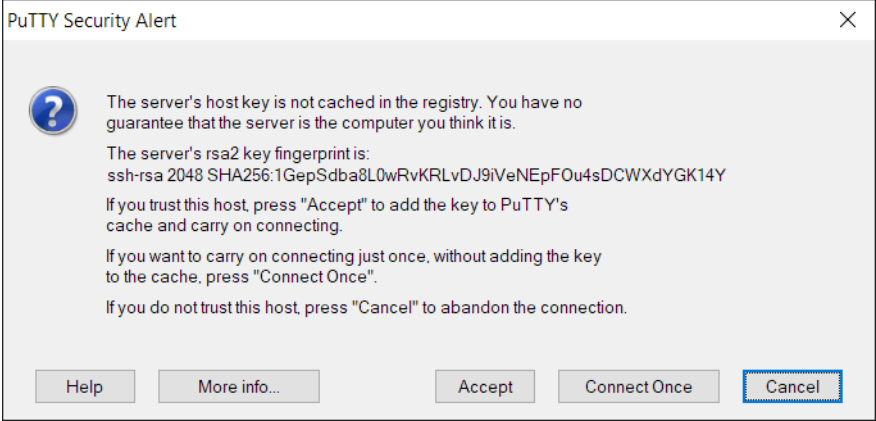


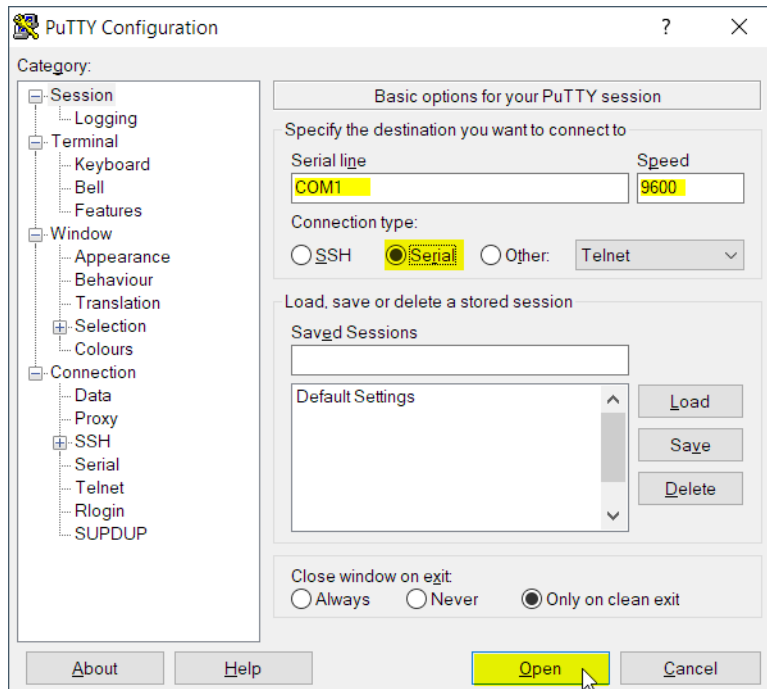


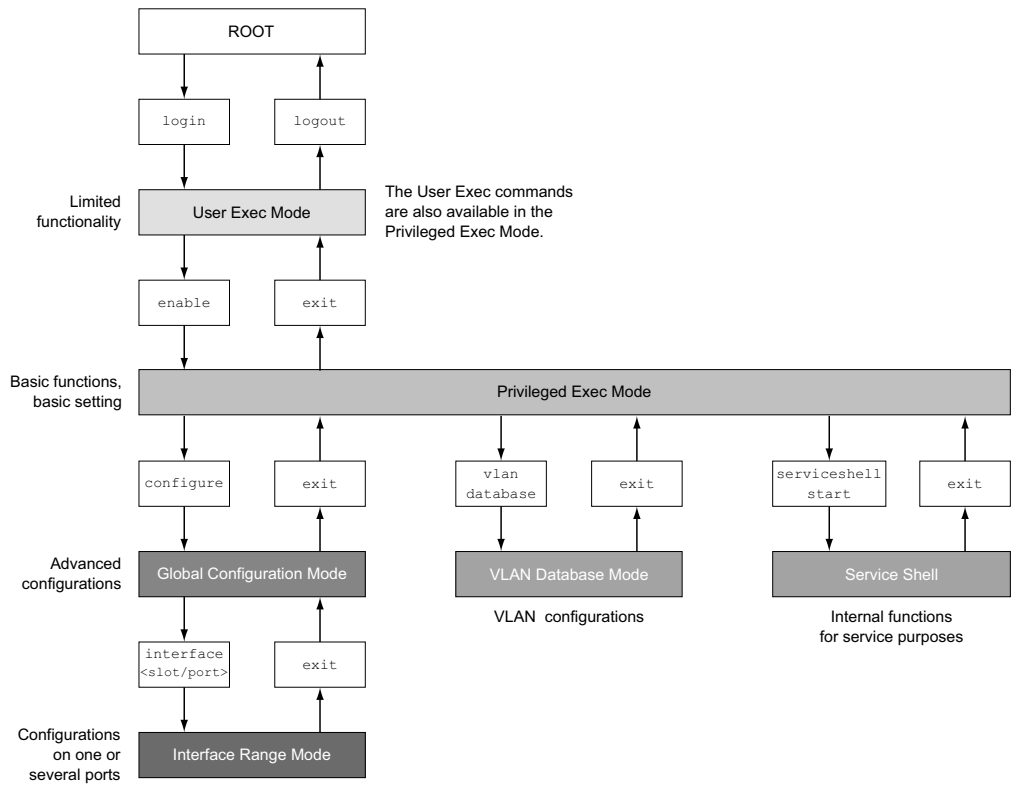


















[Blue shaded bar]





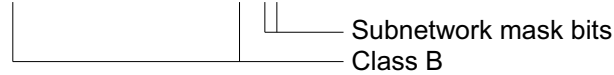




0	Net ID - 7 bits	Host ID - 24 bits	Class A
1 0	Net ID - 14 bits	Host ID - 16 bits	Class B
1 1 0	Net ID - 21 bits	Host ID - 8 bits	Class C
1 1 1 0	Multicast Group ID - 28 bits		Class D
1 1 1 1	reserved for future use - 28 bits		Class E

Decimal notation
255.255.192.0

Binary notation
11111111.11111111.11000000.00000000



Decimal notation

129.218.65.17

└─── 128 < 129 191 > Class B

Binary notation

10000001.11011010.01000001.00010001

└─── Subnetwork 1
└─── Network address

Decimal notation

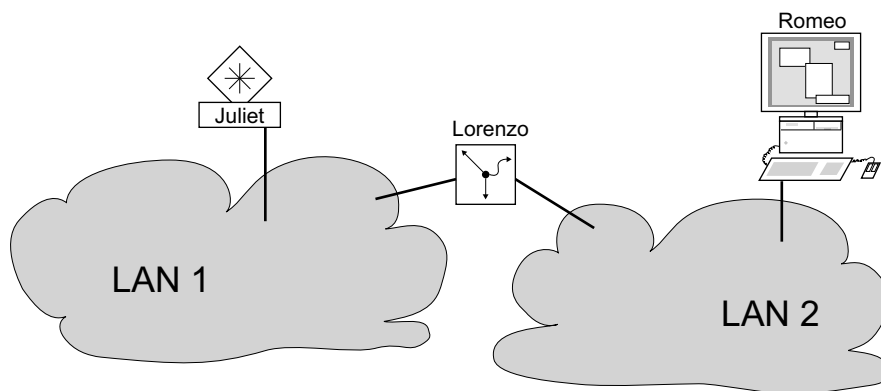
129.218.129.17

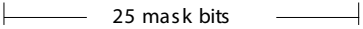

└─── 128 < 129 191 > Class B

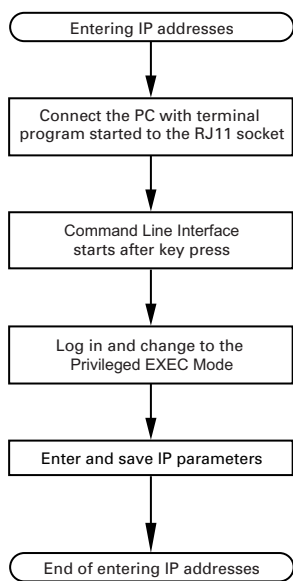
Binary notation

10000001.11011010.10000001.00010001

└─── Subnetwork 2
└─── Network address



IP address, decimal	Network mask, decimal	IP address, binary
192.168.112.1	255.255.255.128	11000000 10101000 01110000 00000001
192.168.112.127		11000000 10101000 01110000 01111111
		 25 mask bits
CIDR notation: 192.168.112.0/25		
	 Mask bits	



```
NOTE: Enter '?' for Command Help. Command help displays all opt
that are valid for the particular mode.
For the syntax of a particular command form, please
consult the documentation.
```

```
! ( )>
```




Id	MAC Address	Writable	IP Address	Net Mask	Default Gateway	Product	Name
1	00:80:63:A4:CC:00	<input checked="" type="checkbox"/>	10.115.0.76	255.255.224.0	10.115.0.3		
2	00:80:63:C0:50:00	<input type="checkbox"/>	10.115.0.33	255.255.224.0	10.115.0.3		
3	00:80:63:A3:40:00	<input type="checkbox"/>	10.115.0.70	255.255.224.0	10.115.0.3		
4	00:80:63:98:14:00	<input type="checkbox"/>	10.115.0.17	255.255.224.0	10.115.0.3		
5	00:80:63:96:E4:00	<input type="checkbox"/>	0.0.0.0	0.0.0.0	0.0.0.0		
6	00:80:63:46:00:06	<input checked="" type="checkbox"/>	192.168.2.181	255.255.255.0	192.168.2.1		
7	00:80:63:A3:40:40	<input type="checkbox"/>	10.115.0.59	255.255.224.0	10.115.0.3		
8	00:80:63:A4:CC:40	<input type="checkbox"/>	10.115.0.81	255.255.224.0	10.115.0.3		
9	00:80:63:6E:38:4E	<input checked="" type="checkbox"/>	192.168.2.174	255.255.255.0	192.168.2.1		
10	00:80:63:18:2A:61	<input checked="" type="checkbox"/>	192.168.2.170	255.255.255.0	192.168.2.1		
11	00:80:63:A3:40:80	<input type="checkbox"/>	10.115.0.66	255.255.224.0	10.115.0.3		
12	00:80:63:A4:CC:80	<input type="checkbox"/>	10.115.0.80	255.255.224.0	10.115.0.3		
13	00:80:63:61:AC:81	<input checked="" type="checkbox"/>	192.168.2.176	255.255.255.0	192.168.2.1		
14	00:80:63:98:10:95	<input type="checkbox"/>	10.115.0.22	255.255.224.0	10.115.0.3		
15	00:80:63:61:AC:AB	<input checked="" type="checkbox"/>	192.168.2.40	255.255.255.0	192.168.2.1		
16	00:80:63:3B:5C:BD	<input checked="" type="checkbox"/>	192.168.2.178	255.255.255.0	192.168.2.1		
17	00:80:63:A3:40:C0	<input type="checkbox"/>	10.115.0.72	255.255.224.0	10.115.0.3		
18	00:80:63:8F:2C:BE	<input type="checkbox"/>	10.115.0.40	255.255.224.0	10.115.0.3		
19	00:80:63:88:38:EC	<input checked="" type="checkbox"/>	192.168.110.92	255.255.255.0	0.0.0.0		
20	00:80:63:9B:11:00	<input type="checkbox"/>	10.115.0.35	255.255.224.0	10.115.0.3		
21	00:80:63:A4:CD:00	<input type="checkbox"/>	10.115.0.77	255.255.224.0	10.115.0.3		
22	00:80:63:99:41:08	<input type="checkbox"/>	10.115.0.13	255.255.224.0	10.115.0.3		
23	00:80:63:17:35:08	<input checked="" type="checkbox"/>	192.168.2.164	255.255.255.0	192.168.2.1		
24	00:80:63:44:19:2E	<input checked="" type="checkbox"/>	10.115.5.130	255.255.224.0	10.115.0.3		

Properties [X]

MAC Address: 00:80:63:A3:40:00

Name: Power Unit 1 Switch 2

IP Configuration

IP Address: . . .

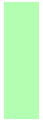
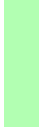
Net Mask: . . .

Default Gateway: . . .

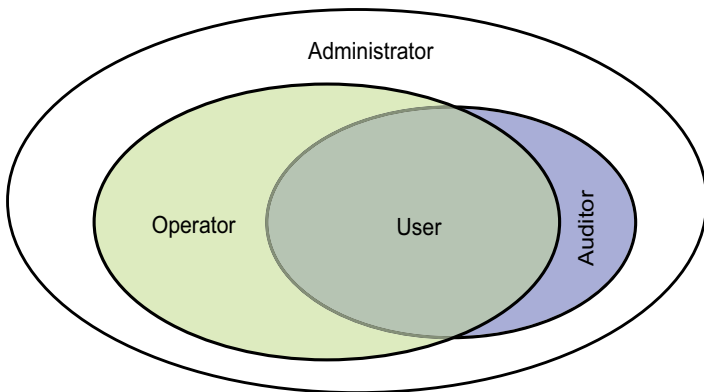


















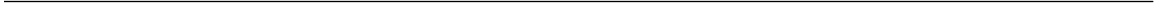
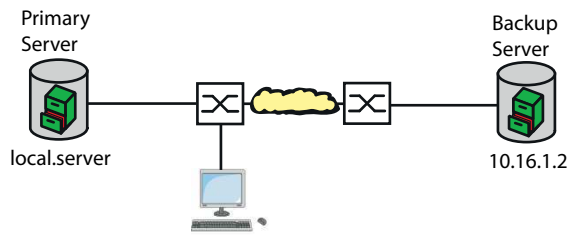


+





0
x





+

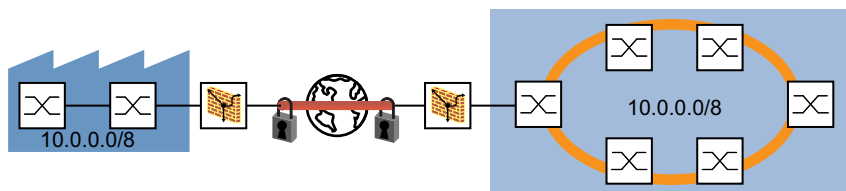
+

+

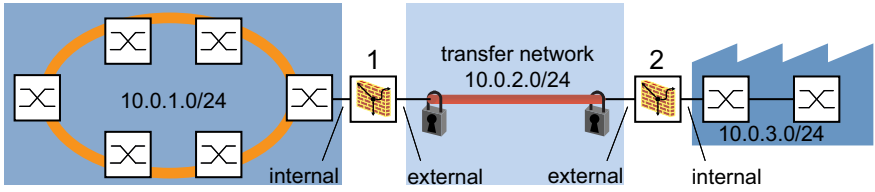














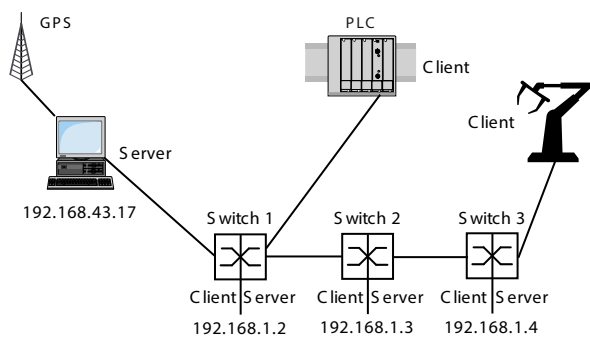














+



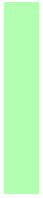


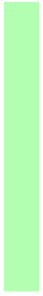
⌘+



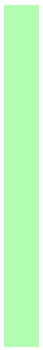








=















≡







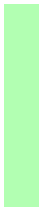




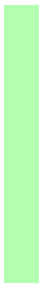




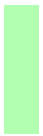
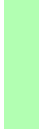
















BE+





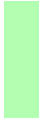
田+

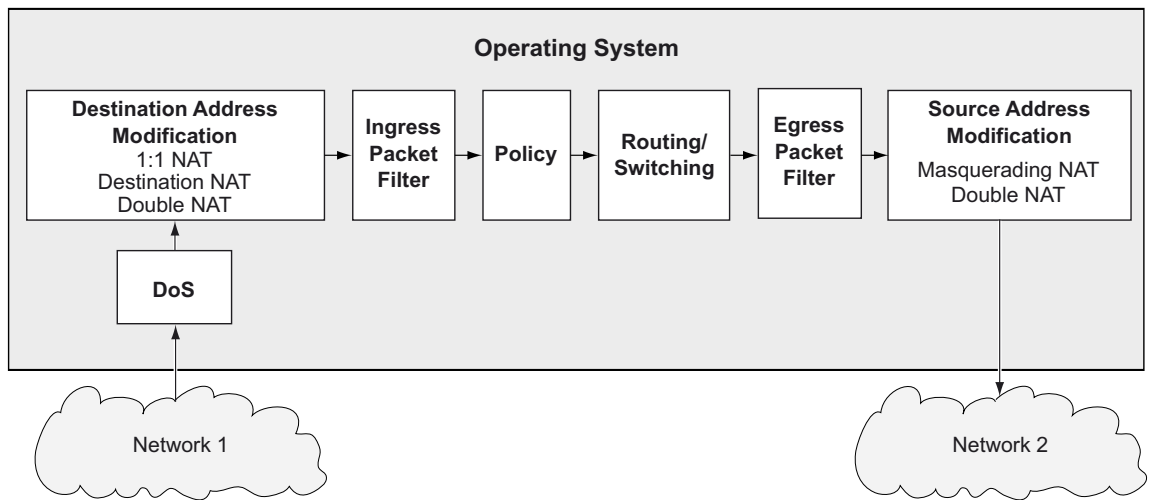
田+

✓





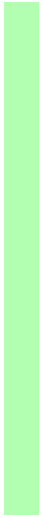






B+



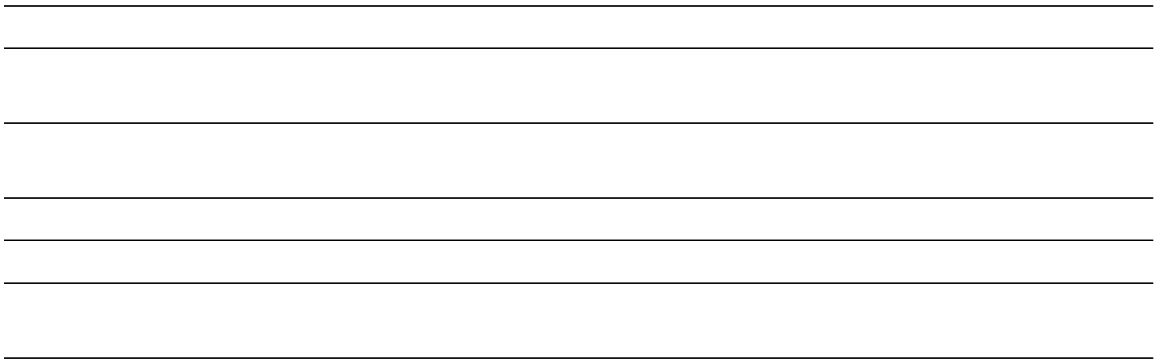




B+

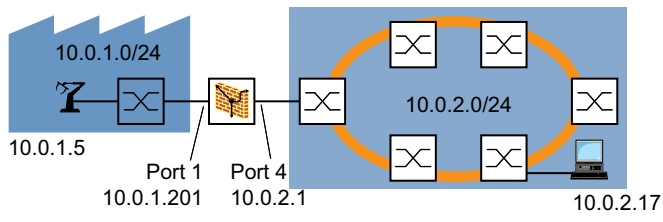








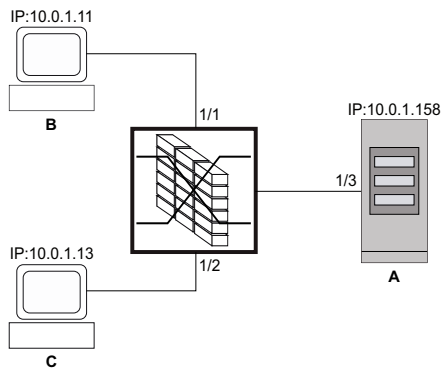










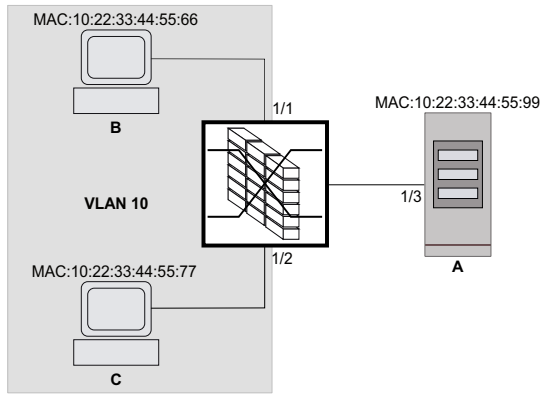






+









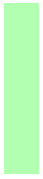
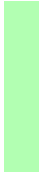
田+



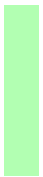
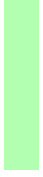




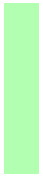
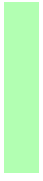


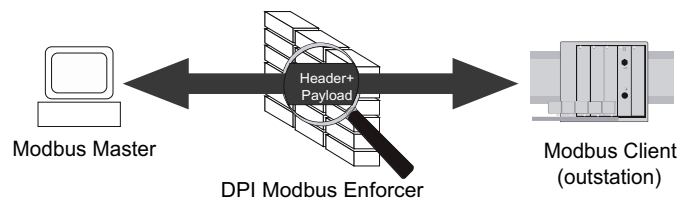










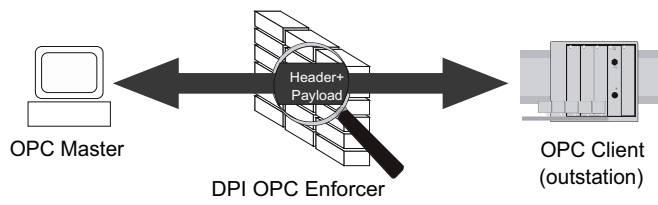




B+





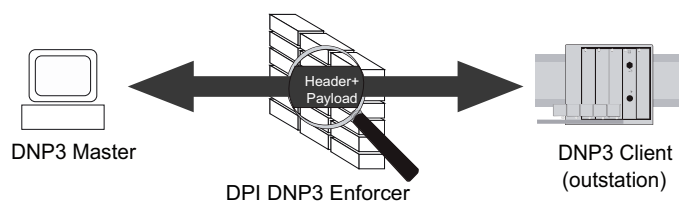




B+







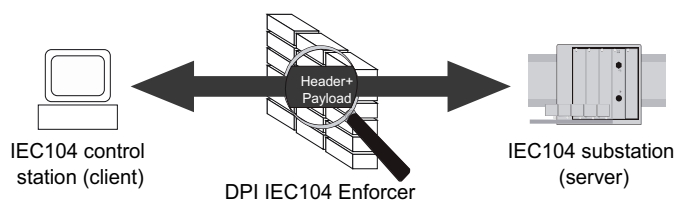


B+





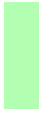


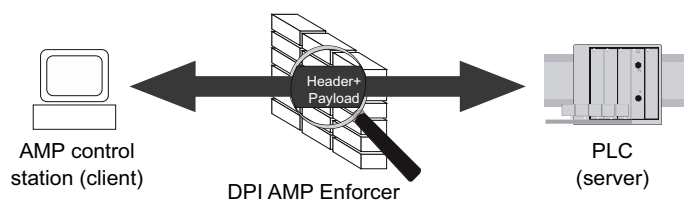




B+



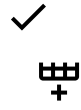
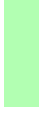




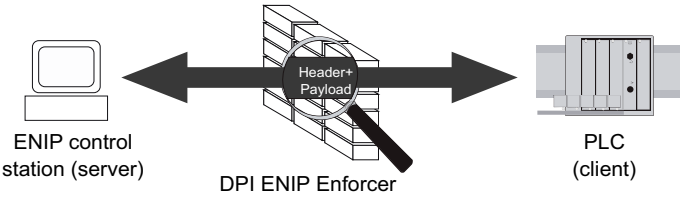


B+











⌘+

⌘+

✓

✓







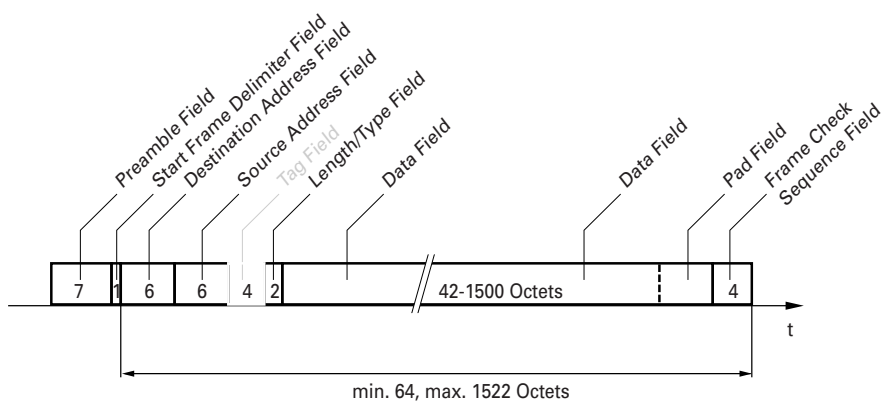
+

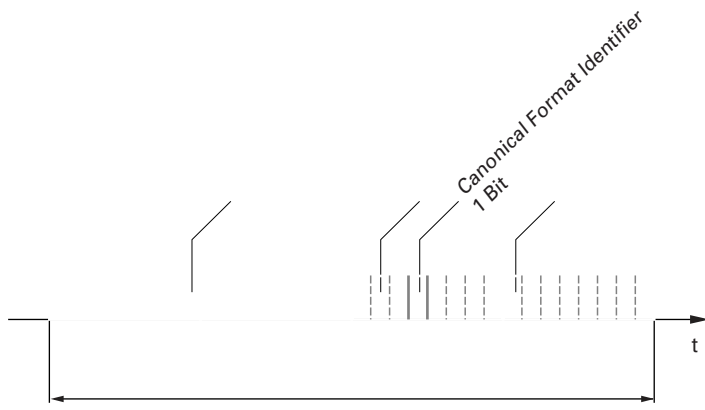


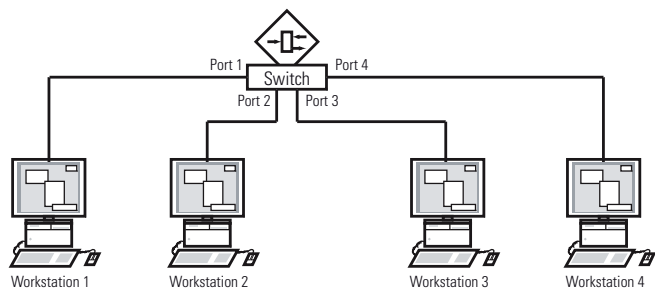






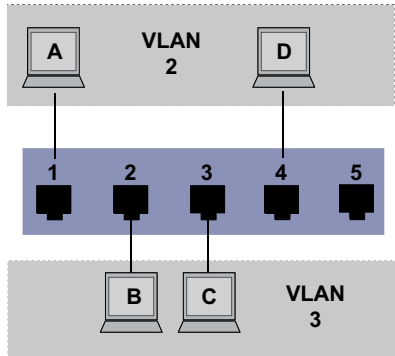










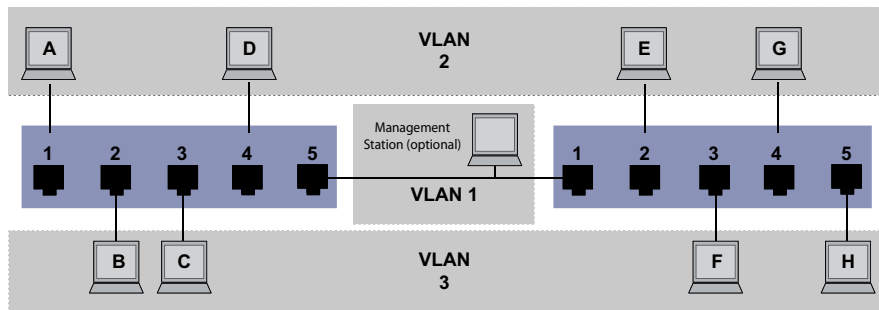




B+



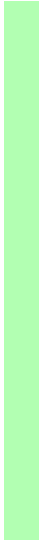




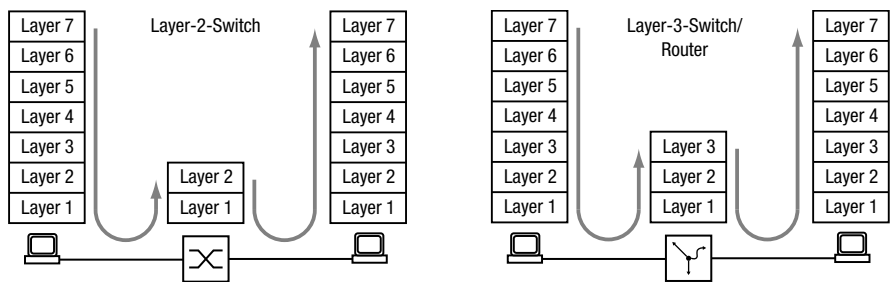


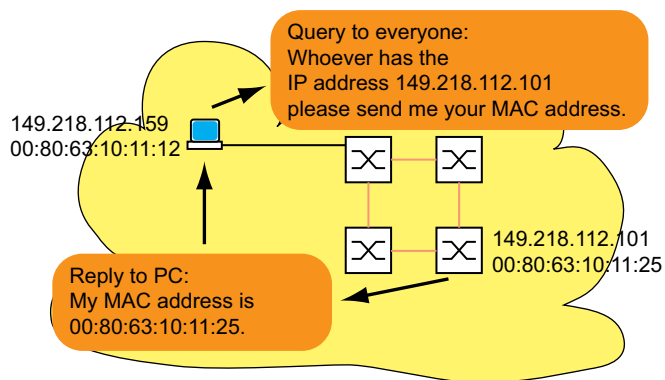


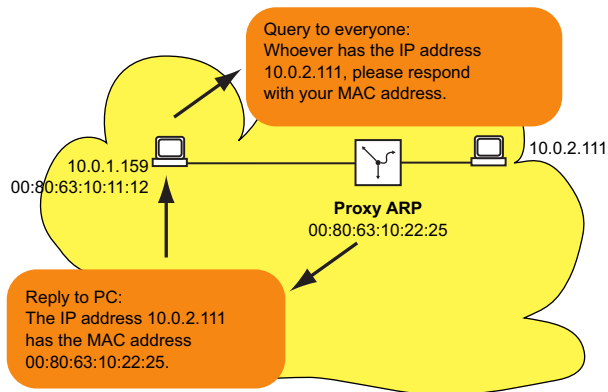
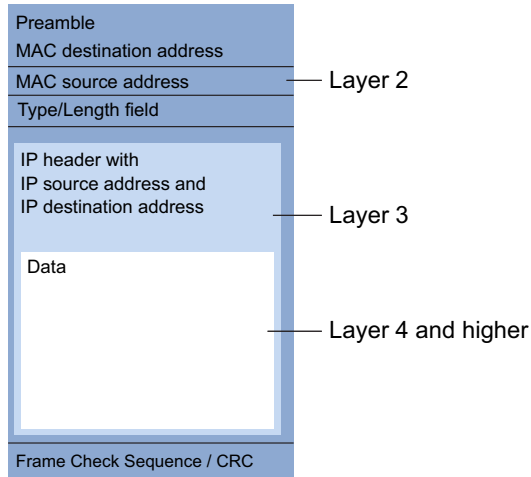




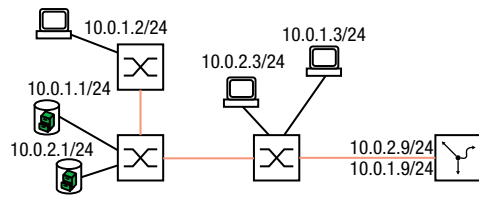


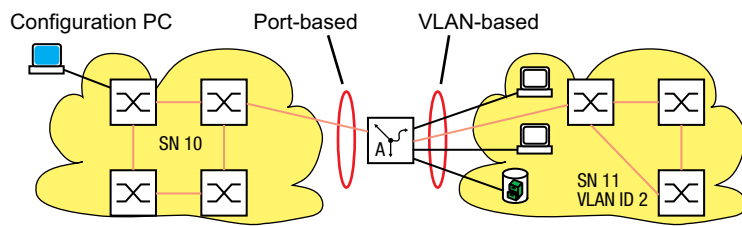


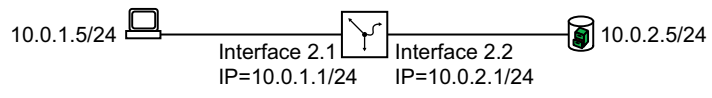


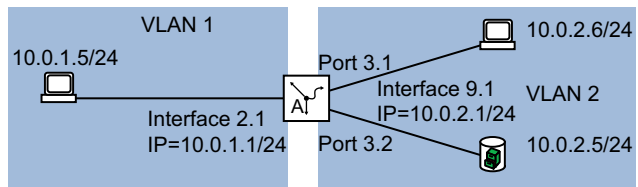


IP address, decimal	Network mask, decimal	IP address, binary
149.218.112.1	255.255.255.128	10010101 11011010 01110000 00000001
149.218.112.127		10010101 11011010 01110000 01111111
		----- 25 mask bits -----
CIDR notation: 149.218.112.0/25		
	-----	Mask bits

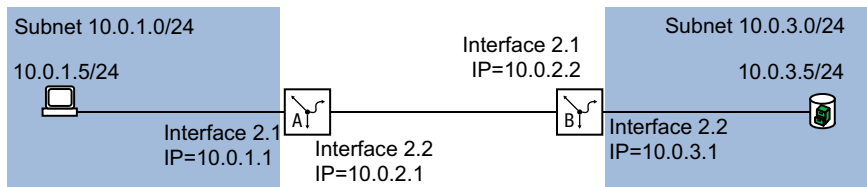


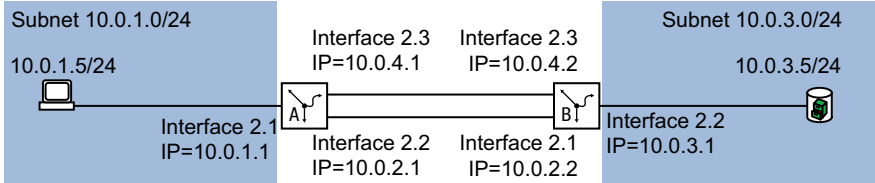




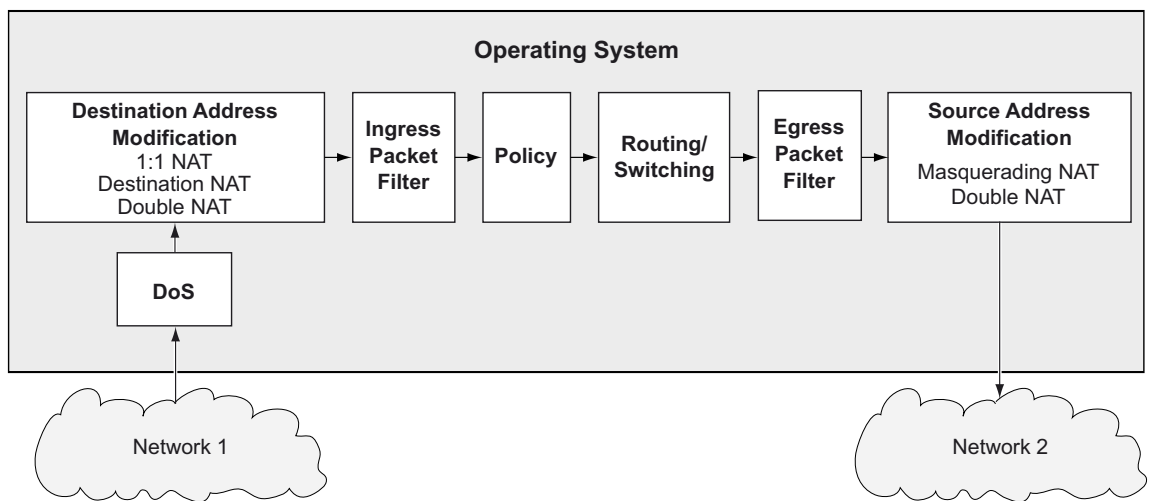


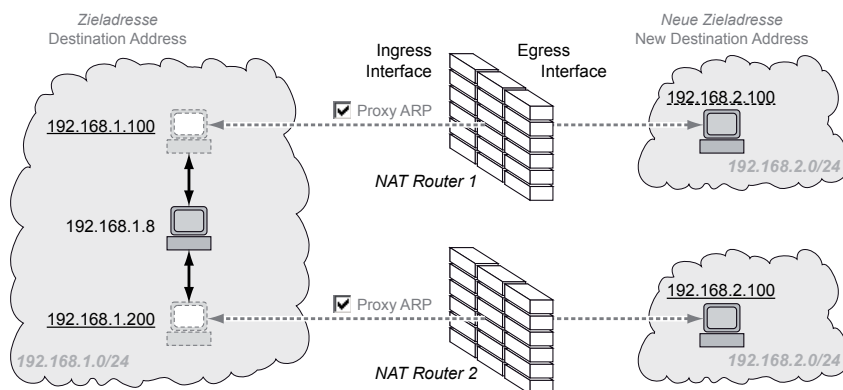
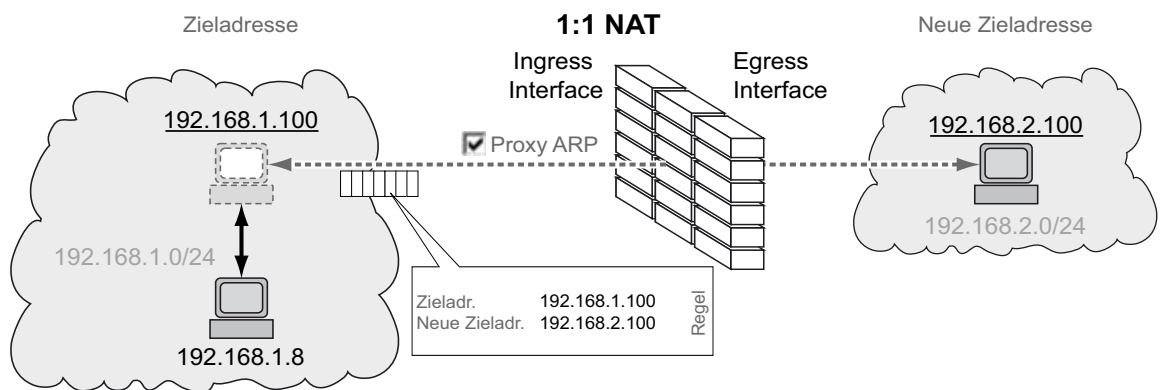




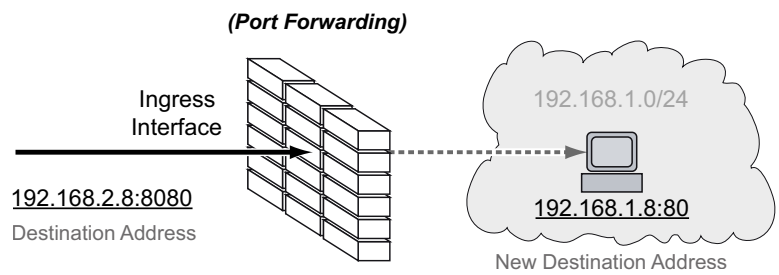
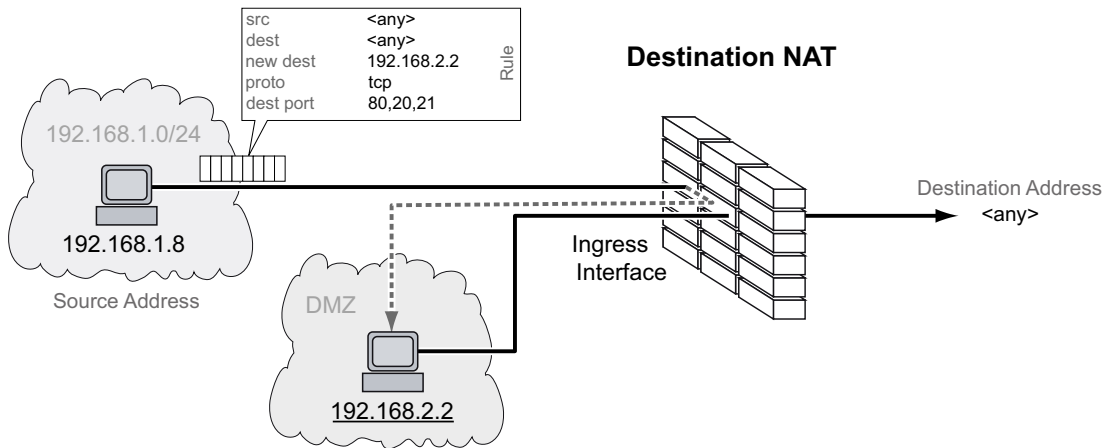










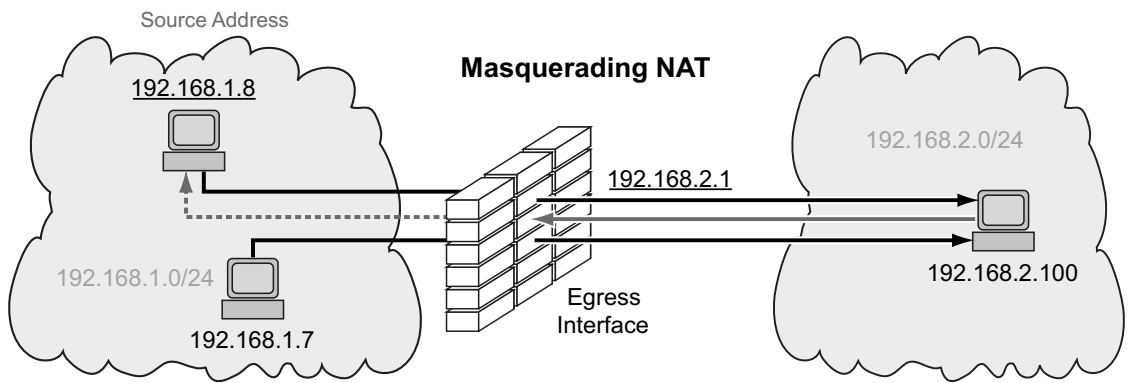


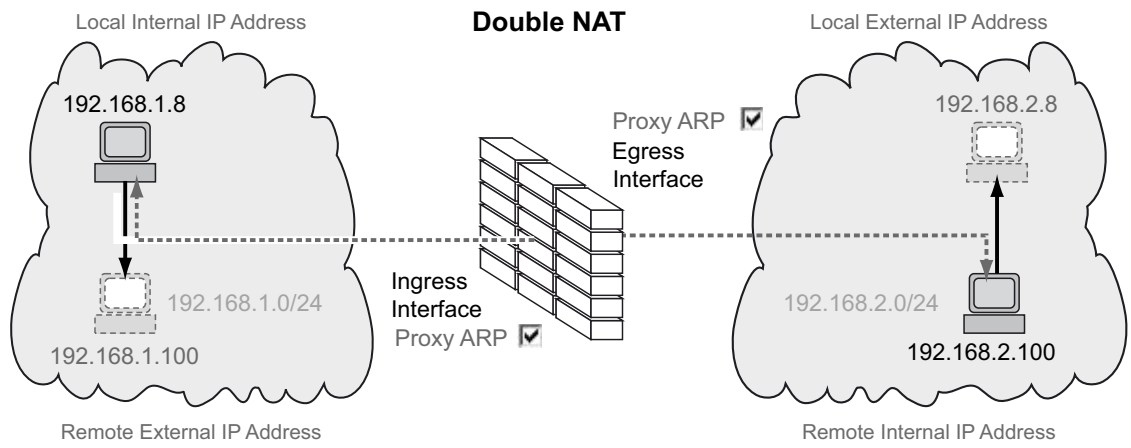


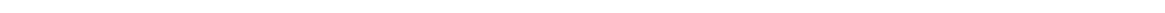


⌘ +



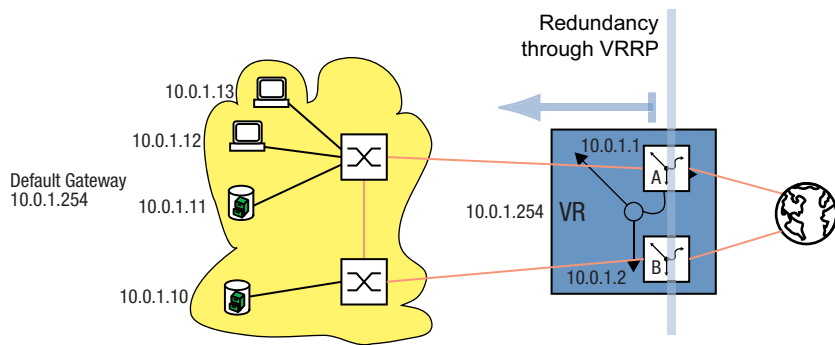








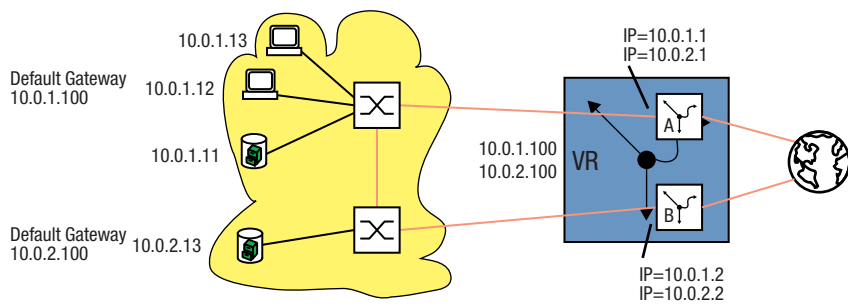
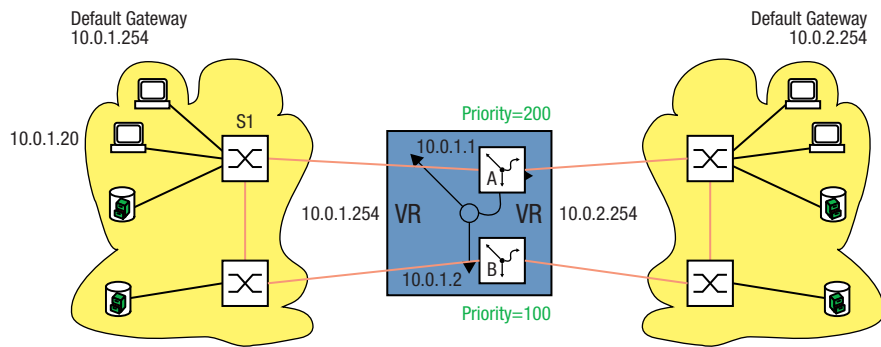




00:00:5e:00:01:xx

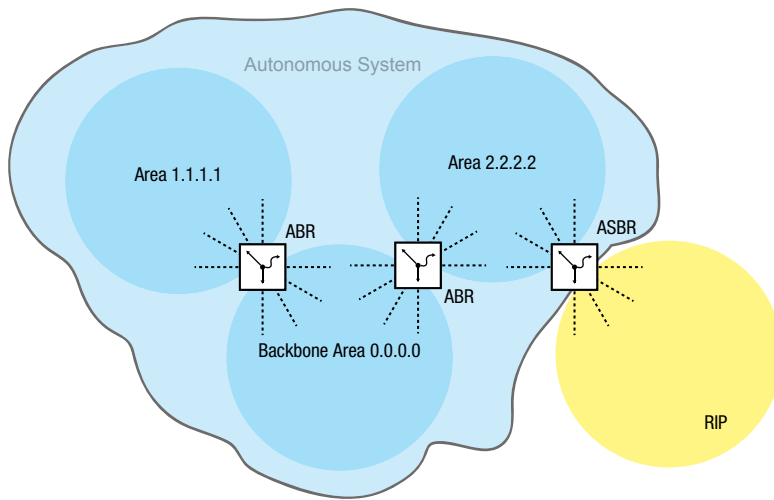
variable element = VRID
constant element

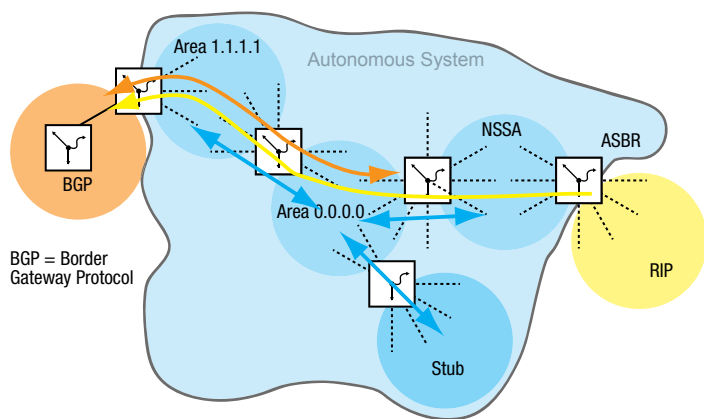


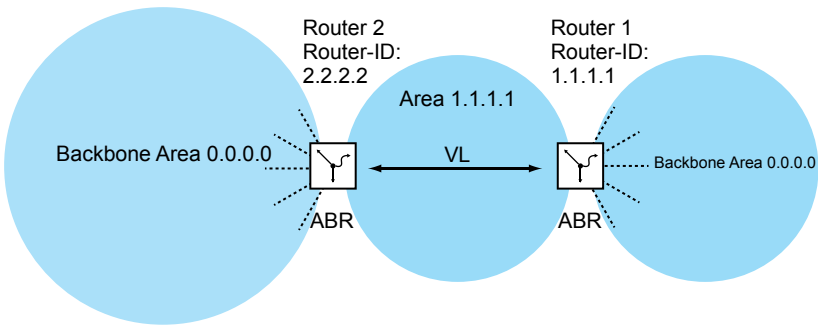
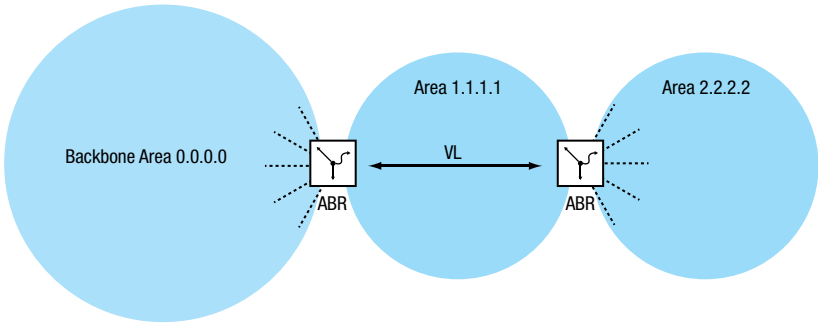


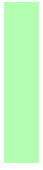


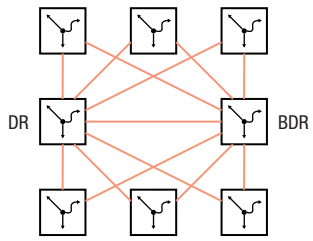








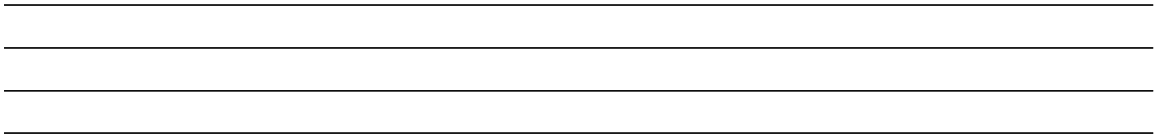


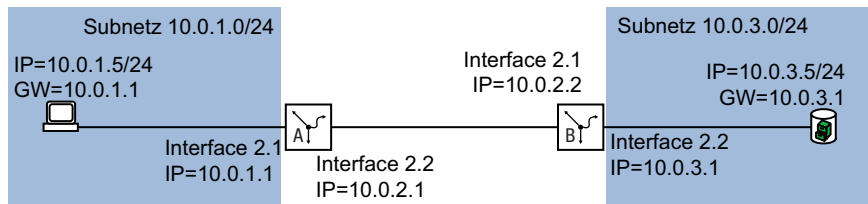






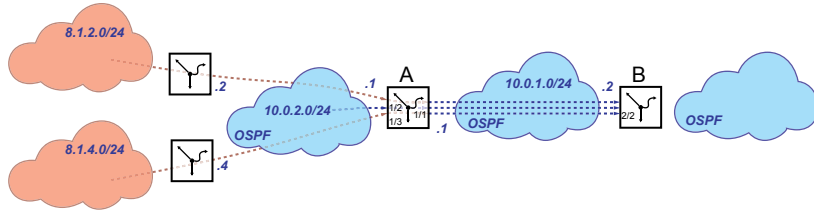




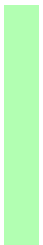
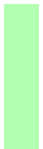
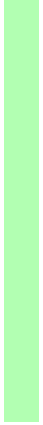
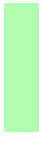


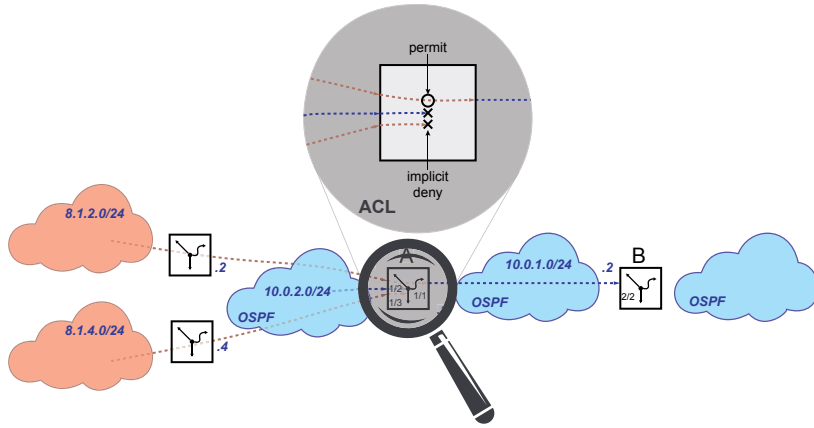




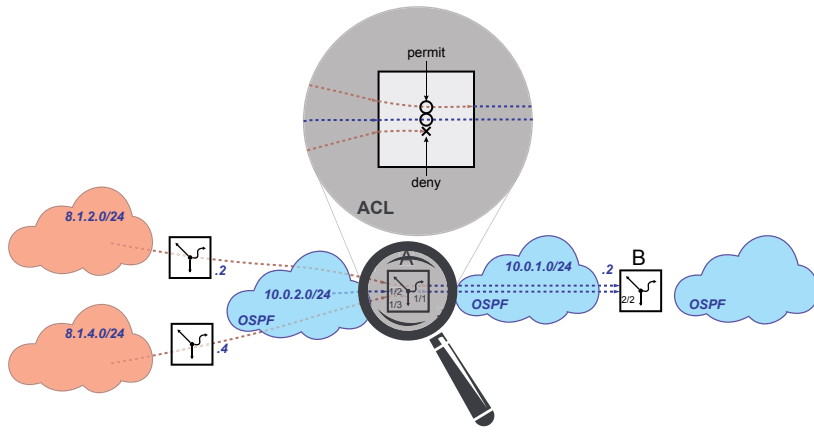


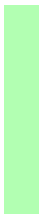
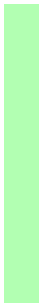








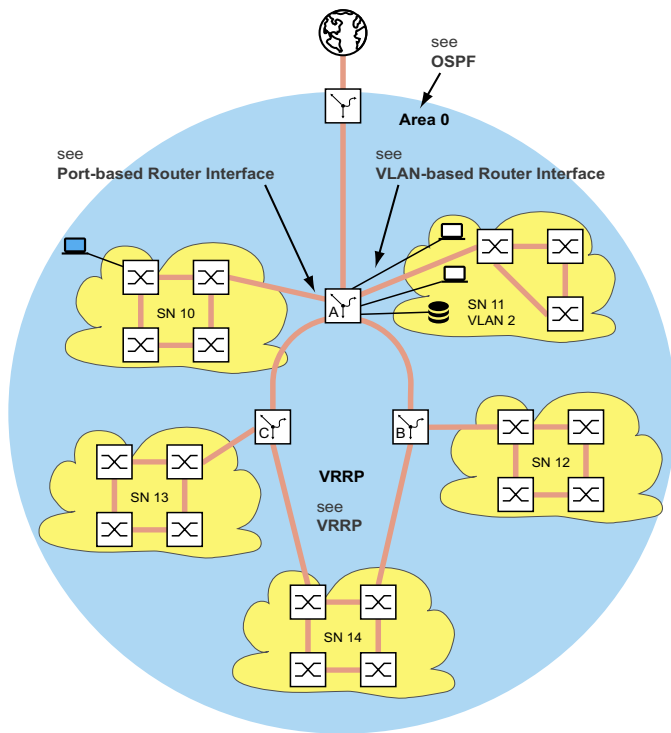


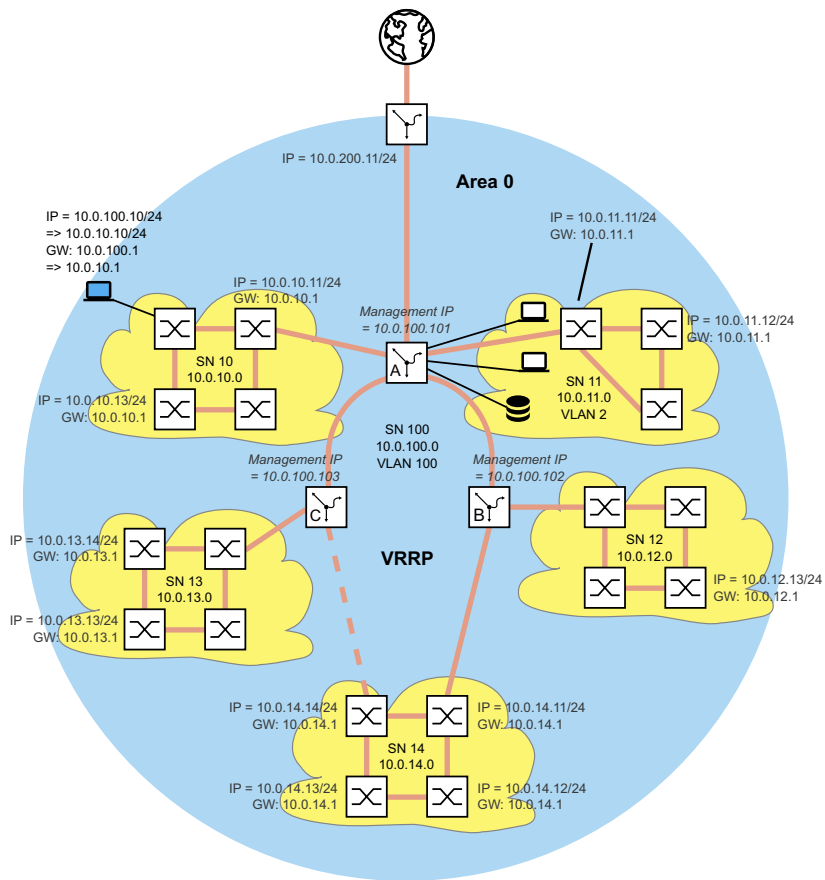


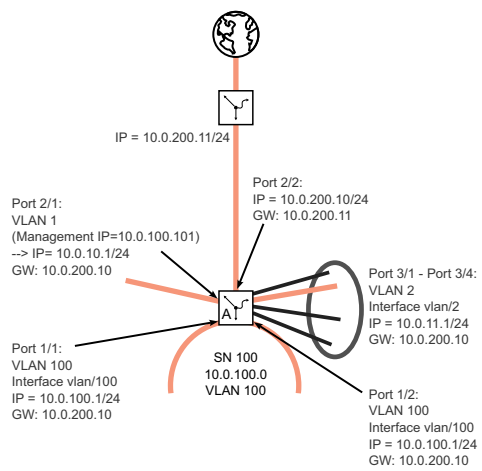


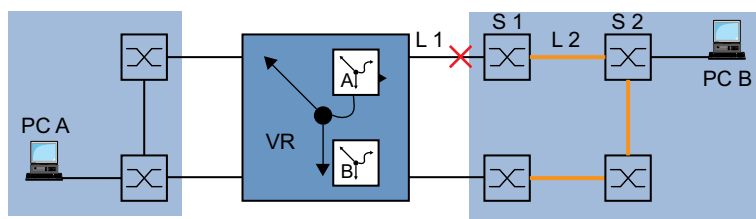


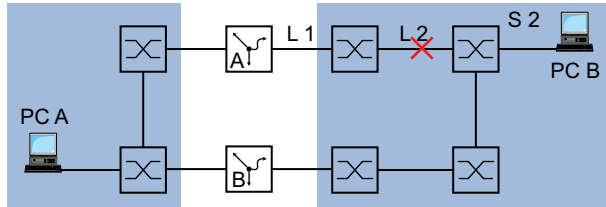














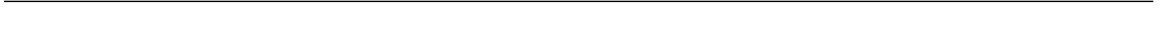
+

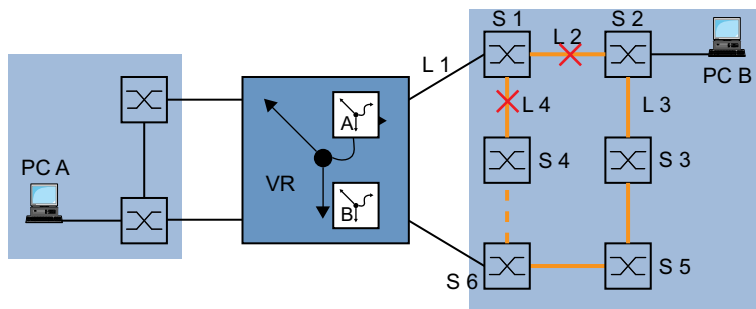
✓



⌘+



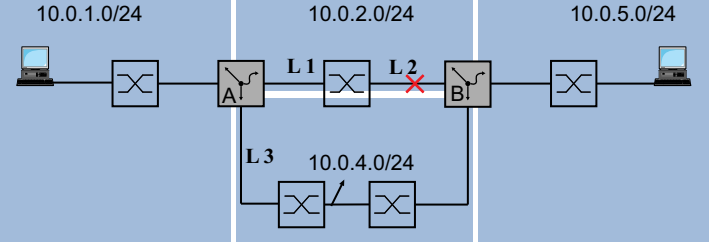


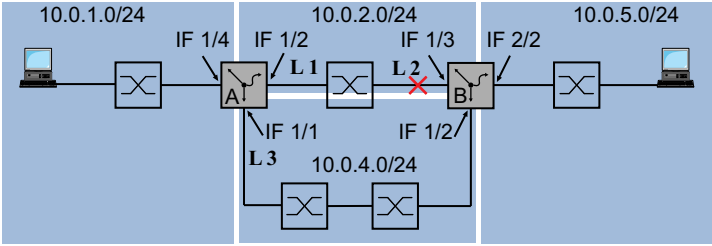


⊞
+









⊞
+

⊞
+

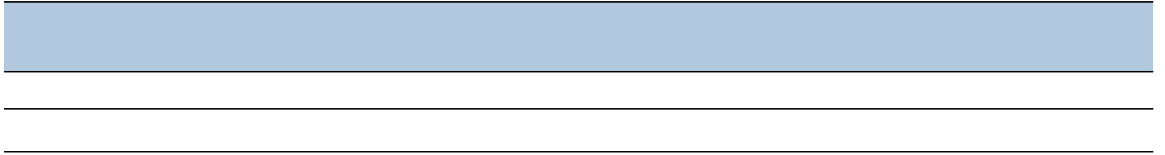




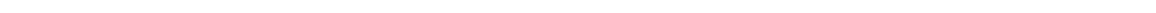
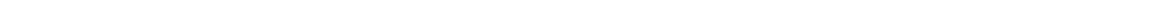
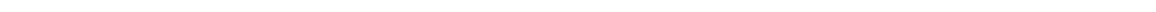
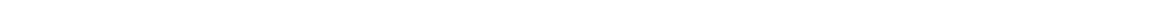
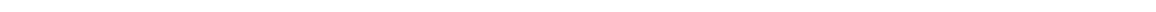
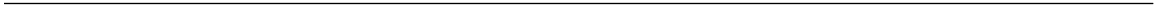
$B+$

$B+$

✓









⌘ +























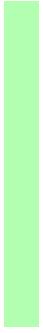


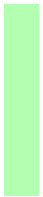
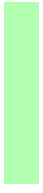




⌘ +





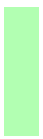


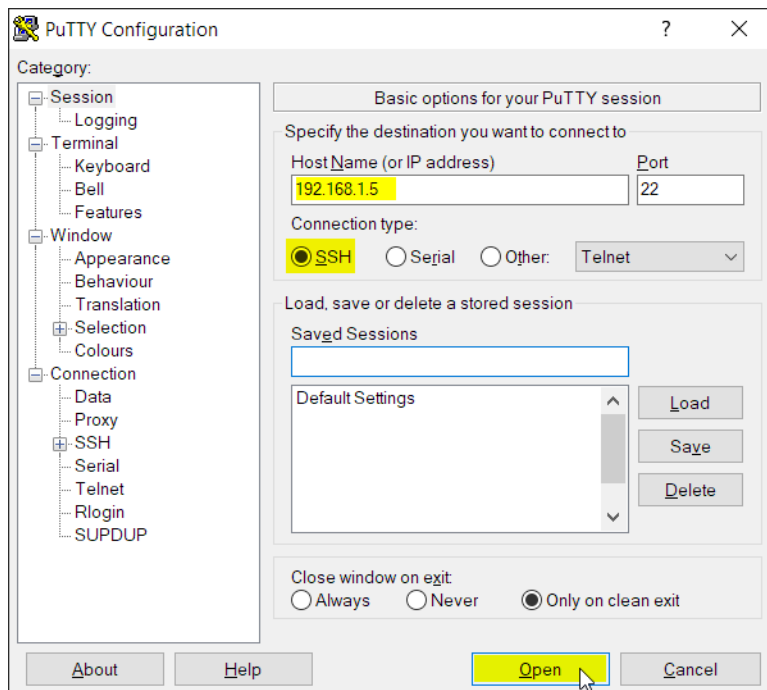


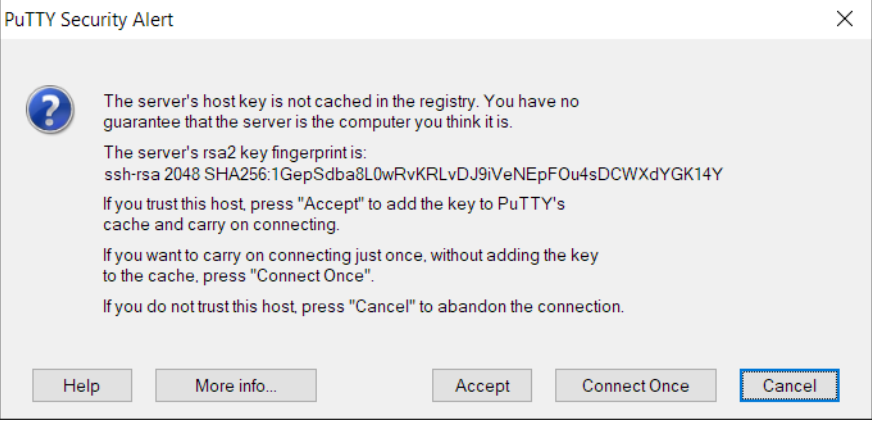
田+



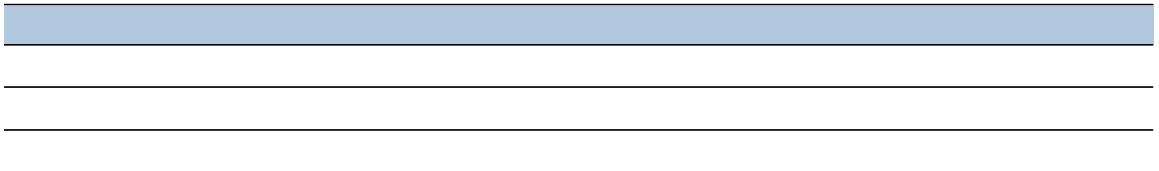










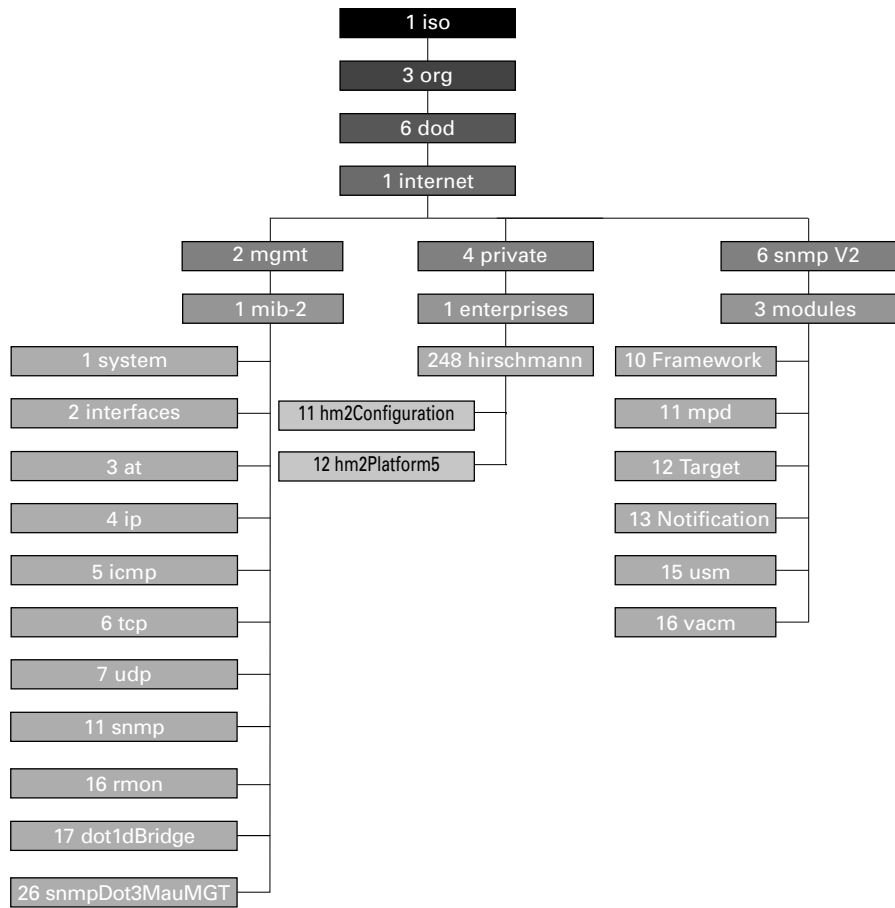








[Redacted Section]





Handwriting practice lines consisting of 20 horizontal black lines, evenly spaced and aligned to the left.





A series of 25 horizontal black lines, evenly spaced, providing a template for writing.









HIRSCHMANN

A **BELDEN** BRAND