What's new

> Reminder

> Support is limited to the most recently published LTSR with the latest CU and the most recently published CR. Please ensure you upgrade to one of these supported versions to benefit from the latest features, updates, fixes, and ongoing technical assistance.

eLux 7 2603 LTSR CU1

Release date: 2026-05-28

New

- This release introduces a new Cumulative Update (LTSR CU1) for the current Long Term Service Release (LTSR)
- Avaya Workplace VDI Client 3.0.16.2
- Cisco Webex VDI Plugins 46.3.2.34954
- Zscaler Client Connector 3.7.541.4.1

Changed

- Configurable notification behavior during deliveries and updates
- Additional system libraries for extended CWA functionality have been added to the System libraries EPM package
- Enabled UFS filesystem support in the kernel

Fixed issues

eLux 7 2603 LTSR CU1

- CapsLock-as-Shift (ForceShiftLock) no longer breaks Numpad input
- eluxd sent an incorrect ELUX_DOMAIN when using OU filter data

- Printers were not visible in the Citrix session after migrating to eLux 7
- Local applications could not be created Cisco Secure Client window was missing minimize/ maximize buttons on first boot
- Fonts in the quick system access menu appeared washed out
- The PulseAudio folder became incompatible, causing CWA to stop redirecting audio devices as CAMV4 after a nightly idle period
- Popup windows blocked HTTP redirects after the first navigation
- The ScoutHandshake RetryTimer was not applied due to an incorrect setter call
- Fixed SCG split-DNS domain handling in eLux
- The EST alias could not be set in est.ini, preventing EST enrollment
- The WLAN profile had autoconnect set to false in NetworkManager
- Various minor bugs fixed

Security

- Security update due to local kernel vulnerability copy.fail (CVE-2026-31431)
- Fixed kernel vulnerability related to dirty.frag (CVE-2026-43284, CVE-2026-43500)
- Updated webkit2gtk from version 2.46.5 to 2.50.4, addressing several security vulnerabilities
- The key storage mechanism for SCG communication has been hardened
- Updated operating system components with security-related patches

Known issues

eLux 7 2603 LTSR CU1

Limitations and known issues

- The persistent user directory feature introduced in eLux 7 2511, which can be activated through configuration, is known to cause undesirable side effects. We strongly advise against enabling this feature. A fix is planned for the next eLux version.
- OIDC and SAML authentication methods for eLux require additional setup via Advanced File Entries. For further details, please refer to the Scout release notes.

Hardware limitations

- Disk encryption only supported for devices with TPM 2.0 module.

- Sleep mode (Suspend to RAM) is not yet available for devices with eKabini chipsets.
- Multi-monitor setup in clone mode requires uniform resolution of all connected monitors.
- Wake on LAN via USB-C docking station is not functional.
- Dell Pro 14 Plus PB14250 Intel: Microphone input via 3.5 mm audio jack is not functional.
- Dell Latitude 3410, 3420: Bluetooth is not functional.
- Fujitsu FUTRO devices: UEFI update only supported for devices without disk encryption via TPM2.
- Fujitsu devices: BIOSset tool for BIOS configuration is not functional
- Hewlett Packard mt645 G8: LTE is not functional.
- Hewlett Packard mt645 G7: Sleep mode (Suspend to RAM) is not functional.
- Hewlett Packard mt46: LTE is not functional.
- Lenovo ThinkPads Gen4: LTE is not functional.
- Lenovo ThinkPads Gen4 AMD: Bluetooth is not functional.
- LG gram Mobile Thin Client 17ZT90P: LTE is not supported.
- LG Ultrabook 14UT70Q: Sleep mode and sound via HDMI are not functional.
- MOStron Touch-Panel-PC TSPC5-24: BIOS update via Scout is not functional.
- NEXterminal XI DP/DP2: Internal speaker is not functional.

Misc - before you start

Migration from eLux RP 6 to eLux 7

The migration from eLux RP 6 to eLux 7 can be carried out easily via a firmware update. For details on the procedure and the system requirements, please refer to the chapter Performing the migration in the product documentation.

Please note the life cycle milestones of eLux RP 6 so that you can plan the changeover to eLux 7 accordingly.

- 2025-03-31 EOM (End of Maintenance eLux RP 6)
- 2025-09-30 EOL (End of Life eLux RP 6)

Update eLux 7 images

The firmware update for eLux 7 2603 LTSR CU1 is carried out as usual via Scout or via the device. The eLux 7 software container required for this must be created via the AllPackages container

archive. Please import the AllPackages container archive into ELIAS 18 or use the installation program eLuxContainer.exe contained in the archive for the legacy ELIAS.

Improved certificate verification for firmware updates via HTTPS protocol

Please note that for firmware updates via the HTTPS protocol, the required root/intermediate certificates of a CA (Chain of Trust) must be available on the devices in /setup/cacerts/ in order to be able to perform the required certificate verification. Self-signed certificates can also be used instead of CA certificates.

Disk encryption

Disk encryption is activated by default in eLux 7 and can be deactivated/activated via the encryption FPM of the BaseOS package or via a configuration parameter. For further information, see the relevant product documentation topic.

User rights for new features

Note that new features are enabled for users or administrators unless explicitly stated otherwise. If required, restrict the relevant rights.

Security updates

Further security updates to fix the Intel and AMD processor vulnerabilities are included in the release of eLux 7 2603 LTSR CU1. The updates consist of patches for the Linux kernel and processor microcode updates for AMD and Intel chipsets. Unicon strongly recommends that you perform the relevant system updates for your eLux devices. Please note that after updating your devices, appropriate tests in your system environment are necessary to ensure the desired functionality.

Client Hardware Compatibility

See **Certified Hardware** on myelux.com for the client models supported by this version of eLux 7.

Deprecation

System requirements

eLux 7

Minimum hardware requirements for eLux 7

- Processor: x86, 64-bit capable and powerful (1)
- RAM: 4 GB (2)
- HDD: 16 GB (3)
- GPU: AMD or Intel chipset
- Network: Ethernet or WLAN
- I/O ports: USB 3.0 or USB 2.0, USB boot support
- Resolution: 1920 x 1080 (Full HD)
- Boot mode: UEFI

(1) 2 GHz (Quad Core) or more recommended, less could affect system performance depending on the use case (2) at least 3.5 GB available for the operating system (3) 8 GB are also supported, but only with limited functionalities

> **Note:

> eLux 7 has been validated on a variety of hardware configurations. However, users should note that entry-level or legacy hardware may not support all features or may have degraded performance. Limitations may result from insufficient CPU capacity, limited memory, slower I/O subsystems or lack of hardware acceleration.

Minimum Scout and SCG version for eLux 7

Please use Scout versions

- 15 2302 LTSR CU1 and later or
- 15 2309 and later and ELIAS versions

- 18 2403 and later and SCG versions
- 1 2209 and later for the management of eLux 7 devices via Scout Console, Scout Board, ELIAS and SCG.

Client Hardware Compatibility

eLux 7 2603 LTSR CU1 has been successfully tested on the following client hardware:

Please note that the devices are tested and released based on UEFI support only. Devices with dual boot mode must be configured to UEFI support. For more information about hardware features, see **Hardware limitations** in the **Release Information**. eLux 7 has been validated on a variety of hardware configurations. However, users should note that entry-level or legacy hardware may not support all features or may have degraded performance. Limitations may result from insufficient CPU capacity, limited memory, slower I/O subsystems or lack of hardware acceleration. If demanding multimedia usage scenarios impair the system performance of older hardware models with minimal system equipment, it may make sense to upgrade the RAM. Models that have been successfully tested and approved exclusively by the hardware manufacturer are listed separately in the **Manufacturer-Certified Devices** section. These devices have not been tested by Citrix.

Amulet Hotkey

- DX1550 - Intel® Processor U300
- DX1554 - Intel® Processor U300
- DX1714 - 13th Gen Intel® Core™ i5-1335U

Asus - Manufacturer-Certified Devices

- NUC 14 Essential Slim PC Kit RNUC14MNK9700002 - Intel® Processor N97

Centerm - Manufacturer-Certified Devices

- S25 Ultra Mini PC - Intel® Processor N150

Dell

- Pro 14 Plus PB14250 - Intel® Core™ Ultra 5 235U vPro
- Latitude 5550 - 13th Gen Intel® Core™ i5-1335U
- Latitude 5540

- Latitude 5440
- Latitude 3440
- Latitude 5530
- Latitude 5520
- Latitude 5420
- Latitude 3420
- Latitude 3410
- Wyse 5470 Mobile Thin Client
- Dock WD19
- Dock integrated in Monitor C2422HE
- Dock integrated in Monitor C2722DE
- Dock integrated in Monitor C3422WE
- Pro Max Micro FCM2250 - Intel® Core™ Ultra 7 265
- Pro Micro Plus QBM1250 - Intel® Core™ Ultra 5 235T
- OptiPlex Micro Form Factor Plus 7020
- OptiPlex 3000 - Intel® Celeron® CPU N5105
- OptiPlex 3000 - Intel® Pentium® CPU N6005
- OptiPlex 3090 Ultra
- Wyse 5070
- Wyse 7020 - - Wyse Z50QQ - up to four monitors
- Wyse 5060
- Wyse 5020

Dell – Manufacturer-Certified Devices

- Dock WD19S

EXTRA Computer GmbH

- exone Thin Client Mobile X14 - Intel® N200
- exone Thin Client Entry X14 - Intel® N100

Fujitsu

- LIFEBOOK E5512 - Intel® Celeron® 7305 @ 1.10GHz
- LIFEBOOK E5511 - Intel® Core™ i7-1165G7 @ 2.80GHz
- LIFEBOOK E5510 - Intel® Core™ i5-10210U CPU @ 1.60GHz
- LIFEBOOK E5410 - Intel® Celeron® CPU 5305U @ 2.30GHz
- LIFEBOOK E559 - Intel® Core™ i5-8265U CPU @ 1.60GHz
- LIFEBOOK E556 - Intel® Celeron® CPU 3855U @ 1.60GHz
- Thunderbolt ™ 4 Port Replicator
- Dock integrated in Display P34-9 UE - for LIFEBOOK E5410
- Dock integrated in Display P27-9 TE QHD - for LIFEBOOK E5410, FUTRO S7011
- Dock integrated in Display P2410 TS CAM - for LIFEBOOK E5410, FUTRO S7011, FUTRO S9011
- FUTRO S9011 with NVIDIA Quadro T400
- FUTRO S7011
- FUTRO S5011
- FUTRO S9010
- FUTRO S9010 with NVIDIA Quadro T400
- FUTRO S9010 with NVIDIA Quadro P400
- FUTRO S7010
- FUTRO S5010
- FUTRO S940
- FUTRO S940 with NVIDIA Quadro P400 - up to five monitors
- FUTRO S740
- FUTRO S540
- FUTRO Q940
- FUTRO S930
- FUTRO S920
- FUTRO S920 with AMD Firepro W2100 - up to four monitors
- FUTRO S720
- ESPRIMO D7010 - Intel® Core™ i3-10100 CPU @ 3.60GHz

- ESPRIMO Q558 - Intel® Core™ i5-8400T CPU @ 1.70GHz

Fujitsu – Manufacturer-Certified Devices

- LIFEBOOK E5512 / MTC - Intel® Core™ i5-1235U up to 4.40 GHz, identical to LIFEBOOK E5512
- LIFEBOOK E5512 / MTC - Intel® Core™ i3-1215U up to 4.40 GHz, identical to LIFEBOOK E5512
- LIFEBOOK E5412 / MTC - Intel® Core™ i3-1215U up to 4.40 GHz, identical to LIFEBOOK E5512
- LIFEBOOK E5412 / MTC - Intel® Celeron® 7305 @ 1.10GHz, identical to LIFEBOOK E5512
- LIFEBOOK E5510 / MTC - Intel® Core™ i3-10110U up to 4.1GHz, identical to LIFEBOOK E5410
- LIFEBOOK E5510 / MTC - Intel® Celeron® CPU 5305U up to 2.30GHz, identical to LIFEBOOK E5410
- LIFEBOOK E5410 / MTC - Intel® Core™ i3-10110U up to 4.1GHz, identical to LIFEBOOK E5410
- Thunderbolt ™ 4 Port Replicator - for LIFEBOOK E5x12
- Dock integrated in Display P34-9 US - for LIFEBOOK E5x10, FUTRO S7011
- Dock integrated in Display P27-9 TE QHD - for LIFEBOOK E5x10
- Dock integrated in Display P2410 WE CAM - for LIFEBOOK E5x10, FUTRO S7011
- Dock integrated in Display P2410 TS CAM - for LIFEBOOK E5x10

HP

- mt645 G8
- mt645 G7
- Pro mt440 G3
- mt46
- mt45
- mt22
- mt21
- EliteBook 850 G6

- EliteBook x360 830 G6
- 2013 UltraSlim Docking Station - for mt45
- Elite USB-C Dock G5
- Elite USB-C Dock G4 - for mt21, mt22
- Elite t755
- Elite t655
- Pro t550
- t740
- t740 with AMD Radeon E9173 PCIe
- t640
- t540
- t730
- t730 with AMD Firepro W2100 - up to six monitors
- t630
- t530
- t430

IGEL

- UD3 M350C

LENOVO

- Thinkpad E14 Gen 7 - Intel® Ultra5 225U 16/512 WUXGA
- ThinkPad L16 Gen1 - Intel® Core™ Ultra 5 125U
- ThinkPad L14 Gen5 - Intel® Core™ Ultra 5 125U
- ThinkPad L13 Gen5 - Intel® Core™ Ultra 5 125U
- ThinkPad L14 Gen4 - Intel® Core™ i5-1345U
- ThinkPad L15 Gen4 - Intel® Core™ i3-1315U
- ThinkPad L14 Gen4 - Intel® Core™ i3-1315U
- ThinkPad L13 Gen4 - Intel® Core™ i3-1315U
- ThinkPad L14 Gen6 - AMD Ryzen 5 Pro 215 with Radeon 740M Graphics

- ThinkPad L16 Gen1 - AMD Ryzen 3 PRO 7335U with Radeon Graphics
- ThinkPad L14 Gen5 - AMD Ryzen 5 PRO 7535U with Radeon Graphics
- ThinkPad L15 Gen4 - AMD Ryzen 3 PRO 7330U
- ThinkPad L14 Gen4 - AMD Ryzen 3 PRO 7330U
- ThinkPad L13 Gen4 - AMD Ryzen 3 PRO 7330U
- ThinkPad L14 Gen1 - AMD Ryzen 5 PRO 4650U
- Lenovo ThinkPad Universal USB-C Dock
- Lenovo ThinkPad Hybrid USB-C with USB-A Dock TYPE 40AF - for ThinkPad L14 Gen4, Dell Latitude 3410
- ThinkCentre M720q Tiny - Intel® Core™ i5-9500T
- Think Centre M710q 10 MS - only certified for eLux 7 2601
- ThinkCentre Neo 50q Gen 4 - Intel® Core™ i3
- ThinkCentre Neo 50q Gen 4 - Intel® Celeron® 7305
- ThinkCentre M70q (Gen 5) - Intel® Core™ i3-14100T
- ThinkCentre M70q (Gen 3)
- ThinkCentre M75q (Gen 5) - AMD Ryzen 3 PRO 8300GE w/ Radeon 740M Graphics ThinkCentre M75q (Gen2)
- ThinkCentre M75n
- ThinkCentre M715q
- ThinkCentre M625q - AMD E2-9000e
- ThinkCentre M625q - AMD A4-9120e
- ThinkStation P3 Tiny - with Lenovo ThinkCentre TIO22 (Gen 5) Touch / TIO22 (Gen 5) Touch / TIO27 LCD Monitor

LG

- gram Mobile Thin Client 17ZT90R-G.AX34G - 13th Gen Intel® Core™ i3-1315U
- gram Mobile Thin Client 15ZT90R-G.AX34G - 13th Gen Intel® Core™ i3-1315U
- gram Mobile Thin Client 17ZT90P-G.AX36G
- gram Mobile Thin Client 15ZT90P-G.AVD11KB
- Ultrabook 14UT70Q-G.AX34G

- Thin Client CL600N-6N
- Medical All-in-One Cloud Device 24CR671N-6P (24")
- Medical All-in-One Cloud Device 24CN670N-6N (24")
- All-in-One Cloud Device 24CQ651N-2P (24") - Intel® Pentium® Silver N6005 @ 2.00GHz
- CQ601N-2P Thin Client Box - Intel® Pentium® Silver N6005 @ 2.00GHz

MOStron

- Touch-Panel-PC TSPC5-24

NEXterminal

- NEXlaptop IV
- NEXstation XI DP/DP2 - Intel® Quad Core 2.0Ghz
- NEXstation X

General information

Administrator's Guides describe all features and functionality of a product or component in detail. They can be used as a reference manual. For some procedures and best practices, we provide **Short guides** to help you perform particular tasks. **Whitepapers** give an overview of benefits and services, standards or technical topics.

This documentation refers to the latest product versions. Any features added or changed for a particular release are listed in the Release Notes on `www.myelux.de`.

Representation

REPRESENTATION	DESCRIPTION
Control element	All graphical user interface controls are displayed in **bold**.
[Menu > menu command]	Multiple sequential steps through the interface such as menus or dialog are separated by `>`.
Value	All data that have to be entered by the user or data that represent a field value are displayed in `Courier New`. Also, file names and path names are displayed in `Courier New`.
[Strg]	Keys to be pressed are displayed in [capital letters].
[<Placeholder>]	Placeholders in instructions and user input are displayed in *italics* and in `<angle brackets>`.
1. Instruction	Procedures to be carried out step by step are realized as numbered steps.
Result	System responses and results are displayed in *italics*.

Abbreviations and acronyms

ABBREVIATION	DESCRIPTION
AD	Active Directory, directory service of Microsoft Windows Server
DMZ	Demilitarized zone. Systems in a DMZ are protected against other networks by firewall rules.
EBKGUI	Interface of the eLux® Builder Kit (Tool for creating eLux software packages)
EPM	eLux package module ([.epm], software package)
FPM	Feature package module ([.fpm], part of a software package)
FQDN	Fully qualified domain name. DNS name that uniquely identifies a computer on the network, example: `client1.sampletec-01.com`
GB	Gigabyte
GHz	Gigahertz (processing speed)
HDD	Hard disk drive (flash memory)
IDF	Image Definition File ([.idf])
IIS	Internet Information Services, Microsoft Web server
MB	Megabyte
OU	Organizational unit. Unit or group within the organizational structure
VM	Virtual machine
VPN	Virtual Private Network

Administrator's guides

An administrator\'s guide is a product manual that describes the features and functions of one of our products, including detailed instructions on how to proceed.

We provide administrator's guides for our eLux operating systems, our management software Scout Enterprise Management Suite® and also for some additional features such as ELIAS, Scout Report Generator and others.

The documentation refers to the latest product versions.

Scout Report Generator

The Scout Report Generator is used to create reports based on devices, OUs, applications or asset entries currently stored in the Scout database.

Overview

The Scout Report Generator is used to create reports based on devices, OUs, applications or asset entries currently stored in the Scout database.

The Scout Report Generator is part of the Scout Enterprise Management Suite and is installed as a component of it. For further information, see Installing Scout Enterprise Management Suite in the **Installation** guide.

Use the menu rights to control whether administrators are allowed to start the Scout Report Generator. Use the basic administrator rights to allow or deny creating, editing and deleting database reports and local reports.

Starting Scout Report Generator

1. In the Scout Console, make sure that **Security** > **Activate administrator policies** is selected. For further information, see Multi-tenancy.
2. In the Scout Console, select **View** > **Report**.

The Scout Report Generator window opens. For database access, the settings of the Scout Console are used.

Multi-tenancy

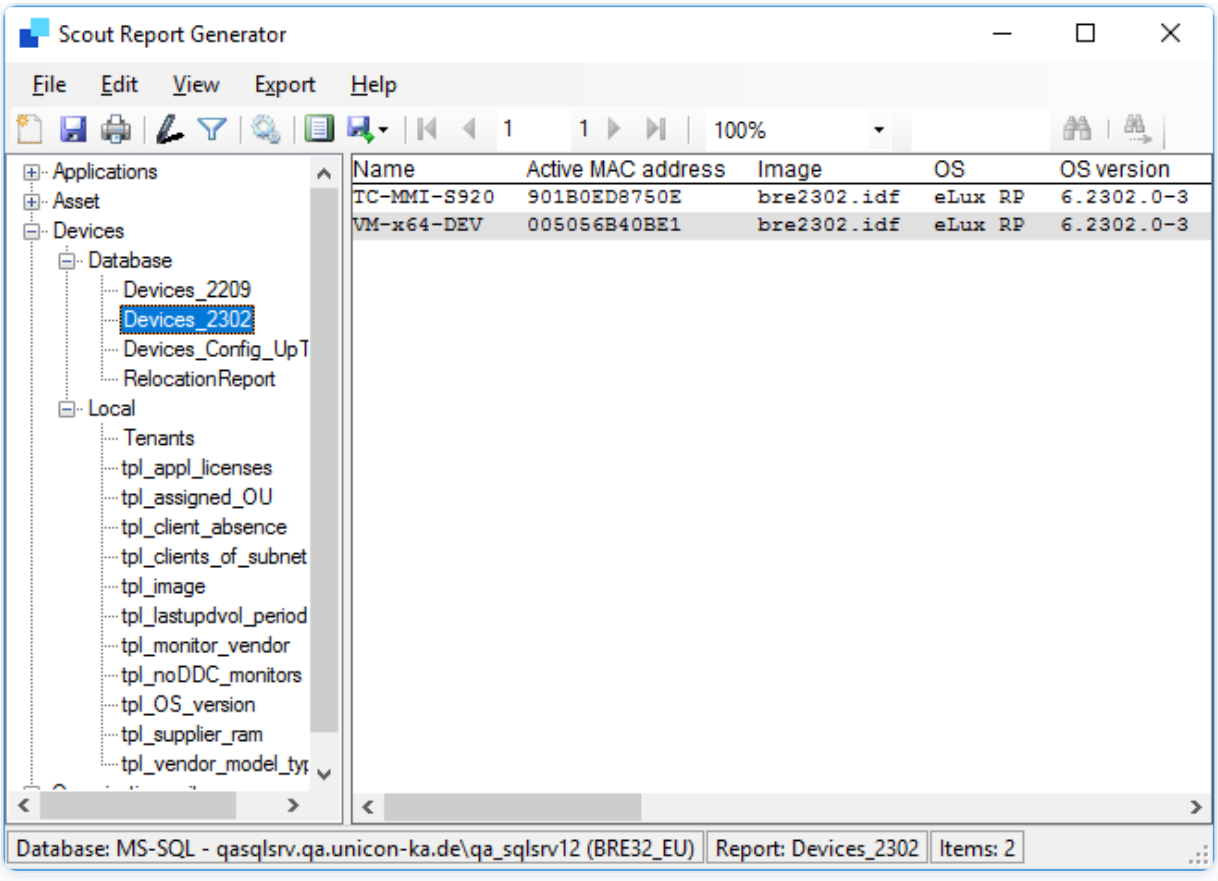
To use the report generator, activate administrator management (**Security** > **Activate administrator policies**). The access rights defined there then are applied for creating reports. This is to ensure that administrators can only create reports based on the devices visible to them.

Interface

The Scout Report Generator application window is divided into two areas. On the left, all reports defined are listed. They are organized into the four types **organization units**, **devices**, **applications** and **inventory**. The predefined reports provided can be used as templates.

On the right, a report output is displayed. The name and number of matching elements are displayed on the status bar.

> **Note:**
>
> The report output on the right does not necessarily correspond to the selected report on the left. Click to refresh or create an up-to-date output of the selected report.



The toolbar buttons help define, create, print and browse reports.

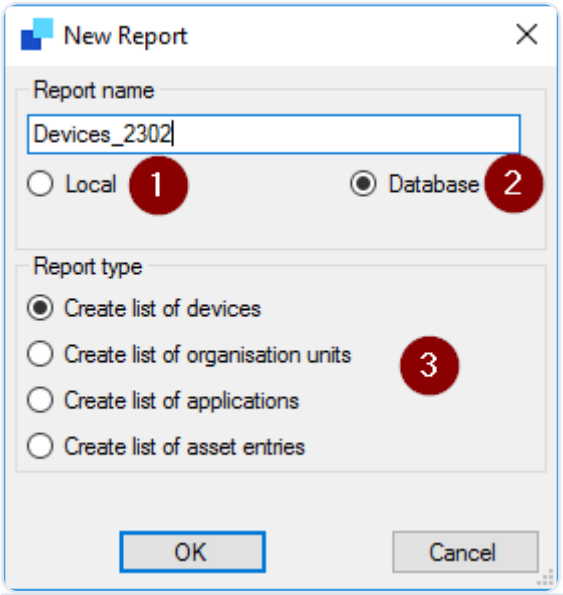
Basic functions

- Creating a new report definition
- Creating a report output
- Applying report templates
- Sharing report definitions
- Saving, printing and deleting a report
- Exporting a report
- Changing language

Creating and saving a new report

REQUIRES
Base right [Create, edit and delete database reports]

1. Click **File** > **New Report** or **click** **New report**.
2. Enter a name, and then select the storage location and report type.



1. The report will be saved to the local user directory of the logged-on user (`<User>\Documents\Unicon\Scout\Report`).

2. The report will be saved in the Scout database and can then be viewed, used and edited by other authorized Scout administrators. See also Sharing reports.

3. The type defines the kind of data you filter on with the report.

Confirm with **OK**.

3. For the selected report, click `Edit > Layout` or, on the toolbar, click the Layout. In the dialog, define the desired layout features. For further information, see Report layout.
4. For the selected report, click `Edit > Filter` or, on the toolbar, click the Filter. In the Report Filter dialog, define the desired filter values. For further information, see Report filter.

Note:

You can easily change the name later on, but not the report type and storage location. To change the storage location, create a new version by choosing **Save as**.

Creating a report output

1. Select a report
2. Click **View > Create report**.

   Or

   Double-click a report definition.

   Or

   Click the **Create report** button.

The report output is created and displayed on the right.

You can vary the display size of the report output, select page view (**View > Page view**), browse through and search the report.

Report outputs can be printed and exported.

Note:

- A report, whose layout does not contain fields, cannot be executed.

- If report outputs contain column titles but no elements, the filter has not been defined correctly.

Report templates

When you install the Scout Report Generator, several report templates are added to the local user directory `Public\Documents\UniCon\Scout\Report`. You can use them whenever suitable.

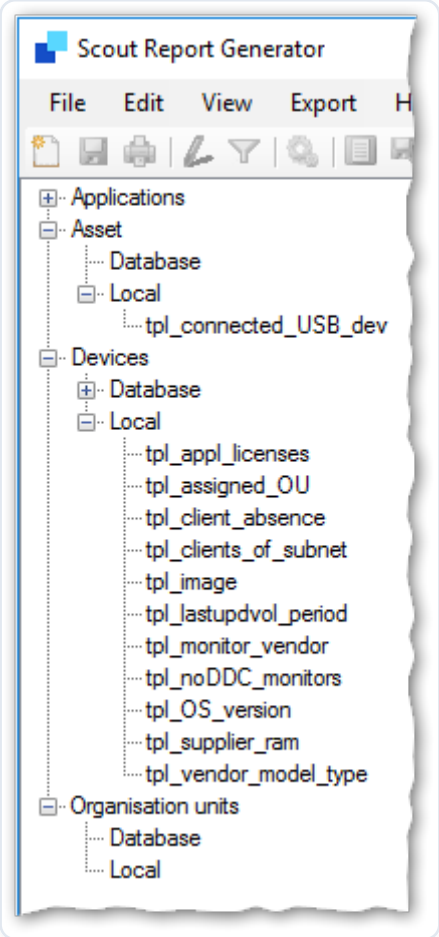
In the tree view of the Scout Report Generator, the report templates are organized into different types. They can be modified as required.

Report type Devices

TEMPLATE NAME	DESCRIPTION
tpl_appl_licenses	Devices with defined application licenses
tpl_assigned_OU	Devices of defined OU
tpl_clients_of_subnet	Devices of defined IP subnet
tpl_client_absence	Devices with latest contact to Scout Server before defined date
tpl_image	Devices with defined image name
tpl_lastupdvol_period	Devices with software update between two defined dates
tpl_monitor_vendor	Devices with monitors of a defined producer
tpl_noDDC_monitors	Devices without DDC information (EDID) of the monitor
tpl_OS_version	Devices with defined OS version
tpl_supplier_ram	Devices of defined producer with RAM less than defined value
tpl_vendor_model_type	Devices of defined producer, model and type

Report type Assets

TEMPLATE NAME	DESCRIPTION
tpl_connected_USB_dev	Devices with connected USB devices of a defined producer



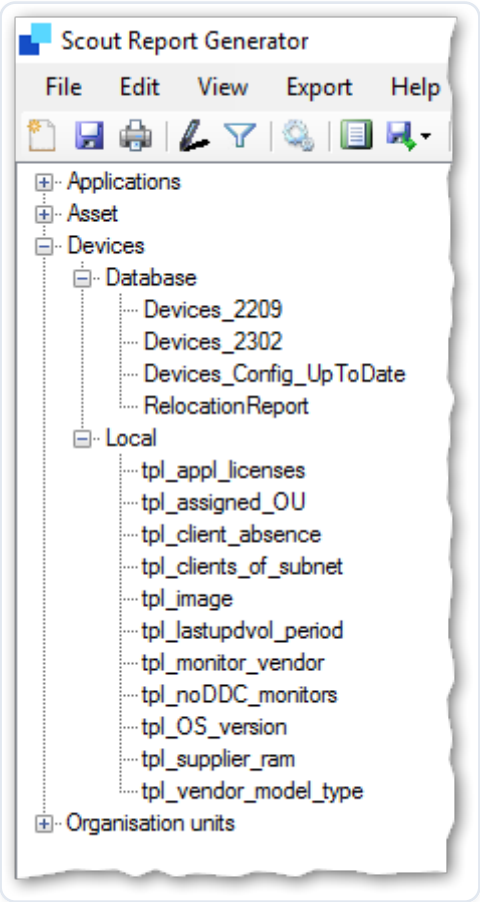
Note:

- Before creating a report output, you must modify the filter values for most templates.
- Use the Save as feature to create copies or variants.

Sharing reports

By using the storage location **Database** when you create a new report, you allow all authorized Scout administrators to use and edit a report definition. For further information, see Creating and saving a new report.

In contrast, any report defined with the option **Local** is only available in the local user directory of the creator.



Note:

Use the **Save as** feature to save locally defined reports to the database and make them available to other administrators.

Required administrator rights

- Administrators with the menu right **View > Report generator** are authorized to run database reports.
- Administrators with the base right **Create, edit and delete database reports** are authorized to edit database reports.
- Administrators with the base right **Allow local reports** are authorized to run and edit local reports.

Saving printing and deleting a report

All file-related features of the Scout Report Generator are provided on the **File** menu.

1. Select the relevant report
2. Click **File** > **Menu command**.

MENU COMMAND	ALTERNATIVE
Rename	[F2]
Delete	[ENTF]
Save	
Save as	--
Print	

> **Note:**
>
> Use the **Save as** feature to create slightly different versions or to make locally saved reports available to other administrators.

Exporting a report

1. Check the page layout for the relevant report. For further information, see Defining page layout for print output.
2. Run the relevant report or make sure that the output shown on the right is up-to-date.
3. On the toolbar, click the Export button.
4. Select the export format Excel or PDF.
5. In the **Save as** dialog, enter a file name and storage location.

The currently generated report output is exported to the specified file.

> **Note:**
>
> Exporting to a Dynamic Client Group makes the result set available in the Scout Console.

> For further information, see Dynamic Device Groups in the **Scout** guide.

> **Note:**
>
> The export entry in the context menu by right-clicking on the report does not work.

Changing language

Click **View** > **Language**.

> **Note:**
>
> The **Default** option uses the language set in the operating system.

Report definition

Defining a report includes two basic steps:

- The layout of a report specifies the relevant database fields and their display properties. In addition, it determines the way the report output is formatted, including the font, page layout and headers.
- The filter is used to filter the selected fields for particular values. You can combine several filter expressions by using operators.

Report layout

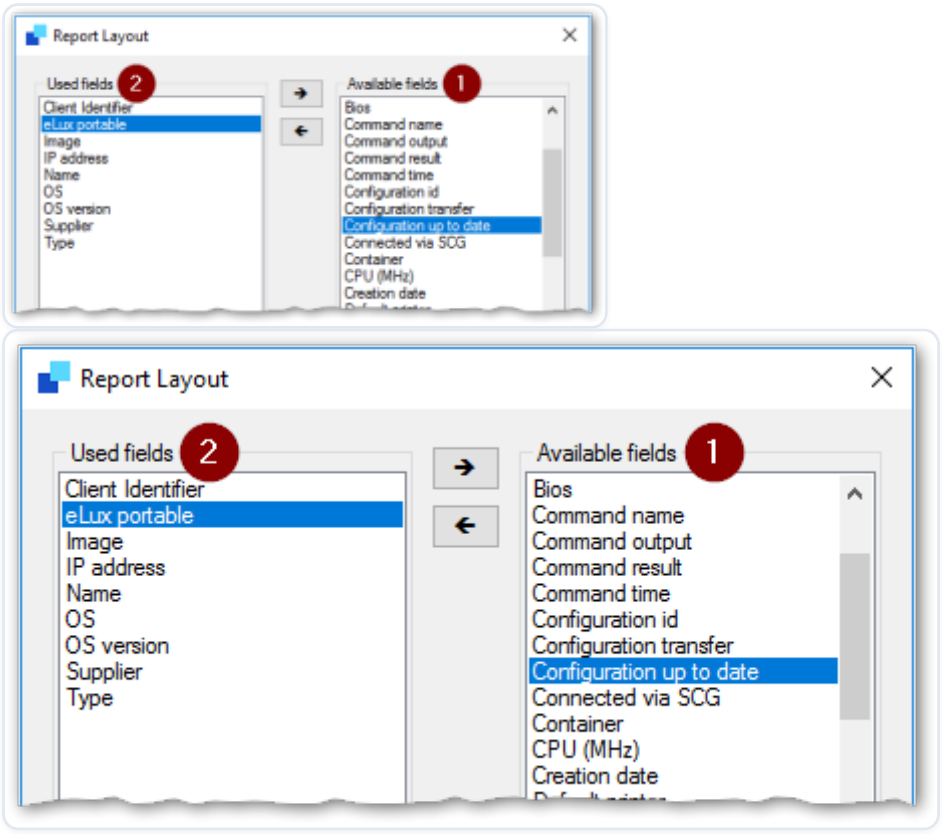
Defining the report layout consists of two steps:

- Select the relevant field that you want to display as columns in the report output (Columns tab).
- Determine the print page layout including the header/footer and page formatting (**Header/ Footer** tab and **Worksheet** tab).

Defining columns

In the report layout, choose the database fields you want to show as columns in the report.For the columns and resulting elements, you can also specify an order and other layout criteria.

Choosing the relevant fields



1. Shows all available fields that you can apply to the report as columns.
2. Shows all fields that will be shown by the report. The order can be defined later on.

Note:

- The **eLux® portable** field shows whether it is a device or an eLux Portable USB stick
- The **Client identifier** field is used for unique identification of devices and elux portable sticks.

1. Select a report and click the button or click **Edit** > **Layout**
2. In the dialog, on the Columns tab, click the **+/-** button.See screenshot above.
3. To add a field, select the relevant field on the right and move it to the left by using the ← button.
4. To remove a field, select the relevant field on the left and move it to the right by using the → button.
5. Confirm with **OK**.

Note:

> Alternatively, to move elements between the two lists, double-click the element or use the and keys.

> In the **Report layout** dialog, in the preview, the selected fields are shown in the **Content** column.

SPECIFYING THE LAYOUT

In the **Report layout** dialog, on the **Columns** tab, define additional properties of the report layout.

1. To change a column header, for the relevant field (row), click into the cell of the first column.Then enter the desired column title and confirm with .
2. Specify further settings:

OPTION	DESCRIPTION
and buttons	Change order of the columns
	Alternatively, use a drag-and-drop operation.
column	Defines whether the text is left-aligned, right-aligned or centered
and columns (column width)	Column width is calculated automatically.You may, however, specify a minimum and maximum width for each column.
	To define a fixed column width, set the same values for minimum and maximum width.
	If the minimum width exceeds the maximum width, the column is not displayed in the report.
	To reactivate the automatic column calculation, set both values to .
	Valid units are millimeters (mm), centimeters (cm), inches (in), points (pt) and pica (pc).

OPTION	DESCRIPTION
Mark rows	Each second row of the report is highlighted to improve readability.
Fill rows	Highlighting of the row is continued until right page margin.
Font	Defines font and font size for headers and rows.
Sorting	Defines field by which the elements of the report are sorted.

3. Confirm with .

> **Note:**
>
> To run the report after you have modified the report layout, click .Only then, the report output is refreshed in the program window.

Merging columns

Under certain conditions it might be useful to merge columns.

EXAMPLE

You have created a list of all ICA® and RDP applications with their names, their assigned organization units and their server addresses.

resulting in the following report:

Organisation unit	Name	ICA Server	RDP Server
Unicon	RDP-Test		rdp-connection
Unicon	ICA-Test	ica-connection	

As only one of the fields **RPP Server** and **ICA Server** contains a value, the two fields can be merged.

MERGING FIELDS

Define the same column title for both fields.

In the example, both titles can be changed to **Server**.

Both fields are assigned to one column. The value of the first column merged that is not empty is displayed:

Organisation unit	Name	Server
Unicon	RDP-Test	rdp-connection
Unicon	ICA-Test	ica-connection

Defining the page layout for print output

Before printing a report, check the page formatting to make sure the page breaks for multiple pages are correct.

You can define headers and footers that are shown on each page.

DEFINING HEADERS AND FOOTERS

1. In the **Report Layout**, on the **Header/Footer** tab, select **Show header** or **Show footer**.
2. Into the text boxes **Left text**, **Centered text** and **Right text**, type the text you want to show on each page, or select one or more fields from the dropdown list:
   - Name of the report
   - Page number
   - Number of pages
   - Date
   - Time
   - Number of items

   You can insert more than one field and you can combine free text with fields.

   Example: Created `<Date>`, `<Time>`
3. If required, click the ... to specify the font and font size.

DEFINING PAGE FORMATS

1. On the **Worksheet** tab, under , select a predefined page size and the page orientation **Portrait** or **Landscape**.
2. If you do not use one of the predefined page sizes, select **User-defined** to type in your individual page dimensions.

3. If required, under **Margins**, modify the predefined values for the margins.

The page layout settings affect direct printing and documents exported to PDF and Excel.

Report filter

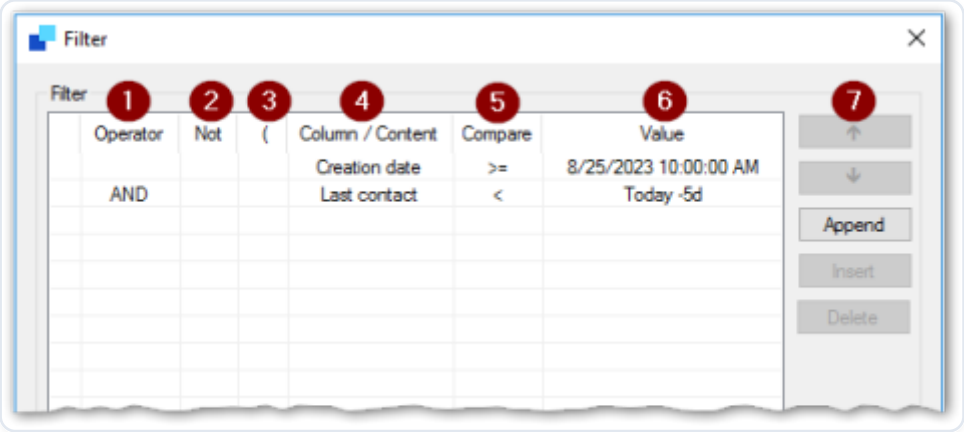
Filtering helps you narrow down the result list pane to those objects matching the specified criteria. If you do not define a filter, all devices, OUs or applications are displayed when running the according report.

> **Note:**
>
> Run the report by clicking after you have modified the filter definition. Only then, the report output is refreshed in the program window.

Report filter dialog

The report filter includes one or more filter expressions. Each filter expression has its own table row.

A filter expression is based on one of the fields that you have selected as columns in the **Report layout** dialog.



LEGEND TO NUMBERS	DESCRIPTION
1	Logical operator AND | OR
	Required for the second and further filter expressions
2	Optional, [NOT] reverses the logical value of the filter expression

LEGEND TO NUMBERS	DESCRIPTION
3	Opening parenthesis (for closing parenthesis, see the logical operator field)
4	Field that has been added as column in the [Report layout] dialog
5	Comparison operator depending on field data type
	= <> < > <= >= LIKE
	LIKE compares two strings. You can use the following wildcard characters
	[_] substitute for a single character
	[%] substitute for zero or more characters
6	Value that matches the data type of the field
7	Functions for editing rows

INSERTING, DELETING AND SORTING TABLE ROWS

Select a row and use one of the following buttons:

BUTTON	DESCRIPTION
Append	Adds a new row. Alternatively, you can click into the next row.

BUTTON	DESCRIPTION
Insert	Inserts a new row above the selected row.
Delete	Deletes the selected row.
↑, ↓	Changes the order of the rows / filter expressions. Alternatively, use a drag-and-drop operation.

Defining filters

1. Select a report, and then click **Filter** or click **Edit** > **Filter**
2. In the **Report filter** dialog, define one or more filter expressions by specifying **column**, **comparison operator** and **value**. To define the column and operator, click the relevant cell and select an entry from the list. For further information, see Report filter dialog.
3. If you define more than one filter expression, join the expressions by using one of the logical operators or in the first column. If required, use parentheses to determine the sequence of operation.
4. Confirm with **OK**.

   The entries are checked and white spaces are removed. If there are any invalid entries, they are highlighted in red. The window can only be closed, if the filter definition is correct.

5. Run the report by clicking .

Special options for date fields

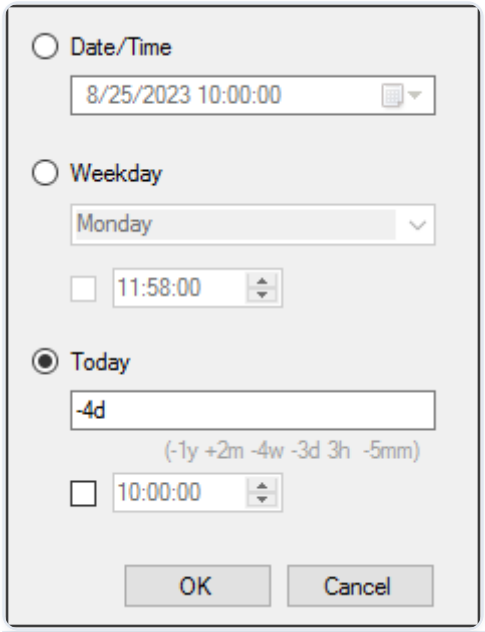
The value of a date field can be set to a fixed date and time. Alternatively, you may refer to relative time periods:

- A period of time to be specified before or after the current day (**Today**)
- A specific weekday within the last week (**Weekday**)

If you work with relative time specifications such as -4d from today, you can run a report, let\'s say each Friday, and always get the results for the last four days without having to change the report definition.

The **Weekday** option refers to the last seven days. The reference to a weekday allows you to query the same time period or point in time with the same report definition every week.

- To define a date or time period for a date field, click in the table cell under **Value**.



OPTION	DESCRIPTION
Date/Time	Fixed date and time
Weekday	If you select Monday, the value refers to the last Monday before the current day at 0:00 o'clock.
	Example
	With =Monday, on a Monday morning, you can show all IDFs of devices that have been updated automatically at the weekend - with status Monday at 0:00 o'clock.
	Example
	With the expressions > Monday AND < Tuesday you will get results referring to the whole Monday (0:00 o'clock to 0:00 o'clock).
Today	Time period relative to the current day, optionally including time
	y Years
	m Months

OPTION	DESCRIPTION
	w Weeks
	d Days
	h Hours
	mm Minutes
	Example
	[-4d] refers to the last four days before the current date

Dynamic Device Groups

Dynamic Device Groups enable administrators to run cross-OU commands for freely definable device groups. For example, you can send a message to all devices with a specific image throughout the whole organization. Or you can run a update on all devices with a specific version, across all OUs. Even to another can be applied to a Dynamic Client Group. The following features can be applied:

- Commands
- Configuration run
- Notifications for software deliveries or firmware updates
- Notifications for relocation

Dynamic Device Groups are based on reports created in the . The reports are exported to the once, and from that point onward, are displayed as a . Commands applicable to OUs or to individual devices can also be applied to .

The report layout must include the field. The report type must be a list of **Devices** or **Assets**. For further information on defining , see Creating Dynamic Device Groups in the Scout Report Generator guide.

Dynamic Device Groups are displayed in the Scout Console in a special window and remain there for re-use until they are deleted. They can be updated any-time with a click.

User rights

When you create Dynamic Device Groups, the user access rights are respected as defined in the administrator management.

By default, executing commands and other functions on a Dynamic Device Group is subject to the object rights configured for the relevant devices and OUs. This can result in a function not being executed because at least one device does not have the required object rights. To avoid this, disable the check for the underlying object rights under **Security** > **Manage administrator**s > **DCG rights**.

Creating Dynamic Device Groups

Creating Dynamic Device Groups consists of two steps that are carried out in Scout Report Generator:

- Creating a report filtered by the desired properties
- Exporting the report as a Dynamic Client Group to the Scout Console

CREATING A REPORT AS BASIS FOR A DYNAMIC CLIENT GROUP

1. Select **File** > **New report** or click the **New report**.
2. In the **New report** dialog, select the **Report type** Create list of devices or Create list of asset entries and confirm with **OK**.
3. For the selected report, on the toolbar, click the **Layout** button to define the report layout.

   In the **Report layout** dialog, on the **Columns** tab, choose those database fields you want to filter by.

   > **Important:**
   >
   > Make sure to include the Client Identifier field. This field is required for unambiguous identification of the devices.
   >
   > Specify the desired layout properties.

   For further information, see Report layout.
4. For the selected report, on the toolbar, click the **Filter** button and define the filter criteria.

   Example: OS version LIKE %4.8% extracts all devices with an OS version 4.8 installed.

For further information, see Report filter.

5. Run the report by clicking the **Create report**.

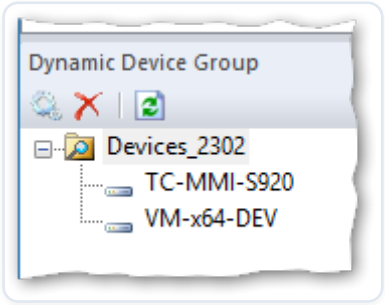
   The report output is created and displayed on the right.

6. Save the report.

EXPORTING A REPORT AS DYNAMIC CLIENT GROUP

1. Select the report that you want to export.
2. Click **Export** > **Dynamic Client Group** to start data export to the Scout Console.

The report is exported and provided as Dynamic Client Group in the Scout Console in the **Dynamic Device Groups** window. All available commands can be applied to the Dynamic Client Group.



For further information, see Using Dynamic Device Groups in the **Scout** guide.

Scout Agent for Windows

Scout Agent for Windows allows you to manage devices running Windows operating systems, such as Windows 10 IoT Enterprise, with the Scout Enterprise Management Suite®.

Windows platforms

Scout Agent for Windows platforms allows you to manage devices running Windows operating systems such as Windows 10 IoT Enterprise by using Scout Enterprise Management Suite. Scout Agent for Windows is charged with the communication between the Windows devices and Scout Server.

In addition to asset management, device configuration and help-desk features you can perform system updates on the devices by using Scout. For version 1.3 and later versions, Scout Agent for Windows can also provide applications defined in Scout on Windows devices.

Licensing

In addition to the Microsoft Windows license for the operating system the client requires a Scout Management license for device management by Scout.

To simplify license distribution, if your devices are not delivered with integral licenses, you can use so-called **loose licenses**. These unallocated licenses can be distributed by the Scout Server to unlicensed devices.

> **Note:**

> Without a management license, a client cannot be run in **Managed** Mode.

> For further information, see Scout Management license in the **Licensing and Subscription** White Paper.

Installation and interface

- Installing Scout Agent for Windows
- Configuring Scout Agent for Windows
- Configuration window

Installing Scout Agent for Windows

1. Download the latest Scout Agent for Windows version from our technical myelux.com portal and unpack the ZIP file.
2. Run the `ScoutAgentBundle.exe` file and follow the instructions of the installation wizard.

After completed installation the Windows taskbar shows the systray icon Scout Agent.

Unattended installation

For unattended installation, use the command-line parameters:

```
install /quiet
```

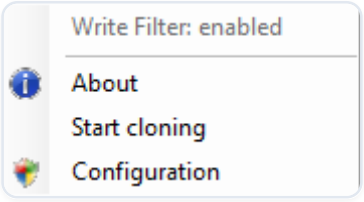
Example:

```
ScoutAgentBundle /install /quiet
```

Configuring Scout Agent for Windows

> **Note:**
>
> The Scout Agent for Windows supports the languages English and German.[1[The shown language is determined by the display language selected in the Windows control panel.]]

1. Show all systray icons of the Windows taskbar and right-click **Scout Agent > Configuration**.



2. Click **Configuration**.

   The Scout Agent for Windows is loaded and requests the device password.

3. Enter the device password of the client.

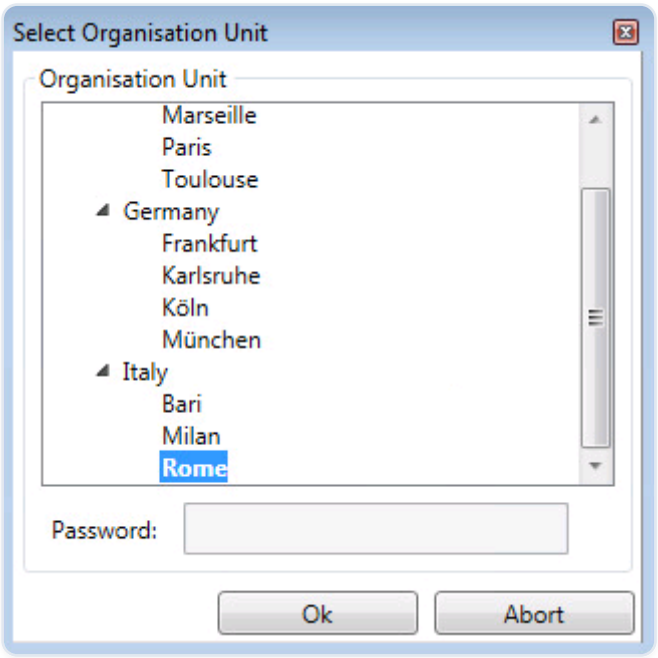
   *For initial start-up the password is elux. During first connection to the Scout Server, the password is changed to the value defined there (see **Scout Console > Options > Base configuration > Security > Local security settings > Device password**).*

   *The **Scout Agent for Windows Configuration** dialog opens. The Scout Server address is predefined as ScoutSrv. For further information, see Self-registration of devices in the **Scout** guide.*

4. In the **Scout Server Address** field, enter the IP address or FQDN of the Scout Server.

   The Scout Agent for Windows sets up a connection to the Scout Server.

5. Next to **Organisation unit**, click the ... button to display the **Select Organization Unit**dialog, and then assign the device to an OU.



6. Confirm with **OK** and **Apply**.

The device receives the configuration data of the selected OU from the Scout Server. The configuration becomes active after the next restart.

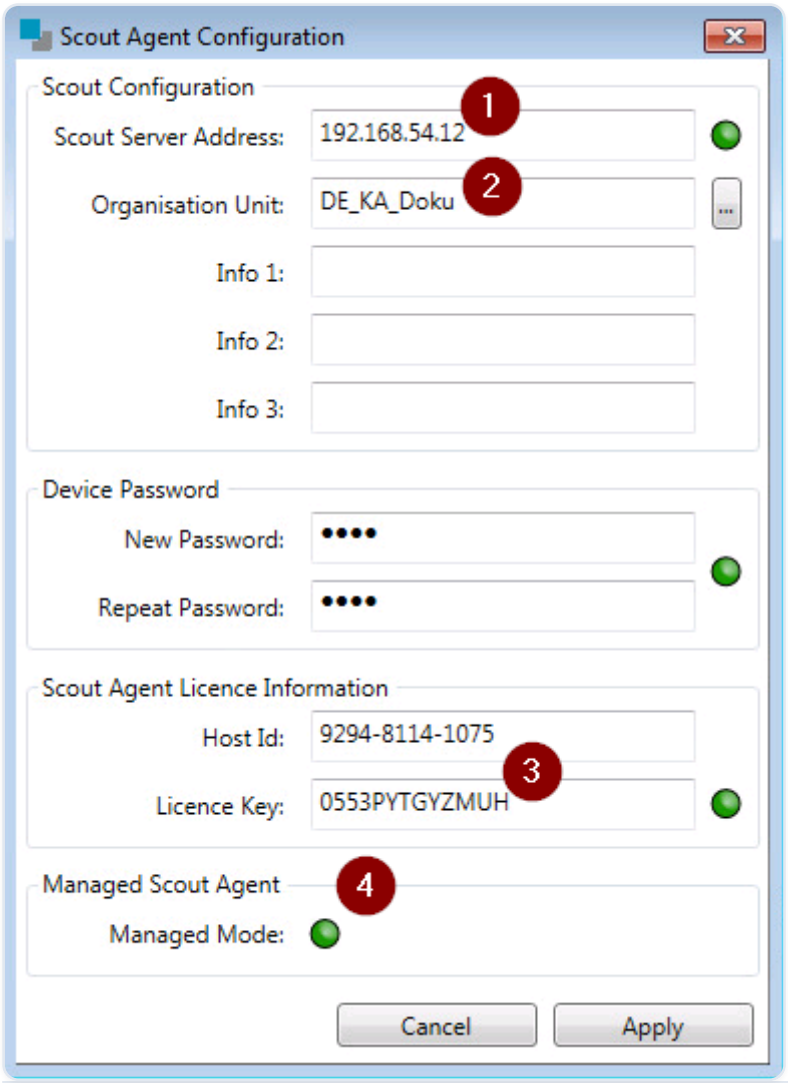
> **Note:**

> On each client restart, the Scout Agent for Windows is started automatically. As long as the Scout Agent for Windows is active, the **Scout Agent** is displayed in green.

Configuration window

After you have installed the Scout Agent for Windows, the Windows taskbar shows the systray icon **Scout Agent**.

To open the configuration window, click the systray icon.



	LEGEND TO NUMBERS
1	IP address or FQDN of the connected Scout Server
2	OU the Windows client is assigned to
3	License key, only required for Embedded 7 clients with pre-installed image
4	If the Windows client is connected to the specified Scout Server, and if it is assigned a Scout Management license, the **Managed** mode is active.

Scout management features

By using Scout Enterprise Management Suite® you can manage different aspects of Windows devices.

REQUIRES
• The devices are provided with the required licenses.
• The devices are connected to a [Scout Server] and are in **Managed** mode.

Scout Agent for Windows controls the communication between the Windows client and Scout. A systray icon helps you access the Scout Agent for Windows configuration at any time. The important point here is that the **Managed** mode is active.

If you have used the client without Scout so far, you only need to specify your Scout Server and, if required, enter a license key in the **Scout Agent Configuration** to enable **Managed mode**.

If, on the Windows system, relevant software components or required access rights are missing, running Scout management features may be restricted.

Security

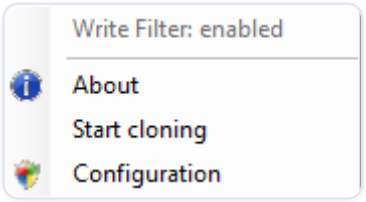
- File-based write filter
- Device password
- Mirroring

File-based write filter

The file-based write filter (FBWF) is a device configuration option that you can enable and disable in the Scout Console (**Setup > Hardware**).

Changing the status of the write filter requires a restart of the client. So do any modifications on the configuration. For further information, see Device configuration.

The current status is shown in the systray context menu of the .



> **Note:**

> The write filter status is determined by the Scout Agent for Windows. If the client is not in **Managed** mode, the Scout Agent for Windows service is not started on the client. In this case, the context menu of the Scout Agent for Windows displays the status `Write Filter: Unknown`.

> Alternatively, you can retrieve the write filter status from the command line running the following command as administrator: `fbwfmgr /displayconfig`

Device password

You can change the locally saved device password in the Scout Console for all devices:

- Click **Options > Base device configuration > Security > Local security settings > Device password**

> **Important:**

> By using this feature you change the device password of **all** devices managed by this Scout Server.

> The password is required to access the **Scout Agent for Windows configuration** dialog (systray icon **Scout Agent > Configuration**).

Moreover, the device password ensures that the client can only be managed by a Scout Server provided with the same password.

For further information, see Local device password in the **Scout** guide.

Mirroring

When you establish a mirror session, all security options apply that have been set in the Scout Console for that OU (for Scout Agent for Windows 1.3 and later versions).

During the mirror session a system message is shown on the client device allowing the user

- to stop mirroring at any time
- to grant only read access to the administrator (for Scout Agent for Windows 1.3 and later versions).

For further information, see Mirroring client screen.

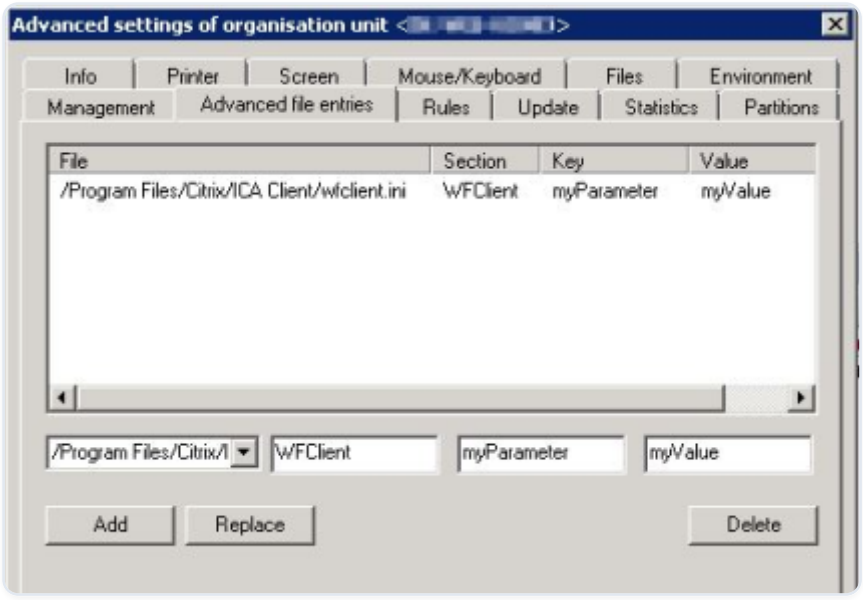
System updating

By using the Scout Console, you can provide your Windows devices with firmware updates. Also, you can modify configuration files and settings of the Windows client, and transfer any files you want to have on the client.

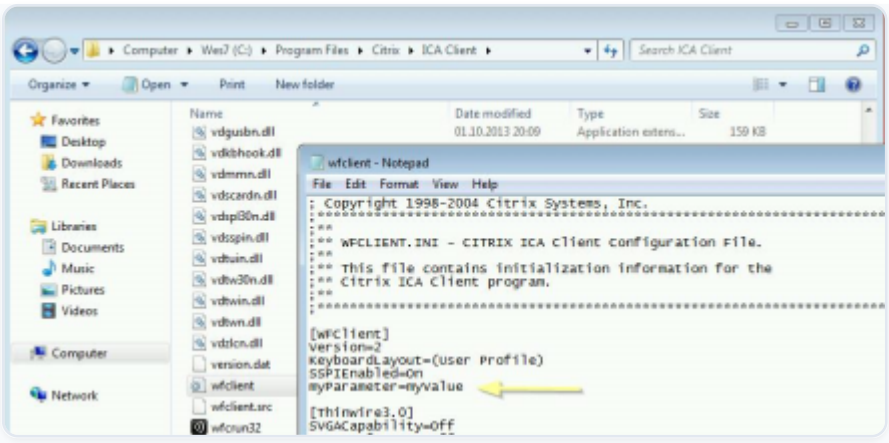
- Advanced file entries
- Transferring files to the Windows client
- Firmware update

Advanced file entries

By using the **Advanced file entries** feature of the Scout Console, you can set and modify parameters and their values in `.ini` files. This feature can also be used to modify particular preferences and parametrize program calls on the Windows client.



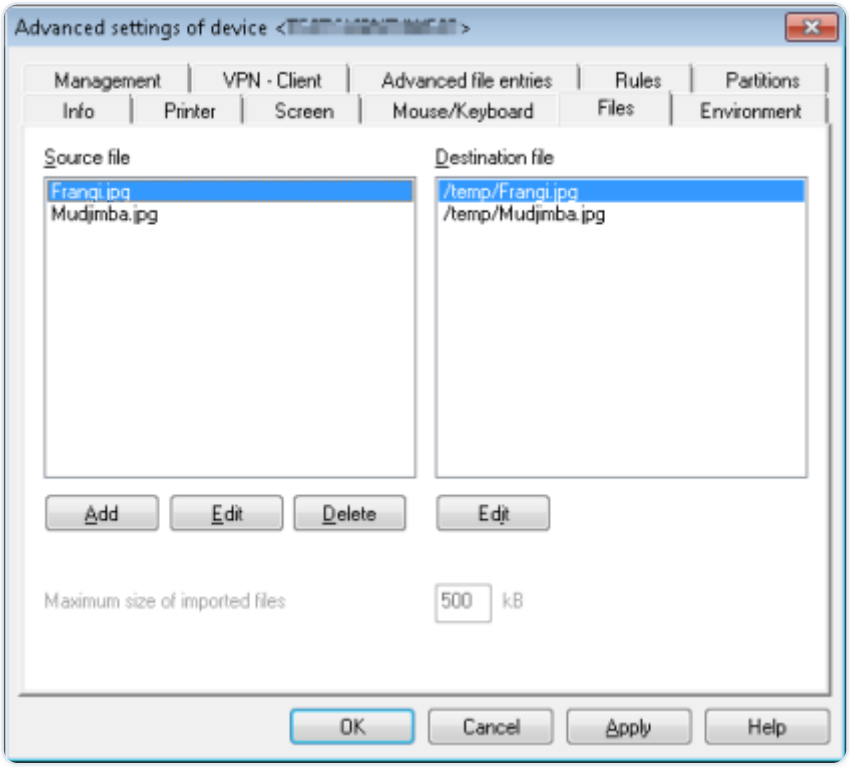
> **Note:**
>
> For the file path on the Windows client, do not enter a drive letter (it is always drive C:) and use slashes instead of back slashes.



For further information, see Advanced file entries in the **Scout** guide.

Transferring files to the Windows client

By using the Scout feature **Files configured for transfer** in the console, you can transfer any files from Scout to the Windows devices.



Note:

For the file path on the Windows client, do not enter a drive letter (it is always drive C:) and use slashes instead of back slashes.

> For further information, see Files configured for transfer in the **Scout** guide.

Firmware update

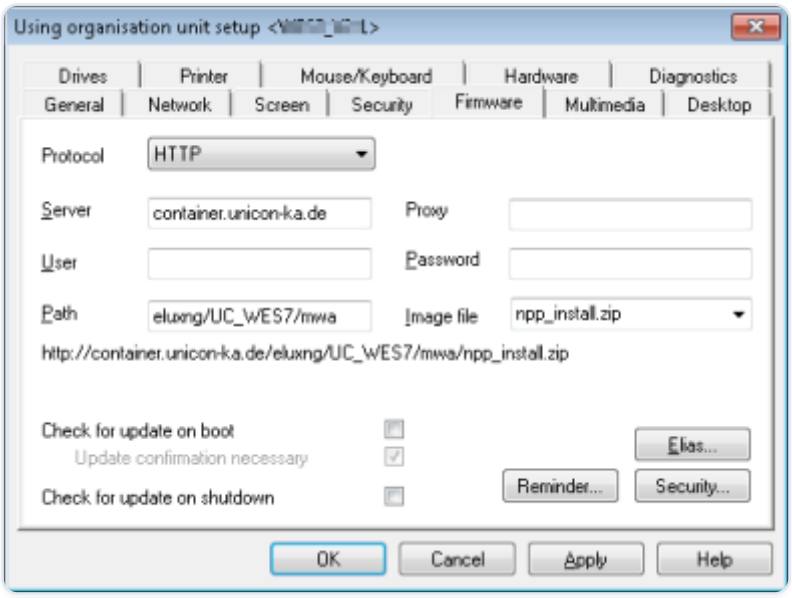
Installing further software on your Windows devices is done by performing a firmware update. For that purpose, a ZIP file is transferred containing an installation method and the software to be installed. The firmware update is configured in the Scout Console, in the device configuration of the relevant devices.

> **Note:**
>
> Due to security reasons, we recommend that you first thoroughly test the installation method in combination with the software to be installed on a reference system.

CONFIGURING FIRMWARE UPDATE

1. In the Scout Console, for the relevant devices, open **Device configuration > Firmware**.
2. In the **Image file** field, specify a ZIP file that contains an installation method in the form of the file `install.cmd` and the software to be installed.

   If the complete path name is required, you must use the environment variable %CD% in the installation method `install.cmd`. Example for installing a Microsoft hotfix with Windows Update Installer: `wusa.exe %CD%windows6.1-KB28xxxxx-v2-x86.msu /quiet`
3. If you want to allow the users to postpone updates, configure **Update deferment** by clicking **Reminder**...
4. Confirm with **Apply** and **OK**.



For further information, see Firmware update in the **Scout** guide.

5. For the relevant devices, perform an **Update** command. For further information, see Remote commands.

UNINSTALLING SOFTWARE

1. For the relevant devices, in **Setup > Firmware > Image file**, specify a ZIP file that contains only an installation method but no software.
2. For the relevant devices, perform an **Update** command.

Device configuration

Any modifications to the device configuration are carried out in the Scout Console. They can relate to a particular device, to a particular OU or to all devices.

Modifying device configuration

1. Open the device configuration (Setup) dialog for the relevant devices. For further information, see Accessing device configuration in the **Scout** guide.
2. Switch to the relevant tab and edit the relevant options.
3. Confirm with **Apply** and **OK**.

On the restart of the relevant Windows devices, the Scout Agent for Windows identifies modifications of the configuration. Normally, the modifications become active after another restart.

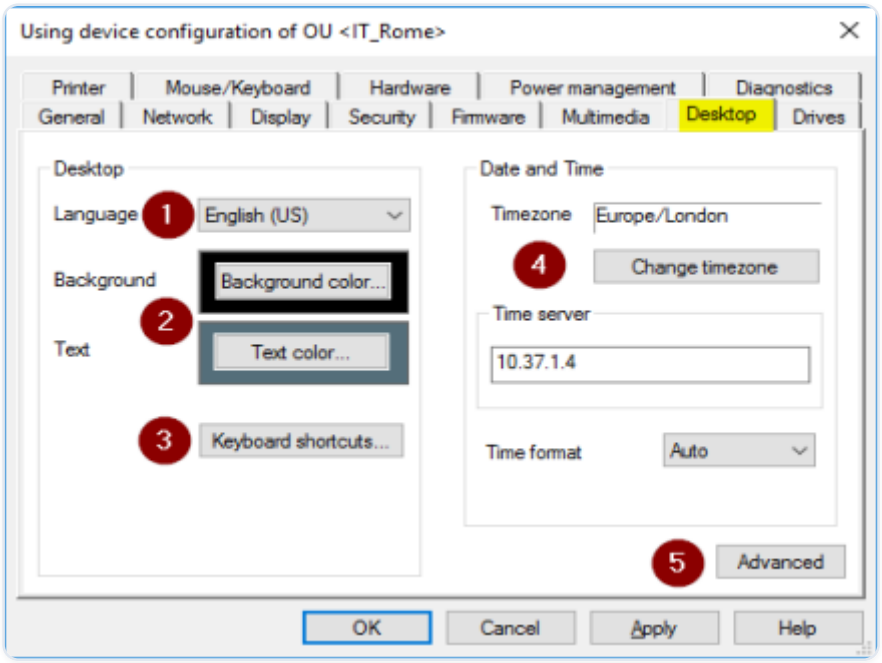
Initial state

Any modifications to the configuration can be undone by resetting the Windows client to initial state by means of the relevant remote command. For Windows Embedded devices, the initial state corresponds to the status right after installing the Windows Embedded image.

Desktop

By using the device configuration in the Scout Console, you can modify the following desktop settings:

- Language
- Time zone



> **Note:**
>
> Language is related to the display of desktop elements but not to text services and input. The Scout Agent for Windows is shown in German if you have selected the German display language. For all other selected languages, it is shown in English.

Screen

By using the device configuration in the Scout Console, you can modify the following screen settings:

- Number of monitors
- Resolution
- Rotation
- Power saving

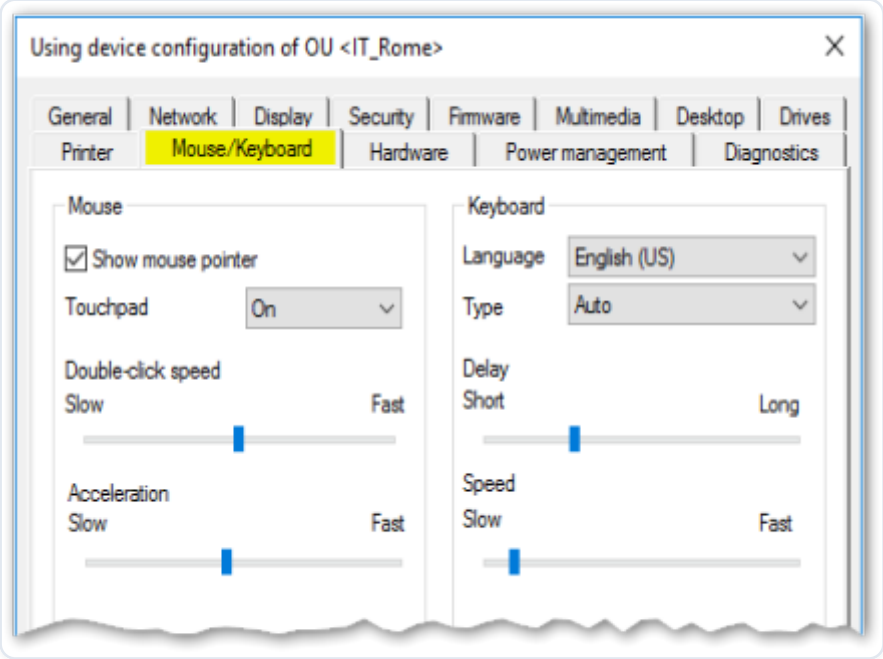
> **Note:**
>
> If the write filter is enabled, modified resolution values are assigned to the Windows client only after the user has logged on.

> For further information, also on multiple monitors, see Screen settings and multiple monitors in the **Scout** guide.

Keyboard

By using the device configuration in the Scout Console, you can modify the following keyboard settings:

- Language
- Delay
- Speed



For further information, see Configuring the keyboard in the **Scout** guide.

Diagnostics

In the Scout Console, in **Device configuration > Diagnostics**, you can enable and disable enhanced debugging on the client.

If the **Debug level** is active, the feature **Device diagnostics** helps you run predefined commands on the client and retrieve a set of configuration and log files to a greater extent than without enhanced debugging.

If you require technical support from Unicon™, enable enhanced debugging before you perform **Device diagnostics.**

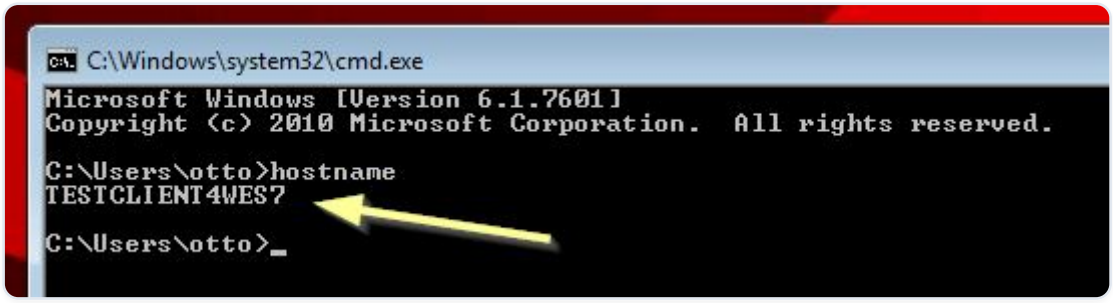
Device diagnostics is performed by using an online command, for further information, see Device diagnosis in the **Scout** guide.

> **Note:**
>
> Make sure to switch off the **Debug level** after having performed device diagnostics. Otherwise you risk to exceed the memory capacity of the device.

Changing the device name

1. In the Scout Console, in the tree view, select a device.
2. Press F2 and enter a new name for the device
3. Press **return**.

The host name on the Windows client is changed accordingly.



Applications

-- for Scout Agent for Windows 1.3 and later versions --

The Scout Agent for Windows can provide the following applications defined in the Scout Console for Windows devices:

- Citrix-Receiver
- Microsoft Remote Desktop Connection
- VMware Horizon Client

- Storefront™
- Browser
- File manager
- Text editor
- Movie player
- On-screen keyboard (English)
- Magnifier
- Custom applications

Installing applications

1. In the Scout Console, for the relevant OU, define the required applications. For further information, see Adding applications in the **Scout** guide.
2. In the Scout Console, configure the application properties. Along with application-specific options the following general options are provided:

OPTION	DESCRIPTION
Start automatically 1 (Selecting or clearing the Start automatically option requires a second restart of the client to become active.)	The application is added to the Windows autostart folder and is automatically started after each system start.
Desktop icon	The application's icon is added to the desktop.

   The Windows system does not provide all of the application options. For further information, see Specifics for applications on Windows clients.
3. In the Scout Console, for the relevant OU, from the context menu, choose **Notifications > Initiate refresh**...

On the next restart of the devices, the application is available on the desktop under **Scout Applications**.

> **Note:**
>
> The applications are provided for all users of the Windows system.

Specifics for applications on Windows devices

Applications on the Windows client that have been defined in the Scout Console can be subject to restrictions or specifics.

GENERAL OPTIONS THAT ARE NOT AVAILABLE FOR WINDOWS DEVICES

- **Application restart**
- Delay for **Start automatically**
- Free parameters
- User-defined desktop icons

MICROSOFT REMOTE DESKTOP CONNECTION

- In the Scout Console, in the application properties **Advanced RPD settings > Advanced**, the **Enable compression** option must be selected.
- pass-through logon is not available
- Passwords sent from Scout are ignored.
- Restricted multi-monitor support

VMWARE HORIZON CLIENT

Scout supports the following options:

- Server and protocol
- Logon with username, password and domain, pass-through logon
- Window size

Further configuration can be done on the related View Connection server.

STOREFRONT™

- Each specified URL receives its own start menu entry (and if defined, its own desktop icon)
- The first start of a StoreFront application might take some time
- The **Advanced options** are not available.

BROWSER

Internet Explorer is used as browser.

The following options are supported:

- Start page
- Enabling kiosk mode

ON-SCREEN KEYBOARD

Requires defining a local Custom application with **Parameter**OSK (no case-sensitivity).

CUSTOM APPLICATION

In the **Parameter** field, make sure to enter only values that exist on the Windows system.

Remote control

Controlling Windows devices remotely means that, in the Scout Console, you schedule and perform commands either on a particular device or on an OU or Dynamic Client Group.

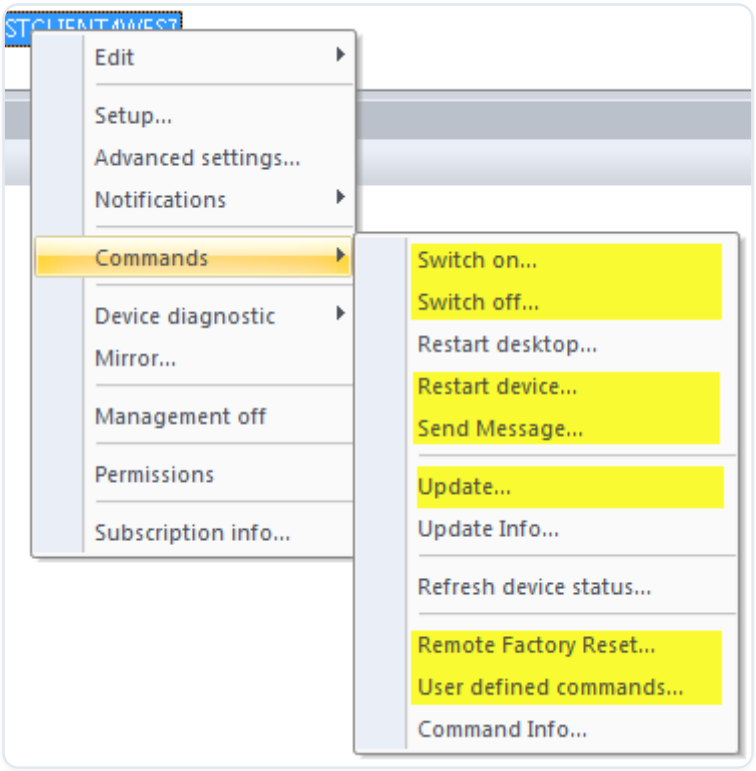
- Remote commands

Remote commands

In the Scout Console, by using the context menu of a device or OU, you can perform the following remote commands:

- Switch on
- Switch off
- Restart device
- Send message
- Update
- Factory reset
- User-defined commands

The commands can be performed immediately, or scheduled to be run once or periodically.



In addition, to an individual client device, you can apply the features **Device diagnostics** and **Mirroring**. For further information, see Helpdesk features.

For further information on how to apply and schedule remote commands, see Online commands in the **Scout** guide.

Asset management

- Device data in Scout

Device data in Scout

On each system start, the client devices connect to their Scout Server and send their asset data such as hardware properties. These data are shown in the Scout Console in the **Properties** window for the selected device.

Also, the asset data can be analyzed with the help of individual queries built with Scout Report Generator. There is information on the hardware such as device type, UEFI/BIOS version, flash size as well as information on the OS version and locally saved licenses.

For further information, see the Scout Report Generator guide.

Helpdesk features

- Mirroring client screen
- Device diagnostics

Mirroring client screen

To launch a mirror session, the administrator performs a **Mirror**... command on the client (for Scout Agent for Windows 1.3 and later versions).

When you establish a mirror session, all security options apply that have been set in the Scout Console for that OU:

- Password for mirror session
- Read access only
- User must confirm
- Encrypt mirror session
- Allow Scout only
- Logoff after disconnect

During the mirror session, a system message is shown on the client device which remains in the foreground but can be moved by the user. The system message provides an **Abort** button that allows the user to stop mirroring at any time and to grant only read access to the administrator.

For further information, see Mirroring devices in the **Scout** guide.

> **Note:**

> If the **User must confim** option is active, the first attempt to connect is refused. Reconnecting a second time does work.

Device diagnostics

Device diagnostics is a useful feature of the Scout Console. It helps you run predefined commands on the client and retrieve protocol and configuration files from the client for diagnostic purposes. The requested client files support the administrator in error analysis and are required when opening a support ticket.

You can also use this feature to request any definable files from the client.

For further information, see Requesting diagnostic files in the **Scout** guide.

Scout Enterprise Management Suite®

The Scout Enterprise Management Suite, or Scout for short, is a lean and easy-to-use end-point management solution for devices running eLux®.

Overview

The Scout Enterprise Management Suite®, or Scout for short, is a lean and easy-to use end-point management solution for devices running eLux. Administrators can easily manage, configure, upgrade, update, secure and monitor even large numbers of devices.

This manual serves as a reference for the Scout Console functionality and related tools. It includes features from version 15.3 onwards. Scout and eLux features that have been added since version 2101 are referred to via footnotes.

- Features of the Scout Enterprise Management Suite
- Communication between devices and Scout Server
- Installation
- Keyboard shortcuts

Features of the Scout Enterprise Management Suite®

Scout Enterprise Management Suite is the management solution for cloud devices, Hybrid Clients, mobile devices and PCs running the operating system eLux. In addition, Windows-based devices can be managed by using basic Scout management features.

Scout Enterprise Management Suite consists of several components. Most of the components listed below are included in the standard installation but can be optionally excluded when choosing custom installation.

COMPONENT	DESCRIPTION	INSTALLATION
Scout Server	The service controls and manages eLux devices as well as Windows devices on which Scout Agent for Windows has been installed.	ScoutInstaller.exe

COMPONENT	DESCRIPTION	INSTALLATION
Scout Console	User interface for the management of eLux devices and Windows-based devices on which Scout Agent for Windows has been installed	ScoutInstaller.exe
	Server communication only via database	
	Multiple consoles can be managed with one Scout database	
Scout Board	Web user interface for the management of eLux devices and Windows-based devices on which Scout Agent for Windows has been installed	ScoutInstaller.exe
Recovery service	Customized TFTP service to realize a PXE recovery environment for eLux endpoints	ScoutInstaller.exe
ELIAS1	Legacy dialog program eLux Image Administration Service (ELIAS) for creating individual image definition files (.idf) for modular firmware updates of the eLux devices. The legacy ELIAS will be replaced by ELIAS 18.	ScoutInstaller.exe
ELIAS 18	New web-based platform-independent ELIAS application for creating individual image definition files (.idf)	separate
		(EliasInstaller.exe)

COMPONENT	DESCRIPTION	INSTALLATION
Scout Report Generator	Tool for creating freely definable reports across all currently existing devices, applications and OUs in the Scout Console	ScoutInstaller.exe
	Is launched from the Scout Console	
Scout Keep Alive service2	Windows service processing keep alive packets from eLux devices	ScoutInstaller.exe
Web API	Application programming interface for the management of eLux devices and for the management of Windows-based devices on which Scout Agent for Windows has been installed	ScoutInstaller.exe
Scout Command Interface	Command line interface for Scout commands	ScoutInstaller.exe
Scout Database Connection Editor	Tool for modifying database connection settings of the Scout Server and Scout Console	ScoutInstaller.exe

Further products

COMPONENT	DESCRIPTION	INSTALLATION
Scout Cloud Gateway	Cloud gateway with VPN backend for convenient connection of devices from the Internet to a Scout infrastructure	separate

COMPONENT	DESCRIPTION	INSTALLATION
Scout Agent for Windows	Service providing an interface for Windows-based devices to be managed through [Scout Enterprise Management Suite]	separate

The features are described in the following guides:

- Scout Enterprise Management Suite: Configuration, control and management of the eLux devices using the Scout Console
- Scout Board
- Scout Cloud Gateway
- ELIAS
- ELIAS 18
- Scout Report Generator
- Scout Command Interface

Recovery procedures for eLux devices are described in the **eLux Recovery procedures** short guide.

> **Note:**
>
> To compose and use your own image files, in addition to the Scout Enterprise Management Suite installation, you need an ELIAS 18 installation. If you use the legacy ELIAS instead, you need an eLux container for the software packages, see Installing a container.

Communication between devices and Scout Server

> **Note:**
>
> The certificate-based management protocol ensures secure communication between Scout Server and devices. For further information, see Certificate-based management protocol in the **Installation** guide.

> During system start, the eLux devices connect to their Scout Server and verify if their configuration data is up-to-date. Updated data may concern device configuration, application definitions, files defined for transfer and advanced file entries. For further information about identifying and transferring updated configuration data, see Configuration method.

The communication between a device and the server may proceed in three ways:

- The device accesses the Scout Server. The Scout Server has no updated configuration data.
  - The device continues starting up with its configuration.
- The device accesses the Scout Server. The Scout Server reports new configuration data and transfers the data to the device.
  - The device restarts using the new configuration.
- The device cannot access the Scout Server due to network or other problems which result in a management timeout, see Advanced network settings.
  - The device continues starting with its configuration. Depending on the handshake settings, the device retries connecting to be able to synchronize the configuration data. For further information, see Optimizing with handshake.

Updated configuration data can relate to device configuration (setup), application definition, files configured for transfer and advanced file entries.

During operation of a user's device there is no data exchange between the Scout Server and device. During shutdown, the device reports its current status to the Scout Server (except for VPN connections).

Installation

All contents relating to installation and setup are covered by the **Installation** guide. For further information, see Installing Scout Enterprise Management Suite in the Installation guide.

Keyboard shortcuts

KEYS	SELECTED ELEMENT	DESCRIPTION
CTRL+SHIFT+INSERT	Individual OU	Opens the **Advanced device configuration** of the selected OU

KEYS	SELECTED ELEMENT	DESCRIPTION
\^	**Applications**	Opens the **Application Properties** dialog to define a new application
\^	**Devices**	Opens the **Information** dialog to enter a MAC address
CTRL+SHIFT+DELETE	Individual OU	Deletes the selected organization unit
\^	Individual application	Deletes the selected application
F2	Individual OU	Rename the selected organization unit
\^	Individual device	Rename the selected device
\^	Individual application	Rename the selected application
F5	_	Updates the configuration of all devices
CTRL+F	_	Activates the Search for devices, OUs or applications field for simple search
CTRL+SHIFT+F	_	Opens the Search for devices, OUs or applications window for advanced search
CTRL+X	Individual device	Cuts the selected device
CTRL+V	**Devices** or individual device	Pastes the device from the Clipboard to the selected position
CTRL+A	Individual application or device in the **List** window.	Selects all applications/ devices in the **List** window

KEYS	SELECTED ELEMENT	DESCRIPTION
CTRL+E	Individual device	Performs a **setup comparison**
CTRL+P	_	Opens the **Print dialog** to print the list of available devices

Interface of the Scout Console

- Organizational structure
- Icons in the tree view
- Windows
- Status bar
- Display settings
- Changing the language
- Search for devices, OUs or applications
- Moving and copying elements
- Switching OU to top-level

Organizational structure

The main window of the Scout Console shows a tree view in the upper left corner that reproduces the organizational structure with all managed devices. When you log in for the first time, you will see the organizational units **Lost&Found** and **Enterprise** which are created by default. The latter serves as the top node of your organizational structure.

At the top level, three applications are provided that you can use to connect to a back-end: **RDP**, **StoreFront** and **VMware Horizon**. For further information, see Defining applications.

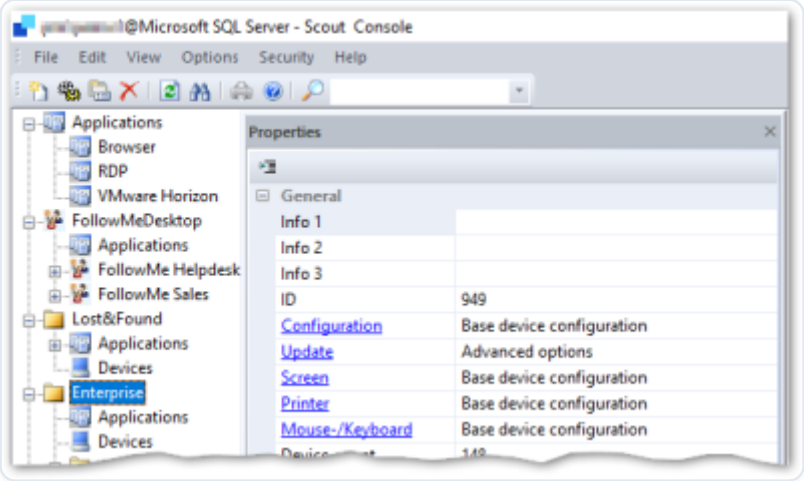
To each organizational unit -- hereafter referred to as OU -- you can add applications, devices and other OUs. Each OU can contain subordinate OUs, applications and devices.

By default, inheritance is active. Application definitions and device configuration data are inherited by subordinate OUs and devices.

If you add a new device to an OU, it will receive all application definitions and device configuration data from this OU.

Individual devices and applications can be moved from one OU to another by using a drag-and-drop operation or the Clipboard. The devices then are assigned the properties of the new OU (if inheritance is active).

For further information, see Concept



For the element selected in the tree view, you can view various details in the **Properties** window.

Adding a new OU

1. For the relevant OU, from the context menu, choose **Add > Organization unit**.

   The **Advanced device configuration** dialog opens.

2. Enter a unique name for the new OU.

   > **Note:**
   >
   > Back slashes are not supported.

3. If required, enter further information into the **Info** fields and edit fields on the other tabs.
4. Confirm with **Apply** and **OK**.

The new OU is shown in the tree structure. It contains the folders **Applications** and **Devices**.

Icons in the tree view

ICON	DESCRIPTION
	Organization unit (OU)

ICON	DESCRIPTION
	Applications
	Device, not connected to Scout yet (Example: Device import)
	Device, running and ready for operation
	Device, switched off or not available
	Device, desktop is initializing or the log-on screen is shown
	Device, firmware update is running
	Device, missing management license

Windows

Click **View > Windows** to show further windows next to the organizational structure:

WINDOW	DESCRIPTION
Properties	Properties of the selected application, OU or device
	For further information, see Properties.
Assets (only for devices)	Hardware information of the selected device
	Is shown as a tab of the **Properties** window
	For further information, see Hardware information.
Dynamic Device Groups	Shows the defined Dynamic Device Groups (list view)
Independent configurations	Shows OUs and devices that do not use the parent device configuration (list view)

WINDOW	DESCRIPTION
	For further information, see Blocking inheritance - independent device configuration
Compare configurations	Shows differences in the device configuration between devices or OUs
OU devices/applications	Devices or applications of an OU view without icons (list view)
	Double-clicking on a device shows the corresponding device in the tree view. This feature can be disabled, see below.
All devices	Shows all devices without icons (list view)
	The device data are only loaded from the Scout database when you click the **Refresh** button. This is meant to prevent unintentional loading of huge amounts of data.
	The context menu offers the same functions as in the tree structure. Several functions such as commands can be applied to multiple devices. To do so, select the devices by pressing CTRL or SHIFT
	Double-clicking on a device shows the corresponding device in the tree view. This feature can be disabled, see below.
	To search window content, use the **Search** field of the toolbar, type (/en-us/unicon-elux-scout/2603-ltsr-cu1/admin_guides/scout/interface_/interface_search.html).

Sorting columns

Click a column header to sort rows.

Showing or Hiding properties

Click the button to define which properties you want to show. Alternatively, use the context menu.

Disabling the Double-click shows device feature

By default, double-clicking a device within a device list causes the tree view to show the corresponding device. This behavior can be disabled.

Define the following registry entries with value type DWORD:32 and value 1:

HKEY_CURRENT_USERSoftwareUniConScoutSettings

DisableDoubleClick_OUDevices_View

DisableDoubleClick_AllDevices_View

DisableDoubleClick_DCG_View

Properties

The properties of the selected application, OU or device are displayed.

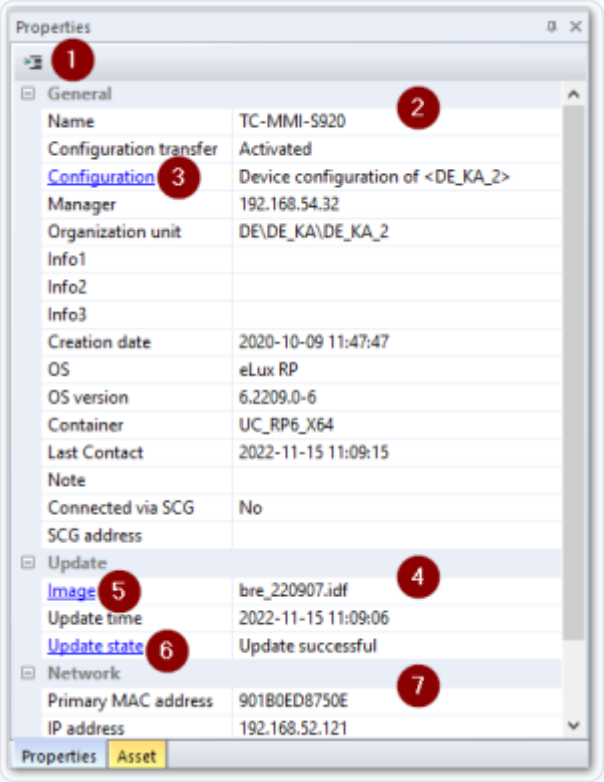
Many properties come from the device configuration (/en-us/unicon-elux-scout/2603-ltsr-cu1/admin_guides/scout/setup/config_method.html).

Some device properties come from the Scout Server and are updated dynamically.

Date and time fields are shown according to the international standard ISO 8601.

> **Note:**

> By default, not all fields are displayed. To show or hide fields, click the icon



The figure shows the properties of a selected device.

1. Show, hide or move fields.
2. View general information on the device.
3. Click to open the device configuration the device is assigned to.
4. View information on the last firmware update.
5. Click to open the assigned image in ELIAS (provided **Options > ELIAS settings** is configured).
6. Click to open the update log.
7. View network information on the device.

DEVICE PROPERTIES

Examples:

OPTION	DESCRIPTION
Name	Host name of the device
Configuration	Origin of the device configuration, mostly inherited from parent instance
Manager	IP address of the Scout Server the device is assigned to

OPTION	DESCRIPTION
Info1-3	The **Info** fields are shown on the device in the **Configuration panel** under **Information** and can be enabled for users for editing (user rights). They are already provided in the First Configuration Wizard.
OS version	Version of client operating system
Container	eLux® container configured for the device in the firmware device configuration
Update time	Time stamp of the last firmware update
\^	For further information, see Command results and update information per device
Last contact	Time stamp of the last contact between server and device
\^	The field is not only updated on a device restart but also on each successful connection set-up from the server to the device.
Status	Example: Switched on
\^	The status of a device is triggered by the 'keep alive' mechanism and by active status messages of the device on start, shutdown, logoff, performing updates and more.
Status time	Time stamp of the last status refresh
Connected via SCG[1] [from Scout 15 2209 and eLux RP 6 2209]	Indicates whether a device is connected via [Scout Cloud Gateway]
SCG address[2] [from Scout 15 2209 and eLux RP 6 2209]	For SCG connections: Shows the FQDN and IP address of the Scout Cloud Gateway
Primary MAC address	Device address of the hardware (MAC=Media Access Control)

OPTION	DESCRIPTION
Simple device identifier	Temporary device identifier users may request for support cases
Client identifier	Globally unique identifier for a device or eLux Portable USB stick
Public address	If a device is connected via VPN, its public IP address is displayed next to its actual **IP address**. You can filter by the public IP address in the Scout Report Generator.
Partitions	The system, setup and update partitions are displayed with their respective sizes.
Screen (Screen info)	Shows the configured resolution and frequency of all connected monitors
\^	For multiple monitor layouts, additionally the output port and defined properties such as primary screen, rotation and the monitor's position (row/column) are displayed.[3] [from Scout 15 2204] If a monitor is not active, this is shown as a dedicated property.[4[[ ]from Scout 15 2209]] The values for row and column are then shown as [-1].

OU PROPERTIES

Examples:

OPTION	DESCRIPTION
OU	Shows the ID of an OU. In addition to the decimal value, you can show the hexadecimal value. This requires a new registry entry. Key: [HKEY_CURRENT_USERSoftwareUniConScoutSettings] ,Value name: DisplayHexOUID, Value type: DWORD:32, Value data: 1
Device count	Number of devices in the OU and the subordinate OUs

QUICK LINKS IN THE PROPERTIES WINDOW OF DEVICES AND OUS

- Make use of the links shown in blue to quickly browse the relevant configuration and information in each context. Double-click the links:

SELECTED ELEMENT	OPTION	DESCRIPTION
Device	**Configuration**	Opens the relevant **Device configuration**
Device	**Image**	Opens ELIAS with the image configured for this device in the relevant container
\^	\^	The connection to ELIAS is made with the data configured in the ELIAS settings of the [Scout Console]. For further information, see Access to applied images
Device	**Update State**	Double-click or [...] opens the [Update-Info] for the device providing information on performed updates. For further information, see Command results and update information per device
OU	**Configuration**	Opens the relevant **Device configuration**
OU	**Update**	Opens the relevant **Update** settings in the **Advanced device configuration** or **Advanced options**.
OU	**Screen, Printer, Mouse/ Keyboard**	Opens the relevant configuration (**Device configuration or Advanced device configuration**) for

SELECTED ELEMENT	OPTION	DESCRIPTION
		Screen, Printer or Mouse/ Keyboard

APPLICATION PROPERTIES

The details of an application depend on its type (/en-us/unicon-elux-scout/2603-ltsr-cu1/admin_guides/scout/app_definition/app_general/app_add.html).

Hardware information

For the selected device, in the **Properties** window, on the **Assets** tab, device-specific hardware information is shown:

- Hardware information of the device
- Hardware information of connected devices such as USB devices and monitors

> **Note:**
>
> By default, not all fields are displayed. To show or hide fields, click the icon.
>
>

1. Information on the managed client device
2. Information on connected devices
3. Show, hide or move fields

HARDWARE INFORMATION OF A DEVICE

Examples:

OPTION	DESCRIPTION
Device type	Product details provided by the hardware manufacturer (character string)
Main memory	Main memory size
Flash memory	Flash memory type

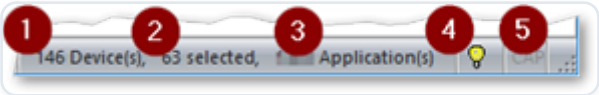
OPTION	DESCRIPTION
Flash size	Size of the flash memory

HARDWARE INFORMATION OF CONNECTED DEVICES

Examples for a USB device:

OPTION	DESCRIPTION
Product Type	Type of USB device
Vendor	Vendor name
Serial number	Serial number of the USB device
Revision number	Firmware version of the USB device
Port number	Number of the port the USB device is connected to: bus number>-port number>

Status bar



1. Number of all devices
2. Number of devices you have selected in a list, for example, in the **All devices** window
    - Use this information as feedback before applying any function to the devices.
3. Number of all applications
4. Alert messages (Error, Warning, Info) such as **Scout Server terminated** or **Could not write Scout server log file**
    - The color of the lamp icon changes to yellow as soon as there is a new entry.
    - To show the alert messages, double-click the icon.
5. Indicates whether the Caps Lock key is active

When you perform large operations such as importing, moving, exporting or deleting a large number of devices, the process is displayed on the status bar. .

Display settings

Some preferences regarding the view may be set via the **View > Settings** menu:

OPTION	DESCRIPTION
Refresh display of devices	
In the tree view	Time span in seconds for periodically refreshing the display of devices, OUs and applications in the tree view
In the list view	Time span in seconds for periodically refreshing the display of devices, OUs and applications in the list view
Show new devices automatically in tree view	New devices after onboarding are automatically added to the tree view (disabled by default).
Status bar/ Refresh totals	Time span in seconds for periodically refreshing the number of all devices and applications across the infrastructure
On console start / Show recently used element	After the console is restarted, the last selected item in the tree view is highlighted.
Show confirmation messages / Before view is refreshed	After you press F5 or click **View > Refresh**, the system prompts you before refreshing the view.
Check independent configurations	When you modify a device configuration, all subordinate independent configurations are checked. You then receive a list of the relevant parameters and can conveniently determine whether and to which instances the modifications are to be transferred.
	For further information, see Blocking inheritance - independent device configuration.
Verbose level	For level 1, a confirmation message is shown before commands are executed, indicating the number of affected devices.

Changing the language

The console is started with the display language you have chosen for the Scout Enterprise Management Suite® installation. The language setting refers to the display of interface elements and can be changed at any time:

1. Click **View > Select language** and choose your language.
2. Restart the Scout Console.

The console is started with the selected language.

> **Important**
>
> Individual window layouts will not be restored after a language change.

Search for devices OUs or applications

The Scout Console offers various search options for objects within the infrastructure.

Search for devices, OUs or applications:

- Quick search in the tree view
- Search via dialog and use search options

Search for devices

- Quick search in the All devices window
- Advanced search

> **Note:**
>
> These search functions refer to the current Scout infrastructure and are not to be confused with the **discovery** function which is meant for devices in the entire network and outside the infrastructure.

Devices or Search for MAC addresses

All described search functions allow you to find devices based on their MAC address. To do so, use one of the following formats:

12-digit MAC address without separators	001122334455
12-digit MAC address with the 6 pairs separated by colons	00:11:22:33:44:55
12-digit MAC address with the 6 pairs separated by hyphens	00-11-22-33-44-55

- Searching the tree view (quick search)
- Searching the tree view including search options (dialog)
- Searching for devices in the All devices window
- Searching for devices using the Advanced search view

Searching the tree view (quick search)

1. Click into the tree view to set the focus.
2. Press CTRL+F or click into the **Search** field of the toolbar.
3. Type the name of an application, device or OU.

   If configured accordingly, you can type partial words.

4. Press RETURN or click the magnifier icon.

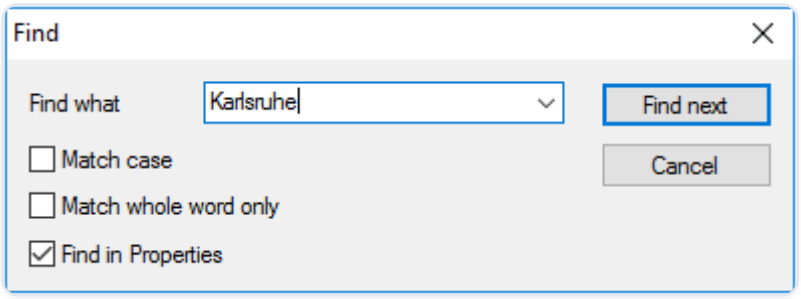
   The first matching object is shown in the tree view.
5. To find the next match, press F3 or click the magnifier icon.

> **Note:**
>
> The search is performed using the search parameters set in the **Find** dialog.

Searching the tree view including search options (dialog)

1. Press CTRL+SHIFT+F or click **Edit > Find**

   The **Find** window opens.



2. Type the name of an application, device or OU.

   If configured accordingly, you may type only partial words.

3. If required, modify the search parameters.

OPTION	DESCRIPTION
Match case	Search is case-sensitive
Match whole word	Only exact matches are found, no partial words.
Find in properties	Search is also applied to **Properties** and **Assets** fields. This allows you to search for a vendor or a MAC address.

> **Note:**
>
> The search parameters remain active after the search and are also applied to the **Quick search**

> The first matching object is shown in the tree view.

4. To find the next match, click **Find next** or press F3.

Searching for devices in the All devices window

1. Click into the **All devices** window to set the focus.
2. Press CTRL+F or click into the **Search** field of the toolbar.
3. Type the name of a device.

   If configured accordingly, you can type partial words.

4. Press RETURN or click the magnifier icon.

The first matching object is shown in the **All devices** window.

5. To find the next match, press F3 or click the magnifier icon.

> **Note:**
>
> You can also search the **All devices** window when it does not have focus: Press SHIFT+RETURN to start searching for the specified object. Press SHIFT+F3 to find the next match.

Searching for devices using the Advanced search view

The **Advanced search** window provides further options that allow you to search for devices including multiple filter criteria and wildcard search. You can perform commands and notifications on search results.

1. Show the **Advanced search** window. To do so, click **View > Window > Advanced search**.

   Some fields are shown as columns including a search field.

2. To show or hide fields as columns, click the button.

   The **Adjust** dialog with all available fields opens. Configure which fields to show.

   Fields that cannot be used for a search contain the entry N/A.

3. To change the order of the columns, also use the **Adjust** dialog.

   The column order is relevant if you want to define multiple filter criteria.

4. In the desired column, in the search field, enter a search term. To replace characters, you can use the wildcard character % at the beginning or end of the search term.
5. To start the search, click the filter icon.
6. To narrow the search, use additional filter criteria in allowed columns.

Use the context menu to perform commands and notifications on the search results.

Moving and copying elements

Devices, OUs and applications can be moved from one OU to another OU within the tree view of the organizational structure. If inheritance is active, after you have moved a device or OU, it receives the properties of the new parent OU.

Moving devices, OUs or applications

1. In the tree view, show the source and target position of the relevant element.

   The target position can be the icon of the target OU or any valid position subordinate to the target OU.
2. Use a drag-and-drop operation to move the element from the source to the target position. or, Move the element via context menu or CTRL-X to the Clipboard and paste it via context menu or CTRL-V at the target position.
3. Confirm with **Yes**.

Confirm your changes again and the element is moved to the target OU.

Moving devices to another OU via context menu

1. In the tree view, in the **OU devices** window or in the **All devices** window, for the relevant device, open the context menu.
2. Click **Edit > Move...**
3. In the dialog, expand the organizational structure and select the target OU.
4. Confirm with **OK..**

Confirm your changes again and the element is moved to the target OU.

Copying applications

> **Note:**

> Applications in the tree view are application definitions and do not include software. The software must be configured and provided separately via IDF.

1. Show the source and target position of the relevant application in the tree view.

   The target position can be the icon of the target OU or the **Applcations** node subordinate to the target OU.
2. Use a drag-and-drop operation while pressing CTRL to move the application from the source to the target position.

   or,

Copy the application via context menu or CTRL-C to the Clipboard and paste it via context menu or CTRL-V at the target position.

3. Confirm with **Yes**.

The application is copied to the target OU.

> **Note:**
>
> Applications can also be copied from any client device to a Scout OU. For further information, see Uploading applications from a device to Scout .

Switching OU to top-level

> **Note:**
>
> This feature can only be applied to an OU.

> For the relevant OU, from the context menu, choose **Edit > Convert to base-OU**.

The relevant OU is moved to the highest level. It becomes one of the base-OUs. Device configuration and inheritance remain as defined. If inheritance is active, all settings are adopted from the base device configuration.

Device management

To be able to manage client devices with eLux or other operating systems, Scout must know their MAC address. There are several approaches to register new devices such as

- Self-registration of devices
- Searching for devices (Discovery)

New devices must be assigned to an organizational unit (OU). You can configure whether new devices

- are added to a specified OU (**default OU**)
- are assigned automatically by the OU filter according to definable criteria
- are created in terms of proxy profiles even before connecting Reserving device profiles

The way you want to deal with new devices is mainly defined under **Options > Advanced Options >** Devices.

> **Note:**
>
> As the devices are organized hierarchically in OUs, you can use Dynamic Device Groups to apply commands to several devices irrespective of their OUs.

- Self-registration of devices
- DHCP configuration
- Searching for devices (Discovery)
- Reserving device profiles
- Device names
- MAC address
- Secure device management with Scout
- OU filter
- Dynamic Device Groups
- Device relocation between Scout Servers

Self-registration of devices

By default, the first time a device boots, it automatically searches for an available Scout Server. The device requires the IP address of the Scout Server.

Requirements for self-registration:

- The device must be in initial state (either upon delivery or by performing a factory reset)
- The device must be connected to the network
- The Scout IP address must be configured in one of the following ways:
- DHCP: A configured DHCP option is set to the IP address/name of the Scout Server. You can also specify more than one Scout Server and a destination OU. For further information, see DHCP configuration.

or

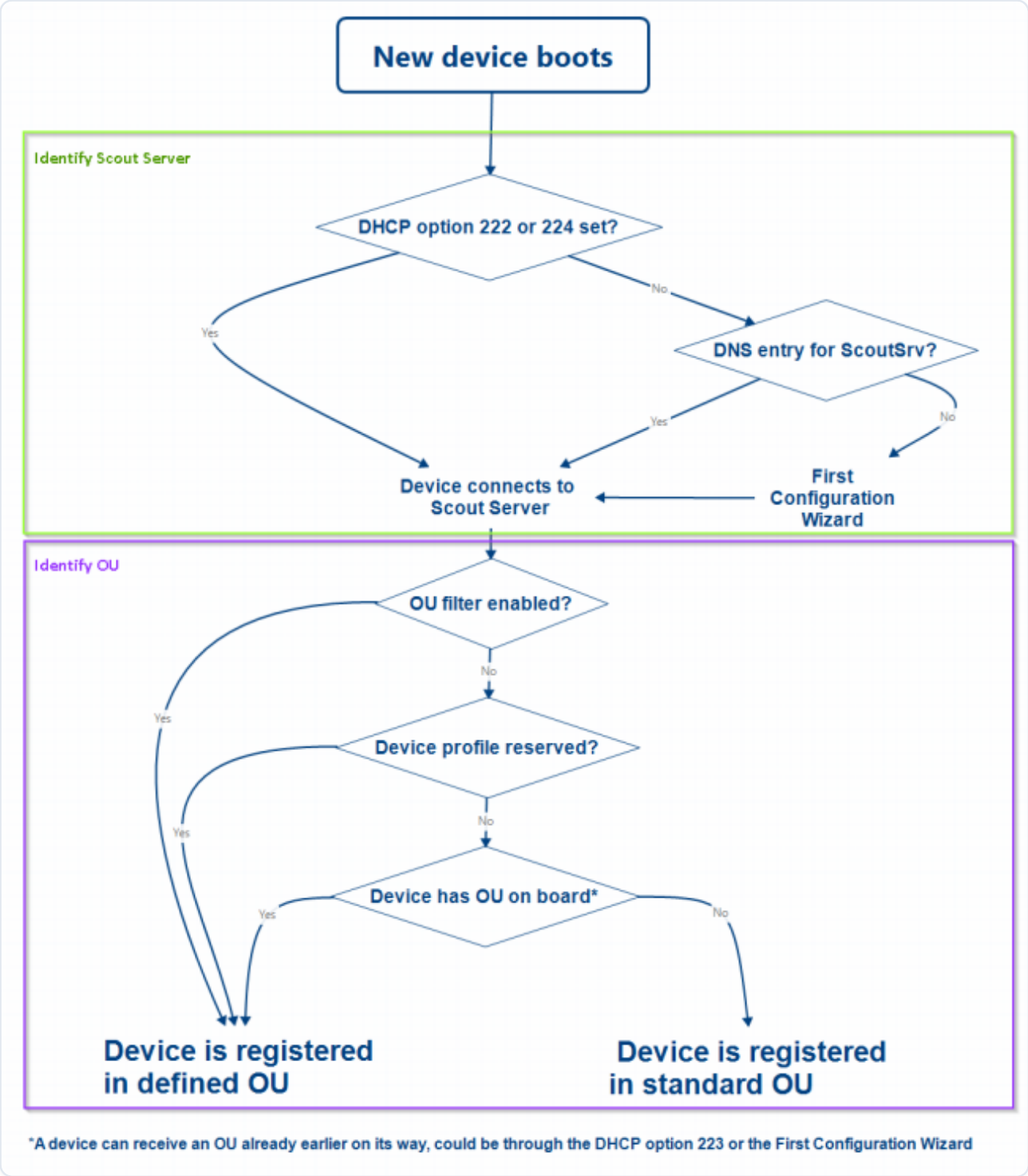
- DNS: The DNS server resolves the host name ScoutSrv (no case-sensitivity).

If the Scout Server's IP address cannot be determined, neither by the DNS nor by DHCP, a First Configuration Wizard automatically runs on the device to help the local user through the initial configuration.

Registering a device automatically

- Turn the device on.

If the requirements for self-registration are met, the device contacts the Scout Server and enters itself in the defined OU or the standard OU. It receives the configuration of the OU and is restarted with the new configuration.



The flow chart roughly shows the way a new device is assigned to a Scout Server and to an OU. Details such as the **Accept only known devices** have not been considered.

DHCP configuration

-- optional --

> **Note:**

> DHCP options can only be applied to eLux® devices.

> A new device booting for the first time can retrieve the following information from a DHCP server:

- IP address or name of the Scout Server (option 222)
- List of Scout Servers (option 224)
- ID of the destination OU on the Scout Server (option 223)

This requires configuring the DHCP server via one of the two following methods.

In method 1 (recommended), you define a new vendor class, set the new options, and then apply the values. Method 2 uses the DHCP Standard Options 222, 223 and 224.

The following instructions are based on the DHCP manager of Windows Server 2012.

Method 1: Defining user-defined vendor class

REQUIRES
DHCP server compliant with RFC 2132, supporting user-defined vendor classes. Otherwise use method 2.

1. Open the DHCP manager.
2. Select the relevant DHCP server, and then click **Action > Define**
3. Click **Add** to create a new class:

OPTION	VALUE
Display name	eLux NG
Description	eLux specific options
Code	ELUXNG
(in ASCII column)	The entry is automatically extended with the related hexadecimal number (45 4C 55 58 4E 47).

4. Click **Action > Set Predefined Options**. Then, in the **Option class** list field, select eLux NG.

5. To define one Scout Server, click **Add** and edit the new option as follows:

OPTION	VALUE
Name	Scout Server
Data type	String
Code	222
Description	Name or IP address of the Scout Server

6. To define more than one Scout Server, click **Add...** and edit the new option as follows:

OPTION	VALUE
Name	Scout Server list
Data type	String
Code	224
Description	Server names/IP addresses, comma-separated

7. To define a specific OU that you can assign new devices to, click **Add...** and edit the new option as follows:

OPTION	VALUE
Name	Scout OU ID
Data type	Long
Code	223
Description	ID of the destination OU

8. To assign the options, for the relevant DHCP server, select either the **Server Options**, the **Scope options** or the **Reservations**. Then click **Action > Configure Options... > Advanced**.

In the **Vendor class** list field, select . Select each option defined and enter its value into the **Data entry** field:

OPTION	VALUE
222 Scout Server	Name or IP address of the Scout Server
223 Scout OU ID	ID of the destination OU
224 Scout Server list	<Names or IP addresses of the Scout Servers, separated by comma>

Method 2: Using DHCP Standard Options

REQUIRES
The DHCP Standard Options [222], [223] and [224] must be available. Otherwise use Method 1.

1. Open the DHCP manager.
2. Select the relevant DHCP server, and then click **Action > Set Predefined Options...**. In the list field, select .
3. Click **Add...** to create the following Standard Options, as described for Method 1:
   - , String, 222
   - Scout Enterprise server list, String, 224
   - OU ID, Long, 223
4. To assign the options, for the relevant DHCP server, select either the **Server Options**, the **Scope options** or the **Reservations**. Then click **Action > Configure Options... > General**.

   Select each option defined and enter its value into the **Data entry** field:

OPTION	VALUE
222 Scout Server	
223 Scout OU ID	ID of the destination OU

OPTION	VALUE
224 Scout Server list	<Names or IP addresses of the Scout Servers, separated by comma>

Disabling DHCP option 12 as source for host names

If you have configured DHCP option 12 (host name), when connecting new devices, you can have the host names set via DHCP. To obtain the host name not via DHCP but from another source, such as the name template defined in the Scout Console, prevent the take-over from DHCP option 12. To do so, use a terminal.ini parameter:

- **File:** `/setup/terminal.ini`
- **Section:** `[Network]`
- **Entry:** `IgnoreDHCPHostname`
- **Value:** `true` (The default value is `false`.)

Searching for devices (Discovery)

Based on the IP address, you can search for devices throughout the entire network or within specific subnets. Any matching devices are automatically registered to Scout and are added to the specified OU **(Destination group)**. The devices are restarted and receive the configuration of the destination group (device configuration, application definitions, files defined for transfer and advanced file entries).

> **Note:**

> If the OU filter is active, the filter specifies the destination group or groups. For further information, see Devices.

> Requirements:

- The devices are turned on and connected to the network.
- The devices are provided with valid IP addresses.
- The device password is known.

Searching and registering devices

1. Make sure that your destination OU is configured correctly.

2. Select **Options > Search devices**.

3. Edit the following fields:

   a. Enter the first IP address of the range.

   b. Enter the number of IP addresses within the range (restricted to 255).

   c. Enter the last IP address of the range.

   d. Enter the device password (default: **elux®**).

      - The password must match the password currently set on the individual devices.

   e. Select the OU you want to assign the devices to.

      - Default is the predefined **Lost&Found** group with the base device configuration.

> **Important:**
>
> If the field is disabled, the OU filter is active and the matching devices are assigned according to the OU filter rules.

   a. The user is informed by a message about the upcoming device restart.

      - Specify in seconds how long the message shall be displayed. If the time period is set to , the system will wait until the user confirms.

   b. Allows the user to suppress the device restart. The configuration is not updated until the device is restarted.

4. Confirm with **OK..**

The matching devices receive the IP address of the managing . The devices are assigned to their destination group and are restarted. The devices inherit the configuration of their new OU. Local non-protected device configuration is overridden. With immediate effect, on each restart, the devices connect to their and, if available, are given the latest configuration and application definition data.

If a device profile has been reserved for a device, the predefined profile is automatically assigned at Discovery.

To modify response time and maximum search time for the Discovery feature, choose **Options > Advanced options > Devices > Discover devices**.

> **Note:**
>
> Devices already registered to Scout are not modified, but their status is updated when connected.

Reserving device profiles

Devices can be assigned to OUs even before the devices connect to the Scout Server for the first time.

By creating devices manually in the , you reserve a device profile by its MAC address. As soon as such a manually created device contacts its for initial start-up, the registered MAC address is recognized and the device is entered. The configuration data of the relevant OU are transferred to the device.

Reserving device profiles can be applied for the following device registration procedures:

- Discovery
- DNS alias name ScoutSrv
- DHCP option 222 for the Scout Server

> **Note:**
>
> If an OU filter is active, the OU filter is applied prior to the device profile reservation.

Reserving a device profile

1. Select the OU you want to assign the device to, and show its sub tree.
2. Below the OU, open the **Devices** context menu and select **Add**.
3. Enter the 12-digit MAC address of the device without hyphens.

   If the MAC address is valid, the **Device configuration** dialog opens. The **Use parent** option is selected by default.
4. Confirm with **OK**..

Scout reserves a profile for the device with the respective MAC address. The actual registration is made at the time of the first device connection.

> **Note:**
>
> Importing devices also results in the reservation of device profiles within the OU structure. To create new devices in a greater number, we recommend using the feature. For further information, see Import/Export.

Device names

The devices' host names can either be taken from the devices without modification or set from the . In addition, name templates are available that use, for example, the MAC address or the IP address of the devices. The settings for this can be found under **Advanced options > Devices**. For further information, see Devices.

Device names are subject to the following format rules:

- Allowed characters are letters (a-z, A-Z) and digits (0-9).
- Additionally, hyphens (-) are allowed within the string, but not at the beginning and end of a device name.
- A device name must be at least one character long (letter or digit).

MAC address

The MAC address is the hardware address or physical address of a network device. Scout used to identify devices by their MAC addresses. In the meantime, the **client identifier** does this job.

MAC addresses can be used for the following tasks, for example:

- Import devices
- Reserve device profiles
- Create dynamic device groups

In Scout, MAC addresses are normally represented as 12-digit numbers without separators. Example: 901B0E01CE84

From , you can also use colons or hyphens as separators between the 6 pairs:

12-digit MAC address without separators	001122334455
12-digit MAC address with colons	00:11:22:33:44:55
12-digit MAC address with hyphens	00-11-22-33-44-55

> **Note:**
>
> The MAC address usually is determined from the network interface (Lan or WLAN card).

Secure device management with Scout

An enhanced security level is available for adding new devices to .

Clients that are registered with their MAC address in the database Reserving device profiles are accepted by the Scout Server and can be integrated into the Scout management. In contrast, devices having an unknown MAC address are not accepted and therefore cannot be managed by Scout. Unknown devices are not provided with a license from Scout's license pool.

Accepting known devices only

1. In the , select **Options > Advanced options > Devices > New**.
2. Select the **Accept only known devices** option.

If an unknown device tries to contact the Scout Server, an error message is displayed on the device stating that a connection to the Scout Server was denied.

> **Note:**
>
> Only the requests of those devices are accepted whose MAC addresses have been saved to the Scout database by a device import or device profile.

OU filter

OU filters can be used for automatic assignment of devices to an organization unit (OU). The assignment is based on predefined criteria. This is particularly helpful when registering new devices or relocating existing ones.

There are two options for configuring an OU filter:

- The **subnet OU filter** uses the devices' network address for filtering
- The **user-defined OU filter** uses configured asset information of the devices for filtering

You can only use one filter at a time. For each filter, you can define multiple filter rules and specify the sequence you want the rules to be applied in.

Once defined, the filter rules are retained unless deleted manually. Deactivate the filter rules that are currently not required but that you want to keep for future use.

The OU filter has precedence over

- OU assignment of devices via the DHCP option 223
- Discovery of new devices via Scout
- selecting the OU in the First Configuration Wizard locally on the device
- the default OU specified in **Advanced options > Devices**.

For individual devices, you can ignore the OU filter **Advanced device configuration > Management**.

> **Note:**

> OU filters are included when you export the data category **Advanced options**. For further information, see Import/Export.

- Setting up an OU filter as subnet filter
- Setting up an OU filter as user-defined filter
- Editing OU filter rules
- Deactivating OU filter for individual devices
- Re-activating OU filter

Setting up an OU filter as subnet filter

You can use the OU filter to filter on client network addresses and assign the matching devices to an OU.

1. Click **Options > Advanced options > Devices**.

2. Under **New devices**, select the **Assign OU depending on OU filter** option. If required, click ••• to open the **OU assignment** dialog.
3. Under **Filter type**, select Subnet filter (device network address).
4. Create a new filter rule. Edit the following fields:

OPTION	DESCRIPTION
Network address	Enter the scope of IP addresses. Example: 192.168.16.0 covers all IPs starting with 192.168.16.
Netmask	Enter the relevant network prefix to define relevant devices.
Destination OU	To browse the OU list, click and select the OU you want the specified devices to go.

5. Click **Add**.

   The filter rule is displayed in the field below.

6. If required, add more filter rules and configure them. For further information, see Options for OU filter rules.
7. In the bottom section, from the list **Non-matching devices**, select where you want to keep the non-matching devices.

   > **Important:**
   >
   > If you select , all non-matching devices and even devices already assigned to other OUs are reassigned to the default OU.

8. Review all active filter rules thoroughly to avoid unintentional assignments.
9. Confirm with **OK**..

All active filter rules are processed. On the next restart, the matching devices are assigned to the OUs as defined by the OU subnet filter. Any additional user-defined filter rules will not be taken into account.

OU filter as a user-defined filter

You can filter by configured asset information of the devices to assign the matching devices to the appropriate OUs.

FILTER RULES WITH OU FILTER TEXT

eLux® devices send an **OU filter text** field containing their device information to the Scout Server. You can use the **OU filter text** field in the Scout Report Generator and for the user-defined OU filter. It includes the following information:

Host name, OS name, OS version, serial number, supplier, device type, UEFI/BIOS, CPU speed, model, kernel version, flash type, flash size, RAM size, graphics, IDF name, MAC address.

- A filter rule can contain one or more filter criteria.
- A filter criterion consists of three parts:

  asset information string from `OU filter text` | logical operator = | value you want to filter by

Asset information elements:

ELUX_HOSTNAME=Alrin;

ELUX_OSNAME=eLux;

ELUX_OSVERSION=7.2507.0-3;

ELUX_SERIAL=7250O422542;

ELUX_SUPPLIER=WYSE;

ELUX_DEVICETYPE=ZQ Class;

ELUX_BIOS=V4.6.5.4 R1.15.0;

ELUX_CPU=3999;

ELUX_PRODUCT=ZQ Class;

ELUX_KERNEL=6.12.30unicon0;

ELUX_FLASH=16GB SATA Flash;

ELUX_FLASHSIZE=15272;

ELUX_MEMORY=8192;

ELUX_GRAPHICS=ATI AMD Radeon HD8330E;

ELUX_IDF=recovery.idf;

ELUX_MAC=7CD30A22D0AE

ELUX_IP=192.168.10.21

ELUX_NETMASK=255.255.255.0

ELUX_BROADCAST=192.168.10.255

ELUX_NETADDR=192.168.10.0

ELUX_NETCIDR=24

ELUX_DOMAIN=crop.company.com

- Use the logical operators **AND** and **OR** to link multiple filter criteria. Use capital letters for the operators.
- Wildcards are not supported, but all matches will be found that begin with the specified string.

Example filter rules:

`[ELUX_OSNAME=eLux AND ELUX_OSVERSION=7.2507.0-3]` `[ELUX_DEVICETYPE=D3314-A1 OR ELUX_DEVICETYPE=ZQ Class]`

SETTING UP A USER-DEFINED OU FILTER

1. Click **Options > Advanced options > Devices**.
2. Under **New devices**, select the **Assign OU depending on OU filter** option. If required, click ••• to open the **OU assignment** dialog.
3. Under **Filter type**, select User-defined filter (configured asset information).
4. Create a new filter rule.

OPTION	DESCRIPTION
Filter rule	Create a filter rule consisting of one or more filter criteria, see above.
Destination OU	To browse the OU list, click ••• and select the OU you want the specified devices to go.

5. Click **Add**.

   The filter rule is displayed in the field below.

6. If required, add more filter rules and configure them. For further information, see Options for OU filter rules.
7. From the list **Non-matching devices**, select where you want to keep the non-matching devices.

   > **Important:**
   >
   > If you select , all non-matching devices and even devices already assigned to other OUs are reassigned to the default OU.

8. Review all active filter rules thoroughly to avoid unintentional assignments.
9. Confirm with **OK**..

All active filter rules are processed in the specified order. On the next restart the matching devices are assigned to the OUs as defined by the OU user-defined filter. Any additionally defined subnet filter rules will not be taken into account.

BEST PRACTICES

The asset information **ELUX_IP** corresponds to the **(Peer) IP address**.

When eLux endpoints connect through the Scout Cloud Gateway (SCG), the peer IP address reflects the **SCG-assigned VPN IP** (e.g. `10.100.100.0` default range).

When devices connect directly to a **Scout Server**, the peer IP equals the endpoint's **local IP address**

> **Notes**
>
> The Scout IP address always reflects the endpoint's local IP.

Example OU filter expression

If you want to assign eLux endpoints based on local network ranges (instead of SCG assigned VPN IPs), you can use a filter like this:

```
ELUX_NETADDR=192.168.168.0 AND ELUX_NETMASK=255.255.255.0
```

This will ensure OU assignments reflect the device's **true local subnet**, instead of the SCG VPN IP range

> **Important**

> If you are defining more complex and nested expressions by using brackets, these brackets - as control characters - need to be escaped.

> Example with brackets and escaping:

```
\(ELUX_CPU=512 AND ELUX_MEMORY=1024\) OR \(ELUX_OSNAME=eLux AND ELUX_HOSTNAME=elux7-vm\)
```

Options for OU filter rules

Once you have defined OU filter rules they remain until they are deleted explicitly. You can edit the filter rules in the following ways:

OPTION	ACTION	DESCRIPTION
Create new filter rule	Click **Add**	User-defined filter: The filter criteria of the Filter rule field and the selected Destination OU are added as a new filter rule to the list. Syntax of a filter criterion: String from OU filter text=value, You can combine two or more filter criteria by using one of the logical operators AND or OR. Use capital letters for the operators. For examples, see OU filter as a user-defined filter. Subnet filter: The data from the fields Network address, Netmask and OU

OPTION	ACTION	DESCRIPTION
		are added as a new filter rule to the list.
Delete a filter rule	Select filter rule and click [Delete]	The selected filter rule is deleted.
Edit filter rule	Select filter rule and press [F2] or double-click	Modify the elements of a filter rule right in the list.
Activate / deactivate a filter rule	Select/Clear [Activated] option	Deactivated filter rules are not executed. Newly added filter rules are active by default.
Change sequence of processing (user-defined filter)	Edit [Sequence] field	Filter rules with low sequence number are processed prior to filter rules with high sequence number.

1. Review all active filter rules thoroughly to avoid unintentional assignments.
2. For the entire filter, under **Non-matching devices**, select where you want to keep the non-matching devices.

> **Important:**
>
> If you select assigned to the default OU, all non-matching devices and even devices already assigned to other OUs are reassigned to the default OU.

Deactivating OU filter for individual devices

If the OU filter is enabled, all active filter rules are applied and the matching devices are assigned to the specified OU on their next restart. To exclude an individual device from the filter, deactivate the OU filter for that device.

1. For the relevant device, open **Advanced device configuration > Management**.
2. Under **New devices**, select **Ignore OU filter**.
3. Confirm with **OK**..

Or:

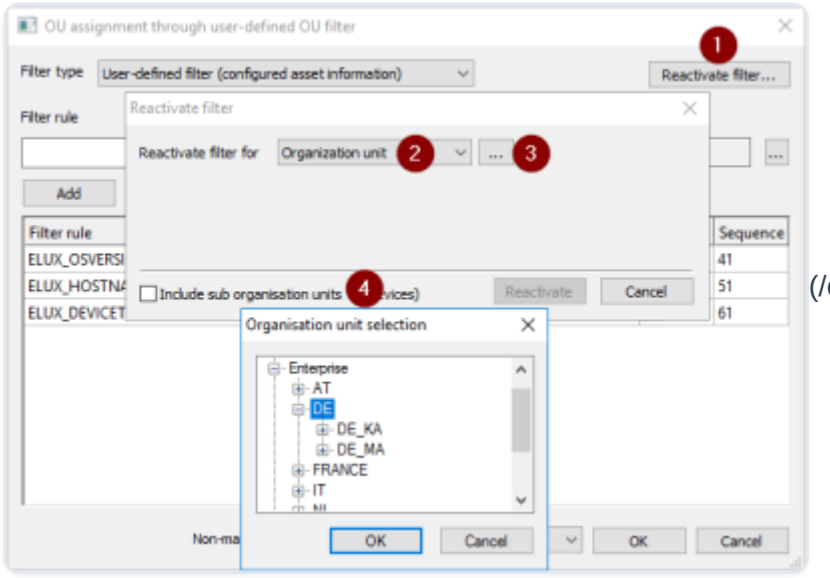
1. By using a drag-and-drop operation, move the device to another OU.
2. Confirm with **OK**..

The device is assigned to the new OU and the OU filter is deactivated for this device.

Re-activating OU filter

The OU filter can be deactivated for individual devices - either by applying the relevant option in the Scout Console or by moving devices by a drag-and-drop operation. To bring the relevant devices back to the OU filter, use the Scout Console option **Re-activate filter**.

1. Click **Options > Advanced options > Devices**.
2. Under **New devices**, select the **Assign OU depending on OU filter** option. If required, click ••• to open the **OU assignment** dialog.
3. In the upper right section, click **Reactivate filter...**(1).



(/en-us/unicon-elux-scout/2603-ltsr-cu1/media/admin_guides/scout/ou_filter_reactivate.png)

4. Under **Reactivate filter for** (2), select the scope of devices for which you want to re-activate the OU filter: for all devices, for an individual OU or for Dynamic Device Groups.
5. To re-activate the OU filter for OUs or Dynamic Device Groups, click ••• (3) next to the list-field and select an OU or Dynamic Device Group.

   To include devices of subordinate OUs, select (4).
6. Confirm with **Reactivate** and **OK**.

The OU filter is re-activated for the selected devices.

Dynamic Device Groups

Dynamic Device Groups enable administrators to run cross-OU commands for freely definable device groups. For example, you can send a message to all devices with a specific image throughout the whole organization. Or you can run a UEFI/BIOS update on all devices with a specific UEFI/BIOS version, across all OUs. Even device relocation to another Scout Server can be applied to a Dynamic Client Group. The following features can be applied:

- Commands
- Configuration run
- Notifications for software deliveries or firmware updates
- Notifications for relocation

Dynamic Device Groups are based on reports created in the Scout Report Generator. The reports are exported to the Scout Console once, and from that point onward, are displayed as a **Dynamic Client Group**. Commands applicable to OUs or to individual devices can also be applied to Dynamic Device Groups.

> **Note:**

> The report layout must include the Client Identifier field. The report type must be a list of **Devices** or **Assets**. For further information on defining , see Creating Dynamic Device Groups in the Scout Report Generator guide.

> Dynamic Device Groups are displayed in the Scout Console in a special window and remain there for re-use until they are deleted. They can be updated any-time with a click.

User rights

When you create Dynamic Device Groups, the user access rights are respected as defined in the administrator management.

By default, executing commands and other functions on a Dynamic Device Group is subject to the object rights configured for the relevant devices and OUs. This can result in a function not being executed because at least one device does not have the required object rights. To avoid this, disable the check for the underlying object rights under **Security > Manage administrators > DCG rights**.

- Using Dynamic Device Groups

- Special form of Dynamic Device Groups by import

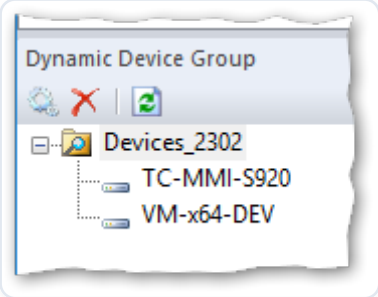
Using Dynamic Device Groups

Dynamic Device Groups are usually based on reports created in the Scout Report Generator and exported to the Scout Console.

For further information on defining and exporting DCGs, see Creating Dynamic Device Groups in the Scout Report Generator guide.

DISPLAYING DYNAMIC DEVICE GROUPS

- In the Scout Console, click **View > Window > Dynamic Device Groups**.



The **Dynamic Device Groups** window is displayed. The Dynamic Device Groups can be expanded to show the matching devices.

> **Note:**

> The Dynamic Client Group shows those devices that have matched the criteria at the time of the latest report generation. Make sure that the Dynamic Client Group is up-to-date.

> For a selected Dynamic Client Group, the **Properties** window shows the **Creation date, Number of devices and Filter** criteria of the used report. The creation date refers to the date of the latest generation of the report the Dynamic Client Group is based on, and thus indicates if the Dynamic Client Group is up-to-date.

If, for example, new devices have been integrated into the database and match the criteria of the report, the Dynamic Client Group is not up-to-date any longer. You can, however, update the Dynamic Client Group by re-creating the report right from the Scout Console.

If a Dynamic Client Group is not needed any longer, use the button to delete it. The report the Dynamic Client Group was based on remains unaffected.

UPDATING DYNAMIC DEVICE GROUPS

1. In the Dynamic Device Groups window, select the relevant device group.
2. On the toolbar of the **Dynamic Device Groups** window, click the **Recreate** button .

The relevant report is re-created and exported. The resulting devices are shown below the Dynamic Client Group as extracted from the database. In the **Properties** window, in the **Creation date** field, the current point of time is displayed.

> **Note:**
>
> The **Refresh** button refers to the view only. The report is not updated by this command.

APPLYING COMMANDS AND NOTIFICATIONS TO DYNAMIC DEVICE GROUPS

1. In the **Dynamic Device Groups** window, select the relevant Dynamic Client Group, and then check the information shown in the **Properties** window.
2. To update the Dynamic Client Group, click the **Re-create** button. This way you ensure that exactly the currently matching devices are affected.
3. Open the context menu of the Dynamic Client Group and select a command or notification.

If the required object rights are available, the commands and notifications are applied to the matching devices, irrespective of their OU. The available commands can also be scheduled for later execution.

Special form of Dynamic Device Groups by import

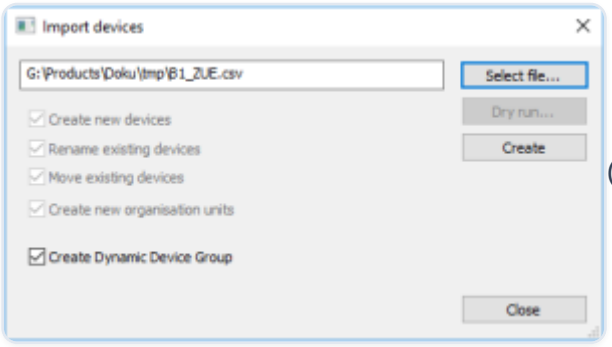
To create a Dynamic Device Group, instead of using the Scout Report Generator, you can use the Import feature of the Scout Console and make up a device group based on a device list containing MAC addresses. The advantage is that you are completely free to select any devices you want. They only have to be registered in Scout. Note that the Import feature here is not used to import devices.

Creating Dynamic Device Groups by import

REQUIRES
The relevant devices must be listed with their MAC addresses in a .csv file. Each line must begin with a MAC address. The lines may contain further information but only the MAC address is evaluated.

1. In the Scout Console, click **File > Import > Devices**.
2. In the **Import devices** dialog, in the bottom section, select the **Create Dynamic Device Group** option.

   All options related to the device import are disabled.

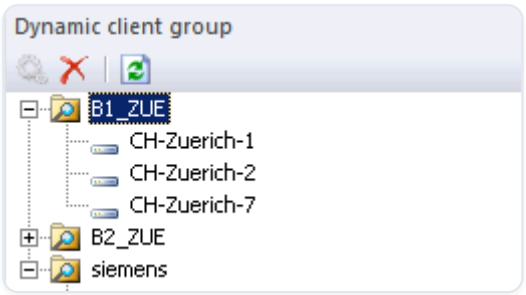
   

   (/en-us/unicon-elux-scout/2603-ltsr-cu1/media/admin_guides/scout/dia_import_dcg.png)
3. Click **Select file...** and select the relevant .csv file from the file system.
4. Click **Create**.

The .csv file is evaluated. Scout creates a new Dynamic Client Group. It contains all devices of the csv list whose MAC address is registered in Scout. The dynamic device group adopts the name of the .csv file.

DISPLAYING DYNAMIC DEVICE GROUPS

- In the Scout Console, click **View > Window > Dynamic Device Groups**.



The **Dynamic Device Groups** window is displayed. The Dynamic Device Groups can be expanded to show the matching devices.

> **Note:**
>
> Dynamic Device Groups that have been created via the import feature cannot be updated with the **Re-create** button. To update the Dynamic Client Group, after having modified the device list, save the *.csv file under the same name and perform a new import as described above.

> For a selected Dynamic Client Group, the **Properties** window shows some information such as the **Creation date** and **Number of devices**. The **Filter** field shows the entry created by device import.

APPLYING COMMANDS AND NOTIFICATIONS TO DYNAMIC DEVICE GROUPS

1. In the Dynamic Device Groups window, select the relevant Dynamic Client Group. Check the information shown in the **Properties** window.
2. Open the context menu of the Dynamic Client Group and select a command or notification.

Commands and notifications are applied to the matching devices, irrespective of their OU. The available commands can also be scheduled for later execution.

Device relocation between s

Relocating devices from one Scout Server to another can be very helpful in different scenarios of device migration, for example when relocating devices from a test/QA server to a production server, or consolidating several Scout Servers to a single server (server fusion).

Client relocation can be performed with and without the devices verifying the availability of the target server. A so-called **offline** relocation does not require the target server to be physically available at the time of relocation. Example: External suppliers, in their environment, set up devices to be used in the customer's environment.

License information and local configuration can either be included in the transfer or left on the source server.

> **Note:**

> You can use the procedure also in and installations. Here, there is no need to carry license information.

- Relocation procedure
- Relocation procedure / offline
- Relocation flow chart
- Initiating device relocation
- Scheduling the relocation process

- Deleting relocation notification

Relocation procedure

The relocation procedure is initiated by the source server (device-releasing server) and, by default, completed by the target server (device-receiving server). The actual relocation procedure, however, is performed by the devices. They check the conditions and transfer the license information.

Relocation for the relevant devices is triggered by a notification named **Initiate device relocation** in the of the source server. The notification includes all required details. On the next device restart, the devices receive the new configuration data of the source server and evaluate the relocation notification.

The devices then check whether the transmitted target server's address is available via the network. Then, the relocation process will be started and the devices will be deleted from the source server.

By default, the devices bring their management and application licenses as well as their proportional Subscription validity with them to the target server. The license and Subscription information is then deleted on the source server and added to the target server.

The new devices are assigned to the specified OU on the target server. If no destination OU has been specified, the default OU or the OU defined by OU filter rules is used (configured in **Advanced options > Devices > New devices**).

The relocation procedure is completed by an automatic device restart which activates the configuration of the target server. If configured, locally defined configuration is retained. If the OU filter is used, an additional device restart is provoked by the system after assignment.

OPTIONS

In contrast to the default procedure, the administrator can configure the following options in the notification:

- Offline relocation

The devices are removed from the source server without checking whether the target server is available. They move to the target server only when they can connect to it.

- Relocation without transfer of licenses

License and Subscription information is left on the source server so it can be used by other devices. This includes management licenses, application licenses assigned to the devices (if any), and the subscription portion of the relevant devices.

- Retain local configuration on new Scout Server1

If on the source server is configured that users' local configurations in unlocked fields should not be overwritten by server-side configurations (), then the local configurations are carried along. The relocated receive the target server's device configuration only where there are no local configurations.

> **Important:**

> Do `not` reserve device profiles by entering MAC addresses of the new devices on the target server before device relocation. If the devices are already registered on the target server, licenses and Subscription will `not` be updated.

Relocation procedure / offline

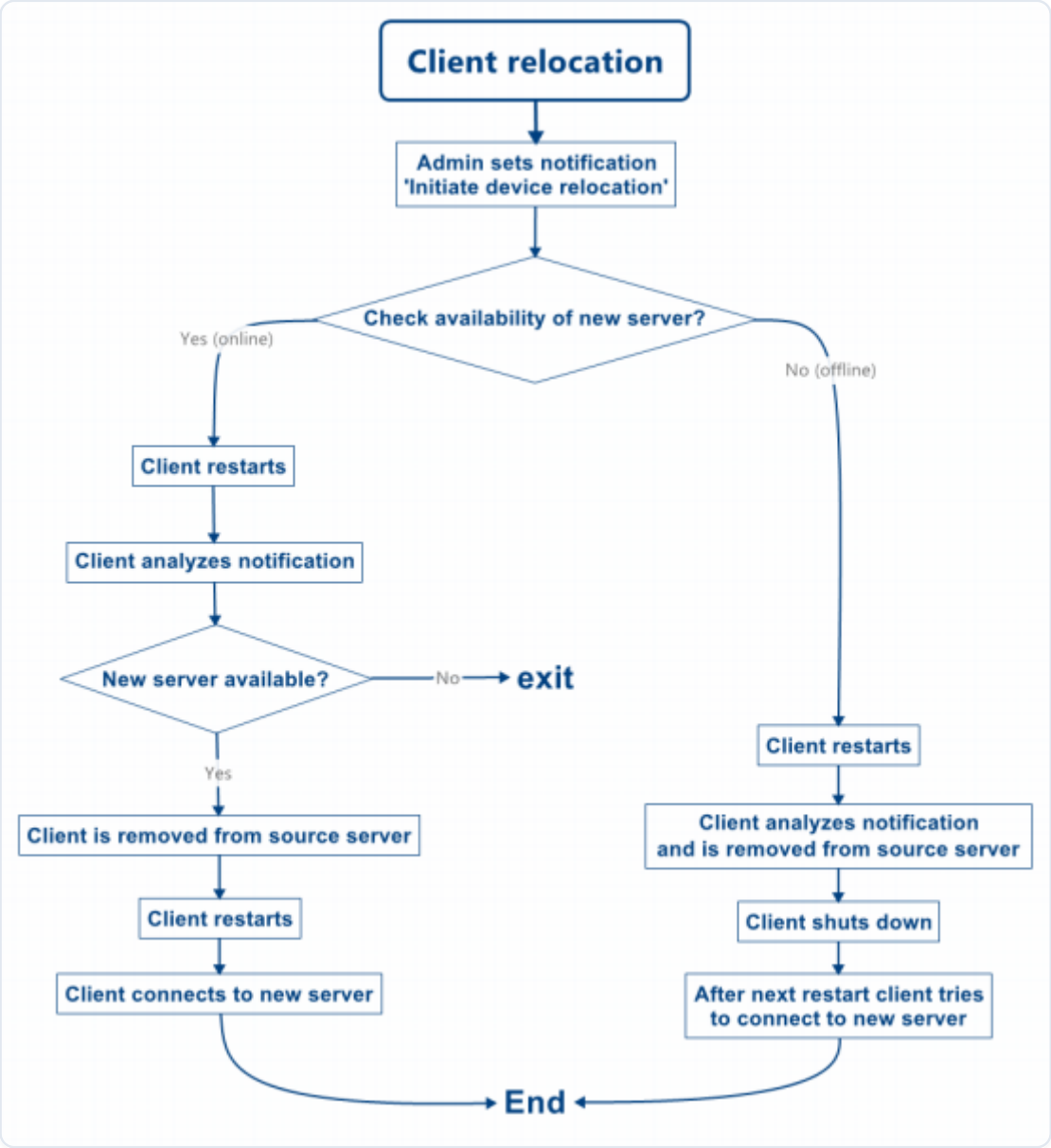
The relocation procedure is initiated by the source server (device-releasing server) and completed by the relevant devices. If you perform an offline relocation, the devices do not verify whether the target server (device-receiving server) is available and ready to accept the devices.

The administrator still triggers the relocation by setting a notification **Initiate device relocation** for the relevant devices in the Scout Console of the source server. The notification includes all required details. On the next device restart, the configuration is synchronized with the source server, the notification is analyzed and the configuration data are updated:

The relevant devices are removed from the source server without further verification. On the next restart, the devices will try to connect to the target server.

License information and local configuration can either be included in the transfer or left on the source server.

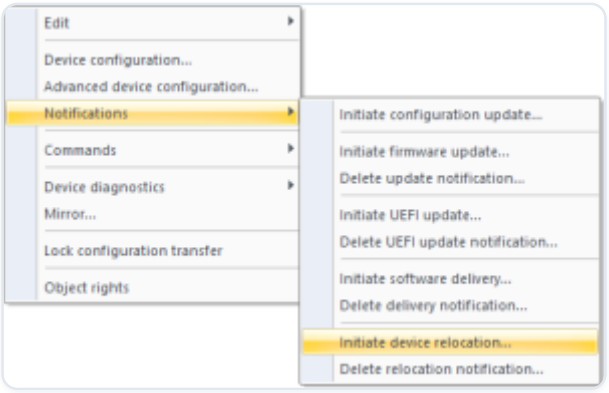
Relocation flow chart



Initiating device relocation

REQUIRES
• **Advanced Options > Devices**, clear the option **Accept only known devices**, if selected
• To ensure relocation success, the DHCP options of the device are not checked during the relocation. If, however, Scout Server DHCP options for the source server have been defined (222/223/224), on the target server, in **Device configuration > Network > LAN > Edit > Advanced select [Ignore DHCP options**

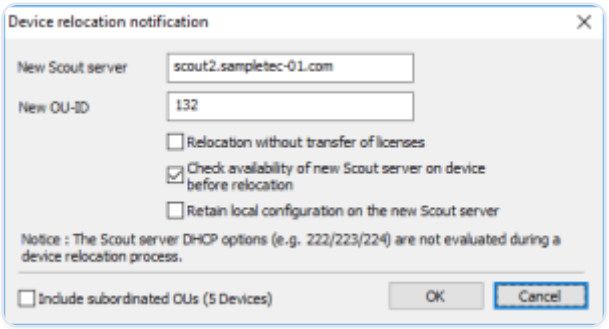
1. Select a device, an OU, a Dynamic Device Groups or devices within the **All devices** window.
2. On the context menu, click **Notifications > Initiate device relocation**.



3. In the **Device relocation notification** dialog, in **New Scout Server**, type the name (FQDN) or the IP address of the target server.

> **Note:**
>
> To connect the devices via Scout Cloud Gateway to their target server, type the name or IP address of the SCG instance. The Scout Cloud Gateway must be fully configured.

4. Edit the following fields:



| New OU-ID | ID of the destination OU on the target server | | | If you do not specify a destination OU, the devices are assigned to the default OU or to the OU defined by the OU filter rules. |

| Relocation without transfer of licenses | The licenses of the relocating devices are left on the source server. The subscription portion for these devices also remains on the source server. |

> **Important:**

> In case you want to move devices from a standard to an MSP or Enterprise Subscription installation, licenses cannot be transferred with them. Select the option to leave the licenses on the source server. Otherwise, the licenses will be lost.

> | Check availability of new Scout Server | 'online' relocation | | | Relocation is only done when the devices can access the target server via the network. | | | selected by default | | Retain local configuration on new 1[from Scout 15 2110]] | User-defined values of the local device configuration in unlocked fields will be retained. This is relevant if the corresponding option in **Advanced Options > Devices** is used. |

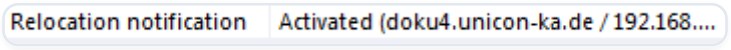
| Include subordinate OUs | All devices located in lower-level OUs will also be moved. |

> **Note:**

> Notifications are always set or deleted for all selected devices regardless of whether they are only available for individual devices.

5. Confirm the notification and confirmation.

The notifications for device relocation will be set. For the relevant devices, in the **Properties** window, the **Relocation notification** field shows the value Activated.



> **Note:**

> To define which fields you want to show in the **Properties** window, click .

> For devices not involved in the relocation, the **Relocation notification** field remains empty.

The devices evaluate the notifications after their next restart and then start the relocation process. Alternatively, trigger the relocation by command to control the time of relocation.

Scheduling the relocation process

To carry out the device relocation outside working hours, for example, and to add a slight delay after each device is processed, proceed as follows.

REQUIRES
The relocation has been initiated and the relevant devices have their relocation notification.

1. In the Scout Report Generator, identify all devices with active relocation notification.

2. Export the identified devices to a Dynamic Client Group.
3. In the Scout Console, for your Dynamic Client Group, schedule a **Restart device** command:
   - In the Command dialog, choose date and time of the execution.
   - Specify a delay in milliseconds, that will be applied after the command execution of each device..

The relevant devices are restarted at defined time and then start the relocation process. They obtain their new configuration data from the target server. If you perform the relocation offline, the relevant devices will be removed from the source server, but will only connect to their new server when they can connect to it.

Deleting relocation notification

1. For the relevant devices, from the context menu, choose **Notifications > Delete relocation notification**.

2. To include the devices of all subordinate OUs, in the **Delete relocation notification** message, select the option **Include sub organization units**.

   The number of devices shown in brackets is updated dynamically.
3. Confirm with **OK**.

After refreshing the **Properties** window, you will see that the **Relocation notification** status for the relevant devices is deleted.

Notifications are always set or deleted for all selected devices regardless of whether they are only available for individual devices.

Device configuration

- Concept
- Configuration method
- Evaluating configuration data
- FollowMe Desktop
- General tab
- Network tab
- Desktop tab
- Display tab
- Mouse/Keyboard tab
- Firmware tab
- Security tab
- User authentication
- Multimedia tab
- Drives tab
- Printer tab
- Hardware tab
- Diagnostics tab
- Power management tab
- Troubleshooting device configuration

Concept

Device configuration is the key to managing a large number of devices efficiently. Configuring as many devices as possible in the same way keeps IT processes simple, and costs low. All the same, numerous different locations, heterogeneous hardware environments and additional requirements do not allow for a unified device configuration.

takes this into account by using an inheritance approach. By default, the base device configuration defined at top level passes its properties on to the devices on lower instances. The concept of inheritance helps you keep your configuration consistent and efficient. To define any variations, simply modify the relevant settings. Scout provides flexibility to override any settings on all levels.

> **Note:**

> Changes to the device configuration take effect on the next device restart.

> For further information, see

Inheritance of configuration

Blocking inheritance - independent device configuration

Supporting local configuration

Initiating a one-time update of all device configuration data

Accessing device configuration

Comparing device configurations between OUs or devices

Locking configuration transfer

> **Important:**

> The device configuration of a device depends on the software packages installed on it.

Inheritance of configuration

The base device configuration and the configuration of OUs can be inherited by lower instances.

The base device configuration is the top-level instance. Lower instances can be other OUs or individual devices.

If the option **Use parent device configuration** is active, the configuration of the superordinate element of the hierarchy is applied to the current instance. By default, the option **Use parent device configuration** is active, so a device inherits its configuration from the base device configuration.

Settings of the configuration can be edited on three levels in the Scout Console:

- Base device configuration (**Options > Base device configuration**)
- OU (**context menu > Device configuration**)
- Device (**context menu > Device configuration**)

On each level, you can inherit the device configuration from the parent level or define independent settings. To be able to override settings, you must block inheritance, that is disable the use of the parent device configuration.

> **Note:**
>
> Pay attention to the **Device configuration** dialog title. It indicates the location of the current configuration. This can be the base device configuration, or a parent OU, or the individual device.
>
>

Example: If inheritance is active and you open the configuration dialog of a device or OU subordinate to France, the title bar shows **Device configuration of OU < France > is used**. To modify any settings, open the France configuration dialog.

Blocking inheritance - independent device configuration

If you want to define independent settings for an individual OU or device, you have to block inheritance for that instance.

1. Open the context menu of the relevant instance (OU or device) and click **Device configuration**.

   The **Device configuration** dialog opens and the title bar shows the currently active device configuration instance. This can be the base device configuration or a parent OU. For further information, see Accessing device configuration.
2. On the **General** tab, clear the **Use parent device configuration** option.

Inheritance is disabled. The title bar of the dialog shows the currently edited instance and the available options are editable. This instance and all subordinate instances can be configured independently of parent instances.

> **Note:**

- The **Independent configurations** window shows all OUs and devices that do not use their parent configuration.
- In **View > Settings**, you can specify that when you modify a device configuration, all of its subordinate independent configurations are checked. You then receive a list of the relevant parameters and can conveniently determine whether and to which instances the modifications are to be transferred.

Supporting local configuration

User rights for modifying the local device configuration can be set for OUs and devices, even for individual fields. You can lock and disable individual fields or tabs for security reasons whereas other features such as monitor management can be allowed. For further information, see User rights.

If individual (local) configuration is allowed, make sure that the relevant configuration data are prevented from being overridden when the Scout configuration is reloaded on the next device restart.

RETAINING LOCAL DEVICE CONFIGURATION

1. Click **Options > Advanced options > Devices**.
2. Under **Field update**, select **Retain local configuration (unlocked fields)**.

When the Scout device configuration data are reloaded, only locked tabs and fields are updated. Local user configuration data in unlocked fields are kept.

RETAINING LOCAL DEVICE CONFIGURATION DURING FACTORY RESET

1. Select the option **Advanced options > Retain local configuration (unlocked fields)**, see above.
2. In the command dialog for the factory reset, select **Retain local configuration (unlocked fields)**.

The device is reset to its initial state and the device configuration of the locked fields is reset. However, the local user configurations in unlocked fields are retained.

For further information, see Factory reset command.

Initiating a one-time update of all device configuration data

For example, in case of a defective user configuration, however, the administrator, in the Scout Console, can set a flag to override all device configuration data on the next device restart.

1. In the Scout Console, for the relevant device or OU, from the context menu, choose **Notifications > Initiate configuration update**.
2. For OUs, in the notification message, select whether you want to apply this on all subordinated devices (default).[1[from Scout 15 2209]]

The relevant devices will be marked to obtain a copy of the relevant Scout device configuration data including unlocked fields, on the next device restart.

Accessing device configuration

Opening the base device configuration

- In the Scout Console, select **Options > Base device configuration**

The **Base device configuration** dialog opens. It contains the global device configuration applying to all devices, unless independent configuration instances are defined.

Opening Device configuration dialog for OUs and devices

- Select an element in the tree view. Then click **Edit > Device configuration >**

or

- For the relevant element, open the context menu. Then click **Device configuration**.

The **Device configuration** dialog of the selected element opens. Possibly, the options are disabled as the **Use parent device configuration** option is selected. In this case, the relevant OU or the base device configuration is specified in the dialog title.

The figure shows the device configuration of a device in the sub-tree of the OU IT_Rome. If the dialog has been opened as described above, all options on all tabs are disabled. Only **Use parent device configuration** can be modified.

OPENING THE RELEVANT DEVICE CONFIGURATION DIALOG (PREFERRED METHOD)

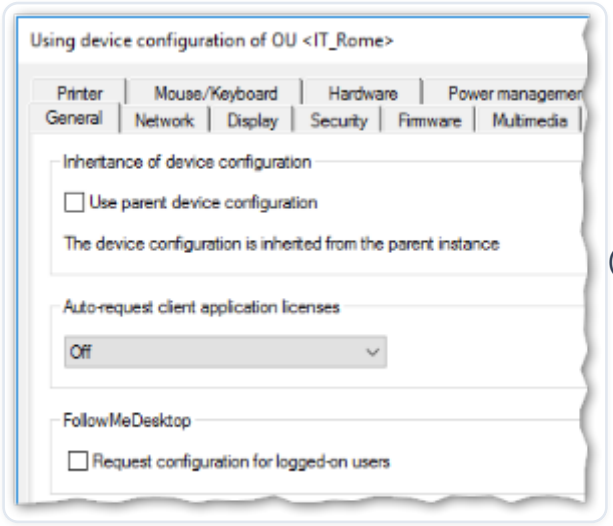
This method allows you to quickly and easily access the point where the device configuration is defined for the relevant devices.

1. In the tree view, select a device (or an OU).
2. To show the **Properties** window, click **View > Window > Properties**.

   In the **Properties** window, next to **Configuration**, the instance is displayed from where device configuration data are applied to the selected element.

3. In the **Properties** window, double-click **Configuration**.

The **Device configuration** dialog of the displayed instance opens. It contains the device configuration data applied to the selected instance. The options of all tabs can be edited as far as permitted by the user rights.



(/en-us/unicon-elux-scout/2603-ltsr-cu1/media/admin_guides/scout/dia_setup_ou.png)

The figure shows the relevant device configuration for the same device. This is the configuration of the OU IT_Rome.

Comparing device configurations between OUs or devices

The device configuration of different OUs or devices can be compared by using a dedicated window.

1. Click **View > Window > Compare configuration**.

   The window Compare configuration is shown as a permanent window in the lower part of the console window.

2. Drag two or more OUs or devices into the **Compare configuration** window by using a drag-and-drop operation.

   Or:

   On the context menu of the relevant OU or device, click **Edit > Add to configuration compare**.

3. On the icon bar of the **Compare configuration** window, click the icon.

   The device configuration of the listed OUs or devices is compared. Differences in the main properties are shown.

4. To view all of the information, on the icon bar of the **Compare configuration** window, click the `All` icon.

All properties are shown.

> **Note:**
>
> To compare actual and target settings of individual devices, use a report. For further information, see Evaluating configuration data.

Locking configuration transfer

Individual devices can be excluded from the process of updating device configuration data.

REQUIRES
Object right Activate/Lock configuration transfer

1. For the relevant device, open the context menu.
2. Click the option **Lock configuration transfer**.

The device is no longer provided with updated device configuration data but remains in the Scout management.

> **Note:**
>
> For newly added devices without management license, the configuration transfer is automatically locked.

Configuration method

During system start of the devices managed by the Scout Enterprise Management Suite®, the devices connect to their Scout Server and check whether any updated configuration data are available, taking inheritance into account. There may be updates for the following data:

- Device configuration
- Application definition
- Files configured for transfer

- Advanced file entries

The Scout Server identifies the relevant configuration data at run-time. That means that all modifications made to configurations in the Scout Console until then are included.

- Configuration run

Configuration run

If the system determines updated configuration data for a device, the relevant modified data are identified, compressed and saved to the database. They are then transferred to the device in one step.

When modifying the configuration of a large number of devices (e.g. when changing the application definitions due to a move to another back-end infrastructure), the administrator can initiate the process of identifying and compressing the relevant data in a data object in advance, for instance at night. To do so, the administrator uses the **Configuration run** command. With this command, the required configuration data can be prepared to be ready for transfer on the next device restart, which might be on the next working day.

PERFORMING A CONFIGURATION RUN

> **Note:**
>
> The configuration run command only prepares configuration data for devices with a configuration delta.

1. For the relevant OU or Dynamic Device Group, from the context menu, choose **Commands**.

2. On the sub-menu, click the **Configuration run** command.
3. Specify a time for execution and confirm with **Schedule**.

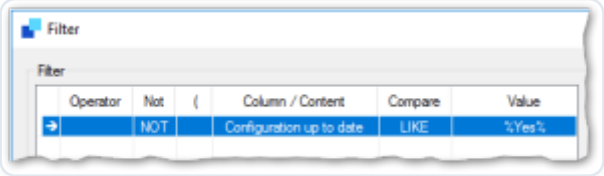
*The processing progress is shown in the **Command history**.*

Evaluating configuration data

The Scout Report Generator provides fields you can use to analyze configuration data:

Configuration ID	ID for a compressed data object holding configuration data. Is created either as a result of a configuration run, or by synchronizing configuration data on the first device to server contact after the configuration settings have been modified.
Configuration up-to-date	Yes -- the device has the configuration that is currently defined for it in the Scout Console. No -- the device is running an earlier eLux® version which cannot evaluate the field.
Configuration transfer	The option **Lock configuration transfer** is active for the device. For further information, see Locking configuration transfer.

Example:



FollowMe Desktop

> **Note:**

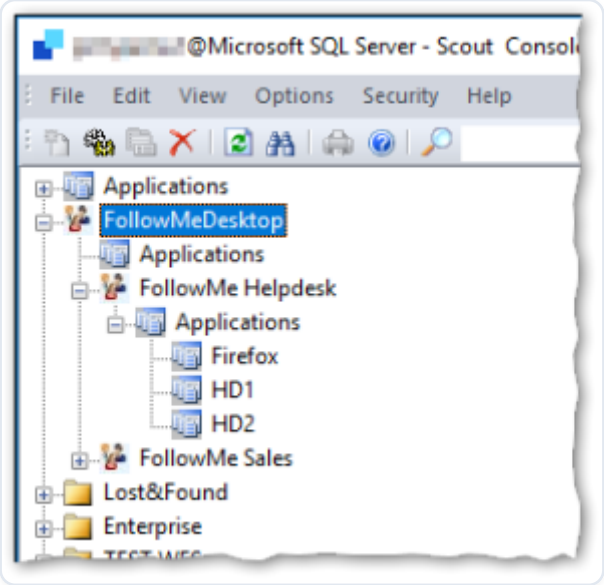
> To use this feature, you will need user authentication via Active Directory.

> FollowMe Desktop allows you to make user-specific configuration settings that are valid across devices.

In addition to the device configuration option, which is bound to devices, the FollowMe Desktop feature allows users to take their configuration settings with them to any device. Users' configured desktop layout and configured applications "follow" them to any device they log on to.

FollowMe Desktop is configured in the tree structure node of the same name and is implemented as one of the top-level OUs. This instance can be used right away or can act as a container for further FollowMe configurations, which are added as subordinate OUs. This is how a device-

independent and cross-hierarchical structure of configurations is mapped, which might be motivated by users' subject-specific functions. For example, a **FollowMe Helpdesk** OU could contain all application definitions and the desktop layout for Helpdesk employees.



A FollowMe Desktop configuration may consist of application definitions and defined configuration values of the device configuration. For further information, see Scope of configurable options.

The FollowMe configurations are assigned to users via AD properties. The administrator applies filter definitions to the FollowMe Desktop instance and all subordinate OUs, which then filter for the relevant AD membership.

When an AD user logs on to a device that is enabled for FollowMe Desktop, the user's AD properties are matched against the filter definitions of the FollowMe OUs. If a FollowMe configuration with matching filter values is found, the desktop is loaded with the layout and applications defined in that FollowMe OU. When working with the FollowMe configuration, users can make changes to it, such as placing application icons freely on the desktop. These changes, however, will not be saved.

After the user closes his session and logs off, the original device configuration is reloaded.

FollowMe Desktop configurations can be exported and imported.

- Configuring FollowMe Desktop
- Scope of configurable options

Configuring FollowMe Desktop

To use the FollowMe Desktop feature, make one or more FollowMe configurations available to certain AD users via filter definitions. Then define on which devices you want to allow the retrieval of a FollowMe configuration.

CREATING FOLLOWME CONFIGURATIONS

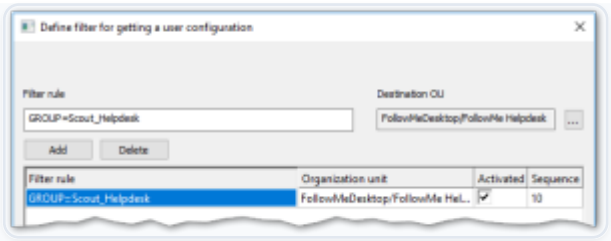
REQUIRES
Administrator base right **Show FollowMe Desktop OU**

1. In the tree structure, below the **FollowMe Desktop** top-level OU, create further FollowMe OUs such as a **FollowMe Helpdesk** OU. To do so, from the context menu, choose **Add > Organization unit**.
2. Edit the device configuration of the new FollowMe OUs (Desktop, Drives, USB options, Power management). Then add the relevant application definitions to the FollowMe OUs.

> **Note:**
>
> New FollowMe Desktop OUs do not inherit configuration values from the base device configuration or base applications. Within the FollowMe Desktop structure, however, inheritance is active by default. Subordinate FollowMe OUs inherit the device configuration and applications from the parent FollowMe instance, unless you disable inheritance.
>
> *The configuration values of a FollowMe-OU overwrite the device configuration values valid on the device.*

3. For each FollowMe OU, define a filter that filters for an AD membership. To do so, from the context menu, choose **Set filter**.

> **Note:**
>
> The filter definitions are independent of any inheritance. For each FollowMe OU, define a separate filter.



In the filter rule, you can filter for the AD properties **User**, **Group** or **Domain**. Syntax: USER|GROUP|DOMAIN=<Value\> Multiple filter rules are allowed.

> **Note:**
>
> For devices with the FollowMe Desktop enabled, the user AD properties are written to the eluxd.log file (search for FollowMeDesktop).
>
> *All active filter rules are processed in the specified order. AD users matching this filter definition will have access to the applications and desktop properties configured in the FollowMe OU after logging on to devices for which FollowMe Desktop is enabled.*

User-related FollowMe configurations are defined. In the next step, define for which OUs or devices these configurations may be retrieved.

ENABLING DEVICES FOR FOLLOWME DESKTOP

Specify the devices that shall have access to multiple configurations. In some cases, users may prefer the device configuration defined in the OU structure on their workstation device. On other devices, for example in the service area, they may prefer a FollowMe configuration.

In the device configuration of an OU / device (or in the base device configuration), you define whether a device should retrieve Follow-Me configurations when authorized AD users log on.

- In the device configuration of the relevant devices, under **General > FollowMe Desktop**, select **Request configuration for logged-on users**.

After AD users log on, their AD properties are evaluated. If the Scout Server finds a matching FollowMe OU, the configuration defined there is loaded.

Scope of configurable options

The following properties can be configured for FollowMe OUs:

- All application types with all application properties
- Software default settings for Firefox and Chromium (browser home directory)

- Device configuration with following options

	OPTIONS
Desktop	All options except date and time
Desktop > Advanced	Advanced Sort Configuration panel
	System bar
	Background picture
Drives	All options
Hardware	All USB options
Power management	All options

Configuration values of a FollowMe OU will overwrite the configuration values of the "normal" device configuration on this device. For options not listed, the values of the "normal" device configuration remain active.

> **Note:**

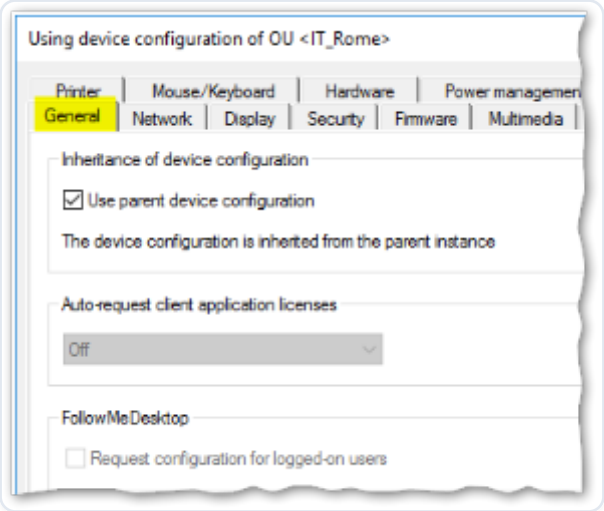
> Within a FollowMe OU, only those applications will be available that have been defined for it.

> The administrator can further limit the scope of the configurable options of FollowMe OUs for operational administrators (**Security > Manage administrators > Default object rights**).

General tab

Use parent

This option of the **General** tab is active by default and ensures that consistent device configurations are used. If **Use parent device configuration** is selected, all other fields of the dialog are disabled: For the relevant device or OU, the device configuration is used that is displayed in the title bar (in the figure IT_Rome). This is where an administrator will normally make any configuration changes.



Note:

In some cases, it might be useful to disable the **Use parent device configuration** option temporarily.

For further information, see Blocking inheritance - independent device configuration.

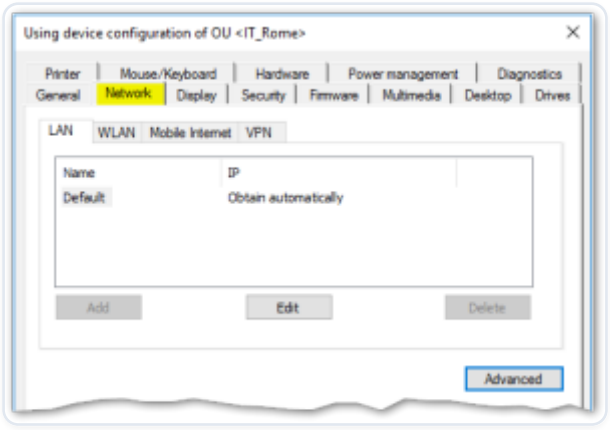
Additional options

OPTION	DESCRIPTION
Auto-request client application licenses	Enables the device to automatically request application licenses when licensed applications are installed or used. To do so, in the **License information** dialog, select **Auto-assign**. For further information, see Automatic assignment of application licenses in the **License management** short guide.
[FollowMe Desktop] / Request configuration	Allows authorized users to use multiple configurations on this device. For further information, see Configuring FollowMe Desktop.

Hardware information for individual devices is listed in the **Properties** window of the Scout Console. For further information, see Properties.

Network tab

Depending on the installed image and the hardware used, users can use different network connections. In the Scout Console, you define network profiles for different network types. The user can then choose from the defined network connections on the system bar.



The following network profiles are available:

- LAN (only one profile, cannot be deleted)
- Wireless LAN
- Mobile Internet (Wireless Wide Area Network)
- VPNfrom Scout Enterprise Management Suite® 15 2101 For earlier versions, see VPN in the Advanced device configuration.

To create VPN network profiles, use the Advanced device configuration for individual devices. For further information, see VPN.

- Defining a LAN profile
- Advanced network settings
- Defining a WLAN profile
- WPA support
- Corporate WLAN
- Defining a Mobile Internet profile (WWAN)
- Defining a VPN profile
- Options for all network profiles
- Proxy configuration
- Internet Protocol version 6 (IPv6)
- Firewall for eLux devices

Defining a LAN profile

1. For the relevant device or OU, open **Device configuration > Network**.
2. Select the **LAN** tab. Then, for the **Default** connection, click **Edit**.
3. In the **Edit network profile** dialog, under **Ethernet**, edit the following fields:

OPTION	DESCRIPTION
Obtain an IP address automatically	The IP address is obtained automatically via DHCP.
	Define a timeout period in seconds.
	Later on, under , specify the behavior for failing DHCP requests.
Use following IP address	Alternatively, specify a fixed IP address and the corresponding options.

> **Note:**
>
> If you do not use DHCP options for Scout, we recommend that you select **Advanced > Ignore Scout Server DHCP options**.

4. To modify the network speed and Maximum Transmission Unit (MTU), edit the **Medium** tab.
5. Under **Advanced**, edit the following fields:

DHCP settings	Specify the behavior for failing DHCP requests.
Proxy	Define a system-wide proxy server for this network profile, see Proxy configuration.
	The proxy setting you define here is used by the option in the browser application definition.

6. Under **IEEE 802.1X authentication**, edit the following fields:

Activate	Enable IEEE 802.1X authentication in general.
Allow LAN connection without, 802.1X if 802.1X fails	Specify whether a connection may be set up if a timeout or authentication error for 802.1X occurs (for Ethernet connections only). If the option is cleared, users can only connect after successful 802.1X authentication.
Number of auto-connect retries	Number of connection retries before aborting
Number of authentication retries	Number of authentication retries for a successful connection before the authentication is aborted
Timeout authentication	Time period in seconds before an authentication try is aborted

> **Note:**
>
> The WPA encryption is performed using the WPA supplicant and the configuration file wpa.conf. For further information, see WPA support.

7. Confirm with **OK** and **Apply**.

Use the **Internet connection test** option to check anytime whether web addresses are accessible via the Internet. For further information, see Options for all network profiles.

Advanced network settings

In **Device configuration > Network > Advanced** you will find a Hosts list as well as features that apply to all network connections.

Defining a timeout for a connection

- Under **Management timers**, in the relevant fields, enter the desired timeout in seconds
- when a connection is set up
- when a connection is in idle state

After the specified time, the connection will be terminated.

The option **Send Keepalive packet** ensures that the device sends keepalive signals to the Scout Server in the specified time interval, provided the Scout Keep Alive service is installed. For further information, see Configuring status messages for devices.

DEFINING A HOSTS LIST FOR NETWORKS WITHOUT DNS SERVER

If the network is not equipped with a domain name server (DNS), host names can be resolved locally by the device.

1. Click **New**.
2. Enter an IP address and one or more host names.

> **Note:**
>
> To specify multiple host names for your IP address, enter the IP names one after the other, separated by spaces. Example: 127.0.0.1 host1.domain.com host2.domain.com host3.domain.com

3. Confirm with **OK**.

The Hosts list is automatically transferred on the next device restart.

Defining a WLAN profile

The following configuration options are provided:

A. In the Scout Console, in the device configuration, you can create a WLAN profile for a device, OU or for all devices, see below.

EAP authentication is not supported with this method.

B. Users can create individual WLAN profiles locally on the device.Local profiles and profiles created in Scout can be merged automatically.This way you can make them connect depending on the user's location.

C. Corporate WLAN: A WLAN configuration can be distributed throughout the entire company network by using a WPA configuration file with and without 802.1X.

Users can additionally create individual WLAN profiles locally on the device.Configured WLAN networks can connect automatically depending on location and priority.For further information, see WPA support.

CREATING A WLAN PROFILE IN THE SCOUT DEVICE CONFIGURATION

1. In the Scout Console, for the relevant OU, open **Device configuration > Network**.
2. On the **Wireless LAN** tab, click **Add**.
3. Edit the following options:

OPTION	DESCRIPTION
Profile name	Freely selectable name for the WLAN profile
Connect automatically	**Note:** If the option is cleared, there is no automatic use of any existing Wifi connection. In this case, the user must start a WLAN manually from the live information on the system bar.
Internet connection test	For further information, see Options for all network profiles.
Medium > Network name/SSID	Wifi name / Service Set Identifier
Medium > Timeout	Time period in seconds waiting to connect
Medium > Channel	Selected automatically by default
Medium > Encryption	Authentication type
	- None
	- with pre-shared key (PSK)
	- with pre-shared key (PSK)
	To authenticate via EAP (Extensible Authentication Protocol), use a WPA configuration file. For further information, see WPA support.
Medium > Hidden SSID	Select this option if a WLAN is hidden.
	Below, the access point MAC address (BSSID) may be entered. If the network has

OPTION	DESCRIPTION
	multiple access points, the MAC addresses are separated by semicolons.
	BSSIDs are required for devices to automatically connect to a hidden WLAN.
IP > Obtain an IP address automatically	The IP address is obtained automatically via DHCP.
	Define a timeout value in seconds.
IP > Use following IP address	Alternatively, specify a fixed IP address and the corresponding options.
Advanced > DHCP settings	Specify the behavior for failing DHCP requests.

Note:

If you do not use DHCP options for Scout, we recommend that you select **Ignore Scout Server DHCP options**.

OPTION	DESCRIPTION
Advanced / Proxy	Define a system-wide proxy server for this network profile, see Proxy configuration.
	The proxy setting you define here is used by the option in the browser application definition.

4. Confirm with **OK**.

Note:

> To create an individual WLAN profile locally on the device (B), apply the same steps in the eLux device configuration, provided you have the necessary user rights. To check the network activities on the device, use the **Diagnostics** feature (Enhanced log level) and the `systemd-journal.log` file.

WPA support

To secure your LAN and WLAN, you can use WPA encryption with the help of the `WPA-supplicant` software. This software provides key negotiation with the WPA authenticator and controls association with IEEE 802.11i networks. WPA uses IEEE 802.1X and WPA2 uses IEEE 802.11i.

Authentication can be performed either with a pre-shared key (PSK) or, for IEEE 802.1X, via the Extensible Authentication Protocol (EAP).

WPA is configured using the text file `wpa.conf` that can list accepted networks and security policies. The configuration file is saved locally on the devices.

`wpa_supplicant` is a free software application. For further information, see http://w1.fi/wpa_supplicant/.

PROVIDING WPA CONFIGURATION FILE

1. Create a text file named `wpa.conf` by using the **wpa_supplicant** program. See below for an example.
2. To transfer the `wpa.conf` file to the devices, use the Scout feature **Files configured for transfer**. Use the following destination:

LAN	setup/scep/
WLAN	setup/wlan/

   For further information, see Files configured for transfer.

EXAMPLE OF A WPA CONFIGURATION FILE WITH 802.1X (WLAN)

```
ctrl_interface=/var/run/wpa_supplicant

ctrl_interface_group=0

ap_scan=1
```

```
network={
    ssid="&lt;WLAN name>"
    scan_ssid=1
    key_mgmt=WPA-EAP
    eap=TLS
    identity="&lt;Common Name as specified in certificate>"
    priority=6
    ca_cert="/setup/cacerts/root-ca.pem"
    client_cert="/setup/cacerts/client.pem"
    private_key="/setup/cacerts/client.key"
        }
```

Note:

Network profiles (LAN and WLAN) that are transferred to the device via a `wpa.conf` file cannot be edited locally on the device.

For further information on WPA configuration and on using variables, see Configuring WPA supplicant in the short guide **IEEE 802.1X authentication**.

Corporate WLAN

A corporate WLAN providing access to internal resources can be secured by 802.1X with firewall policies tailored to specific needs.

After having set up your WPA configuration file, you can deploy it wherever required. For further information, see WPA support.

With a corporate WLAN, you can allow users to create their own WLAN profiles in parallel. For example, a mobile device could use the provided LAN connection when it is attached to the docking station on the job, but change automatically to the corporate WLAN when undocked. Once the device is started in the home office, eLux® connects to the manually configured WLAN.

For further information, see Adding a WLAN profile in the **eLux** guide.

Defining a Mobile Internet profile (WWAN)

For mobile devices equipped with an appropriate SIM card, define profiles that allow users to connect to cellular data networks such as LTE or UMTS.

1. In the Scout Console, for the relevant OU, open **Device configuration > Network**.
2. On the **Mobile Internet** tab, click **Add**.
3. In the **Edit network profile** dialog, edit the following options:

OPTION	DESCRIPTION
Profile name	Name of the new profile
Connect automatically	If the signal strength is sufficient, the device automatically attempts to connect to the cellular network.
APN	Access Point Name: Address used by the device to connect to the Internet when the cellular data connection is used
PIN	PIN of the SIM card (if used)
	If you leave the field empty and the SIM card requires a PIN, the PIN will be requested on each connection setup.
Username	Username for the mobile account
Password	Password for the mobile account
Roaming	The cellular data connection remains intact when the device is outside the mobile operator's network.

4. Confirm with **OK**.

UNLOCK BLOCKED SIM CARD VIA COMMAND

If, for example, a SIM card has been blocked due to incorrect PIN entries, use the PUK to create a new PIN.

1. For the relevant device, open the context menu and click **Commands > User-defined command**.
2. Enter the following command:

   ```
   mmcli -i 0 \--puk=[\<PUK code\>] \--pin=[\<PIN code\>
   ```

3. Select **Run with system rights**.
4. Click **Execute**.

Defining a VPN profile

-- from Scout --

> **Note:**

> To create VPN network profiles in earlier versions, use the for individual devices. For further information, see VPN.

> You can define one or more VPN profiles for entire OUs.

1. In the Scout Console, for the relevant OU, open **Device configuration > Network**.
2. On the **VPN** tab, click **Add**.
3. In the **Edit network profile** dialog, edit the following options:

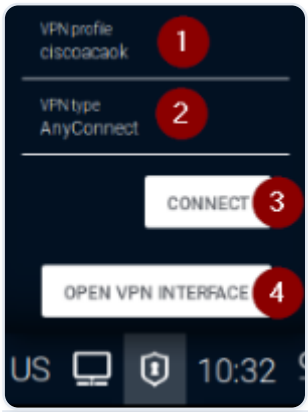
OPTION	DESCRIPTION
Profile name	Name for the new VPN profile
Connect automatically	The VPN client starts automatically and sets up a connection.
VPN client type	Choose between the following options:
\^	- OpenVPN
\^	- CiscoAnyconnect

OPTION	DESCRIPTION
\^	Use this option also for the 1[from eLux RP 6 2302.1000]] -- both Cisco VPN clients can be used to connect to an AnyConnect VPN.
\^	- User-defined VPN client
\^	Specify the ID number of your VPN client.
Configuration (only OpenVPN)	Name of the OpenVPN configuration file without file name extension. The specified configuration file must be available on the devices.

4. Confirm with **OK**.

Note that the eLux devices need to have the VPN software packages and the relevant certificate. A configuration file may be added. For further information on creating OpenVPN and Cisco AnyConnect profiles, see also VPN.

Once users have an active VPN connection, they can use the live information icon on their system bar:



1	Name of the VPN profile
2	Active VPN network
3	Connect or disconnect
4	Open VPN interface

Options for all network profiles

OPTION	DESCRIPTION
Name	Name for the network profile (can be defined freely)
Connect automatically	(except for LAN)
Internet connection test	Each time a connection is set up, the system checks whether addresses on the Internet can be reached via the current network profile (LAN and WLAN). If a connection to the Internet cannot be set up, the system checks for the existence of a captive portal and, if available, redirects to it.
\^	**Auto**: A connection test is automatically performed for WLAN profiles, provided that no central system proxy is defined. (default)
\^	**On**: For LAN and WLAN network profiles, the Internet connection is automatically tested.
\^	**Off**: No automatic connection test is performed. The device does not display any content in the browser and does not attempt to connect to external services or websites.
\^	The option is protected by a dedicated object right **Device configuration > Network > Handling of network profiles > Internet connection test** and by a dedicated user right. [from Scout 15 2101]

Proxy configuration

For each network profile, you can define a proxy server that is used by web clients or browsers. The proxy server can be configured manually or automatically.

If you define the proxy server centrally in the device configuration, it can be accessed from all application definitions (browsers). This central **system proxy** contains the proxy setting which can either be a fixed server setting, automatically determined, or simply .

Using an automatic WPAD configuration, all web clients of an organization can then be configured easily to the same proxy server or servers.

For the **system proxy** setting, in the network profiles, you will find the options described below.

- Scout Console: **Network > Advanced**
- eLux® RP 6: **Network configuration > Advanced > Use proxy > Proxy settings**

OPTION	DESCRIPTION
No proxy	No proxy server is used
Manual (Proxy:Port)	Specify fixed proxy server with port number
	Example: `proxy.sampletec-01.com:3800`
	To define destinations that you do not want to access via proxy, in the **Proxy exception list**, enter the relevant network addresses separated by semicolons.
Auto (URL)	Proxy auto-config (PAC): Determines the appropriate proxy for each URL
	Examples
	`http://proxy.sampletec-01.com/proxy.pac`
	`http://wpad.sampletec-01.com/wpad.dat`
Pass-through logon for proxy (with AD user authentication)	If a central **system proxy** is configured with AD authentication, the AD logon data are used for authentication.
	Proxy authentication may be required if you use browser content redirection under Citrix.
Proxy username	Username for authentication on the system proxy
Proxy password	Password for authentication on the system proxy

Note:

When you define a browser application, the default proxy setting is . The proxy setting defined in the relevant network profile is now active. For further information, see Defining a browser application.

Internet Protocol version 6 (IPv6)

In addition to full support of Internet Protocol Version 4 (**IPv4**), **IPv6** is used by default for local applications including automatic network configuration (DHCP, DNS, NTP).

- To disable IPv6, in the Scout Console, for the relevant devices, configure the following Advanced file entry:

File	/setup/terminal.ini
Section	Network
Entry	DisableIPv6
Value	true (default = false)

For further information, see Advanced file entries.

Firewall for eLux® devices

To secure your eLux devices with a firewall, for example to allow exactly one more connection besides the connection to the Scout Server, use the eLux **Firewall support** package, which allows you to define appropriate rules.

By default, **nftables** are used.Alternatively, the **iptables** syntax can be used.

Once the eLux **Firewall support** package is installed, the firewall rules are applied on system start.By default, only data packets that are required for communication with the Scout Server are allowed to pass.To allow further connections, define filtering rules and transfer them to the devices in a file.

If the **Firewall support** package has been installed on the device without filtering rules, the firewall will only start if the feature package **Strict firewall policy** is installed.Then communication

between the Scout Server and the device is established via the management protocol (port 22125) and no other communication is allowed.

CONFIGURING FIREWALL RULES (NFTABLES)

> **Note:**

> The eLux package **Firewall support** and the included feature packages **eLux firewall plugin** and **firewall nftables programs and libraries** must be installed on the devices. This may require modifications of the image definition file on the web server via ELIAS.

1. Create the nftables.conf file according to the following example:

   ```
   table ip filter {chain input {tcp dport 22 accepttcp sport 80 accept}
   chain output {tcp sport 22 accepttcp dport 80 accept}}
   ```

   In the example, outgoing http and incoming ssh connections are accepted.

2. Transfer the files to the devices to /setup/firewall/nftables.conf.To do so, use the Scout feature **Files configured for transfer**. For further information, see in the **Scout** guide.

CONFIGURING FIREWALL RULES (IPTABLES)

> **Note:**

> The eLux package and the included feature packages and must be installed on the .This may require modifications of the image definition file on the web server via ELIAS.

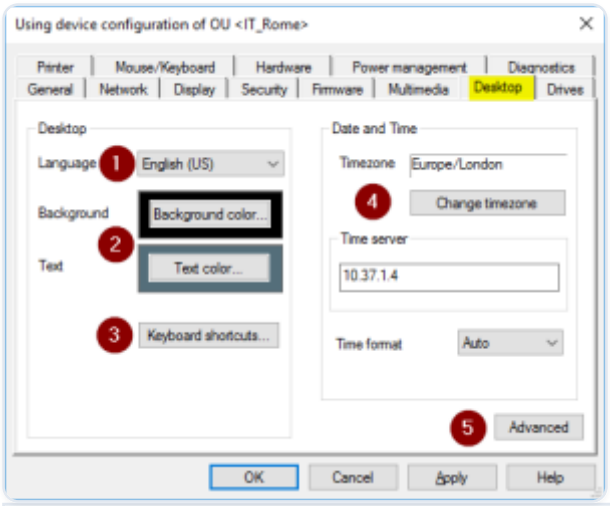
1. Create the file rules.v4 for IPV4 or rules.v6 for IPV6.Use the **iptables** syntax for the rules.
2. Transfer the files to the devices to /setup/firewall/rules.v4 or /setup/firewall/rules.v6, respectively.To do so, use the Scout feature .

   For further information, see in the **Scout** guide.

All rule files are included in the **Diagnostics** feature and are part of the System template.

Desktop tab

On the **Desktop** tab, you can modify the eLux® desktop layout.



1	Display language
2	Background and text colors
3	Keyboard shortcuts
4	Date and time settings
5	Advanced settings

- Configuring language and colors
- Keyboard shortcuts
- Date and time
- Advanced desktop settings
- eLux RP 6 User Interface

Configuring language and colors

1. For the relevant device or OU, open **Device configuration > Desktop**.
2. In the **Language** list, click the preferred desktop and application language.

   The following languages are supported: English, German, French and Spanish

> **Note:**

> The language setting refers to the display of desktop elements. It does not affect text services and input. For a smooth performance, ensure that the applications support the selected language.

3. Click the **Background color** button to select a desktop background color.
4. Click the **Text color** button to select a text color for the application icons. Make sure there is sufficient contrast to the background color.

For further configuration options of the eLux RP 6 desktop, see eLux RP 6 User Interface.

Keyboard shortcuts

For switching applications and locking the screen, keyboard shortcuts are already predefined and you are free to customize them. To define keyboard shortcuts freely, follow the syntax rules given.

DEFINING KEYBOARD SHORTCUTS

1. Open **Device configuration> Desktop > Keyboard shortcuts**.
2. You may select keyboard shortcuts for the following actions:

OPTION	DESCRIPTION	HOW TO DEFINE
Switch applications	Switch between open applications or sessions	Select an option from the list-field.
	The default shortcut Alt+Ctrl↑ helps avoid conflicts with the shortcut Alt+Tab which is used to switch between the tasks within a Windows session.	
Log off	Log off currently logged-on user (AD users)	Specify the desired key combination as free-text, see below.
	The logon dialog is then displayed.	
Lock screen	Activate password-protected screen saver (AD users)	Specify the desired key combination as free-text, see below. Otherwise, the

OPTION	DESCRIPTION	HOW TO DEFINE
		default key combination is active.
	Default: Ctrl+Alt+End	

Furthermore, you can define a key combination for multi-monitor environments that allows users to quickly switch between clone mode and extended desktop.[1[from Scout 15 2101 and eLux RP 6 2101]] For further information, see Multiple monitors.

RULES FOR SPECIFYING KEY COMBINATIONS

- Key combinations consist of a combination of one or more modifier keys and a single non-modifier key
- For the non-modifier key, you can choose from the following keys: Letter keys, number keys, function keys, Windows logo keys, key, position and numpad keys as specified
- Key combinations must be specified in the following format:

```
<modifier key\><modifier key (optional)><modifier key (optional)>additional key
```

No spaces or other characters may be placed between the keys.

- The spelling of the key names must follow the specification, see examples. To receive a complete list of allowed key names and their spelling, use the following command in an eLux® shell:

Modifier key names: Non-modifier key names:

Examples

KEY COMBINATION	DESCRIPTION
Ctrl+Alt+Escape	
Shift+Ctrl+l	
Mod4+Alt+F1	Mod4 corresponds to the Windows logo key
Ctrl+Mod4+Alt+End	

KEY COMBINATION	DESCRIPTION
Mod4+Super_R	(= right Windows button if used as key) Super_R corresponds to the Windows logo key on the right if used as a key
Mod5+KP_End	Mod5 corresponds to Alt Gr, KP_End corresponds to the End key of the numpad

Important:

If you define a key combination for eLux that is already defined within an application/session, this key combination will only work for eLux. Avoid using the same key combinations in different environments.

BEHAVIOR OF THE CAPS LOCK KEY

Note:

May be configured via user interface from , see Advanced mouse and keyboard settings.

In most environments, pressing the () key in combination with letter keys results in the display of uppercase letters, while the number keys above the letter block output numbers despite the key. To display the special characters of the number keys, the key must be pressed.

- To let users write special characters instead of numbers while pressing , for the relevant devices, configure the following Advanced file entry:[2[from eLux RP 6 2107]]

File	`/setup/terminal.ini`
Section	`Keyboard`
Entry	`ForceShiftLock`
Value	true, Default: `false`

For further information, see Advanced file entries.

The Caps Lock key then behaves like the Shift key.

Date and time

OPTION	DESCRIPTION
Time zone	Click [Change time zone] and select the required time zone from the list.
Time server	Under [Time server], specify the relevant server name or IP address.
	The time server must comply with the Network Time Protocol (RFC 1305) or the Simple Network Time Protocol, a simplified form of NTP. Microsoft Windows operating systems include the [W32Time] service which communicates via SNTP in older versions such as Windows 2000, and uses NTP in later versions. The time service is started automatically.
	The service runs on port 123 and uses the UDP protocol.
	For further information on the Windows Time Service, see the Microsoft documentation.
	For further information on NTP, see http://www.ntp.org.
Time format [from Scout 15 2101 and eLux® RP 6 2209]	The time can be displayed in 24 hour or 12 hour time format.
	• Auto (default): The displayed time format depends on the configured display language (see same dialog).
	• 12 hour

OPTION	DESCRIPTION
	• 24 hour

Advanced desktop settings

SYSTEM BAR

OPTION	DESCRIPTION
Show system bar	The system bar will be displayed on the devices. Below, select its behavior and the icons you want to show on the system bar. (selected by default)
Hide automatically	The system bar is hidden by default. As soon as users point the mouse to the bottom of the screen, the system bar is displayed.
Show Desktop icon	Users click this icon to minimize all open windows and show the desktop. (selected by default)
Show live information icons	These icons allow users to view current status information such as plugged USB devices via right-click. (selected by default)
Show time	Shows the current time. When users point the mouse to the time, the current date is shown. (selected by default)
Show Config panel icon	Allows users to open the device configuration (Configuration panel). (selected by default)

Important:

Only when the Configuration panel is displayed, can the administrator unlock the configuration via device password locally on the device.

QUICK CONFIG: QUICK ACCESS TO CONFIG PANEL DIALOGS VIA SYSTEM BAR

All options are selected by default.

OPTION	DESCRIPTION
Volume	Volume control for input and output devices
Keyboard	Keyboard language and key speed
Display	Screen settings
Peripherals	Settings for peripherals such as USB and Bluetooth devices, and COM ports
Network	Network information and setup, disconnect/ connect
Device information	Information on the device
Date and time	Date and time settings

BACKGROUND

Different background images can be defined for primary/secondary monitors and for the time before/after the AD logon.

- Select a background image from the list-field. Then configure it via the buttons.

OPTION	DESCRIPTION
Background picture	Desktop wallpaper (default). If further pictures are defined, this one will be used only for the primary monitor and after AD logon.
Additional picture	Desktop wallpaper for secondary monitors
Background picture AD	Desktop wallpaper until AD logon (primary monitor)
Additional picture AD	Desktop wallpaper until AD logon (second and more monitors)

OPTION	DESCRIPTION
Load	Browse the file system and select a picture file. The picture file will be imported into the database.
	• The following file formats are supported: [.svg], [.png], [.jpg] (Note: .jpg cannot be previewed)
	• Maximum file size: 500 KB
Delete	Remove the current background image from the database.
Set URL	As an alternative to a picture file from the file system, specify a web address for loading pictures.

Note:

Make sure you have enough flash memory on the devices. The background image is stored in the directory of the flash card.

FURTHER OPTIONS

OPTION	DESCRIPTION
Sort Configuration panel	The Configuration panel dialogs are displayed in alphabetical order (active by default)
Timer for shutdown confirmation	Before the device is shut down, a message is displayed for the specified time period (in seconds). This allows users to prevent the shutdown by clicking the **Cancel** button or pressing the ESC key. (deselected by default) This option has its own user right and object

OPTION	DESCRIPTION
	right (for administrators). Both rights are enabled by default.
Desktop sort order	Sort order of the desktop icons. The administrator can specify the sort order for users without the user right **Sorting desktop icons**. Users who have this user right can freely place desktop icons on the desktop.

eLux® RP 6 User Interface

eLux RP 6 devices come with a new desktop interface that can be customized to your needs and that provides a personal desktop view. For further information, see eLux RP 6 User Interface in the **eLux** guide.

CUSTOMIZING THE LAYOUT OF THE ELUX RP 6 USER INTERFACE

1. For the relevant OU, use the Advanced file entries feature of the Scout Console to modify the client file in the section. Add the following new entries and specify the relevant values:

ENTRY	VALUE RANGE	DESCRIPTION
DesktopLogo	none	Option 1 replaces the eLux Logo in the lower right section by the specified picture file. Example: Option 2 removes the eLux Logo in the lower right section.

> **Note:**
>
> To show an individual picture file, transfer the picture to the devices. For further information, see Files configured for transfer.

ENTRY	VALUE RANGE	DESCRIPTION
DesktopTextColor	#<>	Text color of application icons

ENTRY	VALUE RANGE	DESCRIPTION
DesktopHighlightedTextColor	#<>	Text color of application icons when the mouse pointer is moved over them
DesktopTitleTextColor	#<>	Color of the folder/tab title (All Applications, StoreFront store name)
DesktopSearchTextColor	#<>	Text color in the search field
DesktopSearchBackgroundColor	#<>	Background color of the search field
DesktopSearchIconColor	#<>	Color of search icon (magnifier)
DesktopSortIconColor	#<>	Color of sort icon (A-Z)

2. To display a background image, configure the relevant picture file in the **Advanced** desktop settings of the device configuration. For further information, see Advanced desktop settings.

   For multiple monitors, the image will be used as a background image on each monitor.

> **Note:**
>
> The background image overrides the **Background** color defined on the **Desktop** tab.

3. To set a solid background color for the desktop, use the **Background** option on the **Desktop** tab.

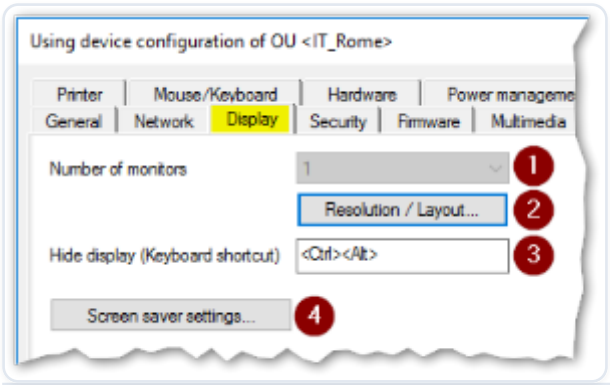
   To set a gradient background color, use the RGB values .

4. To configure the position of the system bar, define the following entry in the file in the Layout section:

ENTRY	VALUE RANGE	DESCRIPTION
SystembarPosition	0/1	0:bottom of the screen
		1: top of the screen

Display tab

On the tab, you can choose between display settings and screen saver settings. To define display settings for one or more monitors, open the dialog.



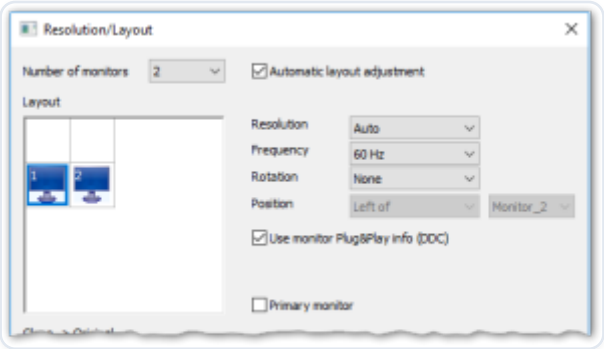
1	Define the number of monitors together with further display options in the [Resolution/ Layout] dialog.
2	Configure the display
	• Display options per monitor
	• Multiple monitors
3	Keyboard shortcut to hide/show the content of a monitor
4	Select and configure a screen saver

> **Note:**

> Power saving options can be found on a dedicated tab, see Power management tab.

Configuring the display

1. If you have defined more than one monitor, in the **Resolution/Layout** dialog, select a blue monitor icon.



2. For the selected monitor, use the list fields on the right to specify the screen resolution, frequency, and rotation.

Resolution	Screen resolutions that are not listed may be added to the database table `dbo.Resolution`. After modifying the table, restart the Scout Console.
Frequency	Refresh rate
Rotation	The screen display can be rotated 270° (left), 180° (inverted) and 90° (right).
Position	Only for multiple monitors

3. To have the values supported by the monitor processed by the device, select **Use monitor Plug&Play Info (DDC)**.

   Clear the option to activate the **Monitor class** field.

> **Note:**
>
> If you decide to use adapters or the analog VGA port to connect monitors to devices, warranty for the operation of these devices will be excluded. These types of combinations are not part of functional acceptance tests.

4. To define the selected monitor as the primary one, select **Primary monitor**.

5. Confirm with **OK** and **Apply**.

> **Important:**
>
> If your monitors do not support the settings you have defined, you may have to perform a factory reset of the device and modify the desired screen settings.

Multiple monitors

Up to eight monitors can be configured.

DEFINING MULTIPLE MONITORS

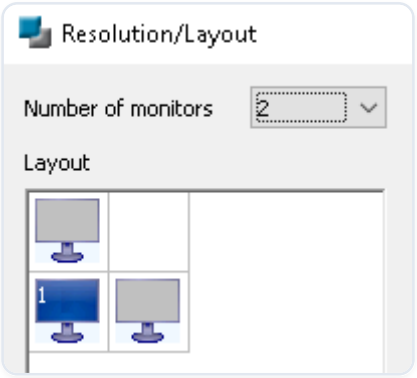
1. In **Device configuration > Display**, click **Resolution/Layout** to open the dialog of the same name.

   The field **Number of monitors** specifies one monitor by default. In the field below, this monitor is represented by a blue monitor icon with the number 1. By default, the first monitor is defined as the primary monitor (see option in the lower section).

   For a different setting, see the instructions below.

2. In the **Number of monitors** list, select how many monitors you want to connect to the devices.

   Once you have defined more than one monitor, their possible positions (horizontal and vertical) are shown as gray monitor icons.

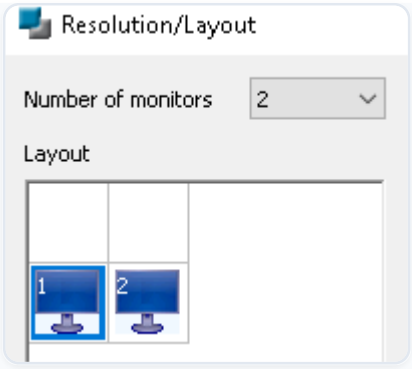


3. Double-click the gray monitor icon that shows the position of the second monitor.

   The selected monitor icon is shown in blue with the number .

> **Note:**

Alternatively, right-click a monitor position to assign a monitor to it.



4. If you have specified more than two monitors, double-click the desired gray monitor icons, one after the other.

   Each of the defined monitors is shown as a blue monitor icon with its number.

5. To automatically adjust the layout after one of the monitors is removed, select the option **Automatic layout adjustment**.

   If the option is not active, the current layout is retained regardless of the actual situation.

Note:

A four-monitor configuration is supported on the following devices: Dell Z50QQ, Hewlett-Packard t620 Plus and Hewlett-Packard t730. A five-monitor configuration is supported on the following devices: Fujitsu FUTRO S940 and Fujitsu FUTRO S9010.

DEFINING POSITIONS OF ALL MONITORS FREELY

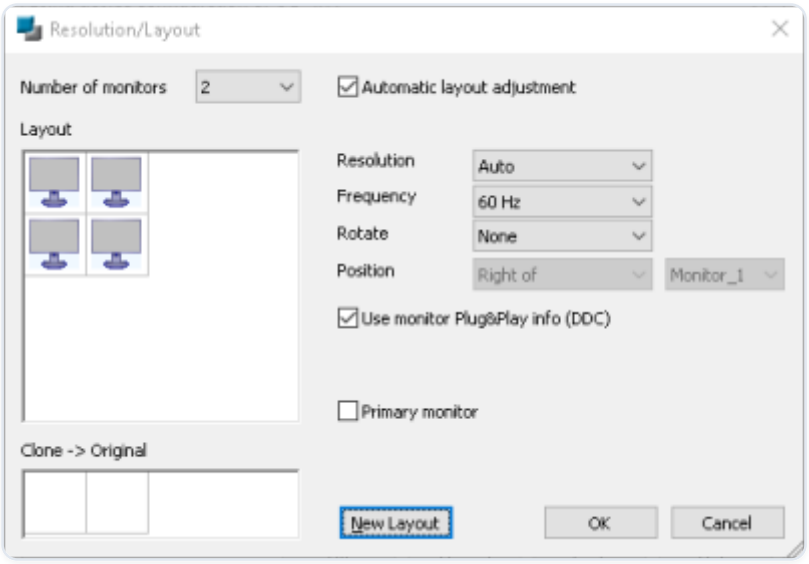
To define the position of the first monitor, use a new layout.

1. In the **Resolution/Layout** dialog, in the **Number of monitors** list, select how many monitors you want to connect to the devices.

   The first monitor is shown as a blue monitor icon. For each additional monitor, their possible positions (horizontal and vertical) are shown as gray monitor icons.

2. Click **New layout**.

   For the number of monitors selected, all possible positions are shown as gray monitor icons:

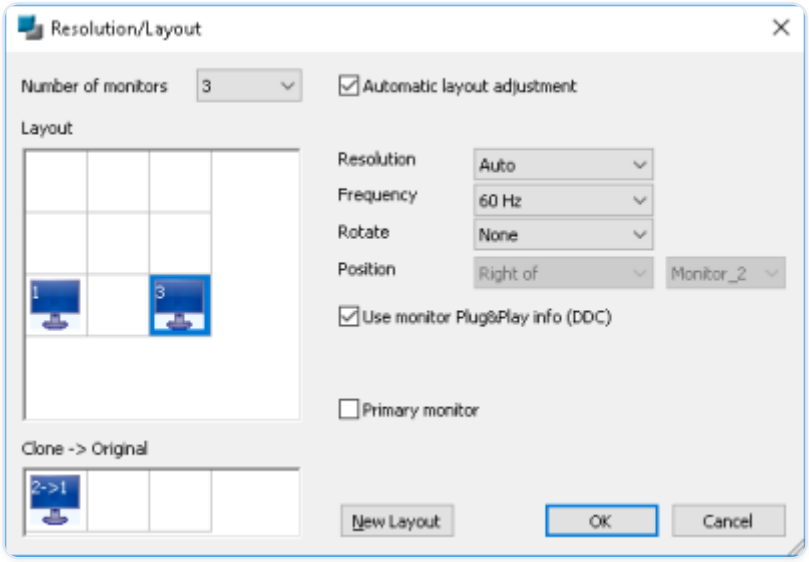


3. Double-click the desired monitor position for the first monitor. Then, double-click the desired monitor positions of the remaining monitors.

CLONING MONITORS (CLONE MODE)

If you have specified more than one monitor, by default the system configures the monitors as extended desktops (one continuous desktop over all monitors). Alternatively, for individual monitors, after having defined them, you can activate the Clone mode (same display on multiple monitors):

- Right-click the last defined blue monitor icon, and from the context menu, choose **Clone of** .



To deactivate the Clone mode, use the **New Layout** feature (see above).

KEY COMBINATION FOR SWITCHING BETWEEN CLONE MODE AND EXTENDED DESKTOP

-- from Scout and --

- To allow users to quickly switch between the two modes, define the following Advanced file entry for the relevant devices:

File	`/setup/terminal.ini`
Section	Hotkeys
Entry	CloneDisplays
Value	`<Mod4>p`
Value	`<Mod4>` corresponds to the Windows logo key

For further information, see Advanced file entries in the **Scout** guide.

Screen saver

CONFIGURING THE SCREEN SAVER

1. Under **Display > Screen saver settings**, choose between a black screen, a specific screen saver or multiple screen savers.
2. Depending on the option chosen, select one or more screen savers from the list. To select multiple entries, press `Shift` or `Ctrl`

> **Note:**
>
> The **HTML** option allows you to choose a website.

3. To configure each screen saver, use the settings on the right.

ENABLING THE SCREEN SAVER

- On the **Power management** tab, for each profile, select **Enable screen saver after** and specify a waiting time in minutes.

LOCKING THE SCREEN ON THE DEVICE

If the screen saver is enabled, eLux® users can lock the screen before the configured waiting time with the following key combination:

- Press CTRL+ALT+END
- Password-protected screen saver
- Downloading picture files for screen saver

PASSWORD-PROTECTED SCREEN SAVER

Note:

When user authentication is enabled, password protection of the screen saver becomes active and cannot be turned off.

The password is set to $ELUXPASSWORD. For further information, see Where to apply user variables.

The screen saver becomes active after the defined time period and the system is locked. By pressing a key or moving the mouse, a dialog is displayed for unlocking. It provides the following options to users:

OPTION	BUTTON	DESCRIPTION
The logged-on user unlocks the screen by entering his/her password / smart card	Unlock	Default
Another person leaves a message for the logged-on user	Message	The screen remains locked. The logged-on user receives a notification with the message when he or she unlocks the screen. This function is enabled by default and can be disabled by the following Advanced file entry: File: [/setup/terminal.ini], section: [xscreensaver_dialog], entry: [MessageEnabled], value: [false]
Another user authenticates to log off the previous user (and to log on), restart or shut down the device.	Log off	Useful if devices are used by multiple users: Allows users to reuse devices that have been left without logging off and therefore are blocked. Once the new user has authenticated, the **Restart**,

OPTION	BUTTON	DESCRIPTION
		Shut down and **Log off** buttons become active. In any case, the previously logged-on user is logged off. This function is disabled by default and can be enabled by the following Advanced file entry: File: [/setup/terminal.ini], section: [xscreensaver_dialog], entry: [ShowSysCommandButtons], value: [true]

Important:

Data loss may occur if the **Log off** option is used followed by a restart, shutdown, or logoff. The user currently logged on is logged off regardless of whether the documents or data last edited have been saved.

DOWNLOADING PICTURE FILES FOR SCREEN SAVER

Optionally, Scout administrators may configure direct download of picture files to the devices into the screen saver's picture directory. To do so, use the **FileFetch** tool, which downloads graphic files via **wget**.

REQUIRES
• The eLux® package **FileFetch** must be installed on the devices. This may require modifications of the image definition file on the web server via ELIAS.
• The picture files must be located in the specified directory and have the file extension .gif, .jpg, or .png. The filename must be numeric. Examples: 0001.jpg, 0002.jpg, 0003.png, 0004.gif

- For the relevant devices, configure the web server and the directory of the picture files.To do so, choose the **Advanced file entries** feature of the Scout Console:

File	[/setup/terminal.ini]
Section	[FileFetch]
Entry	[URL]
Value	URL of the web server including path
Example	[http://webserver.sampletec-01.com/eluxng/pictures]

For further information, see Files configured for transfer.

The **FileFetch** tool checks on each device restart whether new picture files are available on the web server for download.

Hide display

-- from Scout --

Users can temporarily hide the content of a monitor using a predefined key combination. This can be helpful when multiple monitors are used and certain information should not be visible to others present.

The following options are provided:

OPTION	DEFAULT KEYBOARD SHORTCUT
Hide display of primary monitor	Ctrl > Alt > 1
Hide display of second and all other monitors	Ctrl > Alt > 2

The digits **1** and **2** are fixed. The modifier keys Ctrl > Alt may be defined differently by you.

DEFINING ALTERNATIVE MODIFIER KEYS

- Next to **Hide display (Keyboard shortcut)**, enter the desired modifier keys. For further information on how to spell the key names, see Keyboard shortcuts.

Mouse/Keyboard tab

- Configuring mouse

- Configuring keyboard
- Advanced mouse and keyboard settings

Configuring mouse

- On the **Mouse/Keyboard** tab, under **Mouse**, edit the following fields:

OPTION	DESCRIPTION
Show mouse pointer	To hide the mouse pointer, clear the option.
	The mouse type is automatically identified.
Touchpad (for mobile devices)	On - Enables touchpad (default)
	Off - Disables touchpad
	Auto -Disables touchpad when a mouse is plugged in
Double-click speed	Double-click speed defines the time interval between the two clicks of a double-click.
Acceleration	The faster the mouse pointer, the smoother the movements.

Configuring keyboard

- On the **Mouse/Keyboard** tab, under **Keyboard**, edit the following fields:

OPTION	DESCRIPTION
Language	Keyboard layout
Type	If the value is set to [Auto] (default), the keyboard type is identified automatically by the system.
Delay	The delay controls how long a key needs to be pressed until the letter is retyped.
Speed	The speed controls how fast a letter is retyped when a key is pressed.

Advanced mouse and keyboard settings

1. On the **Mouse/Keyboard** tab, click **Advanced**.
2. Edit the following fields:

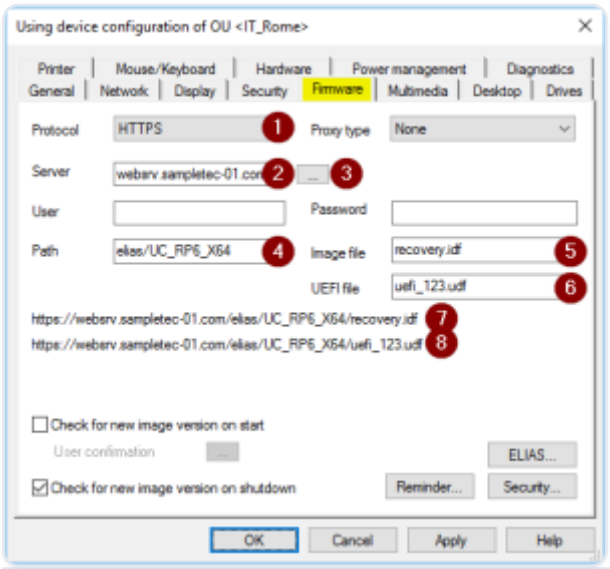
OPTION	DESCRIPTION
Left-handed	Switches primary and secondary mouse buttons
Dead Keys	Dead keys only produce visible output when they are followed by a second keystroke. Accent keys are dead keys as they need to be pressed before you press a character key (` + A => à). Note: Some hardware platforms and some applications do not support this option.
Console switch	Users can use key combinations to switch between consoles. If the option is not selected, console 1 (/en-us/unicon-elux-scout/2603-ltsr-cu1/admin_guides/elux7/1_introduction_e7/1_shortcuts.html) in the **eLux** guide.
Extended keys	Enables multimedia keys and other keys with special functions on the keyboard.
Handle Caps Lock always like Shift [1[from Scout 15 2204]]	With being activated, the number keys above the letter block output special characters instead of numbers. So these keys behave in combination with in the same way as in combination with . For further information, see Keyboard shortcuts.
Num Lock	On: Enables the numeric keypad of the keyboard on device start (default). Off: Disables the numeric keypad of the keyboard on device start. Auto: Enables the numeric keypad on mobile devices and disables it on other devices.

3. Confirm with **OK** and **Apply**.

The modifications become active on the next device restart.

Firmware tab

The **Firmware** tab provides the relevant information required to perform a firmware update (software update) of the devices via network.



1	Network protocol for software package transfer from the web server to the devices
2	Name or IP address of the web server providing the eLux® software packages and the image files
3	Optional: alternative web server for devices connected via VPN[\^1]
4	Directory path of eLux software packages on the web server (container path)
5	Image file on the web server, also called image definition file or IDF, defines the software packages to be installed on the devices
	The image is created with the ELIAS application and then made available on the web server.

6	Optional: UEFI file in eLux container with assignment of device types and UEFI firmware to be installed[\^2]
7	From 1-4 a URL is generated, which is used by the devices to update the firmware.
	If the web or FTP server is password-protected, the user name and parametrized password are also included in the URL.
8	If a UEFI file is specified, the system generates a URL which is used by the devices to update the UEFI system.

Note:

The image file and container path can be parametrized if required.

The image file, UEFI file and container path can be predefined globally by the administrator so that you may select them from the respective list-field. For further information, see Predefined IDFs and containers.

- Configuring firmware updates
- Protecting firmware configuration
- Different hardware models
- Different eLux versions
- Update to new partition layout (1.77 / 1.84 GB to 2.35 / 2.41 GB)
- Update to new partition layout (2.35 / 2.41 GB to 3.75 GB)
- Firmware security through signature
- Update deferment by users

Configuring firmware updates

Note:

> The fields **Protocol**, **Server**, **Path** and **Image file** are used to build a URL used by the devices for firmware updates. The URL address is displayed below the **Path** field.

1. For the relevant device or OU, in the Scout Console, open **Device configuration > Firmware**.
2. Edit the following fields:

OPTION	DESCRIPTION
Protocol	Network protocol of the web server for software package transfer to the devices (HTTP, HTTPS, FTP, FTPS)
Server	Name (FQDN) or IP address of the web server containing the eLux® software packages and the image definition file
2nd web server for VPN devices (optional)	Click … to specify an alternative web server for devices connected via VPN: from Scout 15 2107 and eLux RP 6 2204. Choose the protocol (HTTP or HTTPS) from Scout 15 2204 and enter the server name as FQDN or IP address. The system displays a message if the name cannot be resolved or the IP syntax is incorrect.
Proxy	Static.
User and Password (optional)	Username and password (if required) to access the eLux software container of the web or FTP server
Path	Directory path of eLux software packages on the web server. Use slashes / to separate directories.For ELIAS 18, specify the path name defined during installation. Example: elias/UC_RP6_X64, For the legacy ELIAS, use `eluxng/<container name>`, To account for different eLux major versions, use a specific parameter. For further information, see(/en-us/unicon-elux-scout/

OPTION	DESCRIPTION
	2603-ltsr-cu1/admin_guides/scout/setup/firmware/firmware_macro_container.html).
Image file	Name of the image definition file (IDF) on the web server which is used for firmware updates, Depending on the object rights, an image file name may be specified or an image must be selected from the list-field. For further information, see (/en-us/unicon-elux-scout/2603-ltsr-cu1/admin_guides/scout/setup/firmware/firmware_protect.html). To account for specific hardware models, use the release parameter. For further information, see Different hardware models. To update devices with an earlier partition layout to a current version with an extended system partition, see Update to new partition layout.
Check for new version on start / shutdown	The device checks during start or shutdown whether any firmware updates are available and necessary. To allow users to decline or defer an update on start-up, specify the **User confirmation** options. For further information, see Updates initiated by the system.
ELIAS button	Starts the ELIAS tool and opens the image definition file indicated in the **Image file** field
Security button	The **Security settings** allow you to define a signature check before update through the device. Signature checks can be performed for the image definition files and/or eLux software packages.
Reminder button	The **Reminder Settings** allow you to define whether a user is allowed to defer a firmware update and for how long. Moreover, you can specify time intervals for the update

OPTION	DESCRIPTION
	reminder. For further information, see Update deferment by users.

3. Test the **Firmware** settings on a device . To do so, on the eLux RP 6 device, on the [Command panel] click **Update**. For further information, see Updating the firmware in the eLux guide.

   If the settings have been defined correctly, a connection to the is set up to check whether an update is necessary.

Protecting firmware configuration

Image definition files (IDF) are provided on the web server in an eLux® container. They must be specified in the device configuration under **Firmware** in order for the devices to access the intended image in the event of an update request. Depending on the object rights defined, in the **Firmware** configuration, administrators are allowed to enter individual IDF names as free text or need to select one of the predefined IDFs from the list-field. The same applies to the software container (**Path** field) in the **Firmware** configuration.

To protect such critical firmware configuration parameters, the IDFs and the container paths to be selected or configured for a firmware update can be defined in advance. In combination with the relevant object rights, operational administrators can then only choose between predefined values.

For and later versions, the firmware of UEFI systems[1[from eLux RP 6 2107]] can be updated via the same mechanism as the software (firmware update). Therefore, a **UEFI file** field can be found in the same dialog that behaves accordingly.

SETTING OBJECT RIGHTS FOR FIRMWARE CONFIGURATION FIELDS

The object rights for the **Image file**, **Path**, and **UEFI file** fields are each divided into **predefined** and **user-defined**. If you grant an administrator both rights, he can add new entries as free text as an alternative to selecting a predefined entry from the list-field.

REQUIRES
Administrator policies are enabled.

1. For the relevant OU, from the context menu, choose **Object rights...**
2. Select the relevant administrator /administrator group and click **Edit object rights...**
3. For **Device configuration > Firmware**, change the object rights as required by double-clicking or pressing the `Space` bar:

| Image file (predefined) | The administrator can only select one of the IDFs provided in the **Image file** list-field on the **Firmware** tab. | | | The list-field contains predefined IDFs (see below). If predefined IDFs are missing, the list-field shows the recently used IDFs. |

| Image file (user-defined) | The administrator is allowed to enter any IDF name into the text field. |

| Path (predefined) | The administrator can only select one of the paths provided in the **Path** list-field. | | | The list-field contains predefined paths (see below). If predefined paths are missing, the list-field shows the recently used paths. |

| Path (user-defined) | The administrator is allowed to enter any path into the text field. | | | The path must correspond to a software container on the web server. |

| UEFI file (predefined) | The administrator can only select one of the files provided in the **UEFI file** list-field. | | | The list-field contains predefined UEFI files (see below). If predefined files are missing, the list-field shows the recently used UEFI files. |

| UEFI file (user-defined) | The administrator is allowed to enter any UEFI file name into the text field. |

4. Confirm with **OK**.

For further information, see Administrator policy.

PREDEFINING FIRMWARE CONFIGURATION VALUES

1. On the menu, click **Options > Advanced options > Predefined IDFs**.
2. To add additional IDF names, click the **Add** button and edit the new entry. Note that the spelling must match the actual names.
3. For all entries you want to share in the firmware configuration, select the **Valid** option.
4. Confirm with **Apply** and **OK**.

All valid IDFs, container paths and UEFI files are provided in the device configuration under **Firmware** and can be used by authorized administrators.

> **Note:**

> Scout does not check the physical existence of files or container paths on the web server.

> For further information, see Predefined IDFs and containers.

Different hardware models

-- from eLux® RP 6 2103 --

To update the firmware of devices with an alternative image depending on the hardware model, the release parameter is available. For example, you can update newer model types to a CR version, while older models remain on their previous (LTSR) image. If there are no changes to the previous image, as in the LTSR image example, all devices whose type is not whitelisted are unaffected by the update command.

For the release parameter, in the firmware configuration, an `__RM__` string is inserted into the image file name.

Before an update is executed, an appropriately configured device resolves the parameter using a whitelist:

- If the model type of the device is part of the whitelist, the `__RM__` parameter is replaced by a string you define. So the device will pull the alternative image with the newly created name.
- If the model type of the device is **not** part of the whitelist or the whitelist cannot be loaded, the `__RM__` parameter is removed from the image name. This leaves the device on the previous image (image name as in the firmware configuration, but without the `__RM__` string).

CREATING A WHITELIST

REQUIRES
The web server must support the file extension [.mee] in the MIME type settings.

1. Create a text file named `elux.mee`. Then, enter the section name `__RM__`.

   > **Note:**
   >
   > Make sure you use the correct spelling for the section name: Two underscores followed by `__RM__` (uppercase) followed by two more underscores.

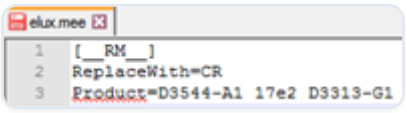
2. Begin the second line with `ReplaceWith=` and then define a short string.

   This string must be included in the image name for the alternative image, see below. If you do not specify anything, the string `CR` is set by default.

3. Begin the third line with the string `Product=`. Then, enter all model types you want to receive the alternative image in the same line.

   Separate the model types by white spaces.

   Enter type names that contain white spaces without the white spaces.

   You can retrieve the model type of a device from the Scout Console. It is shown in the **Properties** window in **Asset > Hardware information > Type**. On the devices, the model type can be found in the `terminal.ini` under `HWInfo.Product`.

4. Copy the `elux.mee` file into your `UC_RP6_X64` container on the web server.

PREPARING THE SOFTWARE CONTAINER ON THE WEB SERVER (ELIAS)

1. Leave the existing image as you want the devices outside the whitelist to receive it.

   Example: `recovery.idf`

2. In ELIAS, create an alternative image that you want the devices in the whitelist to receive.

   For example, to be able to update newer models to a CR version, create an image containing the eLux packages of the latest CR version.

3. Assign the same name to the alternate image, but include the string defined in the whitelist under `ReplaceWith=`

   Example: `recoveryCR.idf`

The software container now contains two images whose names differ only by the defined string, and the whitelist `elux.mee`

MODIFYING DEVICE CONFIGURATION

1. For your OU, open the device configuration under **Firmware**.
2. Under **Image file** insert the `__RM__` string into the file name, see screenshot above. The file extension `.idf` must be kept.

   Example: `revovery__RM__.idf`

> **Note:**

> Make sure you use the correct spelling: Two underscores followed by (uppercase) followed by two more underscores.

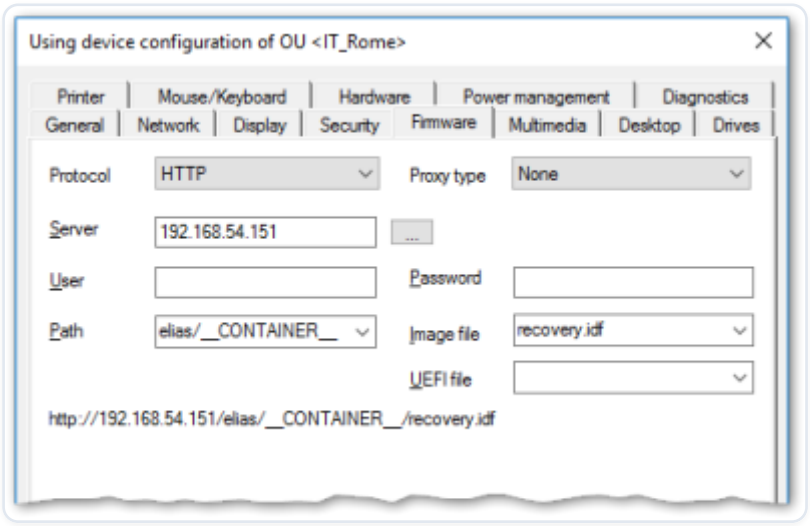
3. Confirm with **Apply** and **OK**.

The next update command on your OU will apply the alternative image to the whitelisted devices.

Different eLux® versions

The `__CONTAINER__` parameter is useful if you have more than one major eLux version in use and you want to update the firmware of multiple devices regardless of their major version. All updated devices remain within their respective major version.

The parameter replaces the container path in the URL



The `__CONTAINER__` parameter is part of the directory path and parametrizes the relevant software container (directory) on your web or FTP server. When the administrator sends an update command to a device configured with a container parameter, the parameter is resolved by a macro according to the installed eLux version.

Example:

If you run devices with both, eLux RP 6 and eLux 7, the eLux RP 6 devices require the `UC_RP6_X64` container, while the eLux 7 devices require the `UC_ELUX7` container. To provide all devices with their appropriate software, in Device configuration > Firmware > Path of all devices , use the `__CONTAINER__` parameter. The devices then resolve the container parameter according to their installed major version to `UC_RP6_X64` or `UC_ELUX7`, respectively. The advantage is that an image file with the same name can be used for both platforms. So, in ELIAS, you would create an image with the same name for eLux RP 6 and for eLux 7.

> **Note:**
>
> In some cases, it can be useful to replace the container parameter by a fixed container name. In this case, the entry in the Path field must correspond to the container name on the web server.

ELIAS 18 CONTAINERS

To use the container parameter with ELIAS 18, choose the following names for your containers:

MAJOR ELUX VERSION	CONTAINER NAME
eLux RP 6	UC_RP6_X64
eLux 7	UC_ELUX7

For further information, see Creating a container in the **ELIAS 18** guide.

SPELLING OF THE CONTAINER PARAMETER

When replacing a fixed container name by the container parameter, make sure you use the correct spelling:

Two underscores followed by the word `CONTAINER` (all uppercase) followed by two more underscores.

> **Note:**
>
> You can use the container parameter in the firmware configuration and in the recovery settings **Options > Recovery setttings**.

Update to new partition layout (1.77 / 1.84 GB to 2.35 / 2.41 GB)

Earlier eLux RP 6 versions up to eLux RP 6 2104 LTSR have created smaller system partitions during installation than current eLux versions. Beginning with eLux RP 6 2107, the system partition is created with 2.35 GB with encryption and with 2.41 GB without encryption, so that larger images can be included.For further information, see eLux RP 6 partitions.

For updating the firmware of devices with eLux® RP 6 2104 LTSR to the current eLux RP 6 version,[1[from eLux RP 6 2110]] the system partition must be repartitioned before the actual update installation. In order to still be able to perform an update installation in one step, the

solution described below uses two images to which devices are updated, one after the other. The second firmware update is automatically triggered.

Note:

When you start from eLux RP 6 2104 LTSR CU3 or CU4, the intermediate and final image will be based on the latest eLux version and you can follow the instructions below. When you start from eLux RP 6 2104 LTSR CU1 or CU2, the intermediate image must be based on a different eLux version. In this case, please contact our support for more details.

TWO IMAGES AS PREREQUISITES

	IMAGE WITH PARTITION RESIZE PARAMETER	FINAL IMAGE
Size	Corresponds to the old world with a maximum of 1.77 GB	May have a size over 2 GB
Content	• Current **BaseOS** package	• All packages from the first image
	• Specific eLux software package named **Update Support Utility** which automatically triggers the second update on the devices	• Additional packages that did not fit into the first image (optional)
		• No **Update Support Utility**-package
Name	Must contain the _PR__ parameter as a string so that the macro can be executed later on	Same name as first image, but without the _PR__ string
	Example: [recovery_PR__.idf]	Example: [recovery.idf]

Both images must be in the same container.

PROCEDURE

For the update installation of your devices you only start the update to the first image with **Partition Resize** parameter. After that it's the system's turn:

- The first update process installs the current eLux version without running the macro. The devices now have the first image on board. After installation, the **Update Support Utility** package triggers another update.
- On the second update request, the current eLux version on the devices identifies the **Partition Resize** parameter and runs the macro:
- The *_PR* string will be removed from the update URL.
- An update to the final image will be performed.

> **Note:**
>
> The automatic update to the final image is always performed without formatting the system partition. This is independent of whether the option was set during the first update.

- The **Update Support Utility** package will be uninstalled because it is not included in the final image.

CONFIGURING FIRMWARE UPDATES WITH PARTITION RESIZE PARAMETER

1. In ELIAS, create an image with **Partition Resize** parameter. Follow the specifications from the table above. The image file name must contain the string *PR on any position. Example: recovery__PR_*.idf
2. In ELIAS, in the same container, create a second image which will be used as the final image. Follow the specifications from the table above. The image name must match the first image name, but must contain the string *_PR*. Example: recovery.idf
3. In the Scout Console, for the relevant OU, open **Device configuration > Firmware**. In the **Image file** field, enter the name of the first image (the one with *_PR* parameter).

   The image file specified in the figure above requires the existence of a second image named which is the final image.

Edit the other fields of the **Firmware** tab. For further information, see Configuring firmware updates.

4. For the relevant OU, perform a firmware update.

The devices update to the first image with current eLux version. Then, the **Update Support Utility** package triggers a second automatic update that installs the final image on the devices. This automatic update is retried on each system restart until a successful installation uninstalls the **Update Support Utility** package.

For devices connected via VPN, the automatic update is triggered only after the VPN tunnel is established. The installation is performed without user interaction.

SPELLING OF THE PARTITION RESIZE MACRO STRING

Make sure to use the correct spelling:

Two underscores followed by the string PR (all uppercase) followed by two more underscores.

> **Note:**
>
> You can use the **Partition Resize** parameter in the firmware configuration or in the recovery settings **Options > Recovery setttings**.

Update to new partition layout (2.35 / 2.41 GB to 3.75 GB)

eLux® RP 6 2302 LTSR CU2 contains a modified partition layout that provides more space on the system partition for additional software packages. To upgrade to eLux RP 6 2302 LTSR CU2, please note the following information.

SYSTEM REQUIREMENTS FOR THE NEW PARTITION LAYOUT

- Memory size at least 8 GB
- System partition size 2.35 GB / 2.41 GB

Devices with less than 8 GB memory size can be updated to eLux RP 6 2302 LTSR CU2, but the repartitioning will not be performed.

DOWNGRADE TO THE PREVIOUS PARTITION LAYOUT

Once the repartitioning is complete, it is no longer possible to downgrade to an eLux version with the previous partition layout via firmware update. Please use PXE recovery or USB stick recovery if a downgrade is required.

PERFORM THE FIRMWARE UPDATE

The update to eLux RP 6 2302 LTSR CU2 is done exclusively via a specially configured firmware update, which automatically triggers a second update. The image macro *_P4* and the service package **Custom updater** (part of the eLux RP 6 2302 LTSR CU2 release) are used for this purpose.

The process is started by a firmware update to an image that has the image macro `__P4_` in its name. This `_P4__` image must contain the package and must not be larger than 2.35 GB / 2.41 GB.

After the installation of the *_P4* image on the device a restart is performed and the **Custom updater** starts the second firmware update automatically.

The second update installs the final image, which must have the same name, but without the image macro *_P4* in the name and without the package **Custom updater** in the image. The size of this final image can be up to 3.75 GB.

PROCESS

a. Currently installed image with system partition 2.35 GB / 2.41 GB Example: MyCurrentImage.idf

b. Firmware update to 2 LTSR CU2 image with image macro *P4 in its name and the **Custom updater** package Example: MyNewImage__P4_*.idf / max. 2.35 GB / 2.41 GB

c. Automatic update to the final eLux RP 6 2302 LTSR CU2 image without image macro and without **Custom updater** package Example: MyNewImage.idf / max. 3.75 GB

> **Note:**

> The image with the image macro *P4_* in the name can be kept in the firmware tab of the device configuration until the update to the next eLux image. eLux RP 6 2302 LTSR CU2 ignores the macro and always checks the update against the final image without image macro.

INSTALLING THE UPDATE STEP-BY-STEP

1. Create an IDF for eLux RP 6 2302 LTSR CU2 with the image macro *P4_* in the name containing the package **Custom updater**. Image size <= 2.35 GB / 2.41 GB. Example: MyNewImage__P4__.idf

2. Via ELIAS perform **Save as** for this IDF and remove the image macro *P4_* from the name and remove the package **Custom updater** from the image. Add more packages if needed. Image size <= 3.75 GB. Example: MyNewImage.idf
3. In **device configuration > Firmware** tab enter the name of the image with the image macro *P4_* in its name. Example: MyNewImage__P4__.idf
4. In Scout Console perform a firmware update via **update command** or **update notification**.

   Recommendations:

   - Send the **update command** only to devices with the status "Switched on" (see parameters of the update command) or send a separate command to switch on the units before the update command. Otherwise, the devices could be switched off after the first update.
   - Using **Check for new image version on start / on shutdown** may result in a failed second update on some devices and is therefore not recommended.

CHECK THE RESULTS

In Scout Console > **Properties** after a successful update to the final image UEFI devices should display the following values:

- Partition info | system 3,75 GB | setup 256,00 MB | update 3,75 GB (up to 20,00 GB)
- Image | Image name without image macro `__P4__` in its name Example: MyNewImage.idf

> **Note:**

> Due to a missing boot partition, legacy BIOS devices should display the Partition info | system 4,00 GB | setup 255,00 MB | update 3,75 GB (up to 20,00 GB).

TROUBLESHOOTING

ERROR / PROBLEM	REASON	SOLUTION
Properties in Scout Console	The second automatic update via **Custom updater** was not carried out	Restart the device

ERROR / PROBLEM	REASON	SOLUTION
• Partition info / system 4,07 GB or 4,17 GB / setup 120,00 MB /		
• Image / `MyNewImage_P4__.idf`		
• Image name with image macro `__P4__` in its name		
Properties in Scout Console	A second update was carried out, but not via the **Custom updater**.	Perform a manual update via Scout Console with a format of the system partition before update.
• Partition info / system 4,07 GB or 4,17 GB / setup 120,00 MB		
• Image / `MyNewImage.idf`		
• Image name without image macro `__P4__` in its name		

Firmware security through signatures

You can configure the firmware configuration in the or on the device to have the device check signatures each time before an update is performed. An update is then only performed if the signature of the image definition file (IDF) and/or the signature of the eLux software packages have

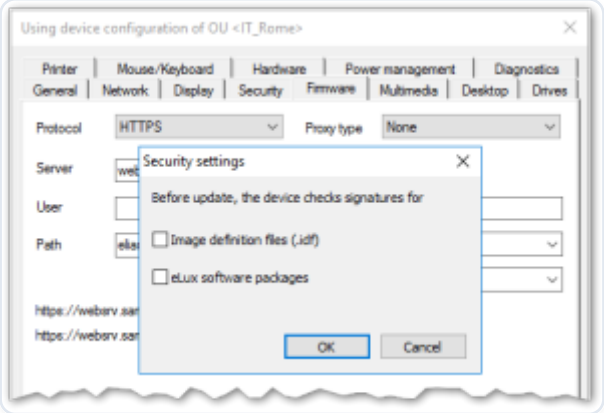
been successfully verified. The update cannot be run, however, if the IDF or one of the eLux software packages to be installed does not have a valid or verifiable signature.

Important:

A signature check of eLux software packages requires an update partition on the device. On devices without an update partition, signatures can only be checked for image definition files but not for eLux software packages. For further information on update partitions, see eLux partitions.

ACTIVATING SIGNATURE CHECK

1. In the , under **Device configuration > Firmware**, click **Security** On the eLux® RP 6 device, select **Configuration panel > Firmware > Check signatures before update**.



2. Under **Signature check before update**, select the **Image definition file** option and/or the **eLux software packages** option.
3. Confirm with **OK** and **Apply**.

Note:

In eLux, both options are provided in the Config panel, under **Firmware**.

The signature verification results are documented in the update log file on the device. After an update has been performed, the update log file is sent to the Scout Server. To view it for the selected device, in the **Properties** window, double-click the **Update status** field.

CERTIFICATES

Verifying the IDF signature on the client side requires the root certificate, but also the signature certificate in the local device directory /setup/cacerts. If you use own certificates for signing IDFs or individually composed eLux packages, configure their transfer to the devices. To do so, use the Scout feature **Files configured for transfer**. For eLux packages provided by Unicon, all required certificates are included in the BaseOS.

> **Note:**

> When updated code signing certificates are made available on our technical portal, download and import them into ELIAS. Instructions are included.

> For further information on how to create image signatures, see Signing an image in the ELIAS 18 guide.

Update deferment by users

-- applies to firmware updates (software) and UEFI updates[1[from eLux® RP 6 2107]] --

This feature allows users to determine the update time as soon as an update is requested. This allows users to avoid updates while using the device.

With the deferment options, users may postpone updates that are initiated by an administrator's **Update** command or by the system-side **Check for new versions on start** configured in the **Firmware** settings.[2[from Scout 15 2104.3000 and Scout 15 2209 with corresponding eLux versions]]

> **Note:**

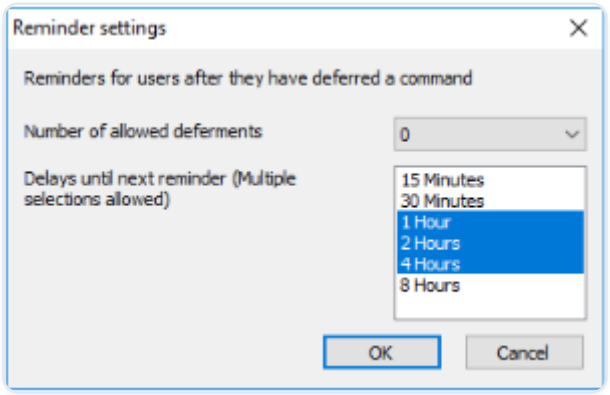
> A device reports the current update process status to the Scout Server. The status can be viewed in the Scout Console in the **Update State/UEFI update state** field of the relevant **Properties** window. In addition, you can use the Report Generator to evaluate the **Update State** field by the value `Deferred (other: [Successful], [Not successful], [Not necessary])`.

CONFIGURING UPDATE DEFERMENT FOR USERS

1. For the relevant devices, open **Device configuration > Firmware > Reminder**.

2. Select the **Number of allowed deferments** from the list.
3. In the **Delays until next reminder** list, click one or more time intervals from which users can choose when they receive the next reminder.

FIGURE: CONFIGURING UPDATE DEFERMENT



Users now are basically allowed to defer updates. When an admin performs an **Update** command with the **Inform user** option, the user will receive a system message including deferment options. The same applies when firmware updates are triggered by the **Check for new version on start** option with user confirmation.[3[from Scout 15 2104.3000 and Scout 15 2209 with corresponding eLux versions]] For further information, see User information before update.

> **Important:**
>
> Update deferment must be configured on the **Firmware** tab. In addition, for each **Update** command administrators run, in the **Command** dialog, they must select the **Inform user** option. For further information, see Performing updates via command.

Security tab

1	Define end-user rights to functions
2	Configure Identity provider and logon procedure for users
3	Configure mirror sessions
4	Configure kiosk mode

Identity Providers

As of Scout version 15 2507, the following Identity Providers can be configured for eLux® 7

IDENTITY PROVIDER	INFORMATION
None	No authentication required and the eLux desktop will show after boot
Active Directory	Provide AD related information
Active Directory + Smart Card	Provide AD and Smart Card related information
Citrix®	Provide Citrix related information
Entra ID	Entra ID authentication for the native AVD client
Evidian	Provide Evidian related information
Imprivata	Provide Imprivata related information

Apart from these authentication methods, which can be configured in the UI, you can also configure `Entra ID` and `OKTA` authentication for eLux 7 using the advanced file entries.

Configuring Entra ID authentication for eLux 7

FILE	SECTION	KEY	VALUE
`/setup/terminal.ini`	`Security`	`AuthType`	`12`
`/setup/terminal.ini`	`OpenId`	`AuthorizeEndpoint`	`<URL>`
`/setup/terminal.ini`	`OpenId`	`RedirectUrl`	`<URL>`
`/setup/terminal.ini`	`OpenId`	`ClientId`	`<ID>`

Configuring OKTA authentication for eLux 7

FILE	SECTION	KEY	VALUE
`/setup/terminal.ini`	`Security`	`AuthType`	`12`
`/setup/terminal.ini`	`OpenId`	`AuthorizeEndpoint`	`<URL>`
`/setup/terminal.ini`	`OpenId`	`RedirectUrl`	`<URL>`
`/setup/terminal.ini`	`OpenId`	`ClientId`	`<ID>`
`/setup/terminal.ini`	`OpenId`	`ClientSecret`	`<Secret>`

User rights

To prevent users from configuring defective or unwanted settings locally on the client, you can disable user rights for individual features.

Functions that you disable via the user rights are not displayed on the device.

User rights can be configured for OUs and for individual , even for individual fields. For example, for security reasons, you might want to disable all tabs, but enable specific options such as some screen settings.

> **Note:**

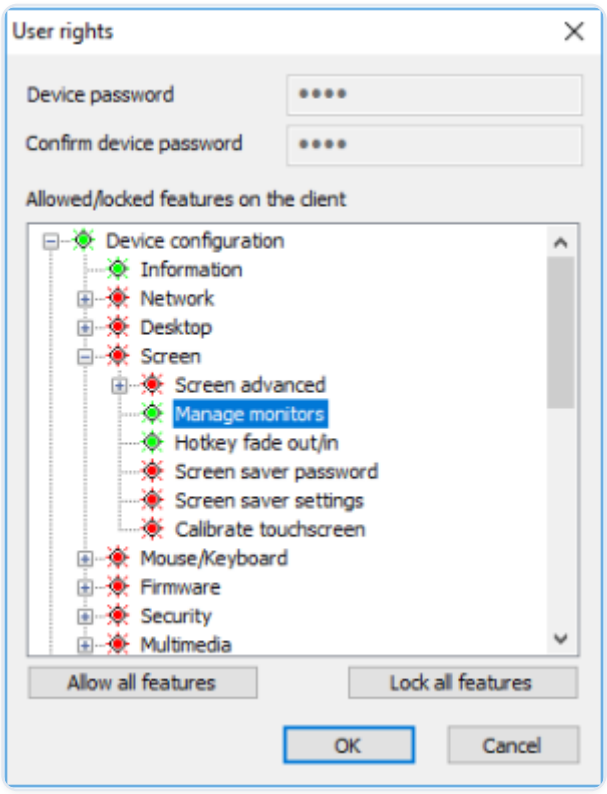
> In addition to configuring user rights, you can hide various elements of the desktop via the device configuration. For example, you can hide the icon for opening the Config panel. For further information, see the **System bar** settings under Advanced desktop settings.

> User rights are available for the following functions:

- Device configuration
- Application definition
- General functions such as **Log off**

MODIFYING USER RIGHTS FOR DEVICE CONFIGURATION

1. On the **Security** tab, under **Local Security**, click **User rights**.



   The **Device configuration** node refers to the devices' device configuration and its structure corresponds to the Configuration panel dialogs of eLux® RP 6.

   In addition, among the user rights under **Device configuration > Security > Scout settings**, you will find options for the fields **Info1, Info2, and Info3**. One user right for all three Info fields. These **Info** fields are shown in the Scout Console in the **Properties** window of a device, and on the eLux RP 6 devices, they are displayed in the **Configuration panel** under **Information**.

2. Expand the **Device configuration** node and navigate to the desired function.
3. To modify the status of a function, double-click it or press the `Space` key.

Allowed functions are displayed in green, locked features are displayed in red.

Modified user rights become active on the next device restart.

MODIFYING USER RIGHTS FOR APPLICATION DEFINITIONS

1. In the **User rights** dialog, expand the **Application definition** node.
2. To allow or block users from creating, editing, or deleting application definitions,change the function's status. To do so, double-click it or press the `Space` key.

   Allowed functions are displayed in green, locked features are displayed in red.

If you lock the **Application definition** node, on the device, the **Applications** tab of the Configuration panel (eLux RP 6) is disabled and users will not be able to view the application definitions.

> **Note:**
>
> If you protect Supporting local configuration and decide to lock the three application functions, we recommend that you also lock the **Application definition** node to ensure that the application definition data are updated correctly.

Modified user rights become active on the next device restart.

LOCAL DEVICE CONFIGURATION

If you allow local device configuration for some features, note that fields configured by users may be overwritten by the values set on the Scout-side when the device configuration is updated. To prevent this, protect the relevant fields or tabs using the provided option. For further information, see Supporting local configuration.

Configuring mirroring

1. On the **Security** tab, under **Mirroring**, select **Enable**.

> **Note:**
>
> After you enable mirroring, the device needs two restarts to be able to start the VNC server.

2. Click **Settings** for configuration:

OPTION	DESCRIPTION
Password	If you define a mirror password, the password will be requested before a mirror session can be started. (optional) The password must have 6 characters minimum and 8 characters maximum.

OPTION	DESCRIPTION
Read-only access	Allows read access only. If not selected, in the mirroring session, the user may still select the option so that the administrator has read-only access.
User must confirm within	Before the mirror session can be started, the user must confirm. Specify how many seconds you want to display the request before the connection is aborted.
Log mirror session	For each mirror session, a log file is created and saved to a sub-directory of the files directory.
Encrypt data transmission	Uses encrypted transmission.
Allow from only	Mirroring is only allowed if the is used.
Log off on disconnection	Automatic logoff as soon as the connection is aborted.

3. Confirm with **OK** and **Apply**.

For further information, see Mirroring.

> **Note:**
>
> The user can cancel a mirror session at any time.

User authentication

User logon and authentication can take place directly on the eLux® device after device start-up or on the back-end server to which the device connects.

eLux supports user authentication with Active Directory. This can be done via username and password or via smart card. Both eLux methods are configured on the tab of the device configuration as described below. Additionally, authentication via Evidian is supported.

Note that for authentication under eLux, the appropriate eLux software packages must be installed on the devices. For smart card check, in addition to the eLux functions, appropriate middleware and hardware drivers for the smart card readers are required.

Configuring user authentication

> **Note:**
>
> The eLux package **User authentication modules** must be installed on the devices. This may require modifications of the image definition file on the web server via ELIAS.

1. In the device configuration, under **Security > User authentication**, choose from the following authentication methods.

None	Disables user authentication
Active Directory	Active Directory (Microsoft directory service)
AD + smart card	Smart card with Active Directory
Evidian	Identity and access management via RFID or smart card

   On the eLux® RP 6 client, under **Security > User authentication**, enable user authentication. Then under **Authentication type**, choose the method.

2. Click **Edit**.
3. On the **AD directory** tab, specify the server, server list or domains to which users may log on. Define multiple entries, if you want to create a selection for users when they log on.

   For Evidian, on the **Evidian server** tab, specify a server or server list.

4. To help users log on quickly, select the **Show last user** option.
5. Only for AD: In the **Domain field** list, choose whether you want to show users the specified domain so they can edit it, or whether you want to hide it.
6. Confirm with **OK**.

After you have enabled user authentication, the users will be prompted for their username and password after the next device restart. The screen saver is automatically protected by password.

> **Note:**
>
> To devices that are not managed by Scout, administrators may log on with the **LocalLogin** username and device password to correct any settings, if required.

ACTIVE DIRECTORY (AD)

Define multiple domains that can be displayed with friendly names. In the logon dialog on the devices, users can then choose between default and alternative domains.

> **Note:**
>
> To enable users to log on to different domains, the following software packages must be installed on the devices: **User authentication modules** and **Security libraries**.

AD directory tab

- Click **Add** to create one or more entries. Then edit the entry (F2 or double-click).

OPTION	DESCRIPTION
Name (optional)	Display name of the domain
Server, server list or domain	IP address or name of the domain controller
	To specify more than one domain/server, separate them by spaces.
	Example: [int.sampletec-01.com dev.sampletec-01.com]
	If the server is not located in the same subnet as the device, enter the fully qualified domain name (FQDN).
	If you define more than one domain, users will be able to choose a domain from a list. The domains are shown with their display name. The first entry of the list is the default domain in the AD logon dialog on the device. You can

OPTION	DESCRIPTION
	specify applications that are shown in only one of the domains.

Note:

We recommend using a Windows time server. If the system time of the domain controller and device differ, Active Directory queries cannot be run successfully.

User variables tab

Based on LDAP attributes, you can define local variables and use them in the device configuration and application definition. For further information, see User variables.

Automated logon tab

By using predefined logon data, terminals can, for example, run in kiosk mode under an AD service account.

Username, password and domain can be set as variables.

ACTIVE DIRECTORY + SMART CARD

To enable users to use smart card readers, install the relevant middleware on the devices. `sc/interface` by Cryptovision is a smart card middleware that integrates smart cards and other smart tokens into IT environments.sc/interface supports more than 90 different smart card profiles.For further information, see the Cryptovision web page.

Note:

For smart card authentication, eLux packages for middleware (such as **Cryptovision sc/interface**) and for the hardware drivers (such as **PCSC Lite**) must be installed on the devices. This may require modifications of the image definition file on the web server via ELIAS.

Smart card tab

OPTION	DESCRIPTION
Behaviour of smart card on removal	If you choose **Lock screen**, in the **Screen saver** settings, **Password protected** will be selected.
Allow logon with username+password	Smart card application allows user/password logon via the **Username & Password** link.
Show Username+password dialog by default	Logon via username + password can be forced despite smart card configuration.
	To use this option, enable **Allow logon with username+password**.

Certificate tab
Certificate-based logon requires verification of the user certificate against the root certificate.

- Select one or more root certificates, and then click **Add**.

The selected certificates are transferred to the device.
User variables tab
Based on LDAP attributes, you can define local variables and use them in the device configuration and application definition.For further information, see User variables.

For the **AD directory** and **Automated logon** tabs, see Active Directory (AD).
Enhanced logging for smart card authentication
When using **PCSC Lite**, you can have an additional log file `/tmp/PCSCDlog.txt` created.To do so, temporarily enable enhanced logging via **Device configuration > Diagnostics > Enhanced logging for smart card support**.After diagnosis, we recommend that you disable the enhanced logging function in order to avoid unnecessary strain on the flash memory capacity of the device.

EVIDIAN

> **Note:**

> The eLux® package **Evidian** must be installed on the devices. For smart card authentication, eLux packages for the middleware (such as **Cryptovision sc/interface**) and for the hardware drivers (such as **PCSC Lite**) must be installed on the devices. This may require modifications of the image definition file on the web server via ELIAS.

> With Evidian access management, you can connect via RFID or smart card.Evidian uses the SOAP network protocol.

- On the **Evidian server** tab, create an entry and edit the following fields:

OPTION	DESCRIPTION
Name (optional)	Display name of the Evidian server
Server or server list	Specify your Evidian server depending on whether you use HTTP or HTTPS in the following format
	`http://\<FQDN or IP address\>:9764/soap`
	`https://\<FQDN or IP address\>:9765/soap`
	To specify more than one server, separate them by spaces.
Use smart card	Enables authentication via smart card
Allow logon with username+password	Users may alternatively log on with username and password.
Secret	Copy the secret from the registry entry of the Enterprise Access Management server
	HKEY_LOCAL_MACHINE\SOFTWARE\Enatel\WiseGuard\FrameWork\Authentication from the key ExternalRoamingSessionSecret. Do not encrypt.

You can configure to show users the system bar during logon.This allows them to access the Configuration Panel and Command Panel.For further information, see Starting Configuration panel from logon dialog.

Additional options for AD users

If Active Directory is used for user authentication, users log on with their AD domain account and password on their device. Pass-through logon is supported by applications which provide access to back-end systems (Citrix®, RDP, VMware).

On the device, in the Configuration panel under **Information**, the user logged on is shown.

CHANGE PASSWORD

To change the password, users choose the eLux® command **Logoff** and then click **Change password**:

PASSWORD-PROTECTED SCREEN SAVER

> **Note:**
>
> When user authentication is enabled, password protection of the screen saver becomes active and cannot be turned off.
>
> The password is set to $ELUXPASSWORD. For further information, see Where to apply user variables.

The screen saver becomes active after the defined time period and the system is locked. By pressing a key or moving the mouse, a dialog is displayed for unlocking. It provides the following options to users:

OPTION	BUTTON	DESCRIPTION
The logged-on user unlocks the screen by entering his/her password / smart card	Unlock	Default
Another person leaves a message for the logged-on user	Message	The screen remains locked. The logged-on user receives a notification with the message when he or she unlocks the screen. This function is enabled by default and can be disabled by the

OPTION	BUTTON	DESCRIPTION
		following Advanced file entry: File: /setup/terminal.ini, section: xscreensaver_dialog, entry: MessageEnabled, value: false
Another user authenticates to log off the previous user (and to log on), restart or shut down the device.	Log off	Useful if devices are used by multiple users: Allows users to reuse devices that have been left without logging off and therefore are blocked. Once the new user has authenticated, the **Restart, Shut down and Log off** buttons become active. In any case, the previously logged-on user is logged off. This function is disabled by default and can be enabled by the following Advanced file entry: File: /setup/terminal.ini, section: xscreensaver_dialog, entry: ShowSysCommandButtons, value: true

Important:

Data loss may occur if the **Log off** option is used followed by a restart, shutdown, or logoff. The user currently logged on is logged off regardless of whether the documents or data last edited have been saved.

SERVICE APP

You can enable AD users to start eLux in service mode. To do so, define a service app that can be started from the logon dialog using the **Service** button. For further information, see Defining a service app.

STARTING CONFIGURATION PANEL FROM LOGON DIALOG

You can enable AD users to access the Configuration panel before they log on. This allows them to connect to a WLAN or VPN or to change the language before logging on.

You can show the system bar for users authenticating via Evidian before they log on. This allows them to access the Configuration panel and Command panel.

- For the relevant devices, under **Security > User rights** on the bottom, enable the user right **Start Config panel from logon dialog**.

AD: On the next restart, the logon dialog shows the button for opening the Configuration panel , so AD users can access the options you allow them to.

Evidian: While the Evidian authentication manager is displayed, users may open the Configuration Panel dialogs and access commands such as **Shut down** and **Restart** in the Command Panel.

> **Note:**

> By default, the user right **Start Config panel from logon dialog** is disabled.

User variables

> **Note:**

> If you want to use user variables, the **User authentication modules** and **Open LDAP** packages must be installed on the devices. This may require modifications of the image definition file on the web server via ELIAS.

> The values of user variables are used by the authentication server for the log-on process. User variables can also be used in some fields of the eLux control panel.

Predefined user variables are

$ELUXUSER

$ELUXDOMAIN

$ELUXPASSWORD

The variables are used when users log on and user authentication is active.

Where to apply user variables

Defining new user variables

WHERE TO APPLY USER VARIABLES

Note:

To use this feature, user authentication via Active Directory is required.

When they are applied, user variables must have a leading $. User variables can be applied in the following fields:

Device configuration

	FIELD	USER VARIABLE
Drives	Username	\$ELUXUSER
	Password	\$ELUXPASSWORD
	Directory, Server, Share	Any \$ELUX variable
	Browser home directory	Any \$ELUX variable
Power management	Enable screen saver	\$ELUXPASSWORD
	(also manual activation)	\$ELUXPASSWORD

Application definition

CATEGORY	FIELD	USER VARIABLE
Citrix®	Server	Any $ELUX® variable
RDP	Username	$ELUXUSER
VMware Horizon	Password	$ELUXPASSWORD
	Domain	$ELUXDOMAIN
Browser	Proxy type, Proxy port	Any $ELUX variable

CATEGORY	FIELD	USER VARIABLE
Tarantella	Server	Any $ELUX variable
Local / Custom application	Parameter for all programs run from the command line	Any $ELUX variable
	Example: eluxrdp / vint.sampletec-01.com.de /u:\ $ELUXUSER /p:\ $ELUXPASSWORD	

DEFINING NEW USER VARIABLES

> **Note:**
>
> To use this feature, user authentication via Active Directory is required.

> You can define your own user variables as local variables based on LDAP attributes. The variable definition has the form Local variable = LDAP variable

1. On the **Security** tab, under **User authentication**, select **Active Directory (AD)** or **Active Directory + Smartcard**.
2. Click **Edit**.
3. Under **User authentication > User variables**, edit the following fields:

OPTION	DESCRIPTION
Local variable	The name of the local variable must begin with the string `ELUX` (but without `$`), which can be followed by any characters. Example: `ELUXFULLNAME` , More than one entry can be transferred if you append a `#` sign to the variable name. Example: `ELUXmemberOf#`
LDAP variable	To be able to use the LDAP variables, the relevant LDAP variable names are assigned to the individual variable as an attribute.

OPTION	DESCRIPTION
	Example 1: `ELUXFULLNAME = displayName` , Example 2: `ELUXmemberOf# = memberOf` , If there are several `memberOf` values within the search base on the authentication server, they are assigned to the local variables `ELUXmemberOf_1`, `ELUXmemberOf_2` and so on.

4. Confirm with **OK** and **Apply**.

> **Note:**
>
> User variables are defined without a leading $, but when they are applied they must begin with $.

multimedia tab

The audio **output** devices are grouped in classes depending on their connector:

USB	USB port
Analog	TRS audio jack (phone connector) or integrated devices
Digital	DisplayPort or HDMI

For each device class, you can control the volume level and **Mute** option separately.

By default, the priority is defined: USB -- Analog -- Digital.

- To change priority, move the list entries by using drag-and-drop operations.

The audio **input** devices are grouped in classes depending on their connector:

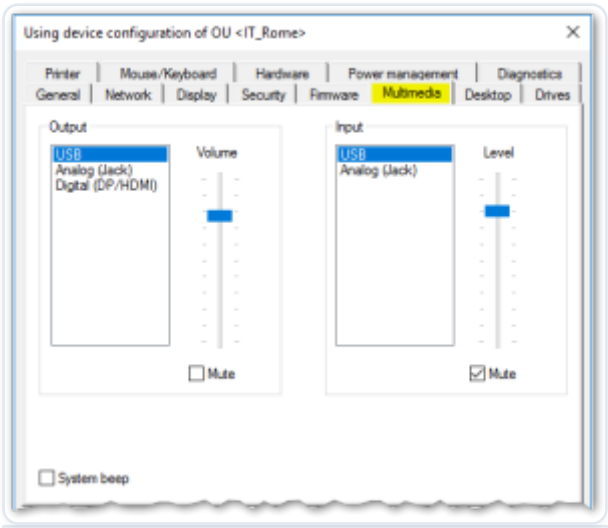
USB	USB port

Analog	TRS audio jack (phone connector) or integrated devices

For each device class, you can control the sensitivity and **Mute** option separately.

By default, the priority is defined: USB -- Analog.

- To change priority, move the list entries by using drag-and-drop operations.



OPTION	DESCRIPTION
Volume (Output)	Slider to control the playback sound level for the selected device class (0 to 100)
Sensitivity (Input)	Slider to control the level of sensitivity for recording for the selected device class (0 to 100)
Mute (Output and Input)	No sound is reproduced / recorded
System beep	Acoustic feedback signal when switching off the device
Priority of device classes	By using drag-and-drop operations, you can change the priority of the device classes for input and output. The top entry has the highest priority.

Drives tab

Define shared network directories on your Windows server as drives that can be accessed by the devices. Any drive defined this way can for example be used as browser home directory.

- Defining a network drive
- Mount points

Defining a network drive

1. In **Device configuration > Drives > SMB Drives**, click **New**.
2. Edit the following fields:

OPTION	DESCRIPTION
Directory	Any name for the directory
Server	Name of the server including the path
Share	Windows share name
Username and password	Windows username and password to access the directory
Domain	Can alternatively be specified in the **User** field: `<DomainUser>` or `<User@Domain>`
AD authentication (only Scout)	The Active Directory logon data are used to access the directory. The fields **Username** and **Password** are disabled.
Test (only eLux®)	Checks if the network share can be accessed with the specified data

> **Note:**
>
> To access network drives with AD authentication, the software package **Network drive share** and the included feature package **Linux Key Management Utilities** must be installed on the devices. This may require modifications of the image definition file on the web server via ELIAS.

3. Click **OK** and **Apply**.

The directory path /smb/ is automatically inserted before the directory name.The data are provided on the local flash drive under `/smb/<Directory name>`.

Example: /smb/share

> **Note:**

> Here, you may apply LDAP user variables.For further information, see Where to apply user variables.

> To make browser settings such as bookmarks permanently available, define a network drive as the browser home directory.For further information, see Browser home directory.

Mount points

Mount points are used to access local resources through an application. The following mount points are provided by eLux®:

Samba	/smb
NFS	/nfs
Internal CD-ROM	/media/cdrom
USB devices	/media/usbdisk

For USB devices, mount points are assigned chronologically: The first device is assigned /media/usbdisk, the second one `media/usbdisk1`, etc.

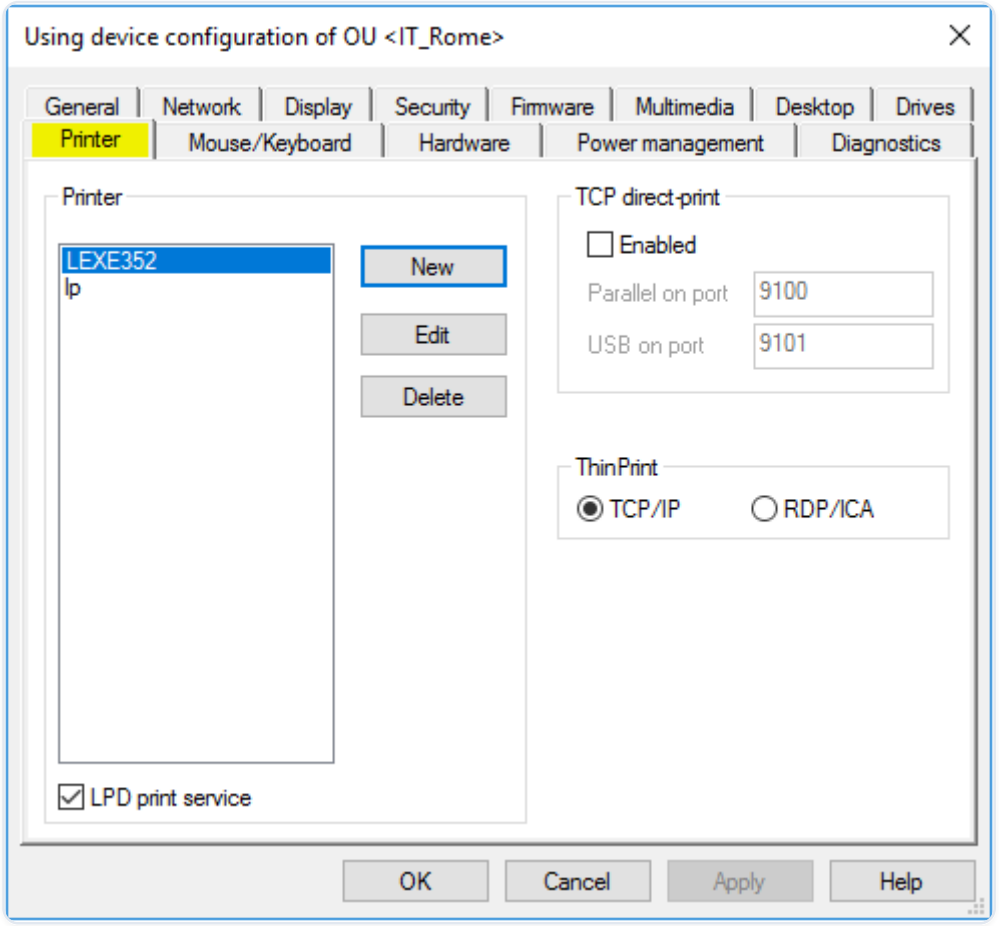
Mounted devices are shown as live information icons. For managed devices, the administrator can suppress the display of live information icons.

> **Note:**

> Due to security reasons, **Allow mass strorage devices** must be selected on the Hardware tab. Drive mapping for access to local resources must be defined in the relevant application definition. For Citrix ICA applications, see Citrix software defaults For RDP applications, see **Advanced application settings / RDP and VMware**

Printer tab

eLux® supports printing from local applications both to locally connected printers and to network printers. In addition, other systems or servers within the network can use a locally installed printer on a device running eLux. Next to the protocols LPR and TCP direct printing, proprietary protocols are also available.



In the Scout Console, in **Device configuration** > **Printer** > **New**, you can define and configure local printers with logical names.

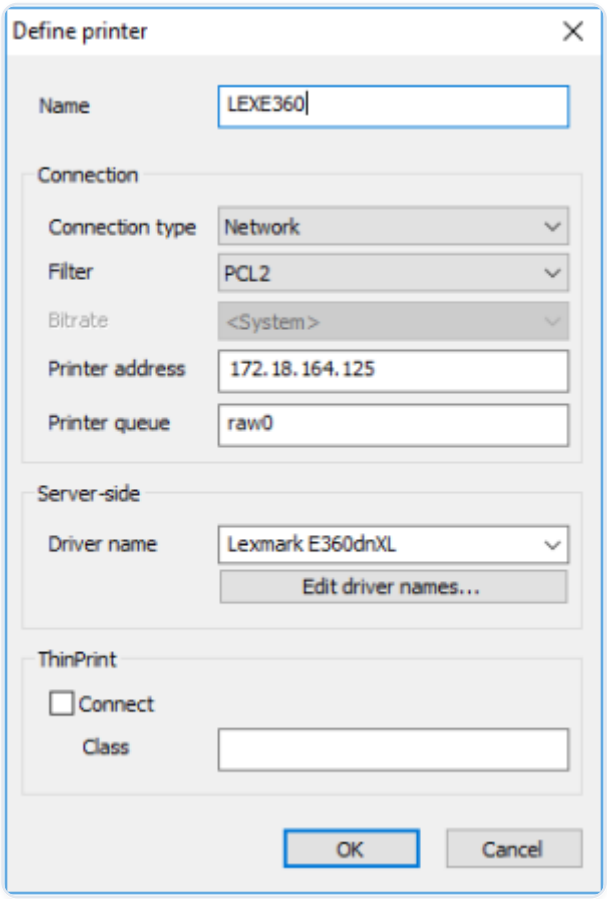
Defining a locally connected printer

1. In the device configuration, on the **Printer** tab, click **New**.
2. In the **Define printer** dialog, type a name for the printer.
3. In the **Connection type** list, choose how the printer is connected to the device.

4. In the **Filter** list, choose whether to use a filter. To print via a Linux Shell, select the text filter. For further information on the filters, see Defining a network printer.
5. Confirm with **Apply** and **OK**.

Defining a network printer

1. In the device configuration, on the **Printer** tab, click **New**.



2. In the **Define printer** dialog, type a name for the network printer.
3. In the **Connection type** list, select **Network**.
4. In the **Filter** list, select one of the following options:

OPTION	DESCRIPTION
None	The printing data from the session are forwarded to the printer in an unfiltered format.
Text	Enables printing from a local shell
PCL2	Enables printing to non-postscript printers in PCL format. If the users do not print from a

OPTION	DESCRIPTION
	Citrix® session, the connected printer must support one of the following languages: **(PCL2), (PS(Postscript)) or (PDF)**.

5. In the **Printer address** field, enter the IP address of the server.
6. In the **Printer queue** field, enter the share name of the printer.
7. In the **Driver name** field, enter the printer's driver name. The driver is used for printing from a Windows session.

> **Important:**
>
> Make sure that the printer driver name is spelled in the same way as the name of the installed driver on the server. The name is case-sensitive and sensitive to white spaces. If the names do not match, the server cannot identify the driver.
>
> For further information, see Citrix auto-created printers.

8. Confirm with **OK** and **Apply**.

For further information, see your printer's manual.

Sharing printers

All printers defined in **Device configuration > Printer** can be shared with other systems via LPD within the network.

1. In **Device configuration > Printer**, select the **Print service activated** option.
2. Activate the Windows LPD service (Line Printer Daemon).

This option ensures that the print service is started at the device. All printers defined in the list can be used to print jobs from network devices.

The printers are controlled by the CUPS server.

Selecting a default printer

1. In the Scout Console, for the relevant OU or device, open **Advanced device configuration > Printer**.
2. In the **Default printer** list, select the printer that you want to be the default printer.

The list provides all defined printers for this element.

CUPS

The CUPS server is installed by default on the devices (**Print Environment (CUPS)** package) and allows printing from local applications and the use of locally attached printers.

The Common UNIX Printing System™ (CUPS™) is a free-of-charge software from Easy Software Products.It provides a common printing interface within local networks and dynamic printer detection and grouping.For further information, see www.cups.org.

The CUPS server can print to serial and parallel ports, USB and the network (LPD).

The CUPS printing system is particularly useful to print from local applications on the device (for example from Adobe Acrobat or a local browser).These local applications have PostScript as output format.If you do not have a PostScript printer, you are required to install a filter such as **PostScript to PCL** on the CUPS server.

CUPS WEB INTERFACE FOR PRINT MANAGEMENT

The eLux package and the included feature package **Web administration service** must be installed on the devices. This may require modifications of the image definition file on the web server via ELIAS.

To manage print jobs, the user can access the CUPS web interface in a local browser with the following URL:

`http://localhost:631`

The web interface can also be used by the administrator to configure the CUPS server.To do this, you must enter the credentials for the local administrator account `[LocalLogin` and device password).

Citrix® auto-created printers

Citrix XenApp® provides automatic configuration of printers (**autocreated printers or dynamic printer mapping**). When the user logs on through a Citrix connection, an automatic printer definition is created on the Citrix server. The printer definition can only be used by the logged-on user and is deleted when the user logs off.

Citrix uses either the specified printer driver or, if not available, the universal Citrix printer driver, which is not tied to any specific device.

Configuring local printer for auto-creating on the device:

1. In **Device configuration > Printer**, specify one or more printers.
2. In the **Define Printer** dialog, in the **Name** box, enter the Microsoft Windows printer's name precisely as listed in the drivers list of the server. The name is case-sensitive.

When the user connects to the Citrix server, the automatically created device printers are shown in the printer settings.

If the specific driver is not installed on the application server or the name is not identical, the printer can not be created and the universal Citrix printer is used.

CITRIX UNIVERSAL™ PRINTING

The universal Citrix printer and various printer settings can be configured on the Citrix server, administrator rights provided.

For further information, see the **Citrix Product Documentation**.

TCP direct print

The print data can be received directly via TCP/IP and sent to the parallel port or USB port to the printer. The data are not modified before printing and there is no spooling of print jobs. TCP/IP handles the flow control.

CONFIGURING TCP DIRECT PRINT

1. In **Setup> Printer**, under **TCP direct print**, select the option **Enabled**.
2. Specify the relevant port number for the communication. The default port numbers are: 9101 for USB printers 9100 for parallel port printers

> **Note:**

> The specified ports are opened on the device.

> To print from a Windows session, for the printer port, choose a standard TCP/IP port. Specify the client IP address and the TCP/IP port selected in the previous step. Select for the protocol in Windows.

ThinPrint

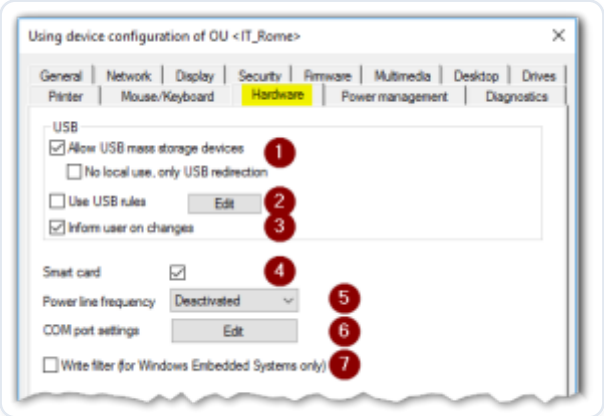
ThinPrint software from ThinPrint GmbH allows optimized network printing across various platforms. ThinPrint is a print protocol that, unlike TCP direct print, LPR or CUPS, allows bandwidth limitation. It is therefore recommended for networks with low bandwidth (WAN).

The software consists of a server component and a client component. The ThinPrint server processes the print data for the target printer and sends them to the device in compressed form. The ThinPrint client receives the print jobs from the server, decompresses and forwards them to the selected printer. ThinPrint server and client are connected via TCP/IP.

CONFIGURING THINPRINT

1. Install the ThinPrint client on the device.
2. Connect a printer.
3. In **Device configuration > Printer > New**, define the printer, and under **ThinPrint**, select the **Connect** option. Optionally, enter a class name of up to 7 characters.
4. If you use Windows CE clients, in **Device configuration > Printer** under **ThinPrint**, select the relevant protocol.
5. Configure the ThinPrint server. For further information, see the ThinPrint documentation on www.thinprint.com.

Hardware tab



1	Allows using connected USB mass storage devices. If the local use of USB devices via mount points is allowed, connected USB devices are shown as live information on the system bar. Select "No local use, only USB redirection" to restrict the use of USB mass storage devices to USB redirection within configured sessions on a backend. There are no mount points then provided locally on the eLux® device.
2	Restricts the use of USB mass storage devices according to defined rules. For example, you

	can restrict to devices with specific VID (Vendor ID) and/or PID (Product ID) such as an individual USB stick model. Moreover, USB rules can be applied to further USB device classes such as smart card readers. To define USB rules, click "Edit".
3	When a USB mass storage device is connected or disconnected, a live information message is displayed.
4	Enables smart card readers on a USB port.
5	Allows changing the refresh rate of webcams to 50 Hz or 60 Hz.
6	Set particular COM port settings such as speed, parity, stop bits.
7	Users are not allowed to store local data on their Windows Embedded device.

Note:

To enable users to use smart card readers, ensure to install the relevant middleware on the devices. `sc/interface` by Cryptovision is smart card middleware that integrates smart cards and other smart tokens into IT environments.sc/interface supports more than 90 different smart card profiles.For further information, see the Cryptovision web page. To use **sc/interface**, the eLux package **Cryptovision sc/interface PKCS11** must be installed on the devices. This may require modifications of the image definition file on the web server via ELIAS.

- Defining rules for using USB devices
- Defining rules for USB redirection
- Safe removal of USB devices
- Enabling Bluetooth audio devices
- Webcams

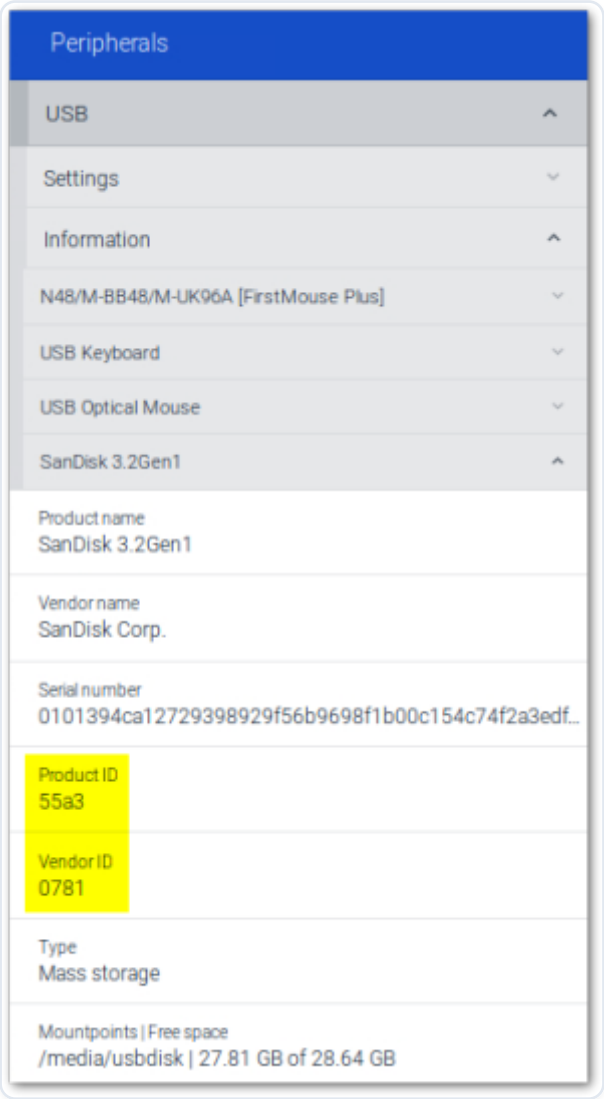
To display hardware information about the device itself or about connected (/en-us/unicon-elux-scout/2603-ltsr-cu1/admin_guides/scout/interface_/interface_windows/interface_win_asset.html).

Defining rules for using USB devices

USB rules allow you to restrict the use of USB mass storage devices to certain models, for example.

1. For the relevant OU or device, open **Device configuration > Hardware > USB > Edit**.
2. In the list-field, select a set of predefined rules as template.
3. Double-click into the relevant line, or select a line and press F2.
4. Modify the rule by using the example rules below.

   The values of the manufacturer ID (VID) and product ID (PID) can be found on the device in the Config panel under (**Peripherals > USB > Information**):



5. Confirm with **OK**.

EXAMPLE RULES

RULE	CODE
Allow a specific USB mass storage device model only	`[ALLOW: VID=0781 PID=5151] # Allow a particular USB model (Example: SanDisk Cruzer Micro)`
Deny all devices of the class MASS STORAGE DEVICES	`[DENY: CLASS=08]`
Deny a specific smart card model only	`[DENY: VID=18a5 PID=0302] # Deny a particular smart card model (Example: Omnikey CardMan 3821)`
Allow all devices of the class SMARTCARD	`[ALLOW: CLASS=0B]`
Deny all printers, mass storage devices, smart card readers	`[DENY: CLASS=07] # Deny all devices of the class [PRINTERS], [DENY: CLASS=08] # Deny all devices of the class [MASS STORAGE DEVICES], [DENY: CLASS=0B] # Deny all devices of the class [SMARTCARD]`
Disable the microphone of a webcam	`[DENY: VID=045e PID=0810 CLASS=01] # Deny audio for the specified USB device`

The syntax of USB rules corresponds to the syntax of Citrix USB policy rules.

> **Important:**

> The USB rules affect all USB device classes including 03 HID (Human Interface Devices). If you deny the 03 HID class, the mouse and keyboard will be deactivated. For specific hardware configurations, you might encounter issues during the boot process of the device. We strongly recommend performing tests before using this option.

Defining rules for USB redirection

For the and 4.1 and later versions, you can define USB filtering rules for USB redirection of connected USB devices from eLux.

1. Type the required USB filtering rules into the appropriate configuration file:

APPLICATION	CONFIGURATION FILE	EXAMPLE RULES
Citrix	`/setup/ica/usb.conf`	`ALLOW: VID=0781 PID=5151`
\^	\^	`DENY: CLASS=08`
VMware	`/setup/elux/.vmware/default-config`	viewusb.ExcludeFamily = "storage"
\^	\^	viewusb.IncludeVidPid = "vid-0781_pid-5151"

2. To transfer the configuration files to the devices, use the Scout feature **Files configured for transfer**. For further information, see Files configured for transfer.

On the next restart of the relevant devices, the USB redirection rules become active.

> **Note:**
>
> The `/setup/ica/usb.conf` file is only required for the Citrix Virtual Apps and Desktops policy **Client USB device redirection rules**. If your environment uses **Client USB device redirection rules (Version 2)**, you do not need to create a `usb.conf` file.

Safe removal of USB devices

Any connected USB mass storage devices should always be removed by using the **Remove safely** feature to ensure that all data are saved.

Users on their device can right-click the USB icon (live information icon) on the system bar and then click **Remove safely**.

DEFINING A KEY COMBINATION FOR SAFE REMOVAL OF ALL USB DEVICES

In the Scout Console, you can define a key combination that allows the users to remove all connected USB mass storage devices safely in one step. The following instructions use the key combination alt+Windows+S as an example:

1. In the Scout Console, for the relevant devices, open **Advanced device configuration > Advanced file entries**.

2.

File	Section
Entry	Value
/setup/terminal.ini	Layout
UsbUnmountHotKey	`<Alt><Mod4><Hyper>s`

For further information, see Advanced file entries.

Enabling Bluetooth audio devices

You can allow the use of Bluetooth audio devices centrally from the console. Users can then search for devices in the eLux® Config panel and connect.

- In the , for the relevant devices, configure the following Advanced file entry:

File	/setup/terminal.ini
Section	Bluetooth
Entry	Enabled
Value	true

For further information, see Advanced file entries in the **Scout** guide.

For further information, see also Connecting Bluetooth audio devices in the **eLux** guide.

Webcams

Webcams on devices are listed under **USB > Information** even if they are built in. To allow users to preview a webcam image, specify an app that can be used to display it.[1[from eLux® RP 6 2107]]The app supports multiple cameras, both built-in and USB-connected cameras.

DEFINING AN APP FOR CAMERA PREVIEW

-- from --

> **Note:**

> In the **eLux Desktop Extensions** package, the feature package **Web camera preview** must be installed on the devices. This may require modifications of the image definition file on the web server via ELIAS.

1. Add a new application and, in the **Application properties**, select the application type [Defining custom applications].
2. Edit the following fields:

OPTION	DESCRIPTION
Name	Name for the application
Application	Custom
Parameter	bash cameraPreview

3. Confirm with **Apply** and **OK**.

When users launch the app, they select one of the connected cameras to get a preview of the camera image.If only one camera is available, the preview window is displayed when the app is started.

> **Note:**

> To disable the microphone of a webcam, define a USB rule.For further information, see Defining rules for using USB devices.

Diagnostics tab

The **Diagnostics** tab allows you to configure the **Device diagnostics**, which can either be requested by the Scout admin or sent from the device.

OPTION	DESCRIPTION
Debug level	If the **Debug level** is set to On (Enhanced debugging), by using the **Device diagnostics** feature you run predefined commands on the device and retrieve a set of configuration and

OPTION	DESCRIPTION
	log files to a greater extent than without enhanced debugging.
	If you require technical support from Unicon™, switch on enhanced debugging before you perform **Device diagnostics**.

Note:

Make sure to switch off the after having performed device diagnostics. Otherwise, you risk exceeding the memory capacity of the device.

Send files to	FQDN of a [Scout Server] or URL of an FTP server to be used as the destination server to which the devices send their diagnostic files. By default, the diagnostic files are transferred to the Scout Server the device is connected to. This option allows you to specify, for example, a specific [Scout Server] or FTP server that will receive the diagnostic files for all devices regardless of their assigned Scout Server. The specified destination server is also active when diagnostic files are sent from eLux.
Enhanced logging for smart card support	Creates an additional log file [/tmp/PCSCDlog.txt] when PCSC Lite is used

Device diagnostics is performed by using a Scout command. For further information, see Device diagnostics.

Power management tab

Especially when using mobile devices, you will need options for power management. By using profiles, on the **Power management** tab, you can pre-define power management settings for your computer. These settings become active when you or the system enable the relevant profile:

- High performance: Favors performance, but may use more energy
- Power saver: Saves energy by reducing computer performance and screen brightness

You can either explicitly activate one of the power management profiles or, for mobile devices, you can let the system choose by selecting the Auto option: If the device is plugged in, the profile **High performance** will be active. If the device is on battery power, the profile **Power saver** is activated.

For both profiles, you can additionally distinguish between working and non-working hours. You can further optimize energy consumption by making more rigorous use of energy saving options outside working hours. To do so, you are required to specify your working hours first. Once you have defined working hours, based on these, the system automatically switches between the **Working hours** profile and the **Default** (Non-working hours) profile. Important: If you choose not to define working hours, the system will use the **Default** profile settings, regardless of the day of the week and time.

Setting a power management profile

- On the **Power management** tab, from the [**Active power management profile** list, select a profile or the Auto option.

The settings defined in the profile become active. The Auto option activates the **High performance** profile if the device is plugged in, and the Power saver profile if the device is on battery power.

Profile-independent options

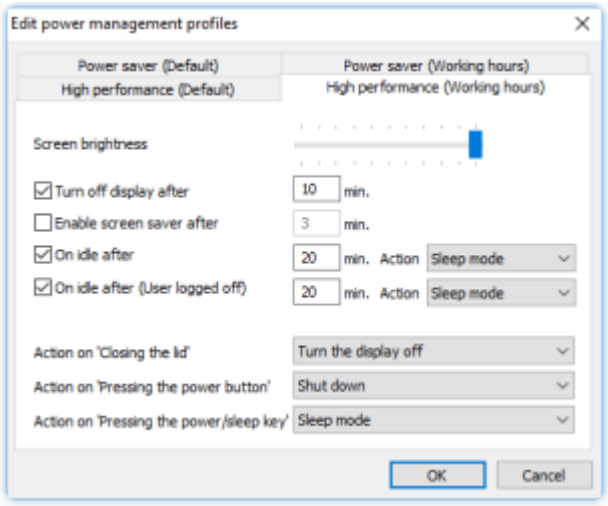
OPTION	DESCRIPTION
Low battery notification	For devices without power supply: Battery status in percent, from which the user is notified
Log off user before going to sleep mode	Users must log on after the computer wakes from sleep mode.

- Configuring power management profiles
- Defining working hours
- Sleep mode (Suspend)

Configuring power management profiles

> **Note:**

If you define your working hours, you can also specify which energy-saving options you want to apply outside of working hours. As long as you do not define working hours, the energy-saving options apply non-stop (24/7).



1. Next to **Power management profiles**, click **Edit** and then select the tab for the required profile. To define profiles for non-working hours, first define the working hours.
2. For each profile you want to use, edit the following fields:

OPTION	DESCRIPTION
Screen brightness	Screen brightness in percent for the selected profile. Move mouse pointer over the slider to display the percentage.
Turn the display off after	Determines whether, after a specified number of minutes, the display is turned off when the user is not using the device (idle state).
Enable screen saver after	Determines whether, after a specified number of minutes, the screen saver is enabled when the user is not using the device (idle state).
On idle after – Action	Determines whether, after a specified number of minutes, the selected action is performed when the user is not using the

OPTION	DESCRIPTION
	device (idle state): - Shut down, - Sleep mode, - Restart, - Log off
On idle – Action (User logged off)	When the user is logged off: Determines whether, after a specified number of minutes, the selected action is performed when the user is not using the device (idle state).
Action on 'Closing the lid'	Action that is performed when the user is closing the lid: - No action, - Turn the display off, - Shut down, - Sleep mode
Action on 'Pressing the power button'	Action that is performed when the user presses the power button: - No action, - Turn the display off, - Shut down, - Sleep mode
Action on 'Pressing the Power/Sleep key'	Action that is performed when the user is pressing the Power/Sleep key on their keyboard (requires a suitable keyboard): - No action, - Shut down, - Sleep mode

3. Confirm with **OK**.

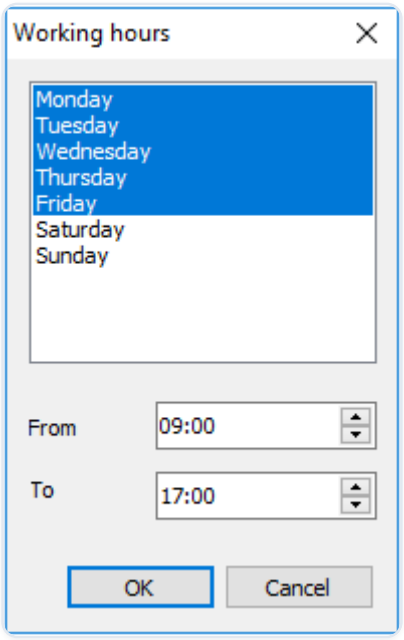
> **Note:**
>
> The sleep mode corresponds to **Suspend to RAM (/en-us/unicon-elux-scout/2603-ltsr-cu1/admin_guides/scout/setup/power_man/suspends3.html).

Defining working hours

REQUIRES
User right **Define working hours**

1. On the **Power management** tab, next to **Working hours** click **Edit**.
2. Select all weekdays that are working days and that are relevant to working times. To select multiple entries, press Shift or Ctrl.
3. Select the earliest time for the start of work (**From**). This time refers to all defined working days.

4. Select the latest time for the end of work (**To**). This time refers to all defined working days.
5. Confirm with **OK**.



Once you have defined working hours, the **High performance** and **Power saver** profiles are divided into sub-profiles for **Working hours** and **Default** (non-working hours). Switching between the two sub-profiles is based on the working times you have defined.

Sleep mode (Suspend)

The sleep mode corresponds to **Suspend to RAM (S3)**.

To put their device into sleep mode, users click a button in the extended Command panel. The sleep mode can also be activated by the system if the power management options are configured accordingly.

When the device goes into sleep mode, the eLux® user remains logged on by default. After reactivating the device, users can continue working right away. This assumes that the backend is configured accordingly: When an application is disconnected, the user must not be logged off automatically.

If you want users to log on and authenticate again after the device wakes from sleep mode, select **Log off user before sleep mode** in the **Device configuration > Power management**. The user is then logged off and when the device wakes from sleep mode is given the eLux logon dialog.

> **Note:**

> In the Scout Console, the **Properties** window shows the status of a suspended device as Switched off (Suspended).

Troubleshooting device configuration

The solutions provided below refer to the Scout Console in the first place.

ERROR / PROBLEM	REASON	SOLUTION
When you use USB multimedia devices such as headsets or webcams, the screen freezes or the window cannot be focused.	The USB operating elements register themselves as keyboard or mouse devices in the system.	Prevent the registration as input devices by defining a `terminal.ini` entry. The basic functionality of the operating elements is not affected. For further information, see Preventing registration of USB multimedia components.
Multimedia USB devices, connected via DisplayPort to eLux RP 5 devices with an AMD processor, do not play back sound.	Sound reproduction via DisplayPort is disabled.	Enable sound reproduction by defining a `terminal.ini` entry. To do so, use the Scout feature Advanced file entries: **File:** `/setup/terminal.ini`, **Section:** `Screen`, **Entry:** `Radeon.Audio`, **Value:** `true`, Alternatively, use a separate audio cable.
Monitor via DisplayPort with AMD GPU: After changing to lower resolution the monitor brings an "Out of range" error message.	The resolution on this monitor interferes with the configured sound reproduction via DisplayPort.	Disable sound reproduction via DisplayPort. This will fix the monitor error. To do so, use the Scout feature Advanced file entries: **File:** `/setup/terminal.ini`, **Section:** `Screen`,**Entry:** `Radeon.Audio`, **Value:** `false`
When you use a touch screen, the location of a	The monitor has not been calibrated precisely enough.	To calibrate the monitor, configure a custom

ERROR / PROBLEM	REASON	SOLUTION
fingertip touch is not recognized precisely.		application by using the parameter `calibrator`. Then start the application.
Only eLux RP 5.7.x: In dual monitor mode, if the second monitor is configured to vertical, the desktop icons are not displayed (correctly).	For some resolutions, the desktop icons on the primary monitor cannot be displayed when the second monitor is vertically aligned and the lower screen area is referenced.	For eLux RP 5.7.3000 and later versions: Use a new parameter to configure the vertical alignment to the upper screen area (top). To do so, use the Scout feature Advanced file entries: **File:** `/setup/terminal.ini` , **Section:** `Screen` , **Entry:** `VerticalAlignment` , **Value:** `top` , The default value is `bottom`.
Display/general graphics issues	The feature package for hardware acceleration `HwVideoAccDrivers` is not installed.	Activate the `HwVideoAccDrivers` FPM within the `XOrg` package in the IDF.
	Hardware acceleration (installed with the `HwVideoAccDrivers` FPM) is not supported by the device and causes problems.	To exclude individual device types from hardware acceleration, create a blacklist that is transferred and locally saved to the devices by using the Scout feature Files configured for transfer: **File:** `/setup/hwaccBlacklist` , In the text file `hwaccBlacklist`, list the relevant device types, one per line. The name of the device type must be identical to the string that is shown in the Scout Console, in the Properties window under `Asset > Hardware information > Type`. ,**Example:** , `FUTRO`

ERROR / PROBLEM	REASON	SOLUTION
		`S920` , `D3314-B1` , `HP t620 Dual Core TC` . For all device types listed in the blacklist, hardware acceleration is disabled.
AD logon to eLux RP 6.x does not work.	Port 389 is configured for the authentication server.	Do not define a particular port for the authentication server.

Note:

After the `terminal.ini` file has been updated on the client, another client restart might be required to enable the new setting.

Preventing registration of USB multimedia components

The registration of USB devices as input devices can be prevented by defining a terminal.ini entry. To do so, use the **Advanced file entries** feature of the Scout Console.

1. In the Scout Console, for the relevant devices, open **Advanced options > Advanced file entries**.
2. Define the following entry

File	/setup/terminal.ini
Section	Xorg
Entry	IgnoreUsbInput
Value	VendorID_1:ProdID_1,VendorID_2:ProdID_2
(VendorID)	Example: 0b0e:034c,047f:c01e
or	Important: Only use lowercase letters for hexadecimal values!
	You can replace individual characters by the wildcard character ?.

	Example: 0b0e:???? filters all products of a specific vendor.
Value	Alternatively, you can filter by vendor name
(VendorName)	Example: Jabra,Netcom
	- White spaces or slashes in the vendor name must be replaced by an underscore _.
	- The vendor name can be entered as a sub-string.
	Example: Netcom finds GN Netcom and GN Netcom A/S.
	- Vendor names can be OR-linked
	Example: Jabra|Sennheiser
	Note: To identify vendor names or IDs, use the following command
	udevadm info \--export-db | grep -Ew \"(NAME|ID_VENDOR)\"

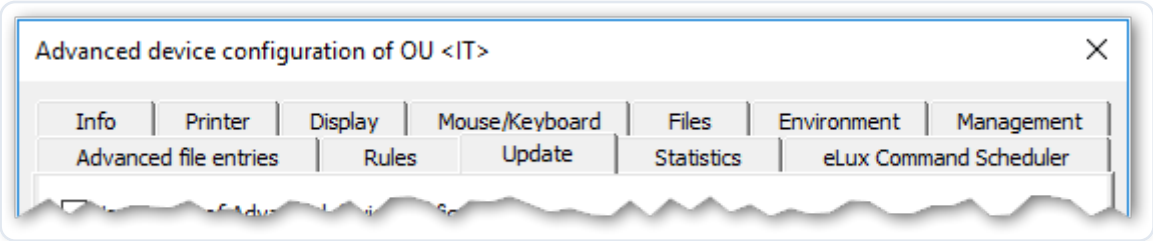
For further information, see Advanced file entries.

> **Note:**
>
> After the terminal.ini file has been updated on the client, another client restart might be required to enable the new setting.

Advanced device configuration and Advanced options

The device configuration defined globally in **Options** > **Base device configuration** or for individual OUs or devices in their **Device configuration** can be extended as follows:

- Override applied options and add further options for individual devices or OUs by using the **Advanced device configuration**

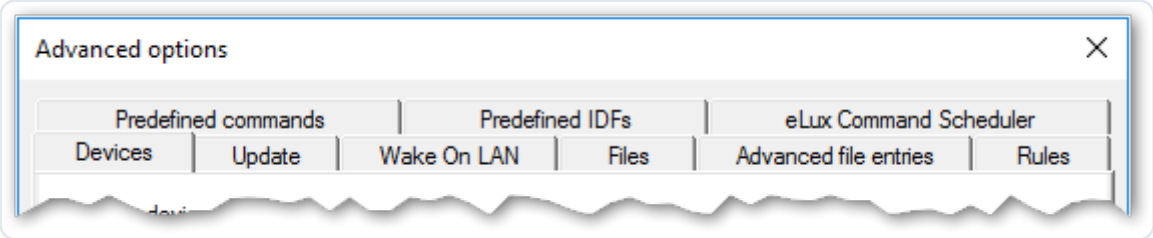


The **Advanced device configuration** is applied on OU or device level.

- Open the relevant context menu and click **Advanced device configuration...**

Inheritance is also used for Advanced device configuration. By default, the option **Use parent of Advanced device configuration** is selected. You can, however, clear this option on some of the tabs.

- Apply further options to all devices by using the **Advanced options**



The **Advanced options** are applied to all devices.

- On the **Scout Console** menu, click **Options** > **Advanced options**.

Some functions such as **Files** (configured for transfer), **Advanced file entries** and **Update** options can be found in both dialogs. Use **Advanced device configuration** to apply the function to individual devices. Use **Advanced options** to apply the function to all devices.

Devices

only globally available for all devices **Options > Advanced options**

Discover devices

OPTION	DESCRIPTION
Maximum ping-time (milliseconds)	Maximum response time in milliseconds by which devices should respond to a `ping` command.

OPTION	DESCRIPTION
Maximum discover time (seconds)	Total time for searching devices for Discovery. After the indicated time has expired, Discovery is stopped.

Field update

OPTION	DESCRIPTION
Retain local configuration (unlocked fields)	Editable fields of the local device configuration are free for individual user configuration and will not be overridden by Scout. During synchronization of configuration data from the Scout Server to the device, only the values of the locked fields will be overridden.
	Which options are allowed to the user for editing (unlocked fields) is determined by the administrator under **Device configuration > Security > User rights**.
	The local user configuration values of unlocked fields can additionally be retained during a factory reset of the device.
	For further information, see Supporting local configuration.

Note:

If users have set defective configuration data, you can, however, override unlocked fields and set a flag for the relevant device in the Scout Console to reload all configuration data.For further information, see Supporting local configuration

New devices

Default OU	OU new devices are assigned to, by default

Assign OU depending on OU filter	Activates the OU filter for new devices
	Click the ... button to configure the OU filter.The OU filter has priority over other methods but can be ignored for individual devices.For further information, see OU filter.
Lock config transfer for new devices	Newly added devices are not synchronized with the server's device configuration
Allow dynamic change of OUs	Allows dynamic assignment of devices via DHCP
Accept only known devices	The Scout Server accepts only devices with registered MAC addresses. For further information, see Reserving device profiles.

Device name

Use the host name of the device	The device name is the client host name and cannot be changed in the Scout Console permanently.
To avoid duplicate names, change name of existing entry	When a new device with an already existing name is added, the name of the existing device instead of the name of the new device is changed.
Name template	Name template for new devices
	Can be overridden for particular OUs (**Advanced device configuration > Management**)
Apply name template only on new devices	Name templates are not applied when you move or relocate devices.

For further information, see Device names.

Update/Delivery

Not available for individual devices

OPTION	DESCRIPTION
Maximum number of parallel updates or software deliveries	Restriction for performance reasons
Maximum time to connect	Time for setting up the connection before the next device is accessed

Note:

The optimum values depend on the individual system.

Management

only available for individual devices and OUs

OPTION	DESCRIPTION
Note	Free text field for internal comments
	Can be shown in the **Properties** window
Visibility	The visibility refers to the list of OUs that new devices request when they initially connect to the [Scout Server] to select an OU.
only for OUs	Hidden OUs are not shown in the list.
	Visible OUs can be selected in the list but can be protected by password.
New devices	When new devices register automatically, they can be subject to a predefined name template.
only for OUs	Device names are configured globally in the **Advanced options** on the **Devices** tab. Please note the parameter **"Apply name**

OPTION	DESCRIPTION
	templates only on new devices". For further information, see Devices.
Ignore OU filter	If the OU filter is active (**Advanced options > Devices**), new devices are assigned to OUs due to the defined criteria.
only for individual devices	Individual devices can be excluded from the OU filter.
	By moving devices within the tree structure by a drag-and-drop operation, for the relevant devices, the option is selected automatically.

Predefined commands

only globally available for all devices (**Options > Advanced options**)

User-defined commands can be centrally predefined and provided as ready-to-use commands for operative administrators managing their devices remotely.

Active predefined commands are shown in **Commands > Predefined command** in the relevant list-field.

In addition, you can set preferences for standard commands used by administrators.

For further information, see Creating predefined commands.

Predefined IDFs and containers

-- only globally available for all devices (**Options > Advanced options**) --

By predefining certain configuration parameters of the **Device Configuration > Firmware**, you specify values to be used by operational administrators. For example, you can restrict the selection of valid image files.

> **Note:**

> To use these functions effectively, configure the relevant object rights in the administrator policies.For further information, see Protecting firmware configuration.

- To create a new list entry, click **Add** and then edit the new entry.The spelling of your entry must correspond to the existing file and path names. Note the following:
- File and path names are case-sensitive.
- Do not use spaces.
- Specify file names with their extensions such as `.idf`

SECTION	OPTION	DESCRIPTION
Predefined IDFs	Image name	Name of an image file (IDF) that if marked as valid can be selected in the firmware configuration under **Image file**
		Example: `recovery_x.idf`
		Used shows the number of devices using the image
		Assigned shows the number of devices to which the image has been assigned
Predefined paths	Path name	Container path for software packages and images, which can be selected in the firmware configuration under **Path** if marked as valid
		Example: `elias/UC_RC6_X64`
Predefined UEFI files	UEFI file	Name of a **.udf** file that if marked as valid can be selected in the firmware configuration under **UEFI file**
		A UEFI file is needed to update a device's UEFI firmware with the appropriate binary data.

SECTION	OPTION	DESCRIPTION
		Example: `uefi.udf`

The **Valid** option allows you to specify for each defined entry whether it is displayed in the **Firmware** dialog.

Wake On LAN

only globally available for all devices (**Options > Advanced options**)

Wake On LAN is a feature supported by Scout that helps you start turned-off devices.

REQUIRES
Wake On LAN is supported by the device and configured in the device's UEFI/BIOS.

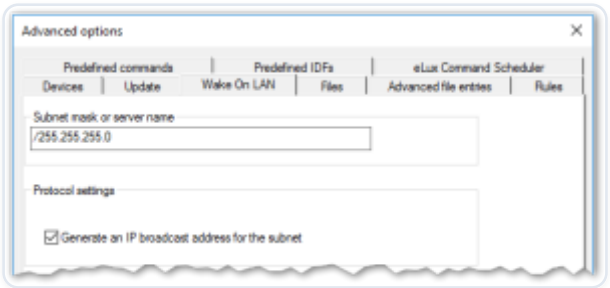
The Scout Server sends a so-called magic packet that is identified by the network adapter of the turned-off devices. The devices are woken via port 67.

The magic packet for Wake On LAN is sent as a broadcast (UDP, eLux port 20000 incoming/ outgoing) within the current network subnet and cannot operate across the entire network. To wake up devices in remote subnets, subnet directed broadcasts are required.

Subnet directed broadcasts

Subnet directed broadcasts can directly address the device's subnet to be woken up via IP. The IP broadcast address of the relevant subnet is determined from the IP address of the device and the configured subnet mask. The magic packet for Wake On LAN is broadcasted (UDP) only within the addressed subnet.

An IP broadcast address for the subnet must be configured once in the Advanced options.



OPTION	DESCRIPTION
Server name	Subnet mask for subnet directed broadcasts (for earlier versions: IP address of Wake On LAN server -- option also available in the Advanced device configuration)
Generate an IP broadcast address for the subnet	The packet is sent to the relevant subnet (dedicated subnet). Requires a subnet address in the Server name field using the format /255.255.255.0 (Note the leading slash).
(only globally available in Advanced options)	Example: To wake up a device with IP address 192.168.10.44, enter /255.255.255.0 in the Server name field. This causes Scout to generate the IP broadcast address 192.168.10.255 for the subnet. By default, this option is disabled.

VPN

only available for individual devices

> **Note:**

> One or more VPN profiles can be defined for entire OUs in **Device configuration > Network** .[1[from Scout Enterprise Management Suite15 2101]]

> Scout supports the following VPN (Virtual Private Network) clients for secure communication:

- Cisco AnyConnect / 2[from eLux RP 6 2302.1000]
- OpenVPN

Depending on the VPN client used, the devices must have a configuration file. You can modify the configuration file by using the Scout feature Advanced file entries.

- Configuring Cisco AnyConnect / Cisco Secure Client
- Configuring OpenVPN

Configuring Cisco AnyConnect / Cisco Secure Client

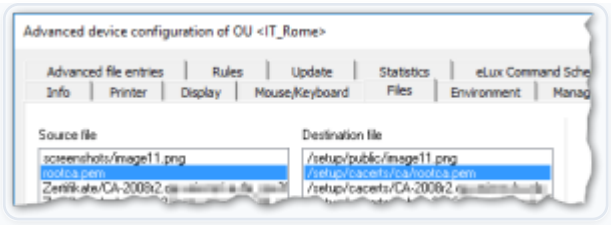
Cisco renamed its VPN **AnyConnect** Client from version 5 to **Cisco Secure Client**. The VPN technology remains the same, and the protocol is still called AnyConnect. Unicon ships the Cisco Secure Client for the first time with eLux RP 6 2302.1000, in parallel with the well-known AnyConnect Client. Their configuration for eLux is the same for both packages.

The following instructions refer to individual devices. To define VPN profiles for entire OUs, use **Device configuration > Network > VPN**. For further information, see Defining a VPN profile.

> **Note:**
>
> The eLux® package **Cisco AnyConnect** or **Cisco Secure Client** must be installed on the devices. This may require modifications of the image definition file on the web server via ELIAS.

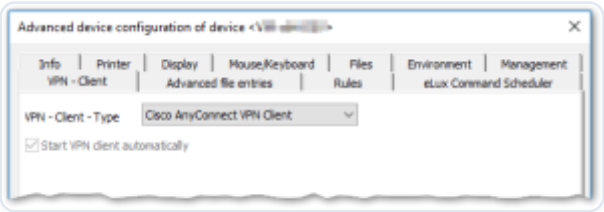
1. Transfer the root certificate to the devices to /setup/cacerts. If you use the Scout feature Files configured for transfer

   > **Note:**
   >
   > Use the `.pem` format for the certificate.

   > **Note:**
   >
   > The certificates that are transferred from the VPN server are stored in /setup/cacerts/client.

2. In the Scout Console, for the relevant device, open **Advanced device configuration > VPN client**. Then, in the list-field, select `Cisco AnyConnect VPN Client`. Use this option also for the Cisco Secure Client.



3. Restart the device.The device might require one more restart to activate the VPN configuration data locally.

The device connects via AnyConnect on the next restart. Users have the active VPN connection as a live information on their system bar.

CONFIGURATION FILE

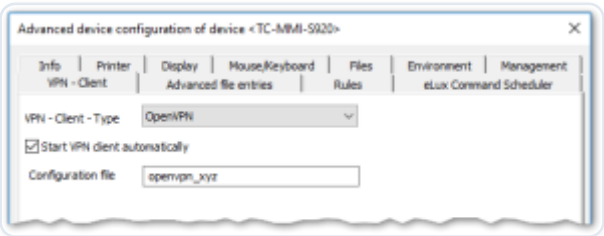
As an option, you can create an AnyConnect or Cisco Secure Client configuration file or copy one from a reference device, and then transfer the file to `/setup/elux/.cisco/profile/` via the Scout feature Files configured for transfer. In the configuration file, you can specify your back-end server address.

Configuring OpenVPN

The following instructions refer to individual devices. To define VPN profiles for entire OUs, use **Device configuration > Network > VPN**. For further information, see Defining a VPN profile.

> **Note:**
>
> **OpenVPN** is an integral part of the eLux® operating system. There is no need to have a separate package installed on the devices.

1. Transfer the .ovpn configuration file and certificates to the devices to /setup/openvpn. If you use the Scout feature **Files configured for transfer**, specify the source and destination file **with** file name extension and the destination path /setup/openvpn. If you use a USB stick, unzip the .zip file to the client directory /setup/openvpn.
2. In the Scout Console, for the relevant device, open **Advanced device configuration > VPN client**.Then, in the list-field, select .



3. Select the option **Start VPN client automatically**.

4. In the **Configuration** field, enter the name of the OpenVPN configuration file **without** the file name extension `.ovpn`.

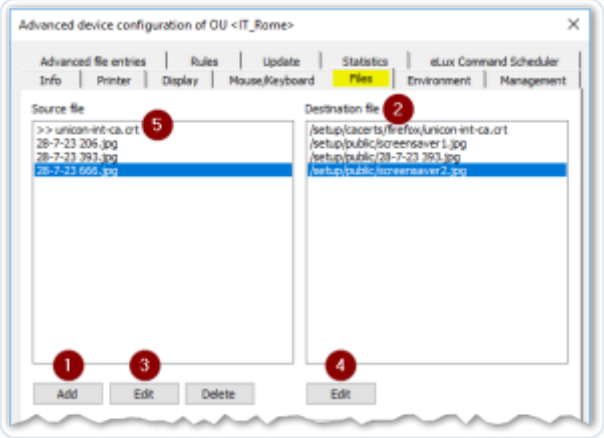
On the next restart of the device, the VPN configuration file is transferred to the device and activated with another restart.The OpenVPN logon dialog is displayed and the user can connect.

> **Note:**
>
> The OpenVPN client can only be used with a valid .ovpn configuration file under eLux. Ensure that the configuration entries are correct.eLux does not accept the setting of an additional default route via the .ovpn configuration file. Some firewall vendors offer different configuration files for different operating systems, with and without a default route. For example, the Sophos configuration file for Android/iOS does not include this default route and can be used for eLux.

Files configured for transfer

This feature helps you transfer files to the devices. The defined files are transferred on the next device restart. You can apply a file transfer to all devices, to individual devices or to OUs.

The source files are imported to the Scout database and therefore are included in an SQL database backup.



Example: Picture files are copied to the devices as screen savers.

1	The source files are selected from the file system via **Add**.

2	For each source file, a destination file on the device is created
	Specify the relevant destination directory and optionally a different file name.
3	Source files of the current level can be edited.
4	For entries of the current level, you can edit the destination file properties.
5	Entries from a parent OU or the global file list are identified by a **>>**.
	You cannot edit these source files. With the corresponding object right, however, you can view them,[1[[ ] from Scout 15 2101]] see below.

Defining files for transfer

1. To configure a file transfer to all devices (global file list), click **Options > Advanced Options** On top level, you can also define destination file templates for subordinate levels.

   To configure a file transfer to the devices of an individual OU or to an individual device (individual file list), on the context menu of the relevant OU or device, click **Advanced device configuration**

   > **Note:**
   >
   > Individual file lists have precedence over global file lists.

2. On the **Files** tab, click **Add**.
3. In the **Add file entry** dialog, to select the source file from the file system, click the … button and then select a file.

   A new entry for the **Source file** and **Destination file** lists is created.
4. Under **Destination file**, if required, modify the directory and file name of the destination file on the device. The destination file name may differ from the one of the source file.

If configured, alternatively use a destination file template from the **Template** list field. The path and name of a destination file from a template cannot be changed later on.

5. Confirm with **OK**

Source and destination are defined. The files are transferred on the next device restart. The files will not be reloaded unless you modify the file configuration.

Re-using imported source files

You can re-use source files that you have imported for one OU for the file transfer in other OUs.

1. In the Advanced device configuration of the source OU, on the **Files** tab, right-click the relevant entry in the **Source file** list.
2. To copy the entry to the Clipboard, from the context menu, choose **Copy**.
3. Open the Advanced device configuration of the target OU.
4. On the **Files** tab, right-click the **Source file** list. On the context menu, click **Paste**.

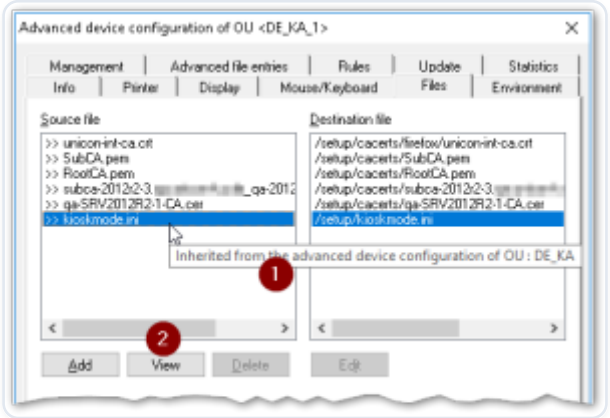
Removing transferred files

- To remove a transferred file from a device, on the **Files** tab, on the relevant level, select the file and click .[2[from eLux® RP 6 2209]]

> **Note:**
>
> To remove all transferred files on all levels, alternatively perform a factory reset. Also use a factory reset for earlier eLux versions.

Source files from parent OUs or the top level

The **Files** dialog also displays entries from parent OUs or the global file list. These inherited source files are identified by a **>>** character and are protected from access by default.



1	To show the origin of a parent entry, move the mouse pointer over the entry.
2	You cannot edit source files of parent entries.
	But, if the file privacy object right has been disabled, you can view the content of a source file. [from Scout 15 2101]
	To disable the object right, for the required OU and for the required administrator, open the object rights, and then navigate to **Edit properties > Files > Inherited file privacy**.
	For further information, see Changing object rights.

Advanced file entries

The **Advanced file entries** tab allows you to set parameters in `.ini` files that cannot be set by using the graphical user interface. For example, you can define additional layout parameters.

Advanced ini

1	Entries from a parent OU or the global **Advanced options** are shown with a **>>**. To show the origin, move the mouse pointer over the relevant entry.
2	For a new entry, first select the relevant file or enter the path and filename.
3	For the new entry, then specify the section, key and value.
4	Click **Add** to add the new entry.
5	Click **Replace** to replace the selected entry in the list by the new entry.

The following applies to the `.ini` files:

- `.ini` files contain at least one section. Each section contains zero or more keywords. The keywords contain zero or more values.
- Each section is headed by a symbolic name enclosed in square brackets.
- Each keyword and its value are in one line. Keyword and value are separated by an equal sign (=). A keyword can have more than one value.
- If a keyword is used more than once in the same section, the last occurrence has precedence.
- Adding individual file entries
- Changing values of individual file entries
- Deleting individual file entries
- Deleting complete sections

Adding individual file entries

1. In the Scout Console, click **Options > Advanced Options**. Or: For the relevant OU, open the context menu and click **Advanced device configuration**
2. Click the **Advanced file entries** tab.

3. Below the list, edit the following fields (1):

OPTION	DESCRIPTION
File	Enter the full path including file name or select from the list:
	Terminal: `/setup/terminal.ini`
	Citrix ICA: `/setup/ica/wfclient.ini` and `/setup/ica/appsrv.ini`
	Cisco VPN: `/setup/ciscovpn/sample.pcf`
Section	Section heading without brackets
Key	Keyword
Value	Value you want to assign to the keyword

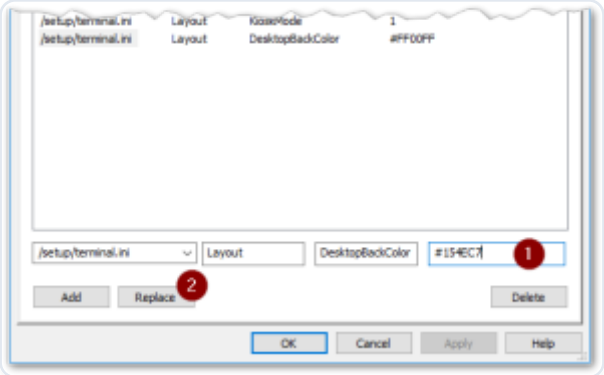
OPTION	DESCRIPTION
	Blanks, hyphens and multiple values are allowed.
	Example: `valueA,valueB,valueC;comment`

4. Click **Add** (2).
5. Confirm with **Apply** and **OK**.

The new entry is written to the .ini file on the next device restart.

Changing values of individual file entries

1. In the Advanced device configuration > Advanced file entries, select the entry whose value you want to change.



2. Below, in the **Value** box, replace the current value.
3. Click **Replace**.

The new values are written to the .ini file on the next device restart.

Deleting individual file entries

1. In **Advanced device configuration > Advanced file entries**, define a new entry:
2. Enter **File**, **Section** and **Key** of the relevant file entry, but leave the **Value** box empty (1).

3. Click **Add**. (2)

The 'empty' file entry overrides previous instructions. The file entry is deleted from the relevant section on the next device restart.

> **Note:**
>
> If you use the Delete button to delete a selected row from the list, then Scout will no longer update the respective entry. However, the entry with its value remains.

Deleting complete sections

1. In the **Advanced device configuration > Advanced file entries**, define a new entry: Specify the **File** and **Section** of the relevant file entry, but leave the **Key** and **Value** fields empty.
2. Click **Add**.

The 'empty' section overrides previous instructions. The section is deleted from the file on the next device restart even if it contains file entries.

Rules

Using this feature helps you define rules which can be executed when closing the last application or using Scout for the first time.

OPTION	DESCRIPTION
After terminating the last application execute the following action	Choose between the options of the list-field
	For OUs and devices, the Use parent action option is set by default to enable the rules defined for a higher OU level.
Display a message on the device for `<x>` seconds	Enter a time period in seconds to inform the user
After first management contact execute the following action	Select Update the device to ensure that the new devices are all up-to-date.

Special case kiosk mode

If you use the browser in kiosk mode to access Citrix applications, the option **After terminating the last application execute the following action** does not work. In this case, you can set an entry in the `terminal.ini` file to determine an action that is performed after the last Citrix application is closed.

- by using the Scout feature Adding individual file entries:

File	/setup/terminal.ini
Section	Global
Entry	ActionAfterLastWfica
Value	
	0 - Use parent action
	1 - Restart
	2 - Shutdown
	4 - Logoff
	8 - Lock
	16 - VPN disconnected

Environment variables

only available for individual devices and OUs

Environment variables can be used locally on the device. They contain strings.

Defining environment variables

1. In the **Advanced device configuration**, click the **Environment** tab.

   Previously defined variables are shown in the list. Entries from a parent OU are shown with a **>>**.

2. Click **New**.
3. Enter the required variable using the format: `Variable name=value` and confirm with **OK**.

   The new variable is shown in the list.

4. To encrypt the value of the variable, right-click the variable. Then from the context menu, choose **Encrypt value**.

> **Note:**

> When you apply the defined variables later on, make sure you prefix the variable with a dollar sign: <Variable name\>. For further information, see also Where to apply user variables.

TPM 2.0 support

A TPM 2.0 chip built into a device can be used for basic security functions:

- Encryption of the setup partition and system partition

The setup partition on a device's flash memory contains the device configuration, application definitions, and certificate store. The system partition holds the software packages of the firmware.

In order to protect the system from manipulation, in addition to encryption, the disk is sealed with security measurements.

- Store the private key of a SCEP client certificate inside the TPM 2.0 module

To store the key inside the TPM 2.0 module, a `scep.ini` entry is required. For further information, see Certificates for SCEP in the `SCEP` guide.

> **Note:**

> If you want to use TPM 2.0 via WLAN, note the special parameters in the configuration file `wpa.conf`. For further information, see Configuring WPA supplicant in the **IEEE 802.1X** short guide.

Requirements for disk encryption

- The devices are provided with a TPM 2.0 module.
- The devices are started in UEFI mode.

Disk encryption via TPM 2.0

If the device-side requirements are met, encryption can be enabled using two different mechanisms:

- Via the configuration parameter **DiskEncryption**
- Via the feature package **Partition encryption** installed with the image

If you install the BaseOS package on the **devices** with the feature package **Partition encryption** enabled, the system will automatically be encrypted. The parameter **DiskEncryption** is then ignored.

The feature package **Partition encryption** is enabled by default.

To encrypt the disk, the partitions must first be formatted. Therefore - as soon as the encryption is activated - a firmware update with previous formatting for the relevant devices is forced.

Encrypting the disk via parameter

1. In the Scout Console, for the relevant devices, open **Advanced device configuration > Advanced file entries**.
2. Define the following entry:

File	`/setup/terminal.ini`
Section	`Security`
Entry	`DiskEncryption`
Value	`true` : Default Value `false`

   For further information, see Advanced file entries.

For the relevant devices, a firmware update is forced with previous disk formatting.

The configuration parameter has no effect on devices without TPM 2.0.

> **Note:**

> You can find information on whether the disk of the device is encrypted in the **Properties** window.

> When new devices with TPM 2.0 chip are on-boarded to the Scout infrastructure and the destination OU is configured with DiskEncryption, it is ensured that the configuration data stored in the is only saved locally on the device after the setup partition has been encrypted.

Update from earlier versions with disk encryption

Updates with disk encryption can only be performed from eLux RP 6.x. Upgrades from eLux RP 5 are not supported.

If you enable encryption when updating to a current eLux® RP 6 version, one more update may be required on the next device restart. This is due to the partition formatting that is required for encryption.

Error handling

If a device fulfills the above-mentioned requirements for encryption and disk encryption still fails during the update, the setup partition will be partially cleaned like it is for a factory reset, without deleting the Scout Server address. The device status in the Scout Console is then displayed with a yellow icon (initialization).

Resetting the disk encryption

REQUIRES
The feature package [Partition encryption] must be uninstalled on the relevant devices. This requires modifying the image definition file on the web server via ELIAS.

- Set the advanced file entry `DiskEncryption` to the value `false`.

or

- Perform a factory reset for the devices. To do so, use the **Remote factory reset** command with the option **Delete Scout Server address on the device**.

During the restart of the relevant devices, the disk is decrypted. That is why the start up process takes longer.

Defining applications

The devices can be supplied with the following types of applications:

- Applications providing access to back-end systems
- Local applications

The definition of applications and the installation of the related software are independent of each other. Defining applications means to configure the applications provided for the users. Additionally, to enable the users to operate the applications, the relevant software packages must be installed

on the device via IDF configuration. For further information, see Creating an image in the **ELIAS 18** guide.

> **Note:**
>
> The term **applications** refers to application definitions. The term **software** refers to the required software packages.

> Applications can be inherited from the top of the organization structure to subordinate OUs. The lowest level to define an application is an OU, the highest level is the root level.

- General
- Connecting to a Citrix farm
- RDP
- Virtual Desktop
- Browser
- Local and user-defined applications
- Emulation
- Applications in kiosk mode
- Local web sites
- Troubleshooting application definition
- Third party software

General

- Adding applications
- Editing application properties
- Defining free application parameters
- Parent applications
- Software defaults
- Defining application templates
- Uploading applications from a device to Scout
- Defining application icons

- Assigning custom application icons
- Defining a service app
- Limiting applications to one logon domain

> **Note:**

> You can define actions to be performed after the last application has been closed. For further information, see Rules.

> Additionally note that application definitions can be

- copied from one OU to another
- exported from one OU and imported to another OU (context menu > **Edit**).

Adding applications

1. In the tree view, right-click the **Applications** icon of the relevant OU.
2. On the context menu, click **Add**.

The **Application Properties** dialog opens. This dialog provides several tabs, each of them relating to a particular application type.

The following options of the **Application Properties** are available for most application types:

OPTION	DESCRIPTION
Name	Name of the application shown in the [Scout Console]
	Important: Applications are identified by their name. Make sure to use a unique name for them.
Display name (optional)	Name of the application shown on the device (control panel, start menu)
Sorting ID	Specifies the sorting order for applications pinned to the system bar
	[1] sorts alphabetically (default)

OPTION	DESCRIPTION
Server	Name of the server the application connects to
Login	The user is automatically logged on to the terminal server by using predefined credentials (username, password, domain).
Pass-through logon	The values of the local user variables `$ELUXUSER`, `$ELUXPASSWORD` and `$ELUXDOMAIN` are used to log on to the authentication server. This allows to use the AD logon data of the eLux® desktop for automatic logon to the configured applications (single sign-on).
	For further information, see User variables.
Application restart	The application is immediately restarted after it has been closed either unexpectedly or by the user.
Start automatically after	The application starts automatically after eLux has been started. Optionally, you can delay the auto-start process by defining the required number of seconds.
Desktop icon	Provides an additional desktop shortcut for the application (icon and display name)
	(except for PN Agent)
Free Parameters	Individual parameters for program start

Editing application properties

- Open the context menu of the relevant application and click **Properties**.

The **Application Properties** dialog for the application opens. Depending on the application, different properties can be configured.

> **Note:**

> Properties of the selected application can be displayed in the **Properties** windows of the Scout Console. They cannot be modified there.

> For each administrator, you can control the object rights for individual application types. Object rights for the advanced settings and free parameters can be assigned separately.

Defining free application parameters

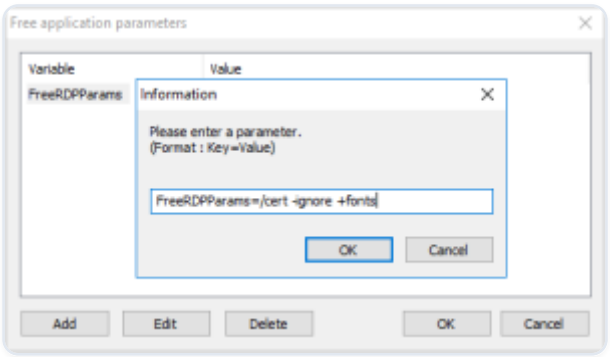
Free application parameters are individual parameters which can be used to start an application. You can define free application parameters for all applications, except for SAP-GUI and Emulation.

1. Open the **Application properties** of the relevant application.
2. Click **Free Parameters**.
3. Click **Add** and enter one or more parameters in the following format. Example:

   `FreeRDPParams=<Parameter> <Parameter> <Parameter>`

   Separate multiple parameters by spaces.
4. Confirm twice with **OK**.

The defined parameters are saved with the application definition. They will be inserted for the relevant application into the file `\\setup\\sessions.ini`.



> **Note:**

> Access to the free parameters can be restricted via the object rights.

> For information on which parameters are available, refer to the description of the respective application definition.

The following parameters can be used across applications:

PARAMETER	VALUES	DESCRIPTION
pinned	true	The application is pinned on the system bar.

Parent applications

By default, applications are inherited to subordinate OUs. This allows you to define applications in only few but central places.

For the subordinate OUs, in the tree view, on the **Applications** context menu, the option **Use parent applications > All** is enabled (check mark). With the check mark set, all applications are active that have been defined for higher-level OUs or for the top-level OU. In addition to these applications, you can define more applications valid for this individual OU (and subordinate OUs).

DISABLING INHERITANCE OF APPLICATIONS

1. For an OU that you do not want to receive higher-level applications, open the context menu.
2. Click **Use parent applications > All** to remove the check mark.

The OU cannot use higher-level applications and cannot inherit them to subordinate OUs any longer. Only applications defined within that OU are active.

INHERITING ONLY INDIVIDUAL APPLICATIONS

1. For the OU that you want to receive some of the applications defined for a higher-level OU or at top-level, open the context menu.
2. Make sure that the option **Use parent applications > All** is cleared (no check mark).
3. On the sub-menu **Use parent applications**, under **All**, select the application you want to inherit from above.

The selected application, on the sub-menu, receives a check mark, and its definition is provided on the next device restart for that OU.

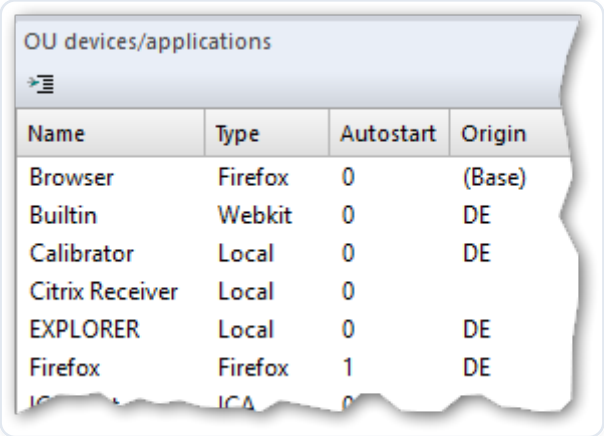
> **Note:**

> Inherited properties of the selected applications will be lost if you move the respective OU to another parent OU. For moved OUs, the system automatically enables **Use parent applications > All**.[1[from Scout 15 2101]]

SHOW DEFINED APPLICATIONS FOR AN OU

1. Click **View > Window > OU devices/applications** to display the relevant window.
2. In the tree view, click the **Applications** icon below an OU.

For the selected OU, all defined applications are listed. The **Origin** column shows the OU from which an application is inherited. Top-level applications show the value Base.



The selected OU of the figure above has one own application (no entry in the **Origin** column), four applications from the higher-level OU **DE**, and one top-level application.

> **Note:**

> To also apply the default settings of the parent applications, on the **Applications** context menu, select the **Use parent defaults** option. For further information, see **Software defaults**(app_defaults.htm).

Software defaults

Software default settings for all applications of the same type can be defined centrally or on OU-level. Software default settings are available for

- Citrix applications (Citrix Workspace-App) For further information, see Citrix software defaults.
- Browsers For further information, see Browser home directory

We recommend to apply the default settings at top level (root applications) to use inheritance across all OUs. If you still need different software default settings, you need to disable inheritance for the relevant OUs.

DEFINING SOFTWARE DEFAULTS

1. In the tree view, for the relevant OU, open the **Applications** context menu and click **Software defaults**

   > **Note:**
   >
   > Citrix Virtual AppsIf inheritance is enabled, you can only open and modify the **Software defaults** of the top-level instance or parent instance. By default, this is the top-level instance.

2. In the list-field, select the relevant software and click **Edit**.
3. Edit the relevant options on the tabs and confirm with **OK**.

The software defaults are applied to all applications of subordinate OUs, provided, on the OU's **Applications** context menu, the **Use parent defaults** option is selected. This is the default.

DISABLING INHERITANCE OF SOFTWARE DEFAULTS

1. In the tree view, for the relevant OU, open the **Applications** context menu.
2. Disable the **Use parent defaults** option.

The parent software defaults are saved for the OU.[1[ab Scout 15 2204]] On this basis, from now on, you may configure the settings individually.

Defining application templates

As an administrator, you can create a template for each application type, for example, a browser template: In the template, you specify values that users will find filled out when they create a browser application. If you create a browser template and specify the browser type **Firefox** and **System proxy** as proxy, as a result users at all levels will see Firefox and system proxy as predefined values when they create a new browser application.

Note that the values can be overwritten. However, if you disable the object rights for the advanced options or free parameters of an application type, these values remain because the relevant users do not have access to them in the application definition.

Application templates are defined at the top level. You can define exactly one template for each application type.

CREATING AN APPLICATION TEMPLATE

REQUIRES
Administrators' base right: **Edit base application**

1. In the tree view, on the top level, right-click the **Applications** icon.
2. On the context menu, click **Define application templates**

   The **Definition of application templates** dialog opens. Each tab contains the application properties of an application type.
3. Switch to the tab that contains the required application type.
4. Edit any options you want to predefine.
5. Confirm with **Apply** and **OK**.

For each new application definition of this application type, the predefined values will be set.

MODIFYING OR RESETTING APPLICATION TEMPLATES

1. In the tree view, on the top level, right-click the **Applications** icon and select **Define application templates**

   The **Definition of application templates** dialog contains for each application type a tab with its application properties. Non-empty fields contain values that are used when new applications of this type are created.
2. For the relevant application type, edit the options you want to change.
3. To reset all values that you have predefined for an application type, click **Reset**.

Uploading applications from a device to Scout

Application definitions of a reference device with an up-to-date eLux® version can be uploaded to the Scout Console and assigned to any OU.

> **Important:**
>
> If you upload applications to an OU, all existing applications in this OU will be deleted.

UPLOADING FROM ANY DEVICE (OUTSIDE OF SCOUT ENTERPRISE MANAGEMENT SUITE®)

1. In the Scout Console, click **File > Application upload**.

The **Application upload** dialog opens.

1. Enter the IP address or name of the device you want to upload application definitions from.
2. Select the **Destination** OU.
3. Click **Start**.

The application definitions of the specified device (or its OU) are uploaded to the specified OU. Already existing applications are deleted.

UPLOADING FROM A DEVICE MANAGED BY SCOUT ENTERPRISE MANAGEMENT SUITE

1. In the Scout Console, select the device you want to upload application definitions from.
2. Click **File > Application upload**.

The **Application upload** dialog opens. The IP address of the selected device is already set in the field **IP-name or IP-address of the device**.

1. Select the **destination** OU to which the application definitions are to be copied.
2. Click **Start**.

The application definitions of the specified device (or its OU) are uploaded to the specified OU. Already existing applications are deleted.

Defining application icons

You can define custom icons for applications to be displayed on the devices\' desktop. For the icon files, the high-resolution formats `.svg` and `.png` are supported.

1. In the tree view, for the top-level **Applications**, from the context menu, choose **Define application icons.**.
2. Click **Add** and select the relevant file from the file system.
3. Confirm with **Open** and **OK**.

The application icon is shown in the dialog. The icon is defined but has not been assigned yet.

Assigning custom application icons

> **Note:**

> Before you assign an application icon other than the default icon, make sure that the icon is already defined. For further information, see Defining application icons.

1. For the relevant application, from the context menu, choose **Properties...**.
2. Select the **Desktop icon** option.
3. Click **...** and select one of the icons.
4. Confirm with **OK** and **Apply**.

The application icon is shown for the selected application on the next device restart.

Defining a service app

> **Note:**

> To use this feature, user authentication via Active Directory is required.

> If you use AD, you can allow users to start eLux® in service mode without logging on. To do so, define one or more service apps. The AD logon dialog then provides an additional **Service** button that starts eLux in a protected mode (service mode). In service mode, eLux only offers the defined service apps on the desktop. In the Configuration panel, only the **Information** dialog is shown.

1. To define a service app, open the **Application properties** of the relevant application.
2. Click **Free Parameters** and add the following parameter:

   `ServiceApp=true`
3. Confirm with **OK**.

After the parameter is transferred to the device, the AD logon dialog contains a **Service** button.

Limiting applications to one logon domain

> **Note:**

> To use this feature, user authentication via Active Directory is required.

> If users have configured multiple AD domains for log-on, you can limit the display of an application to one of the configured domains. This option is defined via a free parameter in the application definition.

1. Open the **Application properties** of the relevant application.
2. Click **Free Parameters** and add the following parameter:

   `ShowInDomain=<AD logon domain>`

   The AD logon domain must match with one of the domains specified in the device configuration under **Security > User authentication > AD directory**.

   Example: `int.sampletec-01.com`
3. Confirm with **OK**.

After the parameter is transferred to the device, the relevant application is only shown if users log on to the specified AD domain.

Control Application Visibility with the ShowApp Parameter

Goal

This guide explains how to use the **ShowApp** parameter in eLux to control whether an application is visible to users based on device properties or environmental conditions. Administrators can define conditional expressions so that applications appear only when specific criteria are met.

This capability helps administrators:

- Restrict applications to certain devices or hardware models
- Show applications only for specific OS versions or configurations
- Hide applications dynamically when device conditions change

The feature was introduced in **eLux 7 2601.0**.

Prerequisites

Before configuring **ShowApp**, ensure the following:

- You have **administrative access to Scout**.
- The application is already **configured in Scout**.
- You can modify **application free parameters**.
- Devices are running **eLux 7.2601 or later**.

How ShowApp Works

The **ShowApp** parameter contains a **boolean expression** that evaluates device properties.

An application will only appear on the user's desktop if the expression evaluates to **true**. If the expression evaluates to **false**, the application is hidden.

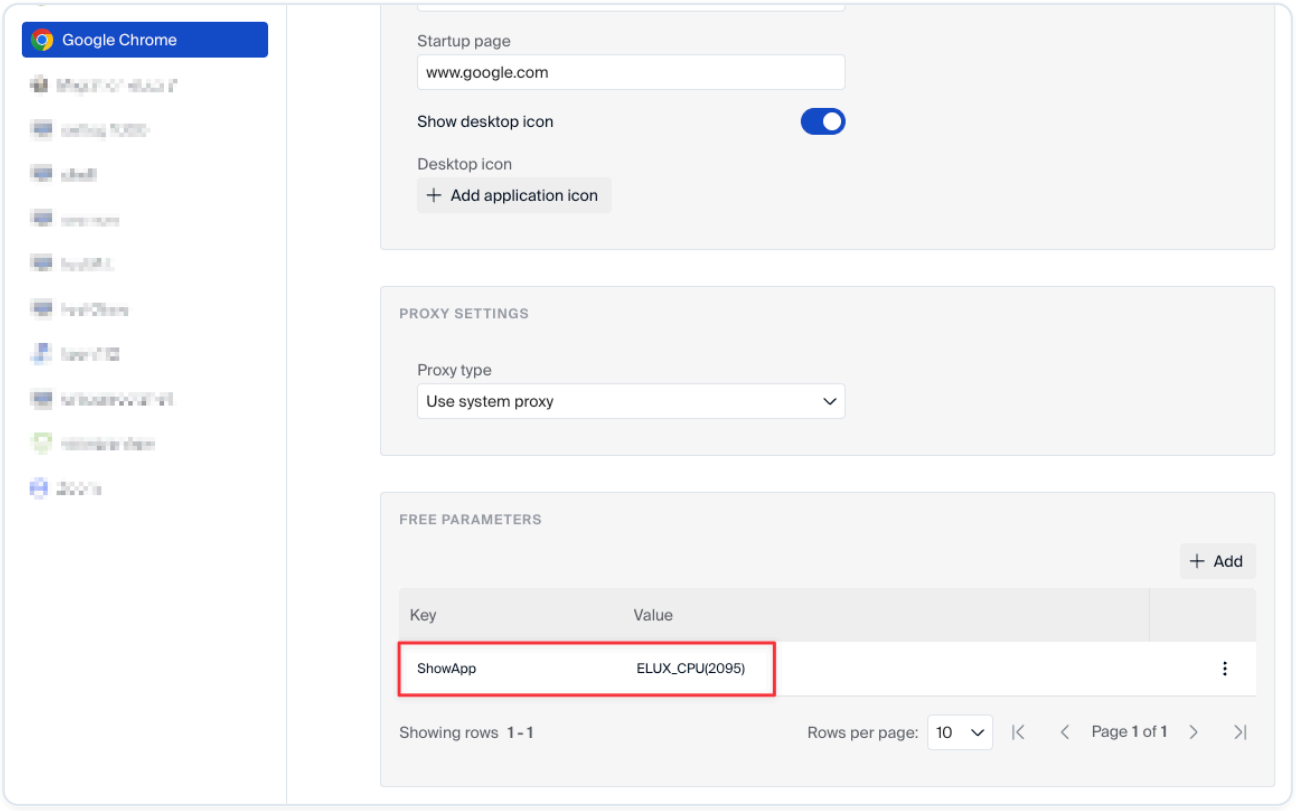
Each condition follows this format of `Property(GLOB)`:

- **PROPERTY** – a device attribute such as hostname, OS version, CPU, or memory.
- **GLOB** – a pattern used to match the value (supports wildcards like `*`).

Multiple conditions can be combined using boolean operators:

- `AND`
- `OR`
- `NOT`
- Parentheses `()` for grouping expressions

Step-by-Step: Configure ShowApp in Scout



1. OPEN THE APPLICATION CONFIGURATION

1. Open **Scout**.
2. Navigate to the desired **device profile**.

3. Select the **application** you want to control.

2. ADD THE SHOWAPP PARAMETER

1. Go to **Free parameters** for the application.
2. Add the parameter: `ShowApp=`
3. Replace `<expression>` with your conditional logic.

Example: `ShowApp=ELUX_HOSTNAME(DellPrecision*)`

This configuration shows the application only on devices whose hostname begins with **DellPrecision**.

3. DEPLOY THE CONFIGURATION

1. Save the profile configuration.
2. Apply the profile to the relevant devices.
3. Restart or refresh the device configuration if required.

The application will appear only when the **ShowApp expression evaluates to true**.

Example Expressions

MATCH A SPECIFIC HOSTNAME

`ELUX_HOSTNAME(DellPrecision3260)`

Application appears only on a device with this exact hostname.

MATCH MULTIPLE DEVICES WITH A WILDCARD

`ELUX_HOSTNAME(DellPrecision*)`

Application appears on any device whose hostname begins with **DellPrecision**.

COMBINE MULTIPLE CONDITIONS

`ELUX_OSVERSION(7.2507.0-3) AND ELUX_SERIAL(71T0ZP3) AND ELUX_MEMORY(8192)`

Application appears only when:

- OS version matches
- Serial number matches
- Device has exactly **8192 MiB RAM**

All conditions must be satisfied because of the **AND** operator.

COMPLEX EXPRESSION EXAMPLE

`ELUX_CPU(3900) OR ELUX_OSNAME(eLux) AND (ELUX_OSVERSION(7.2507.0-3) OR ELUX_SERIAL(71T0ZP3))`

This example combines multiple conditions using both **AND** and **OR** operators.

Available Device Properties

Common properties that can be used in **ShowApp** expressions include:

- `ELUX_HOSTNAME`
- `ELUX_OSNAME`
- `ELUX_OSVERSION`
- `ELUX_SERIAL`
- `ELUX_SUPPLIER`
- `ELUX_DEVICETYPE`
- `ELUX_BIOS`
- `ELUX_CPU`
- `ELUX_PRODUCT`
- `ELUX_KERNEL`
- `ELUX_FLASH`
- `ELUX_IDF`
- `ELUX_MAC`
- `ELUX_IP`
- `ELUX_NETMASK`
- `ELUX_BROADCAST`
- `ELUX_NETADDR`
- `ELUX_FLASHSIZE`
- `ELUX_MEMORY`
- `ELUX_GRAPHICS`

These values are collected by the system during **boot and configuration updates**.

Troubleshooting

The evaluation of ShowApp expressions can be inspected in the following log files.

ELUXD LOG `/TMP/ELUXD.LOG`

Search for entries containing: `showapp`. This log records how the expression was parsed and whether it evaluated to **true** or **false**.

DESKTOP LOG `/TMP/UCDESKTOP.LOG`

This log records when applications are hidden from the desktop because the **ShowApp expression evaluated to false**.

Important Behavior

- Device property values are refreshed **during boot and configuration changes**.
- If device conditions change while an application is running and the expression evaluates to **false**, the application may be **terminated automatically**.
- The eLux desktop hides application icons when the application is marked as hidden.

Best Practices

- Start with **simple expressions** and test them before deploying complex logic.
- Use **wildcards (`*`)** to simplify hostname or device matching.
- Verify behavior using **log files** during initial testing.
- Document expressions used in **Scout profiles** to simplify maintenance.

Connecting to a Citrix® farm

Users can connect to sessions running on a Citrix back-end. Once the connection has been made, the user can access published desktops and applications.

Connecting a device to a Citrix back-end is performed by one of the following applications:

- by a StoreFront application to a StoreFront server
- by the Citrix Self-Service user interface to a StoreFront server
- via a browser to a StoreFront server or Web Interface server, see Browser session to access published resources

Requirements

- The eLux® package **Citrix Workspace-App for Linux** must be installed on the devices.
- To connect via HTTPS, for the application types **Storefront™, Self Service and PNAgent**, the relevant root and intermediate certificates must be available on the devices.
    - Root certificates must be transferred to `/setup/cacerts`.

    - Intermediate certificates must be transferred to `/setup/cacerts/intcerts`.

For further information, see Certificates in the **Installation** guide.

- To connect via HTTPS, for the application type **Browser**, the relevant root and intermediate certificates must be available on the devices.
    - Firefox: Root certificates and intermediate certificates must be transferred to
    - Chromium: Root certificates and intermediate certificates must be transferred to
- The eLux taskbar should be enabled on the devices if published applications are provided as **seamless applications**. Seamless applications behave like local applications and users can only restore them from minimized window size by using the taskbar. For further information, see Advanced desktop settings.
- StoreFront application
- StoreFront / Store provisioning file
- StoreFront / Authentication
- Self-Service user interface
- Custom design for Citrix Workspace-App
- Browser session to access published resources
- Citrix software defaults
- Citrix Connection Center
- Logging for Citrix Workspace-App
- Updating Citrix Workspace-App

Storefront™ application

By using the application type **StoreFront**, users can connect to a Citrix StoreFront server. Virtual desktops and published applications are aggregated and provided through stores. The Citrix products mainly used are XenApp and Citrix XenDesktop. StoreFront sites can be accessed via HTTP or HTTPS.

The StoreFront application enables users to access Citrix resources of one or more stores together with other configured applications, such as **RDP** or **Browser** sessions by using only one interface -- the User Interface. For further information, see eLux RP 6 User Interface.

DEFINING A STOREFRONT APPLICATION

Note:

HTTPS connections require the relevant Certificates on the device.

1. Add a new application see Adding applications and select the application type **StoreFront**.
2. Edit the following fields:

OPTION	DESCRIPTION
Name	Name of the application shown in the Scout Console
Use Provisioning File (.cr)	Use a provisioning file. This option excludes the specification of Store URLs (next option).
Stores	Specify the URL of one or more stores.
	- Click **Add** and replace the automatically created default value by your individual value (double-click or F2).
	Example: `https://CtrXd76.sampletec-01.com/Citrix/Store33/discovery`
	This option excludes the use of a Provisioning file (previous option).
Logon	The user is automatically logged on to the store by using the specified credentials (username, password, domain).
Pass-through logon	The user is logged on to the store via single sign-on. The AD user credentials are used.

Note:

> If you want to use predefined credentials or pass-through authentication, the eLux® package **Citrix Extensions** and the included feature package **Dialog Extension** must be installed on the devices.

> For further information, see StoreFront / Authentication.

Show last user	The user credentials (except for password) of the last logon are displayed in the XenApp® logon dialog. This option has no effect if you specify fixed user credentials for automatic logon under **Logon**
Autostart	Specify the names of those StoreFront applications you want to have started automatically. Make sure to spell the names exactly as in StoreFront. Separate multiple application names by semicolon. Example: `MyApp1;MyApp2`. If only one resource is defined for a store, alternatively use the free parameter .
Application restart	See Adding applications.
Start automatically	
Desktop icon	
Free parameters (optional)	Individual parameters for application start. For further information, see Defining free application parameters.

3. To delete an entry from the **Stores** list, select the entry and click **Delete**.
4. To configure further settings, click **Advanced** and edit the following fields:

OPTION	DESCRIPTION
Windows properties	Desktops can be launched in full-screen or window mode.

OPTION	DESCRIPTION
Timed logoff	To enable automatic logoff from the StoreFront server, select the **Logoff after** option and specify a delay in seconds. Automatic logoff does not affect the launched desktop. Alternatively, automatic logoff can be configured to be performed after the last StoreFront application has been closed.
Application reconnection	Determine the actions to be done on a reconnect to the StoreFront server
	Do not reconnect: The connection to the desktop or the published applications is not restored (default).
	Disconnected sessions only: The connection to a disconnected session is restored.
	Active and disconnected sessions: The connection to a disconnected or active session is restored.
Manual logoff	Determine the actions to be carried out upon logoff from the StoreFront server
	Logoff only server: Logoff is performed only from the StoreFront server
	Logoff server and applications: Logoff is performed from the StoreFront server and from the virtual desktop or published applications.
	Logoff server and disconnect session: Logoff is performed from the StoreFront server but the virtual desktop session is only disconnected. This enables the user to reconnect later on.

> **Note:**
>
> Access to the advanced settings can be defined via the object rights.

5. Confirm with **Apply** and **OK**.

After users have logged on to a StoreFront server or Web Interface server, they can show all provided resources by double-clicking the **StoreFront** icon on the eLux desktop.

StoreFront / Store provisioning file

A Citrix store provisioning file can be created by the Citrix back-end and contains all relevant connection information. Using this file allows to switch automatically from Citrix StoreFront™ connection data to Citrix Access Gateway connection data if StoreFront is not reachable (scenario of switching between company office and home office).

To use a store provisioning file for your eLux® devices, note the following:

- The file must be located on the devices in the directory `/setup/ica/` Transfer the .cr file by using the Scout feature . For further information, see Files configured for transfer.

StoreFront / Authentication

If on a device, smart card packages are installed and the Citrix Workspace-App for Linux identifies smart card middleware on the device, smart card logon has precedence by default. In order to still authenticate via username and password for certain devices, define the authentication method via a parameter:

CONTROLLING THE AUTHENTICATION METHOD VIA ELUX®

The logon method can be changed to username and password regardless of the smart card packages installed.

- Define the following entry by using the Scout feature Adding individual file entries:

File	`/setup/sessions.ini`
Section	`ICADefaults`
Entry	`StoreFrontLogOnWithPassword`
Value	`true | false` , (Default: false)

CONFIGURING SMART CARD BEHAVIOR

If you use smart card authentication for StoreFront™, you can configure the behavior of the smart card when it is removed.

> **Note:**
>
> Using a smart card requires the smart card middleware to be installed on the device. In addition, smart card authentication must be enabled on the Citrix® farm.

- Define the following entry by using the Scout feature Adding individual file entries:

File	`/setup/sessions.ini`
Section	`ICADefaults`
Entry	`SmartcardRemovalAction`
Value	`noaction | forcelogoff` , (Default: noaction)

FURTHER PARAMETERS FOR AUTHENTICATION

To define further parameters for authentication, use the configuration file `/setup/ica/AuthManConfig.xml.template`. This file is transferred to the devices during installation. Using the **Diagnosis** and **Files configured for transfer** features, you can retrieve the file, edit it and transfer it back to the relevant devices. For further information, see Files configured for transfer.

For the function to become active on the device, one restart is necessary to transfer the file, and another restart to activate the new parameters.

Self-Service user interface

The Self-Service user interface (UI) replaces the configuration manager **wfcmgr** and allows access to Citrix services providing published resources.After users are set up with an account, they can subscribe to desktops and applications, and then start them.

DEFINING CITRIX SELF-SERVICE AS LOCAL APPLICATION

> **Note:**

> The eLux® package **Citrix Workspace-App for Linux** and the included feature package **Self-service** must be installed on the devices. This may require modifications of the image definition file on the web server via ELIAS.

1. Add a new application and select the application type **Local**.
2.

OPTION	DESCRIPTION
Name	Name for the application
Local application	Select `Custom`.
Parameter	Enter the following program name to start the application:
(mandatory)	`selfservice`

3. Confirm with **Apply** and **OK**.

> **Note:**

> The `selfservice` application cannot be configured individually. To use configuration options, alternatively use the Citrix Self-Service in kiosk mode.

Custom design for Citrix Workspace-App

> **Note:**

> The eLux package **Citrix Workspace-App** (version 18.08 or later) must be installed on the devices.

> To customize the layout of your Citrix® session, transfer the relevant layout files to the devices into the Citrix directory structure. The files then are merged with the Citrix layout files.

To transfer the files, use the Scout feature **Files configured for transfer**. For further information, see Files configured for transfer.

As destination specify the provided Citrix directories. Example:

Default	`/setup/ica/site_custom`
If **Shared User Mode** is used	`/setup/ica/site_custom/sum_screen`

The Citrix directory structure must be retained.The original structure can be found under `/opt/Citrix/ICAClient/site_orig`.

Browser session to access published resources

Users can access applications and desktops that have been published through a store on the Citrix StoreFront™ server or through Citrix Web Interface by using a local browser.

DEFINING A BROWSER APPLICATION TO ACCESS PUBLISHED RESOURCES

> **Note:**

> To provide the users with a browser application to be used directly on the device, the relevant software package for Firefox or Chromium must be installed on the devices. This may require modifications of the image definition file on the web server via ELIAS.

> HTTPS connections require the relevant SSL certificates on the device.

1. Add a new application see Adding applications and select the application type **Browser**.
2. Edit the following fields:

OPTION	DESCRIPTION
Name	Name for the browser session
Browser type	Firefox or Chromium
Called page	URL of the Web Interface homepage or StoreFront store.
	Examples
	`https://<Servername>/Citrix/StoreWeb`
	`https://<Servername>/Citrix/XenApp`

3. For the remaining parameters, see Defining a browser application.

The local user starts the browser and is forwarded to the defined page. After successful logon to the StoreFront server or Web Interface server, the published applications, desktops and contents available are shown in the browser window.

Citrix® software defaults

For all Citrix applications, in the Scout Console, you can define Citrix Workspace-App software defaults that are applied to all devices of the relevant OU and subordinate OUs if configured.

The following options are available:

- Client drive mapping
- COM port mapping
- Firewall settings
- Citrix keyboard shortcuts
- Window properties
- Connection options
- Bitmap caching

To edit the software defaults, see Software defaults.

> **Note:**

> To define parameters in individual configuration files, use the Advanced file entries feature. All parameters defined by using the **Advanced ini entries** override the software defaults.

> Some of the Citrix default options are described below. For further information, see the Citrix documentation.

GENERAL TAB

OPTION	DESCRIPTION
TW2StopwatchMinimum	Scrolling speed for remote applications (such as Adobe Acrobat Reader, Microsoft Excel)
	The higher the value, the slower the speed when scrolling

OPTION	DESCRIPTION
	Note for Excel: A low value increases scrolling speed but delays as soon as a selection is drawn down out of the visible screen area.
	Default = 25
Client name template	Definition of the client name in the Citrix session
	Note: You can use the Program Neighborhood variables [$ICANAME] and [$ICADOMAIN] to set a unique client session name. This is required for Citrix Roaming and some XenApp programs. For further information, see PNAgent application.

DRIVE MAPPING TAB

OPTION	DESCRIPTION
A-Z	The letters A to Z represent the logical drive names on the terminal server. In the field on the right, you can assign a local resource to a drive letter that is to be shown in the Citrix session.
	Enter the mount point relating to the local access path of the resource. The mount points are provided by eLux: [/media/usbdisk] or [/media/cdrom]
Attributes E / R / W	Type of access right
	• E = enable
	• R = read
	• W = write

OPTION	DESCRIPTION
Enable Drive Mapping	Must be selected to enable the defined drive mappings
Enable Dynamic Mapping	Available mass storage devices are assigned to the next free drive letter.

For further information, see Mount points.

COM PORTS TAB

To connect via COM port, the device name of the client device's COM port is required.

The COM port device name always begins with the string . Device names are case-sensitive.

Examples:

PORT DEVICE NAME	COM PORT
[/dev/ttyS0]	COM1
[/dev/ttyS1]	COM2

The availability of COM ports depends on the hardware platform.

> **Note:**

> The devices' ports must be mapped on the Citrix resource (such as desktops) accordingly. To do so, use a `net use` command. Example: `net use com1: ClientCOM2: /persistent:yes`.

Citrix Connection Center

By means of the Citrix Connection Center, users can see all current server connections and can log off, disconnect or close them without operating the application.In addition, the connection transport statistics can be viewed which might be helpful for slowing connections.

The Connection Center is provided as local desktop application.

DEFINING THE CITRIX CONNECTION CENTER

> **Note:**

> The **Citrix Workspace-App** and the included feature package **Utilities and tools** must be installed on the devices. This may require modifications of the image definition file on the web server via ELIAS.

1. Add a new application see Adding applications and select the application type **Custom**.
2.

OPTION	DESCRIPTION
Name	Citrix Connection Center
Local application	Select `Custom`
Parameter	`/opt/Citrix/ICAClient/util/conncenter`
Desktop Icon	Activate desktop icon

3. Confirm with **Apply** and **OK**.

Logging for Citrix Workspace-App

For the Citrix Workspace-App, you can enable and configure logging via a configuration parameter.

CONFIGURING THE LOG LEVEL

1. In the Scout Console, for the relevant devices, open **Advanced device configuration > Advanced file entries**.
2. Define the following entry

FILE	
Section	
Entry	
Value	0|3|7
	0: Logging disabled (default)
	3: Only errors are logged

FILE	
	7: Logs for all levels are generated

For further information, see Advanced file entries.

As soon as the new configuration is active, logs are written to .

ENABLING DETAILED LOGGING FOR MICROSOFT TEAMS

Note:

The eLux® package **Citrix Workspace-App for Linux** and the included feature package **Microsoft Teams Optimization** must be installed on the devices.

1. Create a configuration file named `config.json` and enter the following:

```
[Copy]



{

  "WebrtcLogLevel" : 0,

  "WebrpcLogLevel" : 0

}
```

2. Transfer the `config.json` file to the devices to `/setup/ica/hdx_rtc_engine`

   For further information, see Files configured for transfer.

When Microsoft Teams (in VDI) is used on the devices, the relevant log files are written.

- To access the log files, use the diagnostics function. To do so, create a diagnostic template that contains the following entries:

  /tmp/hdxrtcengine// /tmp/webrpc//

For further information, see Device diagnostics.

After retrieving the diagnostic files, you can find the log files in the `hdxrtcengine` and `webrpc` directories.

Updating Citrix Workspace-App

- To install later versions of the Citrix Workspace-App on the devices, perform a firmware update.

> **Note:**
>
> The existing `/setup/ica/AuthManConfig.xml` template will be overwritten with the template of the new version. Individual entries must then be set again.

BEHAVIOR OF DEFAULT VALUES

-from Scout 15 2204

When you update to a later version of the Citrix Workspace-App, the vendor's default values are applied. This overwrites individually set values in favor of a consistent situation between updated devices and freshly installed devices.

The vendor's default values are set for the following scenarios:

- Firmware update to newer CWA package (with and without formatting the system partition)
- Firmware update to newer eLux® version including new CWA version (with and without formatting the system partition)
- Factory reset
- PXE or USB installation

RDP

The **RDP** application type uses the Microsoft Remote Desktop Protocol (RDP) to connect to a Microsoft terminal server. The provided RDP client is eLuxRDP that is based on the free software implementation **FreeRDP**.

There are two options for configuration:

- Windows Desktop: The user accesses the desktop of a terminal server by using a remote desktop session. The user can use any application available on the desktop.
- Individual / seamless application: The user can only access one particular application of the terminal server.

- Defining an RDP Windows desktop session
- Defining an RDP application
- Defining a load-balanced RDP session
- Advanced application settings / RDP and VMware
- Configuring RemoteFX

Defining an RDP Windows desktop session

1. Add a new application see Adding applications and select the application type **RDP**.
2. Edit the following fields:

OPTION	DESCRIPTION
Name	Name for the RDP application
Server	IP address or name of the server
Application	Leave the field empty.
Working directory	Leave the field empty.
Logon	The user is automatically logged on to the server by using the specified credentials (username, password, domain).
	The user is logged on via single sign-on. The AD user credentials are used.
Free parameters	Allows to define any parameters supported by **eluxRDP** in the format: `FreeRDPParams=<Parameter> <Parameter> <Parameter>`. Separate multiple parameters by spaces. Examples: `FreeRDPParams=/microphone:sys:pulse +fonts /cert-ignore`. To view the allowed parameters, enter the `eluxrdp` command in a shell. For further information, see Defining free application parameters.

3. Confirm with **Apply** and **OK**.

> **Note:**
>
> Defining a server-independent application as a local hidden application named `RDP_TEMPLATE` allows you to configure a connection template without a back-end. The user starts `rdpconnect` from the shell and subsequently specifies the server to be connected to. This feature requires the eLux® software package **RDPConnect**.

Defining an RDP

application

To configure an individual application via RDP, the Windows desktop definition requires additional data about the relevant application.

1. Add a new application see Adding applications and select the application type **RDP**.
2. Edit the following fields:

OPTION	DESCRIPTION
Name	Name for the RDP application
Server	IP address or name of the server
Application	Name of the Windows application including path name
	System variables are allowed.
	Examples: `c:\Program Files\Microsoft Office\Office\EXCEL.EXE`
	`%SystemRoot%\system32\notepad.exe`
Working directory (optional)	Working directory of the Windows application
Logon	The user is automatically logged on to the server by using the specified credentials (username, password, domain).
Pass-through logon	The user is logged on via single sign-on. The AD user credentials are used.

OPTION	DESCRIPTION
Free parameters	Allows to define any parameters supported by **eluxRDP** in the format:
	`FreeRDPParams=<Parameter> <Parameter> <Parameter>`

3. Confirm with **Apply** and **OK**.

For the users, the application runs full-screen in the session window.

Defining a load-balanced RDP session

If you are running a remote desktop server farm with load balancing, have your `eLux` RDP sessions load-balanced via the following configuration.

1. Define an RDP session/application or open the **Properties** dialog of an existing one. Edit the following options:

OPTION	DESCRIPTION
Server	Name of your RD connection broker
	Example: `RDPLoadMaster.int.sampletec-01.com`
Free parameters	Parameter for the load balancer server collection
	`FreeRDPParams=/load-balance-info:tsv://MS Terminal Services Plugin.1.<[Load balancer collection]>'`
	Example: `FreeRDPParams=/load-balance-info:tsv://MS Terminal Services Plugin.1.RDS-01>'`

2. Confirm with **Apply** and **OK**.

Advanced application settings for RDP and VMware

The settings described below apply to the following applications:

- RDP applications
- VMware applications

If you select a protocol other than RDP, some options are not available.

ACCESSING ADVANCED APPLICATION SETTINGS

> **Note:**
>
> Access to the advanced settings can be restricted via the object rights.

- Scout: In the **Application properties** dialog of an RDP or VMware application, click the **Advanced** button.
- eLux® RP 6: In the **Application properties** dialog of an RDP or VMware application, under **Properties**, expand the relevant section.

VIEW TAB

OPTION	DESCRIPTION
Window size	Full-screen or a specific resolution
Full-screen on monitor	If you have selected the window size `Full-screen`, select if you want to display on one specific or all monitors. Up to eight monitors are supported.
Colors	Color depth for the session (8-32 Bit)

> **Note:**
>
> If you use multiple monitors but wish to display content on only one of them, under **Device configuration > Desktop > Advanced > Windowmanager**, the **Maximize/fullscreen to single monitor** option must be selected.

LOCAL RESOURCES TAB

for terminal servers supporting RDP protocol version 5.2 or later -- The settings take effect only if, on the **Advanced** tab, the value of the **Protocol** field is not set to `RDP V4`.

OPTION	DESCRIPTION
Drive mapping	Select drive, mount point and drive letter that you want to show in the RDP/VMware session. The mount points correspond to the local access paths of the resources and are provided by eLux. For USB devices, define the assignment with `/media`. Internally, the mount point for USB devices is set to /media/usbdisk. If multiple USB devices are plugged in, they use the same mapping. Each USB device is displayed in its dedicated folder. A digit is appended to the folder name and incremented `(usbdisk0, usbdisk1)`. For further information, see Mount points.
Connect printer	Up to four printer definitions can be created automatically for a session. The printers must be configured on the Defining a locally connected printer. The first four profiles can be used with drivers. To define a default printer, choose **Set as default** in the eLux printer configuration.
Sound	**Play local** reproduces the sound locally on the device. **Play remote** causes playback on the remote server.
Connect ports	Makes the defined port connections accessible in the session.
Enable smartcard	Smart cards based on a certificate can be used for logon.

ADVANCED TAB

OPTION	DESCRIPTION
Protocol (only RDP)	Enables you to set the RDP protocol to version 4 or 5. Normally, the protocol is recognized automatically.

OPTION	DESCRIPTION
Keyboard language	Defines the keyboard layout within a session. The default is **Auto** which corresponds to the keyboard setting of the eLux device configuration.

Important:

If you define a specific language, it must be identical to the keyboard language defined in the eLux device configuration, in the **Keyboard** dialog.

OPTION	DESCRIPTION
Deactivate Window-Manager Decorations	The frames of the eLux windows are hidden.
Deactivate encrypting	The server does not accept encrypted sessions. You can use this option to increase performance. By default, the option is disabled.
Deactivate mouse move events	Mouse position data are not transferred to the server constantly, but with every mouse click. This increases system performance and is especially helpful for connections with small bandwidth. By default, the option is disabled.
Show connection bar on full screen	Shows connection list in full-screen mode.
Bandwidth	Choose between `standard, modem, broadband or LAN`.

Configuring RemoteFX

Microsoft RemoteFX™ offers comprehensive functionality for Virtual Desktop Infrastructure (VDI) by providing a virtual 3D adapter, intelligent codecs and the ability to redirect USB devices to virtual machines.

Note:

RemoteFX only works if the server supports RemoteFX and is configured in the right way. The only parameter to be configured on the device is bandwidth.

1. For your **RDP** application, open the **Application properties** dialog and click **Advanced**.
2. On the **Advanced** tab, in the **Bandwidth** field, select `LAN`.
3. Confirm with **Apply** and **OK**.

Virtual Desktop

Note:

For eLux® RP 6, instead of **Virtual Desktop**, the application type **VMware Horizon** is available.

As Virtual Desktop VMware Horizon is supported.

- Defining a virtual desktop
- VMware Horizon

Defining a virtual desktop

1. Add a new application see Adding applications and select the application type **Virtual Desktop**.
2. Edit the following fields:

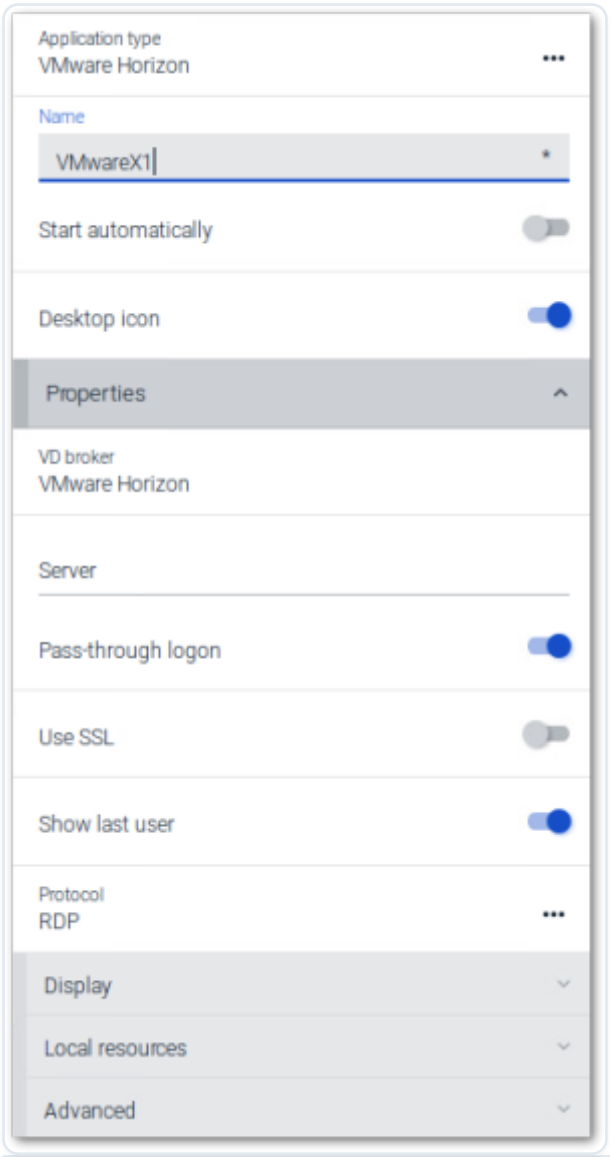
OPTION	DESCRIPTION
Name	Name for the application
VD Broker	Select a virtual desktop application from the list.
Server	Enter the server IP address (or name)
Logon, Pass-through logon	See Adding applications

OPTION	DESCRIPTION
Protocol, (VMware Horizon only)	Choose between the following values: `RDP, PCoIP, VMware Blast`

3. To configure further settings for XenDesktop® or VMware Horizon, click **Advanced**. For further information, depending on the broker selected, see
   - Advanced application settings / RDP and VMware or
   - Advanced XenDesktop settings in PNAgent application
4. Confirm with **Apply** and **OK**.

VMware Horizon

This application type is only available on eLux® RP 6 devices. In the Scout Console, select the Virtual Desktop application type and as VD-Broker VMware View.



OPTION	DESCRIPTION
Name	Name for the application
Auto-start	The application starts automatically after eLux has been started.
Desktop icon	Provides a desktop shortcut on your personal desktop
VD broker	VMware Horizon
Server	IP address or name of the server
Pass-through logon	The user is logged on via single sign-on. The AD user credentials are used.

OPTION	DESCRIPTION
Username, Password, Domain	The user is automatically logged on to the server by using the specified credentials.
Use SSL	Forces the connection via HTTPS
	Note that HTTPS connections require the relevant SSL certificates on the device.
Show last user	The user credentials (except for password) of the last logon are displayed in the logon dialog
Protocol	Choose between the following protocols
	`RDP`
	`PCoIP`
	`VMware Blast`

For information on **Display, Local resources and Advanced** settings, see Advanced application settings.

You may configure the VMware Horizon client by using the application definition in the or locally on the client. To set additional parameters that are not included in the interface, use a configuration file:

- With the help of VMware documentation,[1[Installation guide for VMware Horizon Client for Linux]] create the file `view-userpreferences`. Transfer the file via the Scout feature Files configured for transfer to `/setup/elux/.vmware/view-userpreferences`.

> **Note:**

> The configuration on the Scout or eLux interface has precedence over the configuration file and will overwrite values of the configuration file.

Browser

Supported browsers are Mozilla Firefox and Google Chromium.

In addition, the Builtin Browser is available as a slimmed-down browser. The Builtin Browser is based on the WebKit2 engine which is part of the **Desktop environment** package. By default, the Builtin Browser is run without address and navigation bar. These and some more features can be configured for the kiosk mode.

> **Note:**

> If you use Chromium, we recommend that you equip your devices with 2 GB of RAM.

> For eLux® RP 6, the Java browser plugin is no longer supported.

- Defining a browser application
- Preferences Chromium
- Preferences Firefox
- Preferences Builtin Browser
- Browser home directory
- Kiosk mode for Firefox

Defining a browser application

1. Add a new application adding a new application and select the application type **Browser**.
2. Edit the following fields:

OPTION	DESCRIPTION
Name	Name of the browser shown in the Scout Console
Browser type	Select `Firefox, Chromium or Builtin Browser`.
Start page	Web page (URL) that opens when you click **Home**
Called page	Web page (URL) that opens after starting the browser
Proxy type	- `No proxy`: No proxy server is used

OPTION	DESCRIPTION
	- `Manual (Proxy:Port)`: Specify a proxy server and port number
	- `Auto (URL)`: Use a proxy configuration file
	- `Use system proxy` (default): 'System-wide' proxy setting defined in the device configuration under **Network > Advanced** per network profile
	Note that the setting behind `System proxy` can also be No proxy.
	For further information, see Proxy configuration.

Note:

For the Builtin Browser, the setting must be left on Use `system proxy`.

NAME	INFO
Application restart	See Adding applications
Start automatically	
Desktop icon	
Free parameters (optional)	Individual parameters for application start
	see Defining free application parameters

3. To enable the **Kiosk** mode for Firefox, see Kiosk mode for Firefox.
4. Confirm with **Apply** and **OK**.

Note:

> By default, all browser files (cache, history, bookmarks, etc.) are saved temporarily to the flash memory but are deleted with each restart. We recommend that you configure the browser home directory on a network drive. For further information, see Browser home directory.

> Further browser-specific preferences can be set through policies (Chromium) or configuration file entries (Firefox.). For further information, see the Scout guide:

Preferences Chromium

Preferences Firefox

Preferences Builtin Browser

DEPLOYING SSL CERTIFICATES FOR THE BROWSER

- Use the Scout feature **Files configured for transfer** to transfer certificate files to the required target directory on the device:

BROWSER	FOLDER
Mozilla Firefox	`/setup/cacerts/firefox` or `/setup/cacerts/browser`, The certificates can be located in either directory.
Google Chromium	`/setup/cacerts/browser`

For further information, see Files configured for transfer.

Note that a second restart of the device is required to assign the certificates that have been transferred during the first boot to the certificate store of the browser.

Preferences Chromium

By using policies, you can set mandatory (managed) and recommended preferences for the Chromium browser. Mandatory preferences define fixed values that cannot be changed by the user. Recommended preferences define default values that can be changed by the user. For further information, see https://www.chromium.org/administrators/.

- Use the Scout feature **Files configured for transfer** to transfer policy files (.json) to the required target directory on the device:

NAME	FOLDER
Fixed values	`/setup/chromium/managed`
Default values	`/setup/chromium/recommended`

For further information, see Files configured for transfer.

Preferences Firefox

For version 60 ESR and later versions, Firefox supports enterprise policies that are deployed via `.json` files and are cross-platform compatible. (This method will not work if Firefox is already being managed by using Windows group policies.) Firefox is installed with enterprise policies enabled that block access to `about:config` and other configuration options by default.

SETTING PREFERENCES WITH .JSON FILES (POLICIES)

from Firefox 60 ESR and later versions

You can use all options that are listed in the README on the Mozilla GitHub repository. (Note that the current Firefox version may differ from that of the eLux version you are using.) One or more options are transferred in a `.json` file to the device by using the Scout feature **Files configured for transfer**.

> **Note:**
>
> By default, access to the Firefox configuration is blocked.

1. Create a `.json` file (any file name) and insert one or more options separated by commas. Use the syntax shown in the example. (No "policies" brackets!)

   Example:

   ```
   {
     "BlockAboutConfig": false,
     "DisableBuiltinPDFViewer": true
   }
   ```

2. In the Scout Console, for the relevant devices, open **Advanced device configuration > Files**.

   Define your `.json` file as source file. For the destination folder, use `/setup/firefox/policies/`. Example: `/setup/firefox/policies/custom_A.json`

   For further information, see Files configured for transfer.

On the next device restart, the files are transferred and evaluated.

> **Note:**
>
> You can deploy multiple `.json` files to the device to `/setup/firefox/policies/`. The files are merged in alphabetical order: For identical options, values from files with descending names have precedence (B overrides A).

Preferences Builtin Browser

DISPLAY OPTIONS

By default, the Builtin browser opens with only the title bar, but no navigation bar.

- To display the navigation bar with additional functions, open the properties of your browser application and click **Advanced**. (from 15 2101 and eLux® RP 6 2103)

With which functions the navigation bar is displayed, you also define in the **Advanced browser settings**:

1	Back
2	Forward
3	**Home** button to open home page
4	Refresh
5	Address bar
6	Print

The title bar is always shown.

USER AGENT-STRING

With each page request, the Builtin browser transmits a 'User Agent' string, which is predefined by the WebKit2 engine. It contains information like Mozilla compatibility and operating system.

To transmit individual information, define an Advanced file entry:

1. In the Scout Console, for the relevant devices, open **Advanced device configuration > Advanced file entries**.

File	`/setup/sessions.ini`
Section	`[BuiltinBrowserDefaults]`
Entry	`UserAgentString`
Value	`<name/version> <individual string>`
Example	`Mozilla/5.0 (X11; Linux x86_64)`

   For further information, see Advanced file entries.

> **Note:**
>
> Name and version must be included in the User Agent string.

DNS CACHE

from eLux RP 6 2104

The DNS cache of the Builtin browser is set to 120 seconds by default. To configure the timeout value individually, define an environment variable in the Advanced device configuration:

1. In the Scout Console, for the relevant devices, open **Advanced device configuration > Environment**.
2. To define a new variable, click **New**.
3. Enter the variable using the format: `WEBKIT_DNS_CACHE_EXPIRE_TIMEOUT=<number of seconds>` and confirm with **OK**.

The new variable is shown in the list.

For further information, see Environment variables.

CLEARING THE BROWSER CACHE

- To delete all browser data including cookies, press `CTRL+SHIFT+DELETE`

Browser home directory

By default, the browser settings are temporarily saved to the flash memory. However, they are deleted with each device restart.

Defining a browser home directory on the network, however, allows you to let users save and make available their browser settings such as bookmarks persistently. To do so, use a network share that you have configured for access:

DEFINING BROWSER HOME DIRECTORY

REQUIRES
Configured Windows network share (Defined drive).
Example: `/smb/share`
For further information, see Defining a network drive.

1. In the tree view, for the relevant level, open the context menu and click **Software defaults**

   For further information, see Software defaults.
2. On the list-field, select the relevant browser and click **Edit**.
3. In the **Browser home directory** field, enter the name of one of the defined drives in **Device configuration > Drives**. The name must correspond to the name on the list. Example: `/smb/share`
4. Confirm with **OK**.

The browser settings are saved to the specified Windows directory.

Kiosk mode for Firefox

for Firefox from ESR version 71.0

> **Note:**

> For current eLux RP 6 versions, you can use the Builtin Browser in kiosk mode. For further information, see Browser in kiosk mode (current eLux versions).

> The kiosk mode starts the browser in full-screen mode and with limited user rights. The user cannot open other windows and cannot exit the browser.

By default, the browser window is displayed without address bar and navigation buttons. So users are forced to stay on the predefined web page and cannot exit.

Kiosk mode is suitable if the users are supposed to see only one website and not use further applications on the device. For correct use of the kiosk mode, we recommend that you disable related functions such as restarting the device and opening the Config panel. For further information, see Device configuration > Security.

CONFIGURING KIOSK MODE

> **Note:**

> Firefox supports kiosk mode again starting with version 71.0, but without configuration options. With Scout 15 2110, the Firefox application definition is adapted and offers only the option **Enable kiosk mode**.

1. In the application properties of your browser application, click .
2. On the **Kiosk mode** tab, edit the following fields:

OPTION	DESCRIPTION
Enable kiosk mode	Activates the kiosk mode
Display navigation bar (up to Scout 15 2107)	Allows using browser tabs despite kiosk mode. Users can view multiple web pages of the defined web site concurrently
Add print button (up to Scout 15 2107)	Allows using browser tabs and provides a **Print** feature despite kiosk mode
Add address bar (up to Scout 15 2107)	Allows using browser tabs and provides the address bar including navigation buttons despite kiosk mode

3. Confirm with **Apply** and **OK**.

On the next restart, the Firefox browser opens in kiosk mode.

local and user-defined applications

Defining local commands is particularly important as they enable the definition of applications which can be launched within a shell. This feature assumes knowledge about the commands that average users may not have.

> **Note:**

> Make sure that the user is authorized to start the application. All commands are executed by the UNIX user **eLux®** (UID = 65534).

> Some of the local applications are predefined. If an application is missing, you can define your own application or command via the Custom option of the **Local Application** list-field.

Error messages will not be shown. If your command does not include an X client application, no messages are shown during execution. For this reason, we recommend first running the command within an XTerm session for testing purposes.

- Defining predefined local applications
- Defining custom applications
- Defining Zoom for Linux
- Defining Ekiga SIP Softphone

Defining predefined local applications

1. Add a new application see Adding applications and select the application type **Local**.
2. Edit the following fields:

OPTION	DESCRIPTION
Name	Name of the application shown in the Scout Console
Local application	In the list-field, select a predefined application.

OPTION	DESCRIPTION
Parameter (optional)	Command-line parameters for application start
Application restart	See Adding applications
Start automatically	
Desktop icon	
Free parameters (optional)	Individual parameters for application start
	see Defining free application parameters.

3. Confirm with **Apply** and **OK**.

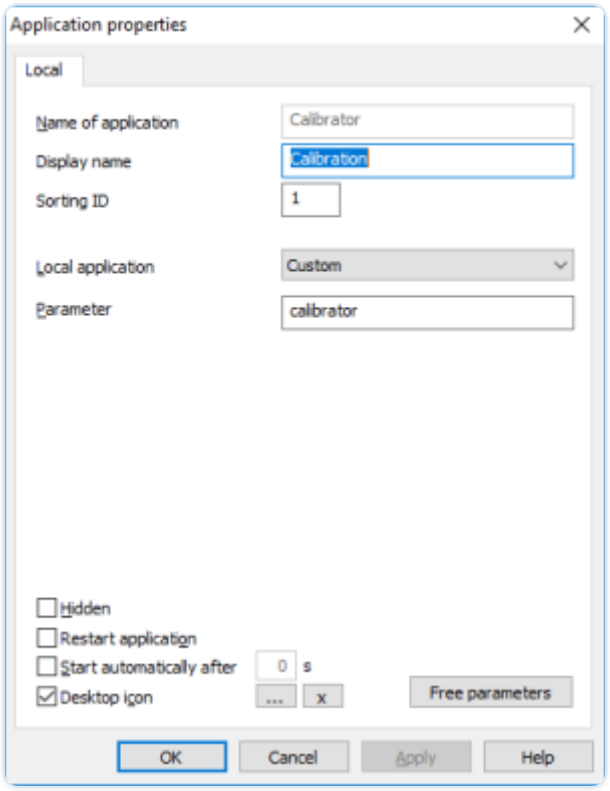
Defining custom applications

1. Add a new application see Adding applications and select the application type **Local**.
2. Edit the following fields:

OPTION	DESCRIPTION
Name	Name of the application shown in the
Local application	Select `Custom`.
Parameter	Enter the program name required to start the application. (mandatory) If required, add start parameters. Example: `calibrator` calls the **Calibrator** tool, `squid` calls the **Squid** application, `squid /tmp/mycache` calls **Squid** using the specified cache directory
Hidden	The application is not shown on the desktop. This is meant for automatically started applications. Select **Start automatically or Application restart** instead.
Application restart	See Adding applications.
Start automatically	

OPTION	DESCRIPTION
Desktop icon	
Free parameters	Individual parameters for application start, see (/en-us/unicon-elux-scout/2603-ltsr-cu1/admin_guides/scout/app_definition/app_general/app_params.html)

3. Confirm with **Apply** and **OK**.



The figure shows the application definition for the **Calibrator** tool . After the next device restart, the **Calibration** application is provided on the device's desktop and can be started (provided that the **Calibrator** tool is included in the image).

Defining Zoom for Linux

The native Zoom client for Linux is a Video Conferencing and Web Conferencing service and offers high-quality video, audio, and screen-sharing experience.

The video and audio devices are configured via the application interface.

1. Add a new application see Adding applications and select the application type **Local**.
2. Edit the following fields:

OPTION	DESCRIPTION
Name	Name for the application
Application	Custom
Parameter	zoom

3. Confirm with **Apply** and **OK**.

Defining Ekiga SIP Softphone

Ekiga is a free software application for audio and video telephony (VoIP) that supports the SIP protocol. Configuration is based on a SIP account.

1. Add a new application see Adding applications and select the application type **Local**.
2. Edit the following fields:

OPTION	DESCRIPTION
Name	Name for the application
Application	`Custom`
Parameter	`ekiga`

3. Click **Free parameters** and then **Add** to define the following free parameters:

VARIABLE	VALUE
account	`<Name of the SIP account>`
server	`<server URL>`
user	`<SIP username>`
password	`<password>`
outbound_proxy	`<proxy URL>`

Example: `password=424242` For further information, see Defining free application parameters.

4. In the **Free application parameters** dialog, right-click the variable name `password` and click **Encrypt**.
5. Confirm with **Apply** and **OK**.

Emulation

Applications of the Emulation type can only be defined in the Scout Console but not locally on eLux® RP 6 devices.

Currently, the **5250 terminal emulation** is still supported.

Applications in kiosk mode

Some applications can be configured so that users can only operate them in kiosk mode.

In kiosk mode, the application opens as a full-screen window. Users cannot open any additional windows and cannot close the application. The connection settings and the resources that users are allowed to access at the back-end, such as Citrix® stores, are predefined in a configuration file. As a consequence, defining applications for the user becomes obsolete.

After system start, the device starts the application defined by the configuration files in kiosk mode, ensuring that the user is directly in the intended work environment and cannot break out.

Version information for kiosk mode support

Configuration via .ini file Configuration via console (GUI)

Citrix Self-Service User interface with extensions	from eLux RP 6.2	from Scout 15 2107 and eLux RP 6 xxxx (not yet specified)
Builtin Browser	From eLux RP 6.5[\^1]	from Scout 15 2107 and eLux RP 6 xxxx (not yet specified)
Configuration file	`kioskmode.ini`	`kioskconfig.ini`

: For earlier versions, use Firefox in kiosk mode.

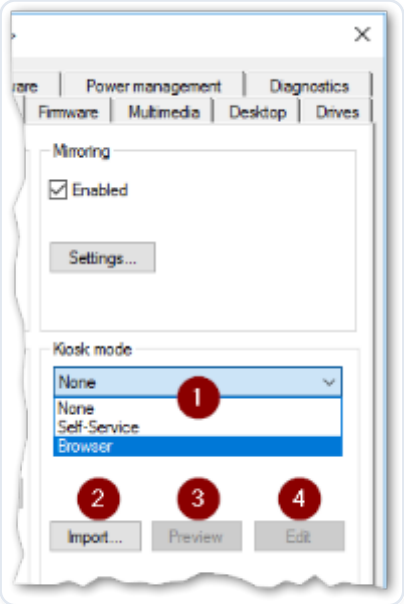
- Setting up kiosk mode via console
- Configuring browser for kiosk mode
- Configuring Citrix Self-Service for kiosk mode
- Citrix Self-Service in kiosk mode (current eLux versions)

- Browser in kiosk mode (current eLux versions)

Setting up kiosk mode via console

from 1Note: Current eLux® versions do not yet support the new settings on the devices.

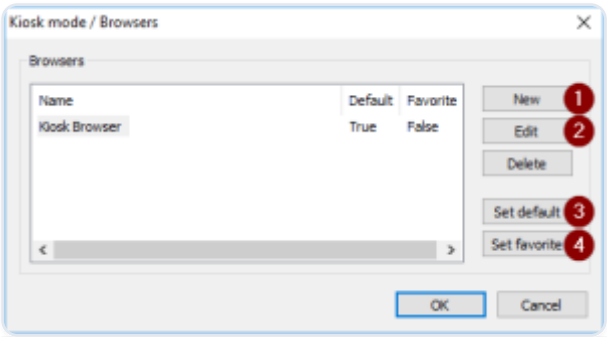
The Scout Console interface allows you to define kiosk mode for Citrix Self-Service (A) as well as for the Builtin Browser (B). The corresponding function is located in the device configuration under **Security**. The kiosk mode can be configured differently depending on the OU.



1	Enable kiosk mode and select type
2	Import existing configuration
3	View configuration
4	Configure kiosk mode for selected type (browser or Citrix)

1. For the relevant devices, open **Device configuration > Security**.
2. Under **Kiosk mode** in the list-field, choose whether you want to activate the kiosk mode for Citrix `Self-Service (A) or Browser (B)`.
3. If you already have a `kioskmode.ini` or `kioskconfig.ini`, click **Import**.
4. Click **Edit**.

   Depending on the selected type, the configuration dialog for Citrix Self-Service stores (A) or for browsers (B) opens.



1	Create new browser
2	Edit selected browser
3	Set selected browser as default
4	Mark selected browser as favorite

5. Create and configure at least one store (A) / browser (B) for kiosk mode.
6. If you have created multiple stores (A) / browsers (B), you can then set one as default.

   On logon, default stores (A) / browsers (B) are pre-selected or started automatically.
7. If you have created multiple stores (A) / browsers (B), you can mark one or more as favorites.

   Favorites are shown to the user on the store selector of the system bar with an asterisk. Favorites for stores can only be defined without **MultiStore**.

Your configuration is written to the `kioskconfig.ini` file. To check it, click the **Preview** button. The next time the server connects to the devices, your selected kiosk mode type (`terminal.ini` entry) will be transferred along with the `kioskconfig.ini`.

(A): On the devices, after start-up, the Citrix Self-Service interface opens in kiosk mode. The configured stores are available to the user. No other applications can be started apart from the Self-Service interface.

(B): On the devices, after start-up, the Builtin browser opens in kiosk mode. The configured functions are available to the user. No other applications can be started apart from the browser.

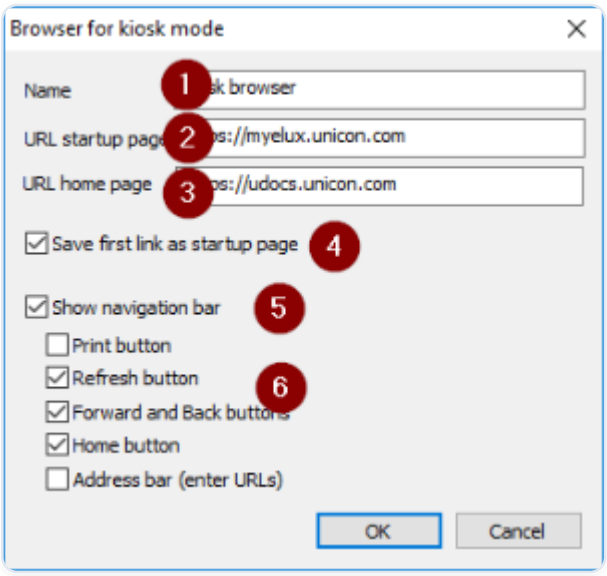
> **Note:**

> For correct use of the kiosk mode, we recommend that you restrict the user rights for the device as required.

Configuring browser for kiosk mode

from 1[The devices support the new settings only from eLux® RP 6 23xx.

1. In **Device configuration > Security**, in the **Kiosk-Modus / Browser** dialog, create a new browser definition. To do so, click **New**.
2. Select and edit the newly created browser definition:



1	Display name of the browser
2	Web page (URL) that opens after starting the browser
3	Web page (URL) that opens when you click **Home**
4	The first URL users load after the home page will be saved as new startup page. How users access the Home page (Home button, link on the startup page, address bar or configured Home page is identical to startup page) is irrelevant. Nevertheless, we recommend that you show the navigation bar with Home button when selecting this option.
5	Unhide navigation bar

6	Elements of the navigation bar to be shown

3. Confirm with **OK**.

Configuring Citrix Self-Service for kiosk mode

from 1[Note: Current eLux® versions do not yet support the new settings on the devices.]

- The eLux package **Citrix Workspace-App for Linux** must be installed on the devices.
- The eLux package **Citrix Extensions 2.x** or later and the included feature package **Self-service wrapper** must be installed on the devices.
- For modifications on the Citrix dialog design, further feature packages must be installed on the devices: **Dialog Extension** and **Self-service dialog themes**.
- This may require modifications of the image definition file on the web server via ELIAS.

1. For Citrix Self-Service, first configure the settings, see below.These apply to all stores.
2. In the **Kiosk mode / Citrix Self-Service** dialog, create a new store.To do so, click **New**.
3. Edit the new store:

OPTION	DESCRIPTION
Name	Display name of the Citrix-Store
Store URL	Web page (URL) that opens immediately after starting the browser
Autostart resources	List of Citrix applications/desktops you want to start automatically after logon
	Currently only available in combination with the **MultiStore** option

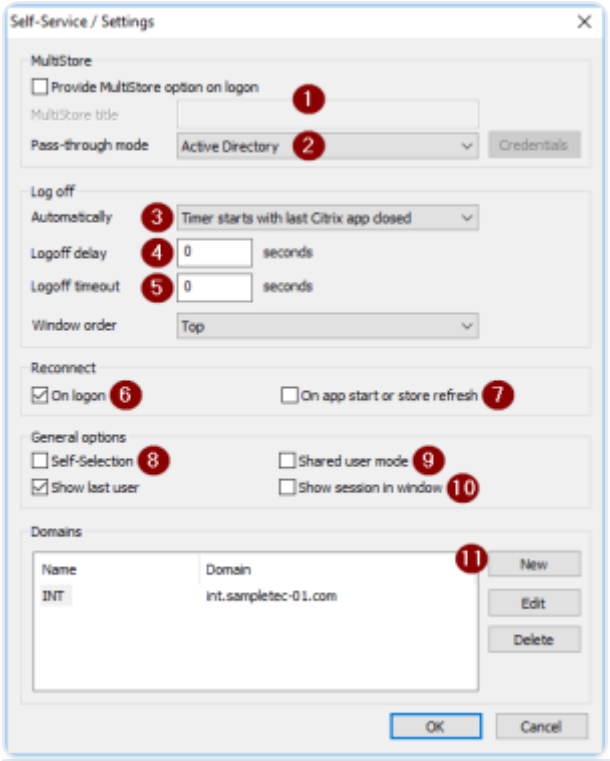
4. Confirm with **OK**.

If multiple stores are defined, users can switch between stores using the **Store** button on the system bar (store selector).Stores marked as favorites are displayed with an asterisk.If **MultiStore**

is configured, users can switch between stores via the logon dialog.For further information on the **MultiStore** option, see below.

SELF-SERVICE SETTINGS FOR KIOSK MODE

- To access the general settings, in the **Kiosk mode / Citrix Self-Service** dialog, click **Settings**.



1	MultiStore: During logon, users can choose between pre-defined stores.
2	Logon can be done without passthrough credentials, via Active Directory or with a defined username+password. The password must be ICA-encrypted.
3	Logoff can be done automatically. Set the timer for it.
4	Delay in seconds when the logoff timer is started
5	Display duration for message on logoff
6	Self-Service tries to reconnect all sessions for a store directly after logon to that store (Citrix Self-Service option).

7	Self-Service tries to reconnect all sessions as soon as an application is started or the store is updated (Citrix Self-Service option).
8	Users are allowed to subscribe to extra applications (Citrix Self-Service option)
9	Use one system account for multiple users (Citrix Self-Service option). The user data are deleted when they close the app or log off. We do not recommend combining this option with **MultiStore**.
10	The session is displayed in windowed mode (instead of full-screen) (Citrix Self-Service option).
11	Define domains
12	Define helpers for **MultiStore**

You no longer have to group stores into environments.

MULTISTORE

The **MultiStore** option allows you to predefine multiple stores - optionally in different domains.Define one of the stores as the default store.Users are then presented with a Citrix Self-Service logon dialog from which they can choose between the preconfigured domains and stores.The default store is pre-selected as the **Home location** with its domain.

1	Configurable title text (**MultiStore title**)
2	Users must log on with their username and password.
3	Predefined domains
4	Predefined stores
5	Users are redirected to your password reset page, to be defined under **Helper**.

If you have defined auto resources for a store, the specified desktops or applications are started automatically after logon.

> **Note:**

> The store selector on the system bar is not available to the users. To switch to another store, users log off and return to the logon dialog.

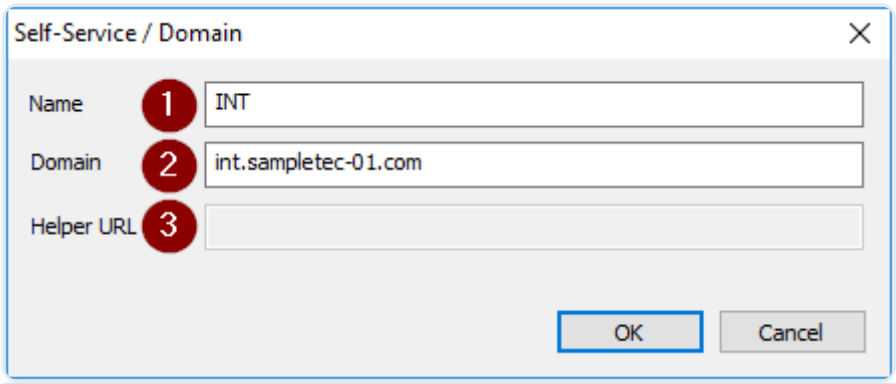
> Note the following for configuration:

- Define one domain as a minimum.
- If you define multiple stores, one store must be set as the default.
- Logon with passthrough is not available.
- We do not recommend combining **MultiStore** with **Shared User Mode**.

DOMAINS

Domain definition is required if you use **MultiStore** or -- without **MultiStore** -- if you configure **Passthrough** for Active Directory.

- In the **Self-Service / Settings** dialog, click **New** to create one or more domains.Then edit the new entries.



1	Display name for domain
2	Domain as FQDN
3	URL for a page that allows resetting passwords (MultiStore only).

HELPER

-only MultiStore --

A helper URL links to an existing password reset page.To access this page, users click a button in the **MultiStore** logon dialog.

You can define one helper URL per domain.This is normally done when you define a domain.

To configure the button for the users and the behavior of the browser with reset page, use the **Helper** dialog.

- In the **Self-Service / Settings** dialog, under **Domains**, click **Helper**.

OPTION	DESCRIPTION
Button text	Button label in the **MultiStore** logon dialog
	Example: `Forgot password?`
	By default, this text is additionally shown in the browser title bar above the reset page.
Timeout	Timeout for launching the Builtin browser and loading the password reset page

OPTION	DESCRIPTION
Options	Optionally, specify additional parameters for the browser start.
	Example: `--title "Reset password"`

If users do not remember their password, they click the relevant button in the **MultiStore** logon dialog.This launches the Builtin browser and loads the helper URL defined for the domain the user is logging on to.

DESIGN OF THE CITRIX DIALOGS

- To change the design of the Citrix dialogs for all Citrix connections, in the **Kiosk mode / Citrix Self-Service** dialog, click **Theme**.

OPTION	DESCRIPTION
Name	Name of the Citrix theme
	Default: `ucselfservice`
Window decoration	The windows are displayed with window decoration.
On hovering...	Background color for list elements on mouse hover (Citrix list selection widgets)
	Default: `#e6f1f7`
Unselected...	Background color for unselected list elements (Citrix list selection widgets)
	Default: `#ffffff`
Selected...	Background color for selected list elements (Citrix list selection widgets)
	Default: `#cce3f0`

Citrix Self-Service in kiosk mode (current eLux® versions)

The Citrix Self-Service user interface with extensions is configured as kiosk mode. All relevant parameters are defined in the `kioskmode.ini` configuration file. Advanced functionality and MultiStore option are also configurable.

Defining Citrix Self-Service UI in kiosk mode

- The eLux package **Citrix Workspace-App for Linux** must be installed on the devices.
- The eLux package **Citrix Extensions 2.x** or later and the included feature package **Self-service wrapper** must be installed on the devices.
- For modifications on the Citrix dialog design, further feature packages must be installed on the devices: **Dialog Extension** and **Self-service dialog themes**.
- This may require modifications of the image definition file on the web server via `ELIAS`.

1. For the relevant devices, open and define the following entry:

File	`/setup/terminal.ini`
Section	Layout
Entry	`KioskMode`
Value	1

   For further information, see Advanced file entries.

   This parameter enables the kiosk mode for the Citrix Self-Service user interface with extensions.

2. Create a text file named `kioskmode.ini` and add the section header `[Parameters]`. Enter the relevant parameters:[1[Default values are displayed in bold]]

PARAMETER	DESCRIPTION
ReconnectOnLogon=true|`false`	Determines whether the Self-Service UI tries to reconnect to all sessions for a given store,

PARAMETER	DESCRIPTION
	immediately after logon to that store (Citrix Self-Service option)
ReconnectOnLaunchOrRefresh=true|`false`	Determines whether the Self-Service UI tries to reconnect to all sessions when an application is launched or the store is refreshed (Citrix Self-Service option)
SharedUserMode=true|`false`	If the Shared User Mode is enabled, the Self-Service UI uses one technical user account for multiple users. The user data are removed from the device when users log off or close the UI. (Citrix Self-Service option) We recommend that you do not combine and
SelfSelection=true|`false`	Is used to disable the search box and the Self-Selection panel (legacy UI). Disabling these UI elements prevents users from subscribing to extra applications.
LogoffMode=`0`|1|2|3	0 = No automatic logoff 1 = Logoff timer is started with logon 2 = Logoff timer is started when the last Citrix app/desktop is closed 3 = Logoff timer is started when the first Citrix app/desktop is opened
LogoffDelay=`<seconds>`	Delay in seconds after the logoff timer is started
LogoffInfoTimeout=`<seconds>`	Shows a message for n seconds during logoff
ShowLastUser=true|`false`	Shows last logged-on username in **User** field
PreLogin=true|`false`	Determines, whether on logon a dropdown list with pre-configured stores is shown (MultiStore). We recommend that you do not combine `SharedUserMode=false` and
PreLoginTitle	Dialog title for the stores list (MultiStore)

PARAMETER	DESCRIPTION
PassThroughMode=`0` |1	0 No pass-through logon data 1 Active Directory UserPassword
Domain=`<Domain display name,domain>`	Each entry contains a domain.

3. To define access to the stores, in the `kioskmode.ini` file, add one or more sections named `[Store<1-N>]` or `[Environment_Store<1-N>]`.

   If you define stores by using `[Environment_Store<N>]`, users can switch between the stores by clicking the **Store** button on the taskbar. The Stores are shown in groups (Environment) as defined.

PARAMETER	DESCRIPTION
Url=`<Store URL>`	URL of the Citrix store
FriendlyName=`<Store display name>`	Display name for the Citrix store
Default=true|`false`	Determines whether this store is displayed as default store
AutostartResources=`<App/Desktop1;App/Desktop2;...>`	List of Citrix applications/desktops to be started automatically after logon (Currently only available with `PreLogin=true`)

`[Environment_Store<1-N>]`.

PARAMETER	DESCRIPTION
Url=`<Store-URL>`	URL of the Citrix store
FriendlyName=`<Display name>`	Display name for the Citrix store
Default=true|`false`	Determines whether this store is displayed as default store
Environment=`<Group name>`	Specifies the group by which the stores are grouped (freely definable character string)

PARAMETER	DESCRIPTION
AutostartResources=`<App/Desktop1;App/Desktop2;...>`	List of Citrix applications/desktops to start automatically after logon (Currently only available with `PreLogin=true`)

4. To change the design of the Citrix dialogs for all Citrix connections, in the `kioskmode.ini` file, add a section named `[Theme]`:

PARAMETER	DESCRIPTION
ThemeName=`<Themes>`	Name of the Citrix theme. Default: `ucselfservice`
Decorated=true|`false`	Determines whether the windows are shown with window decoration
ColorHover=`<RGB color code>`	Background color for list elements on mouse hover (Citrix list selection widgets). Default: `#e6f1f7`
ColorUnselected=`<RGB color code>`	Background color for unselected list items (Citrix list selection widgets). Default: `#ffffff`
ColorSelected=`<RGB color code>`	Background color for selected list elements (Citrix list selection widgets). Default: `#cce3f0`

5. Transfer the `kioskmode.ini` file to the devices to `/setup/kioskmode.ini`. To do so, use the Scout feature **Files configured for transfer**. For further information, see Files configured for transfer.

The `terminal.ini` entry along with the `kioskmode.ini` file on a device will cause the device to open the Citrix Self-Service interface in kiosk mode after start-up. All configured stores are available to the user. No other applications can be started apart from the self-service interface.

> **Note:**

If more than one store is configured, users can switch between the stores by clicking the **Store** button on the taskbar while pressing `Shift`.

EXAMPLE FOR

```
[Parameters]

ReconnectOnLogon=true

ReconnectOnLaunchOrRefresh=true

SharedUserMode=true

SelfSelection=false

ShowLastUser=true

LogoffMode=3

LogoffDelay=10



[Store1]

Url=<https://xd7a/int.sampletec-01.com/Citrix/Store/discovery>

FriendlyName=XenApp A



[Environment_Store1]

Url=<https://xd7b./int.sampletec-01.com/Citrix/Store/discovery>

FriendlyName=XenApp B

Default=true

Environment=PROD
```

```
[Environment_Store2]

Url=<https://xd7c./int.sampletec-01.com/Citrix/Store/discovery>

FriendlyName=XenApp C

Default=false

Environment=INT



[Theme]

ThemeName=ucselfservice

Decorated=false

ColorHover=#b0b0b0

ColorUnselected=#a0a0a0

ColorSelected=#c0c0c0
```

Browser in kiosk mode (current eLux® versions)

For kiosk mode, the Builtin Browser is used. In kiosk mode, the browser is started in full-screen mode and with limited user rights. The user cannot open other windows and cannot exit the browser. Note that the Builtin Browser if defined as a browser application is not run in kiosk mode even if address and navigation bar are hidden.

> **Note:**

> Firefox could be run in kiosk mode up to version ESR 52.8. Mozilla supports kiosk mode again starting with version 71.0, but without configuration options. For further information, see Kiosk mode for Firefox.

> Defining Builtin Browser in kiosk mode[1[Kiosk mode via console (GUI) will be supported by future eLux versions]]

1. For the relevant devices, open and define the following entry:

File	`/setup/terminal.ini`
Section	Layout
Entry	`KioskMode`
Value	2

   For further information, see Advanced file entries.

   This parameter enables the kiosk mode for the browser application.

2. Create a text file named `kioskmode.ini` and add the section header `[Browser1]`.
3. Below, enter the relevant parameters:[2[Default values are displayed in bold]]

PARAMETER	DESCRIPTION
Url	Web page (URL) that opens after the browser is started
Homepage=`<URL of homepage>`	Web page (URL) that opens when users click **Home**
SaveFirstLink = `true\false`	If **true**, the first URL loaded when coming from the startup page will be saved as the new startup page.
Navbar=`true\false`	The navigation bar will be shown.
NavbarPrint=`true\false`	The **Print** button will be shown on the navigation bar.
NavbarForward=`true\false`	The **Forward** button will be shown on the navigation bar.
NavbarBackward=`true\false`	The **Backward** button will be shown on the navigation bar.
NavbarHome=`true\false`	The **Home** button will be shown on the navigation bar.

PARAMETER	DESCRIPTION
NavbarUrl=`true\false`	The address bar will be shown.
NavbarRefresh= `true\false`	The **Refresh** button will be shown on the navigation bar.
Favorite=`true\false`	If **true**, the object is marked as a favorite.
FriendlyName = `<Anzeigename>`	Display name for the user

4. Transfer the `kioskmode.ini` file to the devices to `/setup/kioskmode.ini`. To do so, use the Scout feature **Files configured for transfer**. For further information, see Files configured for transfer.

The `terminal.ini` entry along with the `kioskmode.ini` file on the device will cause the Builtin-Browser to open in kiosk mode after system start-up. The configured functions are available to the user. No applications other than the browser can be started.

> **Note:**

> For correct use of the kiosk mode, we recommend that you restrict the user rights for the device as required.

DEFINING DIFFERENT WEB PAGES FOR DEVICES/OUS

If you want your devices to start with different browser startup pages (`Url`) or have different home pages (`Homepage`), you can parameterize the web pages using environment variables. Carry out the steps described above with the following differences:

1. In the `kioskmode.ini`, in the section, for the first and/or second value, set a variable. Example: `Url=$URL1` `Homepage=$URL2`
2. Define the variables used in the `kioskmode.ini` as environment variables for the relevant devices (Advanced device configuration). Example: `URL1=https://www.unicon.com` `URL2=https://myelux.com`

   For further information, see Environment variables.

Local web sites

from eLux® RP 6 2110

On the devices, browser applications can be configured to run without a network connection. To do so, the required HTML pages and scripts are transferred to the devices and the Builtin browser is configured accordingly.This allows you to provide users with a web application that starts automatically and serves as an entry point.

- Define all allowed links from here as HTML links in your scripts.
- Configure all allowed actions via the browser properties.

Combined with kiosk mode, users are restricted to a specific start application and use it as a kind of landing page.

Providing local web sites

1. Pack all files needed for displaying the local web pages into a file named .
   - Startup page `<start>.html`, example: `index.html`
   - Additional HTML pages and Javascript files (optional)
   - Stylesheet (optional)

   > **Note:**
   >
   > The file name for the archive is predefined and case-sensitive `(landingPage.zip)`. The archive with this name is also required for a single HTML page.

2. Transfer the file to the devices to To do so, use the **Files configured for transfer** feature. For further information, see Files configured for transfer.

   > **Note:**
   >
   > Note case-sensitivity (landingPage.zip).
   >
   > The next time the Builtin browser is launched on the devices, the archive will be unpacked to `/tmp/browser/landingPage/`

3. To make the web pages available in kiosk mode, enable and configure the kiosk mode. When doing so, set the browser startup page to `file:///tmp/browser/landingPage/<start>.html`

   For further information, see Applications in kiosk mode.

4. To make the web pages available without kiosk mode, define a browser application with the following properties:

   - Browser type: `Builtin`
   - Startup page: `file:///tmp/browser/landingPage/<start>.html`

   Optionally, select **Start Automatically** and, under **Advanced**, specify navigation elements you want to display

At the next browser start (if configured: automatically after device restart) the startup page is displayed. From here, users can connect to the link destinations you have configured via the local web pages.

Even in kiosk mode, you are free to define multiple browser applications. Each browser application has its individual startup page, and one of the browser applications is defined as the default. Users can then switch between the applications using the selector on the system bar.

Troubleshooting application definition

ERROR / PROBLEM	REASON	SOLUTION
Missing firmware	The required software is not installed on the device	Install the software on the device. For further information, see Creating an IDF in the ELIAS guide and Firmware update.
Doubled names	Two applications have the same name. This causes conflicts because applications are identified by their names.	Use unique names.
Hidden application cannot be executed	Applications are invisible for the user when they run in hidden mode. This option is available for applications of the **custom** type.	Enable the option **Start automatically** or **Application restart** to start hidden applications on start or to run them non-stop, respectively.

ERROR / PROBLEM	REASON	SOLUTION
Problems with certificates in combination with [VMware] server	Server problem occurred	Create a server certificate in the **Windows-CA** with **FQDN**.
	After successful installation, the VMware server uses a self-signed certificate. If a device is configured correctly, it will not accept. The reason is that the **FQDN** (fully qualified domain name) is mandatory for server certificates.	
		If you use `mmc`: Create a server certificate using the Snap-In `Certificates (Local computer)`.
		The key must be exportable.
		The display name of the server must be `vdm`. The name must be unique in the certificate store **Local computer / Personal**.
COM port redirection in RDP session does not work	Communication errors such as high latencies in the network between your serial device and the virtual desktop do not allow serial communication.	Use the **permissive** mode for the RDP application. This parameter causes communication errors to be downgraded to warnings, and communication becomes more tolerant of timeouts.
		Define a free parameter in your RDP application definition with the **permissive** option.
		Example
		`FreeRDPParams=/serial:COM1,/dev/ttyS0,Serial,permissive`
		For further information, see Defining free application parameters.

Third party software

To install additional applications on the devices, carry out the following steps:

1. From our myelux.com portal, for the relevant eLux version, download the specified software package.
2. In ELIAS, import the software package into your container. For further information, see Importing packages into a container in the **ELIAS 18** guide.
3. In ELIAS, add the package to your image in the **ELIAS 18** guide.
4. For the relevant devices, perform a firmware update to the modified image. For further information, see Firmware update.

   The software package is installed on the devices.
5. Configure the software in the back-end environment.

- Avaya Equinox
- Cisco JVDI
- deviceTRUST
- DriveLock
- Grundig Citrix Extensions
- HDX RealTime Media Engine
- JabraXpress Device Updater
- Microsoft Teams
- Nutanix Frame
- Olympus Dictation
- Philips Speech
- SecMaker
- Zoom

Avaya Equinox

Soft phone application providing access to Unified Communications (UC) and Over the Top (OTT) services

Package name: **Avaya Equinox VDI Client**

Cisco JVDI

Jabber Softphone for VDI JVDI[1[formerly VXME]] extends the Cisco collaboration experience to virtual deployments. With supported versions of Cisco Jabber for Windows, users can send and receive phone calls on their hosted virtual desktops (HVD). The JVDI software routes all audio and video streams directly from one device to another, or to a phone, without going through the HVD.

Package names: Cisco JVDI Client and **Utilities for Cisco JVDI Client**

- Follow the **Cisco Deployment and Installation Workflow** on the Cisco website in order to configure the JVDI system environment.

deviceTRUST®

Dynamic context awareness, allowing users to access their corporate workspace from any location on any device while meeting IT governance requirements

Package name: **deviceTRUST Contextualizing IT**

DriveLock

DriveLock provides endpoint security for USB interfaces on the devices.

Package name: **DriveLock**

Grundig Citrix Extensions

Digital dictation solution by Grundig Business Systems

Package name: **Grundig Citrix Extensions**

HDX™ RealTime Media Engine

The HDX RealTime Media Engine (RTME) enables better audio and video quality for VOIP and video chat.

Package name: **Citrix HDX RTME**

- Configure Microsoft Lync or Skype for Business in the back-end environment.

JabraXpress Device Updater

Solution installed on an end user's device that governs Jabra device configurations such as firmware versions and device settings

A detailed description of how to use the Jabra Xpress Device Updater can be found in the documentation of Jabra Xpress for Linux: `<https://jabraxpress.jabra.com/Downloadables/Linux/UserGuide.pdf>`

Package name: **JabraXpress Device Updater**

Microsoft Teams

Enterprise video conferencing with real-time messaging and content sharing

Package name: **Microsoft Teams Client**

Microsoft Teams Client for Linux is a local application on eLux.

Parameter for the application definition: `teams`

Nutanix Frame

Cloud- hosted Desktop as a Service (DaaS) that empowers any organization to deliver and manage their desktops via a single console for seamless control and administration, providing a true hybrid experience.

Package name: **Nutanix Frame Client**

Nutanix Frame Client for Linux is a local application on eLux.

Parameter for the application definition: `Frame` Parameter with URL: `Frame -url=console.nutanix.com`

Olympus Dictation

Digital dictation solution by Olympus

Package name: **Olympus Dictation Drivers**

Philips Speech

Digital dictation solution from Philips Speech Processing Solutions

Package name: **Philips Speech Drivers**

SecMaker

SecMaker Net iD Enterprise is a middleware supporting SSL 3.0/TLS 1.0 (client identification), PKCS #7 (digital signatures), and PKCS #10 (certificate requests). To use SecMaker Net iD 6.x on the device, a separate license is required.

Package name:**SecMaker NetID**

- Use the Scout feature **Files** to transfer the license file to the devices to `/setup/iid/netidlicense.lic`.

For further information, see Files configured for transfer in the **Scout** guide.

> **Note:**
>
> The license agreement (EULA) is available on the device under `/etc/iid/SecMaker License and Support Conditions 20150202.pdf`

Zoom

Enterprise video conferencing with real-time messaging and content sharing

Package name: **Zoom Client for Linux**

Zoom Client for Linux is a local application on eLux® that enables peer-to-peer connections between connected devices.

Package name: **Zoom Citrix Plugin**

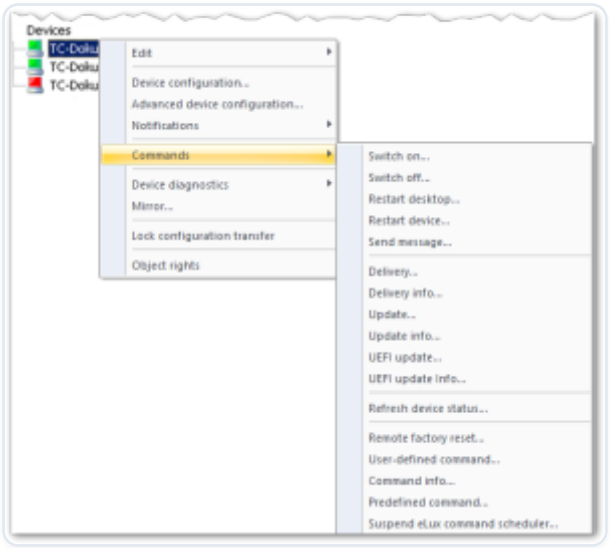
With the Zoom Citrix plugin, ZoomVDI can be used in a virtual desktop infrastructure with a Citrix solution. Peer-to-peer connections are not supported in this scenario.

Client remote management by commands

Administrators can use Scout commands to change the status of the devices, perform updates and send messages. The commands can be executed immediately or can be scheduled to be run once or periodically.

You can apply Scout commands on individual devices, on OUs and on Dynamic Device Groups.

In addition, the context menu of an individual device provides the commands for device diagnostics and for mirroring.



By default, before Scout commands are sent to the selected devices, their current device status is determined. Devices that are switched off are then switched on, if possible, and after command execution are switched off again.

In contrast to commands processed by the eLux® Command Scheduler, commands cannot be "stored" and executed at a later time when a device is up again.

- Available commands
- Executing commands
- Scheduling commands
- Command options
- Command results and update information per device
- Command history
- Factory reset command
- User-defined Commands
- Creating predefined commands
- Defining templates for standard commands
- eLux Command Scheduler

Available commands

The context menu of a device, OU or Dynamic Client Group provides the following **Command** options resulting in the **Execute/Schedule command** dialog for further configuration:

COMMAND	DESCRIPTION
Switch on	Switches on the devices
Switch off	Switches off the devices
Restart desktop	Restarts the eLux® interface.
Restart device	Restarts the devices
Send message	Sends a message to the devices
Delivery	Delivers software for a firmware update
Update	Performs a firmware update
UEFI update	Performs an update of the UEFI firmware
	For further information, see UEFI update in analogy to firmware update in the `UEFI/BIOS` update guide.
Refresh device status	Requests the current device status and refreshes the status of the devices in the tree view
Remote factory reset	Sets the devices back to initial state
	The configuration is deleted, the IDF remains.
	For further information, see Factory reset command.
User-defined command	Enter a user-defined command that will be sent to the devices.
	For further information, see User-defined Commands.
Predefined command	Provides user-defined commands that have been predefined globally. For further information, see Creating predefined commands.

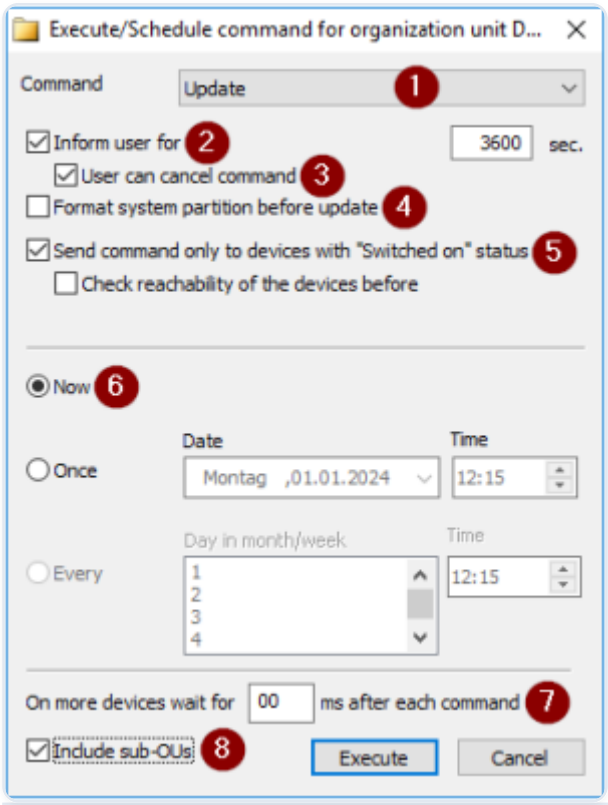
COMMAND	DESCRIPTION
Suspend eLux command scheduler	If recurring eLux commands that are scheduled locally on the device by the eLux Command Scheduler (cron jobs) have been defined by the administrator, the execution of these commands is temporarily stopped until the next restart of the device.
Configuration run. (not available for individual devices)	Prepares the configuration data for an OU or Dynamic Client Group. For further information, see Configuration run. This command is not available for an individual device.

For further information on options you can add to the respective commands see Command options.

The following options open the relevant log file:

COMMAND	DESCRIPTION
Delivery Info	Opens the log file of the latest software delivery
Update Info	Opens the log file of the latest firmware update
Command Info	Opens the log file of the latest user-defined command

Executing commands



1	Allows to switch to other commands
2	Users are informed before command execution, see Command options.
3	Users are allowed to prevent the command execution.
4	Command-specific option
5	Only switched-on devices[1[[ ]from Scout 15 2204]]
6	Time of command execution, see Scheduling commands
7	Delay of execution between devices
8	Apply command additionally on devices in subordinate OUs

1. For the relevant device, OU or Dynamic Client Group, from the context menu, choose **Commands**.
2. From the sub-menu, choose a command.

   The **Command** dialog opens. The options shown depend on the selected command.
3. Check the target devices to which the command is sent: To show the complete title with relevant device or OU, move the mouse pointer over the title bar.
4. Edit the relevant options. For further information, see Command options and Scheduling commands.
5. Confirm with **Execute** or **Schedule**.

The command is executed at the specified time. Depending on the command, you are asked to confirm.

Scheduling commands

The execution of most commands can be scheduled for a specified time instead of being executed immediately. Many commands such as firmware updates can also be scheduled on a recurring monthly or weekly basis.

1. For the relevant device, OU or Dynamic Client Group, from the context menu, choose **Commands**.
2. From the sub-menu, choose a command.
3. To schedule a command to be executed once, select **Once**. Then select the date and time.
4. To schedule repeated execution, select **Every**. Then select the day of the month or week and the time.
5. Edit the other options of the **Command** dialog.
6. Click **Execute**.

The command is executed at the defined time. All scheduled commands (once and repeating) can be displayed, modified and deleted under **View > Schedule...**

Command options

The options you can use to execute a command vary to some extent depending on the selected command.

General and common options

OPTION	DESCRIPTION
Command	The selected command is displayed. , The list-field allows switching to other commands.
Inform user	Users are informed by a notification before the command is executed. , Specify a time period in seconds for displaying the notification. The command will be executed after the time period has expired. , If the time period is set to [0], the notification will be displayed and the system waits until the user confirms the command execution. , For firmware updates or UEFI updates, additional deferment options can be configured in the device configuration. For further information, see Update deferment by users.
User can cancel	Users are allowed to prevent the command execution.
Send only to devices with **Switched-on** status [from Scout 15 2204]	Only for **Update, Delivery, User-defined and Pre-defined commands** , The command is sent only to devices shown with **Switched-on** status. Switched-off devices are skipped. The actual device status is not determined before command execution. , If the **Check reachability** option is also selected, a **ping** command is used to determine the current status of the relevant devices before the command is issued. The command is only sent to the responding devices. This avoids timeouts that can be caused by devices that are not ready for operation despite showing a **Switched-on** status. , Default (both options disabled): The device status of the relevant devices is determined before the command is executed. If possible, switched-off devices are switched on for command execution and then switched off again. If a large number of devices is affected, we recommend that you select the

OPTION	DESCRIPTION
	above options instead. This allows for a faster and smoother operation.
Run with system rights	Some commands require system rights which are checked before execution.
Now/Once/Every	Time of execution, see Scheduling commands
On more devices wait	Delay command execution between individual devices
Include sub-OUs	The command is additionally applied to all devices in subordinate OUs.

Command-specific options

OPTION	DESCRIPTION
Send message /	Select the message to send from the list or enter your message text. To format your message text, use HTML tags.
Message	Optionally edit the title of the message.
Title	Optionally specify the display duration in seconds. With 0 seconds, the system message will be shown until the user clicks **OK**.
Visible	
	Selected or all previously sent message texts can be deleted from the list via button.
Delivery / Clean update partition before delivery	Before writing to the flash memory, the update partition is cleaned up.
	All files of the current software image must be re-transferred from the web server. For further information, see Performing deliveries via command.
Update / Format system partition before update	Before the software is installed, the system partition of the flash memory is formatted.

OPTION	DESCRIPTION
	For the installation of the firmware update, all files of the current software image must be re-transferred from the web server. For further information, see Performing updates via command.
UEFI update / Overwrite identical or higher version	The UEFI system is always overwritten. This is necessary for a downgrade, for example. For further information see Performing UEFI updates via Scout command.
Remote factory reset /	See Factory reset command
Retain local configuration	
Delete **Scout Server** address	
User-defined command	Below the command name, select a command from the list or enter a command. For further information see User-defined Commands.
Predefined command / Selection	Select a predefined command from the list. The function requires that commands have been predefined centrally, see Creating predefined commands.

Command results and update information per device

Feedback on performed Update, Delivery and User-defined commands is available for individual devices

- in the **Properties** window
- in the **Update info/Command info/UEFI update info** dialog (via device context menu)

> **Note:**

> In addition, all executed commands for all devices are displayed in the **Command history** window. For further information, see Command history.

> All processes are recorded, even if the commands turn out to be obsolete and haven't been run or have been aborted. Successfully completed commands are shown with a green symbol.

Viewing command results for an individual device

The following instructions are related to **Update** commands. Viewing results of a UEFI update, delivery or user-defined command is done accordingly.

1. To show the **Properties** window, click **View > Window > Properties**.

   The **Properties** window is shown permanently in the upper right. For the selected device some properties are shown. Properties can be shown or hidden by using the icon.

2. Select the relevant device in the tree view.

   In the **Properties** window, in the Update section, the following fields are provided:

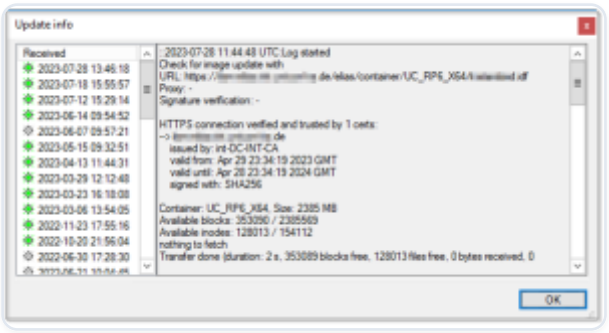
Image	Current image
Update time	Exact point in time of the latest update
Update status	Current status such as , or
Update provider	Origin of software packages (web server or proxy)
Update size	Size of the transferred packages in compressed format

> **Note:**
>
> The properties **Update provider** and **Update size** are evaluated for updates, but not for the migration to a major version or a downgrade.

3. Double-click the term **Update status** or click ... at the end of the line.

   The **Update Info** window is displayed. On the left, you can see all updates that have been processed, aborted or not processed because the image had been up-to-date. For a selected

update command, you can view all logged data on the right side, among them the installed software packages.



Command Info

Information on the last command or update of a device can also be viewed by using the device context menu in the tree view.

- From the context menu, choose **Commands > Commando Info** or **Delivery Info, Update Info, UEFI update Info**.

Scout Report Generator

Use the following fields for your evaluations:

Command name	Executed command for user-defined commands
Command time	Time stamp of command execution
(UEFI) update time	
Delivery time	
Command result	Successful / Failed Not available
Update state	
Delivery state	Further values are available for Update and Delivery commands.
Command output[1[[ ]from Scout 15 2204]]	Only for user-defined commands with specific results

	Note that the **Command result** may be successful for many commands while there is no **Command output**.

For Update, UEFI update and Delivery commands, further fields are provided corresponding to the fields in the **Properties** window.

Command history

All of the executed **Update, Delivery, and User-defined** commands can be viewed in the command history. When calling the command history, the object rights of the administrator management are respected.

- Click **View > Command history**

The **Command history** window opens and displays one job (command for 1 to n devices) per line providing the following information:

OPTION	DESCRIPTION
Type	Type of object the command is applied to. This can be an individual device, an OU with sub units (OU+), an OU without sub units (OU) or a Dynamic Client Group.
Name	Object name (name of device, OU or Dynamic Client Group)
Command	Executed command (Update, Delivery or User-defined command)
Devices	Number of devices concerned
Start	Date and time of sending command to the devices / starting time
End	Date and time of sending command to the devices / ending time
	The ending time of a job is reached when either the devices report back `Successful` or `Failed` or when the timeout of 5 minutes

OPTION	DESCRIPTION
	for feedback is passed. If the administrator terminates a job, the ending time is defined by the terminating time.
Successful	Number of devices that have successfully processed the command
Failed	Number of devices that have reported failure during command processing
Timeout	Number of devices that haven't reported feedback within the defined time period of 5 minutes
Progress %	Progress of command processing in percent, across all concerned devices
Administrator	Administrator who ran the command

Apply the following options to the job list:

OPTION	ACTION
Refresh	Press `F5`.
Sort table rows	Click the column title by which you want to sort.
	A first click sorts the jobs ascending, and the second one sorts descending.
	To reproduce the default sorting order, click `F5`.

Apply the following options to a selected job:

OPTION	ACTION
View details	Click **Details**.

OPTION	ACTION
	The **Command details** window displays all processing details of the concerned devices. Along with starting and ending times you can find the current status and the command processing result for each device.
Search object in **Scout** tree view	Right-click an object name, and then click **Find in tree view**.
	The first result is selected in the tree view.
Terminate running job	Select the running job, and then click **Terminate**.
	A command terminating request is sent to the **Scout Server** and the transmission of the command to the devices is stopped.

Factory reset command

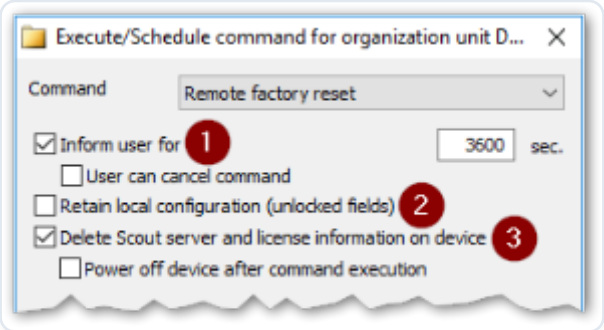
Use the **Remote factory reset** command to reset the configuration of the relevant devices to their initial state. The local device configuration and local application definitions will be deleted.

The following data are retained:

- The installed image with all software packages (firmware)
- Connection data to the Scout Server including the server address and OU ID (default)
- License information (default)

After the device has received the command, the current device configuration and local application definitions are deleted. By default, after a restart, the device reconnects to its and obtains the configuration of the OU to which it is assigned.

Additional options of the Factory reset command



1	Users are informed before command execution, see Command options.
2	Prevent local device configuration from being overridden
3	All configuration data on the device will be deleted. The device then loses connection to its Scout Server. Optionally define to switch off the device afterwards.

OPTION	DESCRIPTION
Retain local configuration (unlocked fields)	User-defined values of the local device configuration in unlocked fields are retained. This only applies to fields that the user is allowed to edit. , -- only available if allowed in **Advanced options > Devices** -- , For further information, see Supporting local configuration.
Delete Scout Server and license information on device	In addition to the configuration data, the following data are deleted
	• Address of the Scout Server machine
	• License information stored on device license lease[1[[ ]from Scout 15 2302 (previously available as a separate

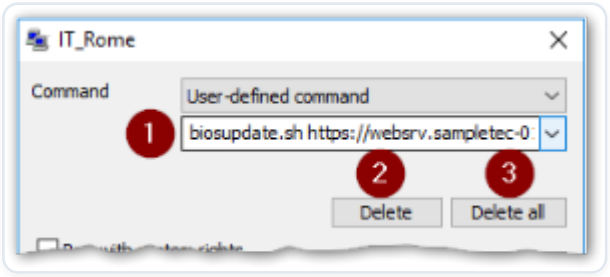
OPTION	DESCRIPTION
	option)]], This option can be used for the resale of devices, for example.
	• Certificates in [/setup/cacerts]
	• WLAN configuration
	• 802.1X configuration
	• Private key of SCEP client certificate stored in a TPM 2.0 module
	An encrypted setup partition (TPM 2.0) will be decrypted. , If you perform the **Remote factory reset** command with this option selected, it corresponds to a local **Factory reset** triggered by the relevant eLux command in the eLux **Command panel** on the device. For further information, see Resetting devices to factory status in the **eLux RP** guide.
Power off device after command execution[2[[ ]from Scout 15 2302 (previously done automatically after deleting the license information)]]	-- only available with **Delete Scout Server and license information on device** --
	The device is shut down and remains switched off.

Note:

The **Remote factory reset** affects the device's setup partition. The system partition with the installed firmware, in contrast, is `not` affected by the factory reset and is only changed using the **Update** command with the option **Format system partition before update**.

User-defined Commands

Authorized administrators may issue individual commands to the devices. A user-defined command can be executed with system rights so that far-reaching operations can be performed. The correct syntax is particularly important here.



1. Select a command from the list (recently used commands) or enter it.
2. Delete the selected command.
3. Delete all user-defined commands.

> **Important:**

> Always test a user-defined command in a test environment before using it productively.

> User-defined commands can be helpful for tasks such as the following:

- Perform a special UEFI/BIOS update
- Update a license lease
- Wipe data disks

> **Note:**

> After having executed a user-defined command, after a time span of 30 seconds, you can run the next user-defined command or update command.

- Wiping the data disk

Wiping the data disk

-- from --

To permanently erase all data of devices from their flash memory, a script is available that you can apply as a user-defined command. The script overwrites all partitions of the primary disk with zeros.

1. Make sure that the devices have the required operating system version.
2. For the relevant device, OU or Dynamic Client Group, from the context menu, choose **Commands > User-defined command**.
3. To run the script, below **User-defined command**, enter the following command: **wipe-device really-wipe**

   The script must be run with the **really-wipe** parameter to prevent accidental execution and wiping of the disk.
4. Select **Run with system rights**.
5. Click **Execute**.

The relevant devices confirm the command execution (last contact to the Scout Server) and shut down. The devices then restart one more time with the core installer, which zeros all partitions of the disk. After that, the devices shut down finally.

Creating predefined commands

User-defined commands can be pre-defined and provided globally to administrators via **Commands > Predefined Command** For example, you can pre-define scripts for the UEFI/BIOS update of particular hardware or a script to renew the license lease of eLux RP 6 devices.

These commands are then available via the **Command** dialog and can be used by authorized administrators. To create a predefined command:

1. On the Scout menu, click **Options > Advanced options > Predefined commands**.
2. Click **Add**.
3. For your new entry, edit the following options:

OPTION	DESCRIPTION
Name	The command name is shown to administrators in the **Commands** dialog. The command itself is not shown.
Command	Syntax for the command to be executed. To enable administrators to set individual values

OPTION	DESCRIPTION
	when executing a predefined command, you can use variables. The values for each variable are prompted when the administrator selects the predefined command in the Command dialog. Example: `ls -%PARAM% %FILES%`. Correct spelling of variables: Free variable name surrounded by percent signs `%variable%`. The variable name may contain upper and lower case letters, numbers, and special characters. To use a percent sign in the command definition outside the variable name, type `%%`.
System	System rights are required to execute the command.
Activated	The command is displayed in the list-field for predefined commands.
Admins	A predefined command is shared with all administrators by default. To restrict to specific administrators or groups, click the cell under Admins and then select the desired administrators/groups from the list. Note that in the default case (no restrictions), the list of administrators displayed does not contain a selection. The command is nevertheless enabled for all administrators.

4. Confirm with **Apply** and **OK**.

You will find all activated predefined commands under in the relevant list-field. They can be applied on individual devices, on OUs and on .

For authorized administrators, the name of the command is displayed instead of the command itself.

- in the **Command** dialog
- in the device properties window under **Command line**

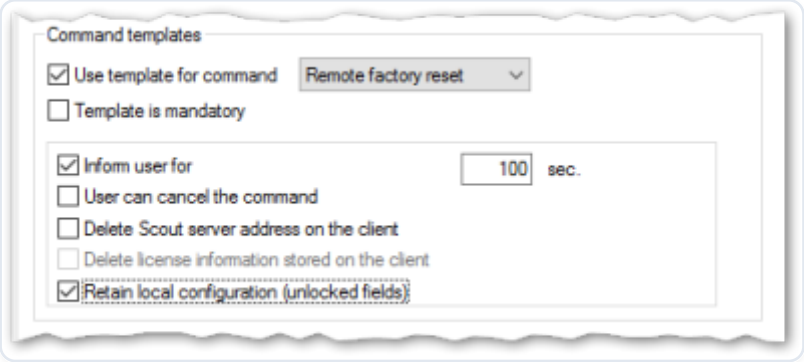
- in the **Command Info** window (command history for relevant device)

Defining templates for standard commands

For standard commands such as **Restart** or **Update**, you can define preferences via templates.

1. On the Scout menu, click **Options > Advanced options > Predefined commands**.
2. Under **Command templates**, select **Use template for command** and then, in the list-field, select the relevant command.
3. To define the preferences as mandatory, select **Template is mandatory**.

   If a template is not mandatory, the operative administrators can overwrite the specified default values.
4. Specify the values for the selected command.



A template may contain command-specific preferences as well as general preferences such as

- Inform user for x seconds
- User can cancel command

> **Note:**
>
> Templates for standard commands are only available in the Scout Console but not in the Scout Dashboard.

eLux® Command Scheduler

-- from Scout 15 2103 --

Recurring commands can be defined so that they are scheduled and executed locally on the device by the eLux Command Scheduler. Like cron jobs, these commands are initiated by the

device at defined points in time, whereas Scout commands are initiated on the server side. The time of execution depends on the time zone in which the device is located.

The eLux Command Scheduler can process the following commands:

- Update firmware
- Synchronize device configuration
- Restart device
- Shut down device
- User-defined command

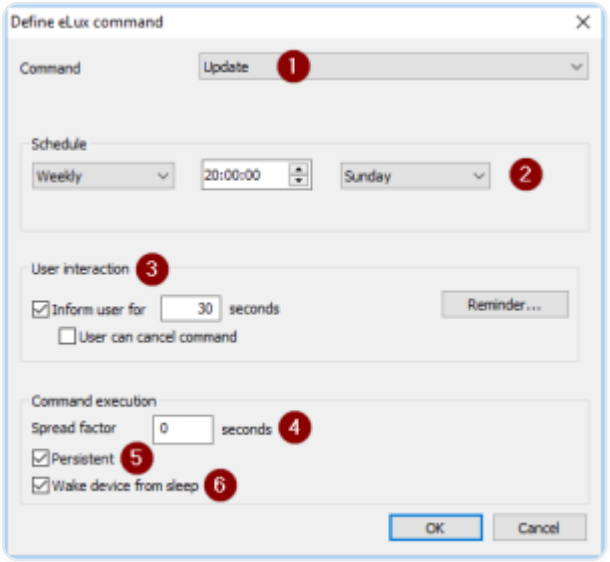
The commands for the eLux Command Scheduler may be defined either on the top level in the **Advanced options** or in the **Advanced device configuration** of an OU. For each OU, you can define whether you want the jobs of the eLux Command Scheduler defined and scheduled above to be applied or not.

- Defining eLux Commands
- Deferment options for users
- Suspending eLux Command Scheduler
- Command definition parameters

Defining eLux® Commands

-- from --

The eLux Command Scheduler is configured in the Scout Console.



1	Select command
2	Configure periodic scheduling
3	Configure user interaction and deferment options
4	Generate random delays between devices
5	Missed jobs are caught up
6	Devices are woken up from suspend to execute jobs

1. Open **Options > Advanced options** or the **Advanced device configuration** of an OU. Then choose the **eLux Command Scheduler** tab.
2. To define a new command, click **Add**.
3. In the **Define eLux Command** dialog, edit the following fields:

OPTION	DESCRIPTION
Command	Type of command
	To execute **user-defined commands**, system rights are required. Therefore, include the password.
Schedule	Periodic scheduling of the command
	If the provided options are still too few, in the , you can define a string using the systemd syntax. For further information, see Command definition parameters.
User interaction	Determine whether and how long you want to inform the users before the command is executed, and whether they are allowed to defer it. For further information, see Deferment options for users.

OPTION	DESCRIPTION
Spread factor	Time span in which random delays are generated to prevent simultaneous execution on many devices
	Example 600 seconds: If the execution time is 6:00 pm, a randomly generated time value for the execution of the command per device, which may be a maximum of 600 seconds, is added to the time 6:00 pm. The command is therefore executed between 6:00 and 6:10 for all devices.
	Default:
Persistent	Missed command executions (jobs) are immediately caught up after the next device start. If multiple repetitions of the same command were scheduled, the command is repeated once.
Wake device from sleep	Devices will be woken up from sleep mode (suspend) to execute scheduled commands.

4. Confirm with **OK** and **Apply**.

   The scheduled command is displayed in the **eLux Command Scheduler** tab of the **Advanced Options** and in the **eLux Command Scheduler** tab of the **Advanced Device Configuration** for all OUs. Creating and editing is only allowed in the **Advanced Options**.

5. Define which OUs you want (not) to inherit the jobs of the eLux Command Scheduler. By default, the settings (jobs) in are inherited from the next higher instance.

The scheduled command is executed by all devices whose OU inherits the jobs of the eLux Command Scheduler at the next due point in time.

> **Note:**

> To view information on the status and the next execution of a command, on the device, use `systemctl status <[jobId]>.service` and `systemctl status <[jobId]>.timer`.

Deferment options for users

Similar to Scout commands you can also define for the command definitions of the eLux® Command Scheduler whether the users are informed before the command is executed and whether they are allowed to cancel or defer it.

OPTION	DESCRIPTION
Inform user	The user is informed by a notification before the command is executed.
	Specify the time period in seconds for displaying the notification. The command is subsequently executed.
	If the value is 0, the notification will be displayed until the user confirms the execution of the command via button.
User can cancel	The user is allowed to prevent the execution of the command via button.
Reminder > Number of allowed deferments	Define how often the user is allowed to postpone.
Reminder > Delays until next reminder	Select one or more time periods users can choose to defer.

For further information, see also User information before update

Suspending eLux® Command Scheduler

The recurring execution of defined commands by the eLux Command Scheduler can be temporarily suspended from the . To do so, the administrator sends a command to the relevant devices, which stops the eLux Command Scheduler until the next restart of the devices. After the next device restart, the eLux Command Scheduler continues to process the scheduled commands as planned.

1. For the relevant device, OU or Dynamic Client Group, from the context menu, choose **Commands > Suspend eLux Command Scheduler**.
2. Click **Execute**.

The execution of all commands scheduled by the eLux Command Scheduler is suspended with immediate effect until the next restart of the devices.

> **Note:**
>
> If commands are defined with `Persistent=true` and the device is switched off at the time of scheduled command execution, these commands (persistent jobs) are started subsequently after the next restart.

Command definition parameters

For each command definition, in the `scheduler.ini` file, a new `Job<N>` section is created. Find all command definition parameters with their possible values below.

PARAMETER	DESCRIPTION
Id	Character string for unique identification of the command
Type	Command type
	1 Update
	2 Synchronize configuration
	3 Restart device
	4[1[[ ]from Scout 15 2103]] User-defined
	5 Shut down
Schedule	The format follows the `systemd calendar events`. Syntax for a timestamp
	Tue 2020-01-21 11:12:13
	Minutely [--* ::00]
	Hourly [---:00:00]
	Daily [--* 00:00:00]

PARAMETER	DESCRIPTION
	Monthly [--01 00:00:00]
	Weekly [Mon --* 00:00:00]
	Examples
	[--* 4:00:00] Every day at 4:00 am
	[Mon..Fri --* 22:30] Every workday at 10:30 pm
	[-*-1,5 11:12] Every first and fifth day of any month at 11:12. am
	[Sun 2020-- 17:15] Every Sunday of the year 2020 at 5:15 pm
	For further information, see https://www.freedesktop.org/software/systemd/man/systemd.time.html
SpreadFactor	Time span in which random delays are generated to prevent simultaneous execution on many devices
	Default: [0]
Persistent	`true`[false]
	If [true], missed command executions (jobs) are immediately recovered after the next device start.
Command	Only for [Type=4]: Command for user-defined command
WakeSystem	true`false`
	If [true], the system will be woken up from sleep mode (suspend) to execute scheduled commands.

Remote maintenance

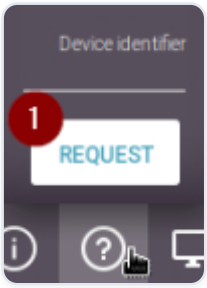
For maintenance, user help-desk and troubleshooting purposes, the administrator can use different tools to access the devices.

- Device identifier for support
- Mirroring
- Device diagnostics

Device identifier for support

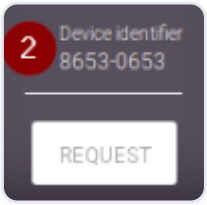
From Scout 15 2101 and eLux® RP 6 2101

To provide support for users, the first step is to identify the relevant device. As an alternative to the IP address or MAC address in the **Information** dialog, eLux provides a temporary device identifier that users can easily request.



Users have a question mark icon on the system bar, provided the user right **Request simple device identifier** is enabled (default).

- To request the so-called **Simple device identifier**, users click (or right-click) the question mark icon and then **Request** (1).



The system generates and transmits a temporary device identifier for the requesting device (2).

The user gives this device identifier to the support staff or administrator. By default, the device identifier has a validity of five minutes.

The validity time and optional additional text to be displayed can be specified by the administrator in the Scout Console under **Options > Advanced options > Rules**.

Mirroring

> **Note:**
>
> This feature can only be applied to an individual device.

Mirroring (Shadowing) allows administrators to either view or take control of eLux® user sessions. On the mirrored device, control of the mouse and keyboard can be given to the mirroring administrator. This can be very helpful in a variety of scenarios such as when administrators assist users or when administrators need to check correct functioning of firmware updates or newly installed software.

- Requirements
- Mirroring from Scout Console
- Troubleshooting mirroring

Requirements

- VNC viewer

  On the administrator's system, a VNC viewer must be installed. This is provided by the Scout Console

- VNC server

  On the target device, a VNC server must be installed. For eLux® devices, the **VNC Server extension** feature package which is part of the **XOrg** eLux package needs to be installed. This may require modifications of the image definition file on the web server via ELIAS.

- Configuration

  For the target device, in **Device configuration > Security > Mirror settings**, mirroring must be enabled and configured. For further information, see **Configuring mirroring** (/en-us/unicon-elux-scout/2603-ltsr-cu1/admin_guides/scout/setup/security/security_mirror.html).

Mirroring from Scout Console

Throughout a mirror session, the user receives a system message which is displayed on both the user's screen and the administrator's screen. The system message remains in the foreground and allows the user to cancel the mirror session at any time by clicking the **Quit** button.

LAUNCHING A MIRROR SESSION

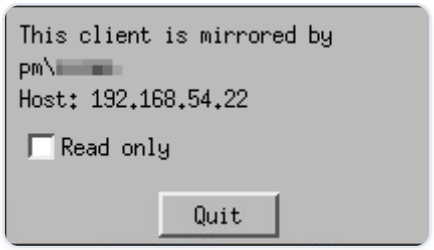
> **Note:**

> If there are two monitors connected to the device, both monitors are mirrored. To obtain the best result, on the Scout Console machine, connect two monitors with the same or a higher resolution. During the mirror session, the keyboard layout of the Scout Console machine is used and not the one of the user.

1. For the relevant device, from the context menu, choose **Mirror**.
2. On the **Mirroring** dialog, confirm with **OK**.
3. Depending on your configuration in **Device configuration > Security**, the session can only be started after
   - the administrator enters the defined password
   - the user confirms the mirror session

   For further information, see Configuring mirroring .

The mirror session is started. On the user's screen, a system message is displayed that can be moved but not closed unless the mirror session is closed.



The user has the following options:

OPTION	DESCRIPTION
Read only	The administrator has only read-access on the mirrored device. Mouse and keyboard input are not transmitted into the mirror session.
Quit	The connection is disconnected and mirroring is stopped.
Quit and logoff	The connection is disconnected and the user is logged off. This option is only available if, in **Device configuration > Security**, the option **Logoff after disconnect** is selected.

The mirror session is closed when

- the administrator closes the session window or clicks the **Quit** button of the system message, or
- the user clicks the **Quit** button of the system message.

If configured in **Device configuration > Security**, the mirror session is logged in a *mirror.txt file and saved to a sub-directory of the Scout Server files directory. The MAC address of the respective devices is also recorded.

Troubleshooting mirroring

ERROR / PROBLEM	REASON	SOLUTION
From a Scout Server with multiple network adapters, establishing a mirror session fails	The mirror session is not set up via the correct network adapter.	In the Registry, set the key BindIPAddress to the IP address of the network adapter the device is connected to: `HKEY_CURRENT_USERSoftwareUniconScoutSettingsBindIPAddress`

Device diagnostics

> **Note:**

> This feature can only be applied to an individual device.

> Device diagnostics help you run predefined commands on the device and retrieve protocol and configuration files from the device. These are then sent to Scout or other destinations for diagnostic purposes. The requested client files support the administrator in error analysis and are required when you open a support ticket.

Depending on the user rights, the reverse way can be used: Users send diagnostic files or screenshots from the device.

Administrators are free to use the **Request diagnostic files** feature to request freely definable files from a device.

> **Note:**
>
> To compare actual and target device configuration settings of individual devices, use a report. For further information, see Evaluating configuration data.

- Diagnostics scope and content
- Log level
- Requesting diagnostic files
- Defining an individual template
- Further diagnostic adjustments

Diagnostics scope and content

The amount of diagnostic files requested from a device is defined by two factors:

- Selected templates in the **Diagnostic files** dialog In addition to the #System template, further individual templates might be active.
- Configured log level Diagnostics tab Only the enhanced log level allows to execute all scripts on the device and retrieve all files defined in the `#System` template.

SYSTEM TEMPLATE

To perform device diagnostics, a predefined template called `#System` is provided. This template includes a file list with relevant configuration and log files plus a script code to be run on the device. Neither of them can be edited. You cannot deactivate the system template.

The system template contains the following files:

- Log files for eLux® and components
- eLux configuration files, including `terminal.ini, user.ini, sessions.ini`
- Configuration files of standard components and standard applications

The defined script contains different system commands that request software and hardware information from the device. Included are for example runtime information, installed packages, active codecs, the update partition, PCI and USB devices. Via `fwupdmgr`, firmware information about installed components such as the UEFI system or the flash memory is also retrieved.

The output of the system commands is stored in the `mytrace.log` file, which is transmitted along with the other files in a `.zip` archive.

> **Note:**
>
> Authorized administrators may define additional custom templates.

Log level

Two log levels are provided in the device configuration of eLux® devices: **Standard** and **Enhanced**. These levels correspond to switching the enhanced log level **On** and **Off** in the Scout Console.

In normal operation, the standard level is adequate. Before performing device diagnostics, however, temporarily enable the enhanced log level on the device to make sure you retrieve all data needed.

> **Note:**
>
> After the device diagnostics, we recommend that you reset the log level in order not to unnecessarily strain the flash memory capacity of the device.

ENABLING ENHANCED LOGGING

1. Open the device's context menu and click **Device configuration...**.
2. On the **General** tab, clear the option **Use parent device configuration**.
3. On the **Diagnostics** tab, set the **Debug level** option to **On**.
4. Confirm and restart the device.

Enhanced logging for the device is active and you can pull off the diagnostic files.

DISABLING ENHANCED LOGGING

1. Open the device's context menu and click **Device configuration...**.
2. On the **Diagnostics** tab, set the **Debug level** option to `Off`.
3. On the **General** tab, select the option **Use parent device configuration**.
4. Confirm with **Apply** and **OK**.

Enhanced logging for the device is reset to standard and device configuration inheritance is restored.

Requesting diagnostic files

Before you perform device diagnostics, we recommend that you temporarily enable enhanced logging on the device. For further information, see Log level.

1. From the device context menu, choose **Device diagnostics > Request files...**

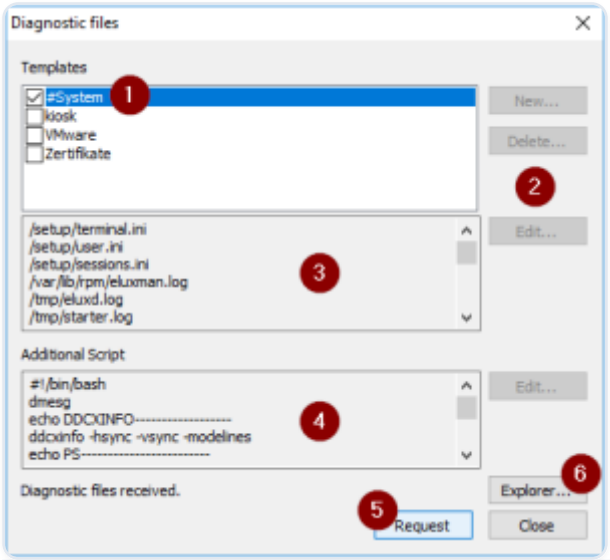
   In the **Edit diagnostic files** dialog, all templates defined so far are shown. Only active templates (check mark) will be processed.

2. The required object right provided, you can select or clear further templates of the list.
3. Click **Request**.

   All scripts defined in the active templates are executed on the device. All files defined in the active templates are retrieved from the device and saved as a .zip file in the local user directory `<userprofile>\Documents\UniCon\Scout\Console\Diag` of the Scout machine.

4. Click **Explorer**.

The Windows Explorer opens showing the diagnostics target directory.



1	Predefined template #System. This template is always active and can neither be deleted nor edited.
2	Provided an administrator owns the Edit diagnostic templates object right, she/he can create, **edit and delete additional templates**.
3	Log and configuration files defined by the selected template and requested by the device

4	Additional commands defined by the selected template to be executed on the device
5	The commands are executed and the defined files requested from the device.
6	Once the requested files are available, open the Explorer with the diagnostics target directory.

Defining an individual template

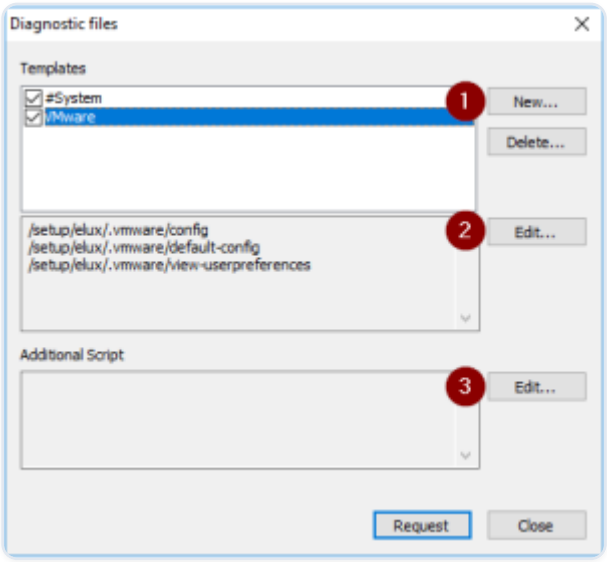
In addition, authorized administrators may define further templates containing files and script. The templates are available globally, no matter where you define them.

REQUIRES
Object right [Edit diagnostic templates] (disabled by default)

1. On the context menu of a device, choose **Device diagnostics > Request files**.

   In the **Edit diagnostic files** dialog, the predefined `#System` template and, if defined, further templates are shown.

2. Click **New** (1). Enter a name for your new template and confirm with **OK**.



3. To define diagnostic files for your template, select the new template and, next to the file list, click **Edit** (2).

In the text box, enter the relevant file names with path, line by line. Confirm with **Save**.

4. To enter script code you want to run on the device, next to , click **Edit** (3). Authenticate with the device password.[1[from Scout 15 2107]]

   In the text box, enter your code and confirm with **Save**.

> **Note:**

> When you perform the device diagnostics feature with **Request**, all selected templates are included. Whether all of the listed files of the `#System` template are written and transferred, depends on the configured log level.

Further diagnostic adjustments

Diagnostic files over multiple device restarts

REQUIRES
Enhanced logging must be enabled.

When you request diagnostic files, a set of log and configuration files is retrieved and packed on the target device. Included is the file `last_boot.zip`, which contains diagnostic files over a device restart. This `.zip` file contains all files defined for the system template plus some additional ones. You can specify the number of cycles for which the file is created and saved on shutdown of the device. The default is set to five.

The most recent file is named `last_boot.zip.0`. The further back a file was created, the later its sequential number in the file name.

- To define the number of cycles yourself, use the **Advanced file entries** feature with the following parameter:

File	`/setup/terminal.ini`
Section	`Global`
Entry	`LogCycles`
Value	`2-n`, The default value is [5].

Note:

That the file is only created, if enhanced logging is active. For further information, see Requesting diagnostic files.

STORE DIAGNOSTIC FILES PERSISTENTLY

By default, the diagnostic files are stored on the device under `/setup/logs`.

- To define a different storage location, use the **Advanced file entries** feature with the following parameter:

File	`/setup/terminal.ini`
Section	`Global`
Entry	`LogPath`
Value	`<Directory on the device>` The default is `/setup/logs`
	Example: `/update/logs` A location on the update partition may also be defined.

If you use disk encryption via TPM 2.0, the setup partition will be encrypted. To have the diagnostic files accessible, define the update partition as their storage location. The update partition is then mounted automatically.

DIAGNOSTIC ADJUSTMENTS VIA SOFTWARE PACKAGE

Provide diagnostic settings for new devices you wish to connect before their initial contact with the Scout Server, during installation. To do so, integrate the eLux® software package **Diagnostic adjustments** into the image to be installed.

The **Diagnostic adjustments** package contains two feature packages that you can activate separately:

- **Enhanced logging**: Corresponds to the device configuration **Diagnosis > Debug level** and enables enhanced logging
- **Logs on update partition**: Sets the location for diagnostic files to `/update/logs`

Note that the diagnostic settings set via software package are only effective until the first contact of the device to its Scout Server. Then the device configuration is synchronized and the device receives the configuration data defined for its OU.

Firmware update

On delivery, the are normally equipped with the operating system and the basic software components such as ICA® client, RDP client, browser and emulations. This software called firmware is stored on the flash drive. Whenever new software versions are available or demands change and software components need to be added or removed, the firmware can be updated.

Basic steps

- Download the relevant software packages from our myelux.com portal.
- Modify the image file (IDF) on the web server via ELIAS.
- Check the firmware configuration of the relevant
- Perform the update
- Deliver new software packages
- Install new software packages

To perform the update in one step, use an command. In this case, the required software packages are delivered and then automatically installed. Alternatively, the two actions can be uncoupled: The software is delivered in a first step when you use a command. It is subsequently installed when you run the command.

To save bandwidth, you can use a proxy client for updates. For further information, see Static proxy.

Ways to initiate a firmware update

Updates can be performed immediately or initiated automatically at a defined point in time:

- Use the command feature to execute or schedule (once or periodically) firmware updates.
- Define an . This will result in a firmware update on the next device restart.
- Configure the devices to automatically check for new image versions on start-up or shutdown. If an updated version is available, the update process will be started.

The option is part of the and can be applied to individual devices, OUs and all devices.

If configured, users can defer the execution of a firmware update to a later point in time.

Updates are only performed, when the relevant IDF has been modified. All update activities are logged.

Relevant devices

Commands and notifications can be applied to the following devices and groups:

- Individual devices
- Multiple devices selected in the window (multiple selection via and)
- OUs
-

Recovery-Installation

To reset your devices to initial state, perform a recovery installation. A recovery might also be required if critical feature packages of the Base OS have been changed, or if your operating system has not been updated for a long time. With a recovery installation, all data on the storage medium are wiped (/en-us/unicon-elux-scout/2603-ltsr-cu1/short_guides/recovery_general.html) in the `Recovery` short guide.

Requirements

The following components are required to perform a firmware update:

- Web server (for example IIS) to provide the eLux software packages and image definition files ( ) via HTTP/HTTPS or FTP/FTPS, with the relevant web server role enabled.
- Software container with up-to-date eLux® software packages on the web server
- ELIAS (eLux Image Administration Service) to create and modify image definition files in a software container
- Scout Console to configure firmware updates for the devices

Scout Server, Scout Console and the legacy ELIAS1 (Choose user-defined installation and select it as a feature) are part of the Scout Enterprise Management Suite. The current software bundle `eLuxversion_AllPackages.zip` and further optional software packages are installed with the eLux container.

Alternatively, install ELIAS 18 and import the eLux software bundle into your container. Then, there is no need for the container installation.

For further information on the installation of the Scout Enterprise Management Suite and the eLux container, see the Installation guide.

Access to applied images

Firmware images are created and modified in ELIAS. They are applied to the devices in the , in the firmware configuration. To open an image used for specific devices directly in the relevant ELIAS container, you have two options in the :

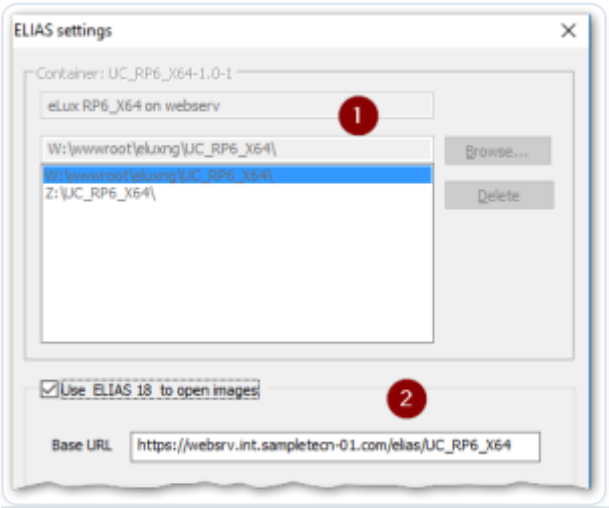
- In the **Properties** window of a device, double-click the **image** link.

- In the device configuration of an OU or device, in the **Firmware** dialog, click the **ELIAS** button.

The connection to ELIAS is made with the data from the ELIAS settings. Here, you set either the legacy ELIAS or 1[from Scout 15 2110]] as the default application for editing images:

Specifying ELIAS settings

1. Select .



1	Legacy ELIAS
2	ELIAS 18

2. If you use the legacy ELIAS, in the top section, click **Browse**. From your web server, for the required container, select the `container.ini` file.

   Optionally specify multiple containers.

3. If you use ELIAS 18, select the option in the bottom section.[2[from Scout 15 2110]]

   Enter your ELIAS 18 URL including the container path. (ELIAS 18 is designed to manage multiple versions in one container.)

4. Confirm with **OK**.

> **Note:**
>
> The menu entry **View > ELIAS** offers another way to call ELIAS from the console.

Planning firmware updates

Providing a new image

1. Download the latest software packages that are not yet included in your container from our portal. In ELIAS, import them into your container. For further information, see Importing packages into a container in the **ELIAS 18** guide or **Importing software packages** in the ELIAS 18 guide.
2. In ELIAS, add the relevant software packages to your image. Then save the modified image file. For further information, see Creating an IDF in the **ELIAS 18** guide or **Creating an image** in the ELIAS 18 guide.

Checking firmware configuration of the devices

1. For the relevant devices, open the **Device configuration > Firmware**. To apply the update to all devices, choose **Options > Base device configuration**.
2. Check whether the following fields are configured correctly for a firmware update: **Protocol, Server, Path and Image file**.

   The URL shown below the **Path** box is generated based on this data. The URL is relevant for the transfer of the image file `(.idf` and eLux® software packages.

   The specified image file name must be identical to the name of the image file updated in ELIAS.
3. For firmware updates via command: To allow users to postpone the update, configure the reminder settings. User can then control the time of execution themselves. For further information, see Update deferment by users.
4. To update the devices automatically on start or shutdown, select the relevant **Check for new version** option in the bottom area.

   Since in this case the update is initiated by the device, the firmware parameters stored locally on the device are used.

5. Confirm with **OK**.

   For further information on the firmware update configuration, see Configuring firmware updates.

Performing firmware updates

With the appropriate firmware configuration of the devices, you only need to provide an updated image to trigger a firmware update:

As soon as an updated image file is available on the web server and if one of the **Check for new version** options is selected, the update is performed on the next device restart or shutdown.

> **Note:**
>
> If the device is connected via VPN, the **Check for new version** options cannot be used. Use an **Update notification** instead.

> Alternatively, you can initiate a firmware update via one of the following procedures:

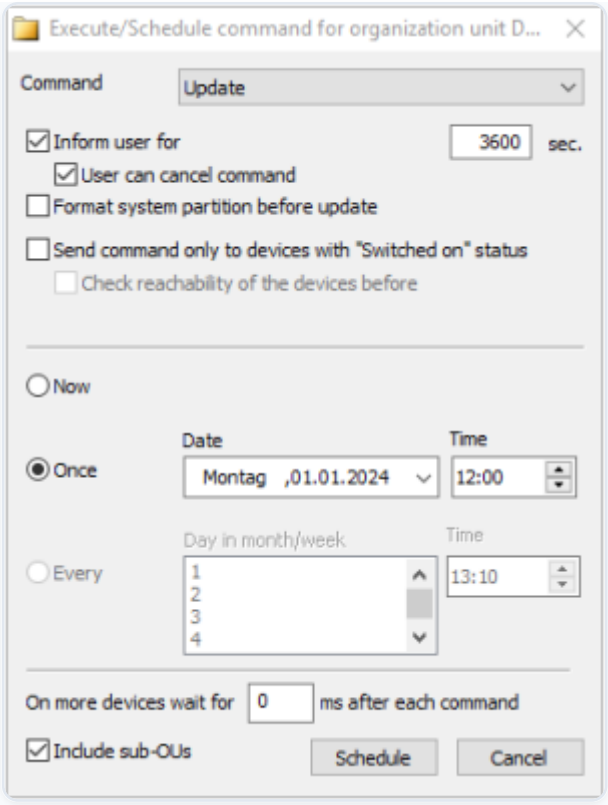
- Perform an update command
- Schedule an update command for a specified time, once or periodically
- Define an update notification

- Performing updates via command
- Updates initiated by the system
- Performing updates via notification

Performing updates via command

Note:

To deliver the software packages in a separate step before performing the update, use the **Delivery** command.



1. Select a device, an OU, a Dynamic Device Group or devices in the **All devices** window.
2. On the context menu, click **Commands > Update**.
3. To inform users before the update will be processed, we recommend that you select the **Inform user** option. For further information, see Command options and Update deferment by users.

   For further information on the impact, see User information before update.
4. To format the system partition of the device's flash memory before writing, select the option **Format system partition before update**.
5. Define the point in time for the update process. For further information, see Scheduling commands and Executing commands.
6. Click **Execute**.

The update process is triggered at the defined time. The update status is displayed for each device in its **Properties** window. During the update process, the status is shown. Detailed information about the currently processed action with time stamp is shown additionally. Example:

Update in progress (Transfer started -- 2022-08-20 11:34:23) Update in progress (Transfer completed -- 2022-08-20 11:35:45) Update in progress (Installation started -- 2022-08-20 11:35:48) Update in progress (Installation completed -- 2022-08-20 11:37:56)

For further information, see Command results and update information per device.

Note that updates are only performed, if the relevant IDF has been modified. If an update fails, no efforts will be made to retry.

> **Note:**

> When you execute an **Update** command, the relevant information is transferred to the devices as a URL. To create the URL, the system uses the values set in **Device configuration > Firmware** at the time when the command is run. Note that if the client initiates the update, the local **Firmware** configuration is relevant.

Updates initiated by the system

The devices can be configured to automatically check on start-up or shutdown whether a firmware update is available and install it.

PROCEDURE FOR SYSTEM-SIDE INITIATED FIRMWARE UPDATE

- Depending on the configuration, the devices check on each system start or shutdown whether a firmware update is necessary.

An update is necessary, if the image definition file on the web server specified by the firmware configuration has changed compared to the local version. The devices determine the delta between the software packages defined by the image definition file on the web server (image) and the locally installed software packages. If a delta exists, a firmware update is necessary.

- If a firmware update is necessary, a device downloads all software packages defined in the image that are not yet on its update partition from the web server or proxy server.
- Subsequently, the installation starts.

> **Note:**

> Since the update is initiated by the device, in this case, the firmware parameters locally stored on the device are used.

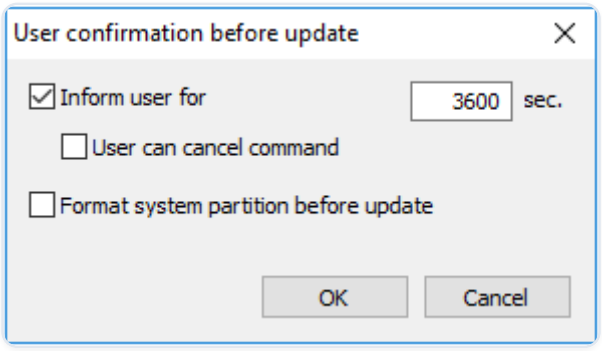
EXCEPTION FOR VPN CONNECTIONS

If **Check for new version on start** is configured and the VPN connection can only be set up after the eLux® desktop is loaded, the firmware will not be updated regardless of whether an update is necessary. Users can continue working without interruption.

For devices connected via the Scout Cloud Gateway, the option can be used without limitation.

CONFIGURING FIRMWARE UPDATES INITIATED BY THE SYSTEM

1. For the relevant devices , in the device configuration, under **Firmware**, select **Check for new version on start** or **Check for new version on shutdown**.
2. If you choose **Check for new version on start**, additionally set **User confirmation** options and whether you want the system partition to be formatted before the update is performed.[1[from Scout 15 2104.3000 and Scout 15 2209]]



For further information, see Configuring firmware updates.

After the devices have received the new device configuration, the option to check for new firmware versions becomes active.

Performing updates via notification

By using update notifications, you can send an explicit one-time update request to selected devices to be evaluated with the next connection. The devices then are updated to the image configured in the Scout firmware configuration.

1. Select a device, an OU, a Dynamic Device Group or devices in the **All devices** window. .
2. On the context menu, click **Notifications > Initiate firmware update**.

   The **Firmware update notification** dialog is shown.

3. Specify whether you want to inform users, and if they are allowed to cancel the command. For further information, see Performing updates via command.
4. To format the system partition before performing the update, select the relevant option.
5. Confirm the notification and confirmation.

   The notifications for firmware updates are defined for the relevant devices.

   For each device, in the **Properties** window, the **Update notification** field shows the value `Activated`.

   If the **Update notification** field in the **Properties** window is hidden, click to define which fields you want to show.

For the relevant devices, a firmware update notification is set. As soon as a device restarts and reconnects to Scout, it receives an update request[1[Note that updates are only performed if the relevant IDF has been modified.]] and the firmware update notification is automatically deleted.

Depending on how you have configured the notification and the device configuration in **Firmware > Reminder settings**, the update is performed immediately or the user receives a system message including deferment options. For further information, see User information before update.

The update status is displayed for each device in its **Properties** window. For further information, see Command results and update information per device. If an update fails, no system-side efforts will be made to retry.

For devices without update partition[2[flash memory less than 4 GB]], an update request might be shown although an update is not required. When the user clicks the **Update** button, the window is closed, no update is initiated.

> **Note:**

> In the Scout Report Generator, you can filter devices by the field **Image update notification**.

DELETING THE UPDATE NOTIFICATION FOR ONE OR MORE DEVICES

An update notification can be deleted before the firmware has been updated:

- On the context menu, click **Notifications > Delete update notification**.

> **Note:**

> Notifications are always set or deleted for all selected devices regardless of whether they are only available for individual devices.

User information before update

> **Note:**

> This feature refers to firmware updates and UEFI updates.[1[from eLux® RP 6 2107]]

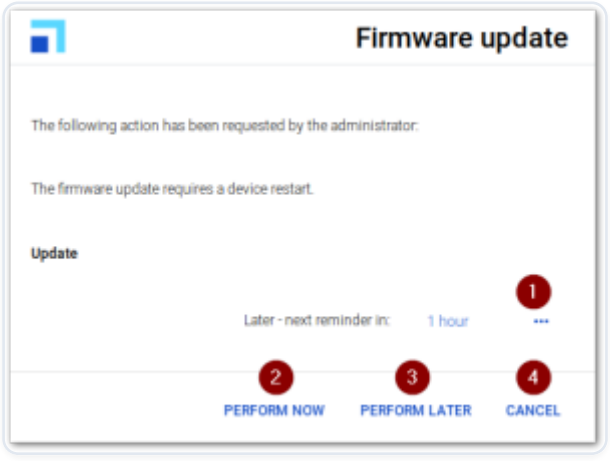
> Before an update is executed, the relevant users can be informed by a system message. Depending on the configuration in **Firmware > Reminder settings**, users are provided with various options for deferring or aborting the request.

The system message is displayed in the following situations:

- The administrator executes an **Update** command with the **Inform user** option.
- In the **Firmware** dialog, the option **Check for new version on start** with user confirmation[2[from Scout 15 2104.3000 and Scout 15 2209 with corresponding eLux versions]] is configured.

While the message is displayed, users may close applications, disconnect sessions, or - if configured - defer the update.

If the display duration for the message has been set to 0, the message will be displayed until the user clicks a button.



Users have the following options:

1	Select a time interval until the next reminder for the firmware update
	The list-field contains the values specified in **Firmware > Reminder > Delays until next reminder**.
	Displayed only, if the **Number of allowed deferments** is set to **1** or higher, and if at least one more deferment is possible.
2	Perform the firmware update immediately
3	Postpone the firmware update by the time period selected (1)
	If the device is shut down before the time interval expires, the update is performed during shutdown.
	Displayed only, if the **Number of allowed deferments** is set to 1 or higher, and if at least one more deferment is possible.
4	Abort the update process definitively
	Displayed only, if the command option **User can cancel the command** is selected.
	There will be no automatic restart of the update process.

Delivering software in advance

Before performing a firmware update, you can deliver the required software packages in a separate step. Only after the software deployment has been reported as successful on all relevant devices, start the installation by using an **Update** command or notification.

- Clients are provided with an update partition, see eLux partitions
- Firmware device configuration of the relevant devices must be configured correctly, see Planning firmware updates

The software delivery can be triggered by a Scout command or a notification.

During delivery, a live information icon is shown on the system bar.

The subsequent installation of the software packages and the update to the new IDF is initiated with an update command or an update notification.

> **Note:**
>
> If you initiate the subsequent firmware update via an **Update** command with **Format system partition**, the system might have to download additional packages. This is because a **Delivery** command only triggers the download of packages that are not installed on the system partition and are not available on the update partition.

- Performing deliveries via command
- Performing deliveries via notification

Performing deliveries via command

1. Select a device, an OU, a Dynamic Device Group or devices in the **All devices** window.
2. From the context menu, choose **Commands > Delivery...**
3. In the **Command** dialog, specify whether and how long you want to inform the user.
4. To allow users to deny the delivery, select **User can cancel command**.

   On the device, a pop-up dialog is displayed that allows users to cancel or start the delivery.
5. To first clean the update partition of the devices, select the **Clean update partition before delivery** option.

   > **Note:**
   >
   > After cleanup, all files of the current software image must be re-transferred and the dynamic proxy cache rebuilt.
6. Specify a time for the delivery. For further information, see Scheduling commands and Executing commands.
7. Click **Execute**.

The delivery is triggered at the defined time. If there is an updated IDF, and if the required software packages do not exist on the update partition of the relevant devices, the delivery of the software packages is started. The system will only download the packages that are missing. If there is less than 30 MB storage space available on the update partition, old packages are deleted before new packages are transferred.

- To check which files have actually been transferred, view the diagnostic file `/var/lib/rpm/eluxman.log`

In the , for each device, the delivery status is shown in the **Properties** window. During the delivery process, the status is shown, including detailed information about the currently processed action with time stamp. Example:

Delivery in progress (Transfer started -- 2018-08-20 11:34:23) Delivery in progress (Transfer in progress -- 2018-08-20 11:35:45) Delivery in progress (Transfer completed -- 2018-08-20 11:35:48)

For further information, see Command results and update information per device.

Performing deliveries via notification

By using delivery notifications, you send an explicit one-time delivery request to selected devices. The request is evaluated with the next connection of the device to its Scout Server. Then, the delivery of all required software packages for the image configured in the Scout firmware configuration is started for this device.

1. Select a device, an OU, a Dynamic Device Group or devices in the **All devices** window. .
2. From the context menu, choose **Notifications > Initiate software delivery**.

   The **Software delivery notification** dialog is shown.
3. Specify whether and how long you want to inform the user, and if the user is allowed to cancel the command. For further information, see Performing deliveries via command.
4. To clean the update partition before performing the delivery, select the relevant option.
5. Confirm the notification and confirmation.

   The notifications for software deliveries are defined for the relevant devices.

   For each device, in the **Properties** window, the **Delivery notification** field shows the value `Activated`.

> **Note:**

> If the **Delivery notification** field in the **Properties** window is hidden, click to define which fields you want to show.

For the relevant devices, a delivery notification is set. As soon as a device restarts and reconnects to its Scout Server, it receives a delivery request and the delivery notification is automatically deleted.

The delivery status of a device is shown in its **Properties** window. For further information, see Command results and update information per device. If an update fails, no efforts will be made to retry.

> **Note:**

> In the Scout Report Generator, you can filter devices by the field **Image delivery notification**.

DELETING THE DELIVERY NOTIFICATION FOR ONE OR MORE DEVICES

Delivery notifications can be deleted before the software is delivered:

- From the context menu, choose **Notifications > Delete delivery notification**.

> **Note:**

> Notifications are always set or deleted for all selected devices regardless of whether they are only available for individual devices.

Dynamic proxy

You may use dynamic proxy devices for the software package distribution to all devices of the same subnet. A dynamic proxy is an automatically selected device in a subnet that downloads the relevant software packages from the configured web server, and then provides them to all other devices in the subnet.

The solution is based on the device roles **Provider** and **Consumer**.

The fully automated provisioning (provider) and discovering (consumers) of the proxy service within subnets is realized in eLux® by using the zero-configuration networking implementation **Avahi**.

- Requirements
- Frame conditions and roles
- Update procedure

Requirements

To be able to perform updates by using a dynamic proxy client, next to the eLux® operating system, the following eLux packages must be installed on the devices of the relevant subnet:

- Dynamic Proxy update
- Avahi
- Squid Update Proxy

The proxy client must have an update partition. For further information on update partitions, see eLux partitions.

Frame conditions and roles

The dynamic proxy client concept is based on the following roles:

PROVIDER

REQUIRES
In the device configuration, under **Firmware > Proxy type**, the device must be configured to `Dynamic`.

The provider is the device that acts as the Dynamic Proxy client. All devices with an update partition can be selected for the provider role. Once a provider is selected, the device remains in the provider role for the upcoming updates. In case the provider is not available at the required point in time, another device with an update partition takes over the provider role. The provider is selected automatically and dynamically.

- To exclude devices from the provider role, under **Firmware > Proxy type**, select None.

CONSUMER

REQUIRES
In the device configuration, under [Firmware > Proxy type], the device must be configured to [Dynamic].

All devices of a subnet that are not selected for the provider role are consumers. The consumers perform their update through the provider of the subnet. The consumers do not need to download software packages from the web server.

- To exclude devices from the consumer role, under **Firmware > Proxy type**, select None.

> **Note:**
>
> In the **Firmware** configuration, if `HTTP` is used, the **User** and **Password** fields must remain empty.

Update procedure

UPDATE CHECK

In the case of an update request coming either from the or from the local configuration (), the consumers download the latest IDF file from the web server and check if they need to perform an update.

DISCOVER PROXY SERVICE

If software packages are required, the consumers try to discover the proxy service in the subnet. If there is no provider existing in the subnet so far, one of the devices with update partition automatically takes over the provider role and provides the proxy service.

DOWNLOAD SOFTWARE PACKAGES

The provider checks the availability of the requested software packages on its update partition and downloads missing packages from the web server specified by the consumers.

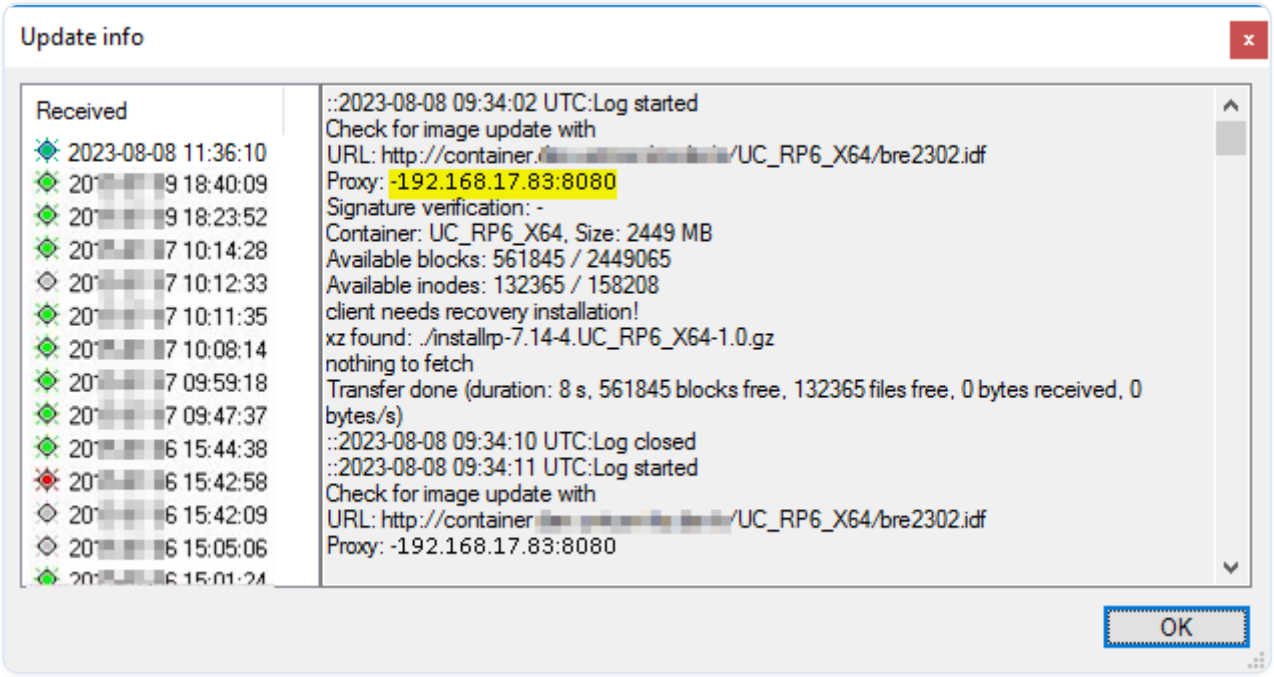
DEPLOY AND INSTALL SOFTWARE PACKAGES

The software packages are transferred from the provider to the consumers, and the consumers install the packages. Devices without update partition use the rhythm 'fetch one package - install one package' while devices with update partition fetch all required packages in one step and install

them subsequently. Only after the last consumer was provided with the relevant software packages, the provider updates its own system, if needed.

Update information is recorded for both, the consumers and the provider:

- For each updated device, the including the provider can be viewed by double-clicking in the window. For further information, see Command results and update information per device.



- The provider has a local file containing the consumers that have been provided with software packages.

Static proxy

from

If you want to update narrow-band connected devices, you might wish to use a proxy device to forward the firmware update. Proxy devices download the required software packages and distribute them to other devices.

The packages are kept locally in the RAM of the proxy device. Depending on the overall size of the packages defined in the image, 2 GB RAM or more are required.

For the proxy device, **Squid** is used as the proxy server software.

> **Note:**

> Squid supports the `HTTPS` protocol. To make the update process more secure, you may additionally use signatures for the image and software packages. For further information, see Firmware security through signatures.

> The configuration in Scout includes three basic steps:

- Creating an application definition for Squid
- Setting up the proxy device
- Configuring the relevant devices for the proxy update

Creating an application definition for Squid

1. Create a dedicated OU which will be configured for the proxy device.
2. For the new OU, add a local application. On the **Local** tab, edit the following fields:

OPTION	VALUE
Name of application	Squid
Local application	Custom
Parameter	squid
Hidden	On
Start automatically after 0 seconds	On

3. Move the proxy device into the OU and restart it.

The device receives the Squid application definition.

Setting up the proxy device

1. Provide the proxy device with a firmware image including the **Squid Update Proxy** package. To do so, modify the relevant image by using ELIAS, and then update the device.

> **Important:**
>
> The **Dynamic Proxy update** package must not be included.

2. For the OU of the proxy device, open **Device configuration > General** and clear the **Use parent** option.

   Inheritance is disabled and the proxy OU can be configured individually.

3. In **Device configuration > Firmware**, on the **Proxy type** list, select **None**.
4. For the proxy device, open **Device configuration > Network > LAN**, and edit the LAN connection. In the **Edit network profile** dialog, select **Use following IP address**. Leave the **Domain** box empty and confirm with **OK**.

   The last obtained IP address is used as the static IP address by the proxy device.

Configuring devices for the proxy update

1. For the OU or devices you want to receive their firmware updates through the proxy device, open the **Device configuration** dialog.
2. On the **Firmware** tab, edit the following fields:

Protocol	HTTP
Proxy type	Static (Consumer)
Provider	<>:3128
User and Password	<>

3. Edit the further fields as usual, see Configuring firmware updates.

Once the modifications have become active, the relevant devices receive their firmware updates from the proxy device.

Troubleshooting firmware update

Error messages

ERROR MESSAGE	REASON	SOLUTION
Bad container	Containers are hardware-specific.	Check whether the container matches your device specifications.

ERROR MESSAGE	REASON	SOLUTION
Bad flash size	IDF size exceeds flash size	Verify whether the image size defined by the IDF matches with the actual flash size of the device.
Bad authorization	Incorrect device password	Correct the entry in **Device configuration > Security**.
Client needs recovery information	If critical feature packages (.fpm) are updated in the baseOS, the device requires a recovery installation before it can be updated.	For further information see, eLux Recovery procedures in the **eLux Recovery procedures** short guide.

Update options

If the update is still faulty, try to modify the update settings. For further information, see Update/ Delivery

Passwords

eLux® users and administrators may be required to enter passwords in various situations, for example

- AD logon for user authentication after eLux system start
- Application logon
- Device password to unlock the device configuration On the Scout side, the device password applies to all devices of an infrastructure and is used for additional functions.
- Accessing a password-protected OU
- Defining a mirror password
- Setting up network profiles
- Further features of the device configuration Example: Defining a network drive
- To view and verify a password after typing, next to the **Password** entry field, click .
- To reset a password entry, click .

Local device password

The device password affects the local devices. All devices managed by a specific Scout Server receive the same device password so that it is used to verify the access rights for the devices. The device password is requested by Scout for management tasks such as discovery.

The device password can only be changed centrally for all devices in the Scout Console. The initial password is set to elux.

> **Note:**

> We recommend changing the password immediately to avoid unauthorized configuration caused by local users.

> Usually, the access rights do not allow users to modify their local device configuration in the **Security** dialog. However, if the user or administrator changes the device password locally for a device, this device can no longer be managed by Scout.

Changing local device password via Scout Console

Changing local device password on a device

For further information, see Device password in the eLux guide.

Changing local device password via Scout Console

> **Important:**

> Using this function, you change the device password of **all** devices managed by this .

1. In the Scout Console, click **Options > Base device configuration... > Security**. Under **Local security**, click **User rights**.
2. In the **User rights** dialog, in the **Device password** field, type the new password and repeat it in the **Conform device password** field.
3. Confirm with **OK**.

The new device password is assigned to the devices on the next restart.

> **Note:**

> To immediately activate the new password, perform a Scout **Restart** command for the relevant devices. For more details, refer to the commands execute guide.

Changing local device password on a device

1. In the eLux® Configuration panel, click **Security > Device password**.
2. Under **Current password**, type the old password, and in the next two fields type the new password.

   To view your entry after typing, click .
3. Confirm with **Apply**.

> **Important:**

> The device can no longer be managed by Scout.

Scout Console password

The default account **Administrator** with console password is only active, if the **Activate Administrator Policies** option is disabled.

In initial state, the Administrator policies are disabled and the console password is set to elux® (all lowercase).

> **Note:**

> We strongly recommend that you change the password immediately in order to prevent unauthorized access.

- To change the console password, log on to Scout as administrator and click **Options > Change console password**

or

- Enable the Activating administrator policies.

As soon as the administrator policies are enabled, the default account and console password are disabled.

We recommend enabling the Activating administrator policies and using your AD accounts for Scout.

Managing administrators

- Activating administrator policies
- Adding administrators
- Deleting administrators
- Administrator policy
- Viewing administrator activities
- Pass-through authentication
- Maintenance windows

Activating administrator policies

Managing more than one administrator requires enabling the **Administrator policies** feature. Scout administrator accounts are based on AD accounts which must be defined before. Scout administrator accounts can be configured in many ways.

By default, the administrator policies are disabled.

> **Note:**

> Enabling the administrator policies requires being logged in as a full-access administrator. The initial account is **Administrator** with the password set to elux.

1. In the Scout Console, click **Security > Activate administrator policies**.
2. Confirm with **OK**.

You are logged off and, from now on, you can only log on by using your Windows AD account. The **Security** menu options then become active. For example, you can enable Pass-through authentication now.

The Administrator default account is not available any longer and the **Change console password** option is disabled.

Adding administrators

You may define any AD users or groups as Scout administrators.

> **Note:**
>
> Check whether the relevant SQL Server permissions are set. For further information, see SQL Server users and application roles in the **Installation** guide.

1. In the Scout Console, click **Security > Manage administrators**...
2. In the **Administrator rights** dialog, click **Add Administrators**...

   The **Initial administrator profile** dialog opens.
3. Select the access range for the new admin and confirm with **OK**.

   The Windows **Administrator rights** dialog opens.
4. Below the Group or usernames field, click **Add**...

   The Windows **Select Users or Groups** dialog opens.
5. Enter the relevant AD username or AD group name, and then click **Check Names**.

   Or:

   Search for the AD user or AD group by using the **Advanced**... button.
6. Confirm with **OK..**

The new user or group is added to the list of administrators. You may assign the appropriate object rights to them now. For further information, see Administrator policy.

The administrators can log on to the Scout Console now by using their Windows account information.

> **Note:**
>
> If you use AD groups only, and if a user is a member of more than one group, the access rights of the groups are not consolidated, but the rights of the first group found apply. If users are authorized with their AD users and if they are authorized with one or more AD groups at the same time, the access rights are not consolidated but the rights of the AD user apply.

Deleting administrators

1. In the Scout Console, click Security > Managing administrators.
2. In the Administrator rights dialog, select the relevant administrator.
3. Click Delete administrator.

The selected administrator is deleted from the Scout administrators list without an 'Are you sure?' verification.

Administrator policy

For all Scout administrators there are three different kinds of access rights:

Base rights	Main access rights organized in functional blocks
Menu rights	Access rights for specific menu commands
Object rights	Access rights on OU or device level for properties, device configuration, applications, and some other functions
Default object rights	Default access rights for all OUs or devices for which no different object rights have been defined

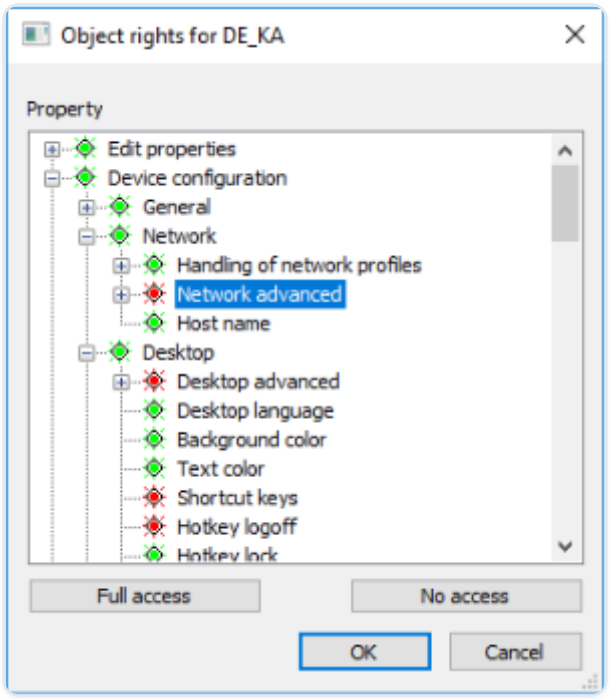
Note that in the dialog you must always first select the relevant administrator for whom you want to edit the access rights. To edit object rights, first select the relevant OU or device.

In the **Administrator rights** dialogs, the provided rights are displayed with a green or red symbol:

Access granted Access denied

To toggle between **Access granted** and **Access denied**, double-click the relevant right or press the space key.

If you click the **Full access** or **No access** buttons, all of the displayed rights are set to green or red, respectively.



Important:

For all kinds of access rights, the following applies: If a right is disabled, the relevant administrator will no longer have access to the related function. For the last or only administrator existing, you cannot disable access rights. This is to prevent being locked out of the Scout Console

- Changing base rights
- Changing menu rights
- Changing object rights
- Changing default object rights
- Defining a Start OU

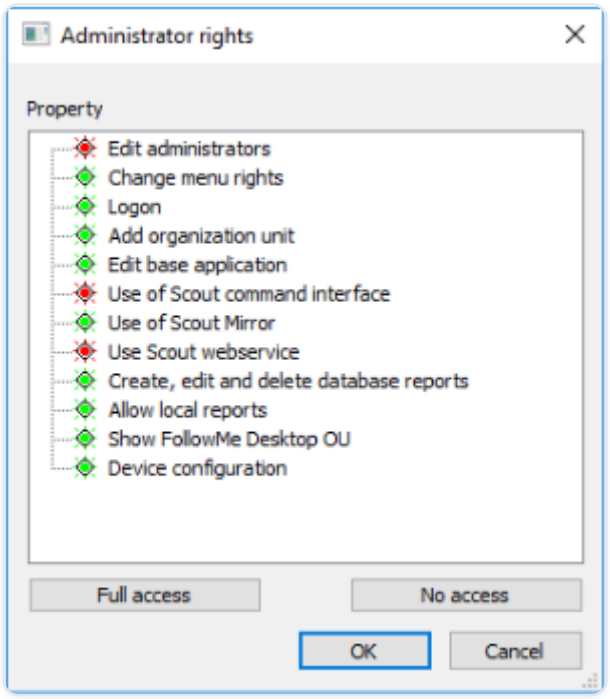
Changing base rights

Administrator base rights refer to entire functional blocks such as using the Scout Report Generator or configuring FollowMe Desktop.

1. In the Scout Console, click **Security > Managing administrators**.
2. In the **Administrator rights** dialog, select the relevant administrator.
3. Click **Base rights**.

   The **Administrator rights > Base rights** dialog opens.

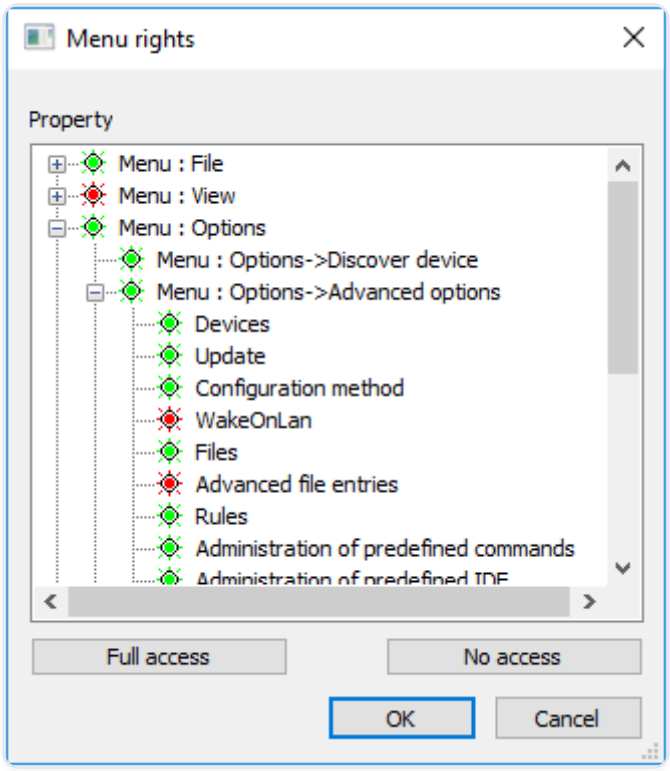
4. Change the relevant rights by double-clicking or by pressing the space bar.
5. Confirm with **OK**.



Changing menu rights

Menu rights refer to the executability of menu functions. If you deny an administrator access to a menu option, the menu item is dimmed and the administrator cannot perform the function.

1. In the Scout Console, click **Security > Menu rights**.
2. In the **Menu rights** dialog, select the relevant administrator.
3. Click **Menu rights**. The **Menu rights** dialog opens.
4. Change the relevant rights by double-clicking or by pressing the Space bar.
5. Confirm with **OK**.

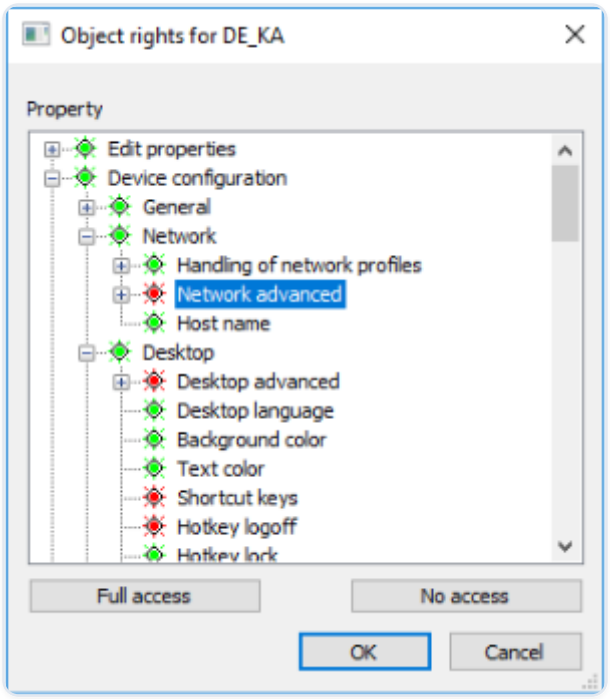


Changing object rights

Object rights refer to an OU or a device. You can define object rights for selected OUs or devices that differ from the general default object rights.

1. In the tree view, select an OU or device.
2. Click **Security > Object rights** or, from the context menu, choose **Object rights**.
3. In the **Object rights** dialog, select the relevant administrator.
4. Click **Edit object rights**.

   The **Object rights for** `<OU>` dialog opens.
5. Change the relevant rights by double-clicking or by pressing the Space bar.
6. Confirm with **OK**.



(/en-us/unicon-elux-scout/2603-ltsr-cu1/media/admin_guides/scout/admin_object_rights.png)

RESETTING OBJECT RIGHTS TO DEFAULT

You can reset object rights that you have defined differently for an OU or a device:

1. In the **Object rights** dialog of the OU or device, select the relevant administrator.
2. Click **Delete object rights**.

Changing default object rights

Default object rights apply to all levels to all OUs or devices for which no specific object rights are defined.

1. In the Scout Console, click **Security > Managing administrators**.
2. In the **Administrator rights** dialog, select the relevant administrator.
3. Click `Default object rights...`.

   The Default object rights dialog opens.
4. Change the relevant rights by double-clicking or by pressing the Space bar.
5. Confirm with OK.

Defining a Start OU

This feature lets you determine that an administrator is allowed to see only a particular start OU including its subordinate OUs.

1. In the Scout Console, click **Security > Manage administrators**.

2. In the **Administrator rights** dialog, select the relevant administrator.
3. Click **Set root OU**.

   The **Root organization unit** dialog opens.
4. Check the **Use the following root organization unit** option.
5. Select the relevant root OU.
6. Confirm with **OK**.

Viewing administrator activities

The activities of all administrators are logged according to the defined monitoring level in **Security > Manage administrators**.

You can view the logs in the Scout Console and you can export them.

1. In the Scout Console, select **Security > Manage administrators**.
2. Under **Monitoring**, click **View protocol**. The Activities dialog opens and shows the activities of all administrators.
3. To export log entries into a text file, select the relevant entries and click **Export selected items**.

Monitoring levels

The monitoring levels 0-3 build on each other. Level 0 only logs changes to the monitoring level itself.[1[from Scout Enterprise Management Suite15 2101]] At each level further activities are added. Logging with monitoring level 3 is the most detailed.

Note:

Use monitoring level 3 only temporarily for diagnostic purposes, since all database transactions are additionally logged.

ACTIVITIES / LEVEL 1	ACTIVITIES / LEVEL 2	ACTIVITIES / LEVEL 3
Log on	Save / Delete / Rename application	All SQL statements executed due to administrator activities in the console

ACTIVITIES / LEVEL 1	ACTIVITIES / LEVEL 2	ACTIVITIES / LEVEL 3
Log off	Upload application	
Deny logon	Save[2[[ ]also applies to Move device]] / Delete / Rename device	
Add / Activate / Delete license	Save[3[[ ]also applies to Move OU]] / Delete / Rename OU	
Add administrator	Save device configuration	
Edit / Delete administrator rights	Save advanced device configuration	
Change administrator policies	Save advanced options	
Request mirroring of a device	Discover devices	
Change device password	Schedule/change/delete Scout command	
	Set / Delete relocation notification	
	Set / Delete update notification	
	Set / Delete delivery notification	
	Set / Delete UEFI update notification	
	MSP or Enterprise Subscription mode: Assign tenant device / Remove assignment	

Pass-through authentication

The pass-through authentication enables Single-Sign-On. Your Windows account information is used to automatically log you on to Scout. The Scout logon window is not shown any longer.

Maintenance windows

A maintenance window is a time period that is scheduled for maintenance work. In the , authorized administrators can define maintenance windows to keep this period free for necessary IT maintenance tasks such as installing server updates. While a maintenance window is active, non-authorized administrators cannot use the and cannot start jobs.

Defining a maintenance window

REQUIRES
Menu right: **Menu: View > System diagnostics > Maintenance windows**

1. In the Scout Console, click the menu entry **View > System diagnostics > Maintenance windows...**
2. Click **Add**.
3. In the **Edit Maintenance window** dialog, specify the following options for your new maintenance window:

OPTION	DESCRIPTION
Name	With descriptive names, you can distinguish multiple maintenance windows.
From	Date and time when the maintenance period will be started for the first time
for	Time period of the maintenance window
	Specify number and unit (hours or minutes)
	Example: 2 hours
Repeat	Repeats the maintenance window periodically, see example below

4. Confirm with **OK**.

The maintenance window is defined and shown in the list of maintenance windows. For the duration of a defined maintenance window, the console will be blocked for administrators who do not have the **Maintenance window** menu right. The affected administrators can then only close the console, other actions are disabled. After the maintenance window has expired, they can restart the console as usual.

> **Note:**
>
> By default, the menu right is active and maintenance windows have no effect. If you disable the menu right for certain administrators, they will not be able to work with the console during the defined maintenance windows. Administrators with active menu right are not subject to any restrictions.

Example of a periodic maintenance window

For a maintenance window, for example defined from Friday, 06.09.2019, 20:00 for 2 hours, you can set the following repeat options:

Every day	Daily, starting on Friday, 06.09.2019
Every week	Every Friday of a week
Every month	Every 6th of a month
Every first weekday in month	Every Friday of the first week of the month
Every second weekday in month	Every Friday of the second week of the month
Every third weekday in month	Every Friday of the third week of the month
Every fourth weekday in month	Every Friday of the fourth week of the month
Every last weekday in month	Every Friday of the last week of the month

Scout Keep Alive service

The Scout Keep Alive service offers an evaluation of configurable status messages of the managed devices. These status messages are called **keep alive messages**.

Within a defined time interval, the configured devices send status messages to the Scout Keep Alive service. These messages allow to refresh the status of the relevant devices in the Scout Console.

1	The device has no network connection or is switched off.
2	The device is ready for operation. The network connection is operational.

For further information, see Icons in the tree view.

The configuration is part of the device configuration and can be applied to the entire infrastructure or to selected OUs and devices.

The **keep alive messages** are transferred from the devices to the Scout Keep Alive service by using the HTTPS protocol. This requires a valid SSL certificate. For further information, see Certificate for Scout Keep Alive service.

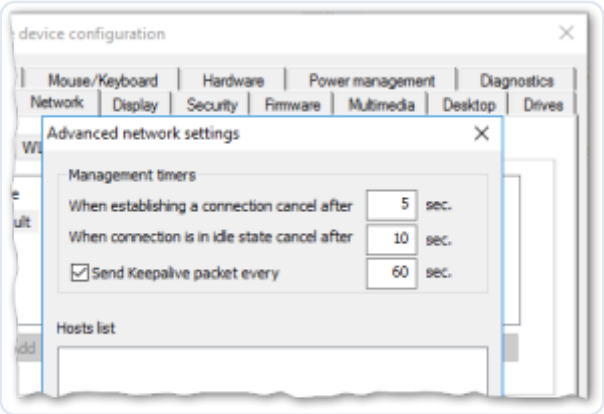
> **Note:**
>
> For Scout versions up to Scout 15 2204, the Scout Statistics Service as an optional component of the Scout Enterprise Management Suite® performed this task. This required an additional statistics database. With Scout 15 2209, the Scout Statistics Service has been replaced by the Scout Keep Alive service: The keep alive data are now stored in the Scout database. The processing of dynamic asset details for statistical analysis is no longer supported.

- Configuring status messages for devices
- Log files of the Scout Keep Alive service

Configuring status messages for devices

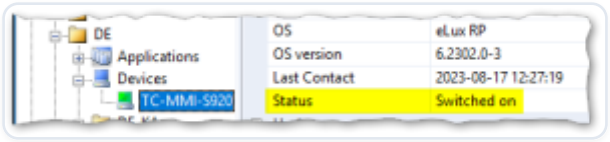
The Scout Keep Alive service helps you configure automatic updating of status of your devices (keep alive messages).

1. In the **Scout Console**, open the relevant network settings. For all devices, choose **Options** > **Base device configuration** > **Network** > **Advanced**. Alternatively, for a specific OU or device, open the **Device configuration** > **Network** > **Advanced** dialog.

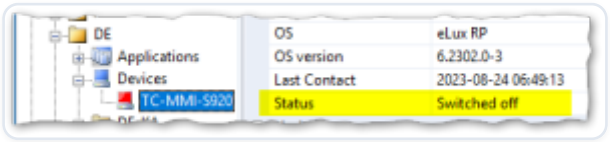


2. Select the option.
3. On the right of it, enter a time interval in seconds.
4. Confirm with **OK**.

Within the defined time interval, the configured devices send their status messages to the . The status messages result in a refresh of the device icons in the tree view and of the relevant device property in the :



If, however, a status message within the defined interval is missing, the device status in the is set to :



Log files of the Scout Keep Alive service

from Scout 15 2302

For the Scout Keep Alive service, log file rotation is active, which writes information about the service start and errors by default. The log level and other log options may be customized via an `.ini` file.

After installation, the log files and the `scoutkeepalive.ini` are located under `%PUBLIC%DocumentsUniConScoutScoutKeepAlive`

The following settings are available in `scoutkeepalive.ini` in the `Logger` section:

OPTION	DEFAULT	VALUE RANGE	DESCRIPTION
MaxLogFiles	10	1..n	Number of log files for log rotation
MaxLogFileSizeMB	100	100..n	File size in MB
Level	1	1..15	Log level
			1 Service start and errors
			2 Information beyond errors
			4 Program flow
			8 Program flow including further details
			To receive error messages, information beyond errors and information on the program flow, we recommend protocol level 7 (bit mask of 1+2+4).
Location	`C:UsersPublicDocumentsUniconScoutScoutKeepAlive`		Destination for the log files

Console communication

- Closing a console
- Sending messages
- Managing consoles

- Managing console commands

Closing a console

1. In the Scout Console, click **File > Console Management > Close console**.
2. Click **Refresh** to receive an up-to-date list showing all active consoles.
3. Click **Find** to filter the list.
4. Select the relevant consoles in the list.
5. To send a message to the relevant users, check the Inform user for option and enter the seconds as desired.
6. To give users a chance to cancel the command, select the **Command can be canceled by the user** option .
7. Click **Close selected consoles** or **Close all consoles**, respectively.

The command is communicated to the consoles. Closing the consoles might take several minutes. The dialog waits up to 5 minutes for receiving the confirmation of all consoles. The list of all active consoles is updated continuously within the time period.

Sending messages

With the help of this function you may send messages to other console instances.

1. In the Scout Console, click **File > Console Management > Send message**.
2. Under **Receiver**, choose which consoles you want to receive the message.
3. Under **Valid until**, define how long you want to show the message.
4. Under **Message**, enter your text.
5. To close the message automatically after a certain period of time, select **Inform user...** and specify a time period in seconds.
6. To allow the users of the receiver consoles to close the message without confirming its receipt, choose **Command can be canceled by the user**.

   When a receiver console is restarted, the message is displayed again within the validity period. If the message display time expires without the user selecting an option, the message is considered accepted.
7. Click **Send**.

The message will be sent to the selected consoles.

- Every console instance shows a message only once.
- If a console instance is not started within the validity period, the message will not be shown.
- If users, within the validity period, start a console instance which was not yet recorded in the database, the message will only be shown in case the option **To all consoles** has been set.

Managing consoles

As soon as administrators open a console, it will be registered in the Scout database. Registered consoles may be displayed via menu command:

- Click **File > Console management > Manage consoles**.

For each console, the logged-on user, computer name and logon domain are shown. The active console is hidden. If an administrator has multiple console instances open on the same computer, the consoles are numbered serially. For example, `mmi #2` is the second console instance of user `mmi`.

Deactivating console instance

- Remove the check mark in front of the relevant entry.

This console instance will no longer be displayed in any of the console communication dialogs.

Deleting console instance

This allows old consoles that are no longer in use to be removed from memory. This function has no effect on currently opened consoles.

- Select the relevant entry and click **Delete**.

The console instance and all commands related to this instance will be deleted.

> **Important:**
>
> This means that you lose part of the command history, and possibly also commands that have not yet been processed.

Check users for AD membership

- Click **Check** and then **Select** or **Deactivate**.

Unknown users are selected or disabled depending on the selected option.

Searching for user, computer or domain

- Click **Search**, and then in the relevant column in the search field, enter a string as search term.

To replace characters, you may use the wildcard characters * and ?. Case sensitivity is not relevant.

Managing console commands

Any console commands that have been run such as **Close console...** and **Send message...** can be viewed. Moreover, in the bottom list, the receiving consoles can be viewed and filtered.

Displaying commands

1. To filter the commands, use one of the options: **All, Active, Inactive, Older than** and **Younger than**.
2. To display a search field for one of the columns, click **Find**.

Changing the validity period of commands

- Select a command, and then under **Valid until**, modify the date and time.

Deleting commands

1. To delete all commands, click **Delete all**.
2. To delete a specific command, select it and click **Delete**.

All changes become valid only when you confirm with **OK**.

List of receivers

The **Recipients** list shows to which consoles a command was sent and when the consoles processed the command.

If * (all) is displayed for user, computer and domain, the command was sent to all consoles. Therefore, it also applies to console instances that are newly logging on to the database. An entry of this form therefore never shows a time in the **Processed** column.

Import/Export

All import and export functions can be performed via the Scout Console or the SCMD interface.

Exported files are saved in an XML format. The file name extension depends on the data category.

DATA CATEGORY FOR EXPORT/IMPORT	FILENAME EXTENSION
Device configuration of OUs	.oustp
Device configuration of devices	.devstp
Properties of OUs	.oupro
Properties of devices	.devpro
Properties of applications	.apppro
Device list	.csv
OU tree	.outree

- Exporting
- Importing

Exporting

1. Select the OU you want to export data from.
2. Click **File > Export** and what you want to export.
3. Select a folder to save and apply with **OK**.

Importing

You can import device configuration data, device properties and application properties. In addition, you can import device lists and OU trees. The import file must have the relevant file name extension.

1. Select the OU you want to import data into.
2. Click **File > Import** and the data category you want to import.
3. Apply with **OK**.

Log files and optimizing

- Log files
- Optimizing

Log files

Scout provides three logging options which are saved as `.log` files on the Scout Server.

OPTION	LOG FILE	DESCRIPTION
Scout Console	`scout.log`	Required for debugging
		Path: `%USERPROFILE%\Documents\UniCon\ScoutConsole`
		In the Scout Console, click **View > System diagnostics > Console log**.
Scout Server	`eluxd.log`	Log file of the Scout service, required for support calls
		Default path: `%PUBLIC%\Documents\UniCon\ScoutServer`
		Previous versions are renamed in elux.log.1, elux.log.2, etc.
		In the Scout Console, click **View > System diagnostic > Server log** (only if the Scout Console is installed on the same machine as the Scout Server).
Server keep alive log	`keepAlive.log`	Log file for keep alive entries of the Scout Server
		For devices, also a keepalive log file exists in the ScoutKeepAlive directory, provided the Scout Keep Alive service is installed.
		Created every 10 minutes

OPTION	LOG FILE	DESCRIPTION
		Default path: `%PUBLIC%\Documents\UniCon\ScoutServer`
Scout Statistics Service	`UniconStatisticService.log`	Rotating log file with configurable path
		Default path: `%USERPROFILE%\Documents\UniCon\ScoutStatisticService`
		The location and the maximum size from which a new file is written can be configured in `%PROGRAMFILES%\Unicon\ScoutStatistic\statisticservice.exe.config`

For further information about file paths, see Program and file directories.

Note:

Click View > System diagnostic > Server files] to open the server files directory in the Windows File Explorer (if console and server are installed on the same machine). The directory contains all configuration and log files organized in their application directories.

The following additional logs are available via the and can be exported:

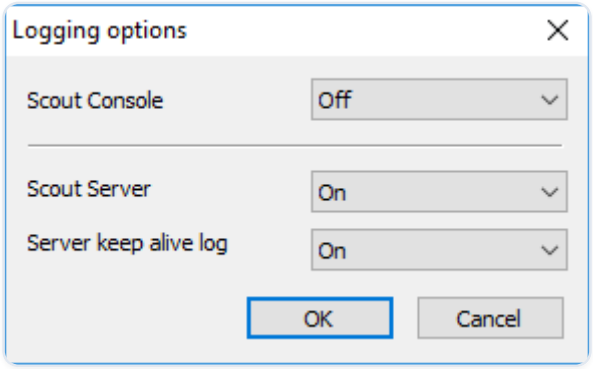
OPTION	DESCRIPTION
License log	All actions related to licenses such as entering, deleting and exporting licenses
	View > System diagnostics > License log

OPTION	DESCRIPTION
Administrator activity log	All administrator activities depending on the defined monitoring level
	Security > Manage administrators > View protocol
	For further information, see Viewing administrator activities.

- Enabling logging
- Configuring Scout Server log
- Cleaning up mirror log files

Enabling logging

1. In the Scout Console, click **Options > Logging options**.
2. For the desired options, in the list field, select On.



The selected log files are created by the system as described.

Configuring Scout Server log

For the Scout Server log file `eluxd.log` the `Scout Server` creates more than one backup (Log rotation). Once a new `eluxd.log` is created, the previous version is saved to the file `eluxd.log.1`, while the old `eluxd.log.1` is saved to `eluxd.log.2` and so on.

The log files continue recording when the server is restarted. The creation of a new log file is triggered by the following parameters:

- log file size
- maximum number of log files

You are free to configure both thresholds by yourself.

In addition, you may modify the location of the server log file and the keep alive log file (by default in `%PUBLIC%\DocumentsUniConScoutServer`). Specify any local directory for them, but no network directory.

MODIFYING LOG ROTATION AND LOCATION

1. In the file system, in `%PUBLIC%\DocumentsUniConScoutServer`, open the `eluxd.ini` file for editing.
2. To modify the rotation parameters, add the following entries:

SECTION	ENTRY	DEFAULT	DESCRIPTION
ELUXD	MaxLogFileSizeMB	100	Maximum size of a log file in MB
ELUXD	MaxLogFiles	10	Maximum number of log files (eluxd.log plus backups)

3. To modify the log file location, add the following entry:

SECTION	ENTRY	EXAMPLE	DESCRIPTION
ELUXD	LogFileLocation	c:\log	Local directory to be used for the log files `eluxd.log` and `keepAlive.log`

> **Important:**
>
> Specify a local directory to which the Scout Server can write. Do not use the UNC (Uniform Naming Convention) format.

After the Scout service has restarted, the log files are written to the specified directory.

If the Scout service cannot access the directory, it cannot start and creates an entry in the Windows Event Viewer. If the Scout service is running but cannot write the log file, it creates an alert message in the Scout Console.

Cleaning up mirror log files

-- from Scout 15 2101 --

If configured for mirroring, Scout creates a log file for each mirroring session and stores it in the mirror directory under the files. For further information, see Configuring mirroring .

You can have these log files deleted automatically via a command.

1. In the Scout Console, click **View > System diagnostics > Mirror log cleanup**.

   The **Command** dialog opens with the command **Mirror log cleanup**.
2. To select the period relative to the current date, under **Delete log files older than**, enter the number of days before which you want to delete log files.
3. Specify the point in time of the execution. To define periodic scheduling of the cleanup process, select **Every**. Then, specify a day of the month or a weekday and then the time.
4. Confirm with **Schedule**.

Periodic scheduling of the command ensures that mirror log files have only a limited lifetime. Example: If you define **90 days** and **Friday**, the log files will remain available for a maximum of one week longer than 90 days.

> **Note:**

> Other log data such as administrator activities, executed commands, or alarms in the Scout Console can be deleted via database cleanup. For further information, see Database cleanup.

Optimizing

To optimize the performance and deal with high network loads you can use the following options:

- Configuring Optimizing with handshake for a device, OU or all devices
- ManagerLoadBalancing to configure load distribution if you use a SQL database For further information, see the `Installation` short guide.

- Configuring the number of ODBC connections if you use a SQL database For further information, see the `Installation` short guide.

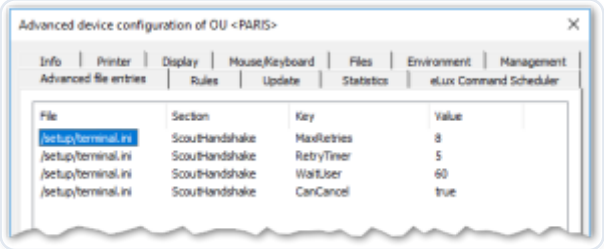
Optimizing with handshake

During each start-up the devices contact their Scout Server and check for new configuration data and application definition data. If they can't access the Scout Server, they retry to connect and synchronize according to their handshake configuration.

Activating new configuration data might require a restart of the client. Then the user is informed and has the chance to suppress restarting.

Handshake parameters can be set in the `terminal.ini` file of the client by using the **Advanced file entries** feature. For further information, see Advanced file entries.

Handshake can be configured for the entire organization or for a particular OU or device.



The values shown in the figure above are examples and can be modified. By default, handshake is not configured.

The `ScoutHandshake` section provides the following configurable parameters:

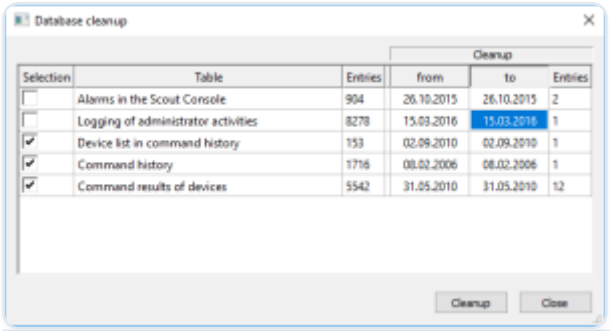
PARAMETER	DESCRIPTION
MaxRetries	Number of connection attempts. The value 0 deactivates handshake.
RetryTimer	Period of time in seconds until next connection attempt (start value). After each attempt, the interval is doubled (+/- random value). Example: Having defined 8 connection retries and a **RetryTimer** start value of 5 seconds, the 8th connection attempt is carried out after about 21 minutes.
PermanentRetriesAfterDays	Number of days (maximum) from the last successful connection until next connection attempt. Ensures that after n days latest the

PARAMETER	DESCRIPTION
	configuration data of device and [Scout Server] is compared. Can be combined with **MaxRetries** and **RetryTimer**.
WaitUser	Waiting time before client restarts to give the user the chance to close applications or log off.
CanCancel	Defines if the user is allowed to suppress a client restart (true\false).

Database cleanup

Scout stores huge amounts of data concerning various processes such as any performed update commands. To purge the Scout database tables, authorized administrators can delete database entries from particular tables for a specified period of time.

The relevant tables are listed in the **Database cleanup** dialog, each table providing the total number of entries and the creation date of the first entry. The administrator can modify only the fields **Selection** and **to**.



Alerts in the [Scout Console]	Alert messages (Error, Warning, Info), can be viewed by double-clicking the lamp icon on the Scout Console status bar	☐
Logging of administrator activities	Log file entries about the activities performed by an administrator according to the defined monitor level in **Security > Manage administrators**	☐

	For further information, see Viewing administrator activities.	
Device list in command history	History data of commands shown in **View > Command history** of the Scout Console (entries for particular devices)	
Command history	History data of commands shown in **View > Command history** of the Scout Console (entries for OUs)	
Command results of devices	Results of commands performed on the devices (Update, Delivery, user-defined command). The log data can be viewed in the Scout Console in the **Properties** window or by using the context menu of a device **Commands > Update/Delivery/Command**.	
	For further information, see Command results and update information per device.	

PERFORMING A DATABASE CLEANUP

1. Select **View > System diagnostics > Database cleanup**.
2. In the **Database cleanup** dialog, for the relevant table, in the "to" field, specify a date that indicates the end of a time span for the deletion of the entries (all entries up to and including this date are deleted).

   The **Entries** column at the right shows the number of entries to be deleted.
3. Click into the field of the `Selection` column on the left to activate this table for cleanup.

   A check mark indicates that entries from this table are selected for cleanup.
4. Click **Cleanup**.

A message shows the total of all entries in all tables which are intended for cleanup.

5. Confirm with **Yes**.

From the selected tables, all entries up to the specified dates are deleted.

> **Note:**
>
> Before you can delete command history entries, you are required to delete the corresponding device list entries.

Appendix

Program and file directories

eLux RP 6 partitions

IP ports

SNMP

Program and file directories

Program directory

The Scout Enterprise Management Suite® by default is installed to

`%PROGRAMFILES%\Unicon\Scout`

ELIAS 18 is installed according to your specifications during installation, for example on the web server IIS.

The eLux® container (only required for the legacy ELIAS) is installed on the web server to

`< root directory>\eluxng`

File path for Scout Server files

Scout log files, configuration files and more are saved to a subdirectory of

`%PUBLIC%\Documents\Unicon`

- To open the server files directory in the Windows Explorer, in the Scout Console, click **View > System diagnostics > Server files** (only if console and server are installed on the same machine).

File path for user files

User files are saved to a subdirectory of the local user directory in

`%USERPROFILE%\Documents\Unicon`

Diagnostic files that are requested via the console are saved to

`%USERPROFILE%\Documents\Unicon\Scout\Console\Diag`

Diagnostic files that are requested via Scout Board are downloaded by the browser used and saved in the download directory, if configured.

> **Note:**
>
> If you use anti-virus software on your Scout Server, we recommend that you exclude the specified directories from the virus scan to avoid side effects.

eLux® RP 6 Partitions

An eLux device's flash memory is generally divided into three or four partitions when eLux is installed. Each partition is reserved for a dedicated purpose and is only touched when you perform special tasks related to this partition.

All partitions are created during a recovery installation.

PARTITION	REQUIRES	PURPOSE	RECREATED WITH	OTHER
System		Reserved for the firmware (software packages)	Scout Update command with option Format system partition before update	Size for eLux RP 6 2104 LTSR and earlier versions: 1.77 GB / 1.84 GB with/without encryption, Size for eLux RP 6 2107 and later

PARTITION	REQUIRES	PURPOSE	RECREATED WITH	OTHER
				versions: 2.35 GB / 2.41 GB with/without encryption
Boot	only UEFI and USB	Boot section	–	
Setup		Device configuration Local application definitions	Factory reset command	Does not affect the system partition with installed firmware
Update	4 GB flash memory	Software delivery in advance (before firmware update) via Scout command or notification Signature check for eLux software packages	Scout Delivery command with option Format update partition before delivery	The size of the update partition complies with the storage space provided. The update partition is no larger than the storage space provided. Devices with less than 4 GB flash memory are not provided with an Update partition.

Note:

In the Scout Console, in the Properties window of a device, the system, setup, and update partitions are listed including their sizes.

Extended System Partition Starting with eLux RP 6 2107

When you perform an update installation or a new installation (recovery) to eLux RP 6 2107 or later, the system partition is created with 2.35 GB / 2.41 GB (with/without encryption) instead of the previous almost 2.0 GB. This creates more space for the firmware and allows larger images to be used.

UPDATE INSTALLATION

An update installation (firmware update) is still based on the previous partition sizes. The image size is thus still limited to the earlier values. Afterwards, the extended system partition is available and you can install images that may be up to 2.35 GB / 2.41 GB in size. This means, to install larger images on the freshly resized partition of the devices, a second firmware update is required.

RECOVERY INSTALLATION

Provided an up-to-date recovery system is available, with a PXE or USB recovery installation the system partition can be partitioned to the new size directly during the installation process and a larger image with up to 2.35 GB / 2.41 GB can be written in the same process. A new installation or recovery installation thus allows the partition to be resized and used in one step.

DOWNGRADE

To downgrade devices with the extended system partition (eLux RP 6 2107 or later) to an earlier version that only supports the previous system partition with less than 2 GB, you will have to go back to eLux RP 6 2104 LTSR.

We therefore recommend that you update test devices to eLux RP 6 2107 or later as the first step to thoroughly test functionality.

For further information, see Update to new partition layout (1.77 / 1.84 GB to 2.35 / 2.41 GB) in the Scout guide.

Network requirements

All contents relating to ports, protocols and network requirements are covered by the **network requirements** guide. For further information, see Network requirements guide.

SNMP

SNMP (Simple Network Management Protocol) is a network protocol for monitoring and controlling network devices.

For eLux® RP 6 and newer, version SNMPv3 is used.

> **Note:**

> The command line program **snmpget** is not included in the software package. To query SNMP status information, please use third party software.

> Configuring SNMP

SNMPD and SNMP Configuration Directives

Configuring SNMP

1. From our myelux.com portal, under **eLux Software Packages**, for your eLux version, under **Add-On**, download the package SNMP Environment and deploy it to the devices.
2. If there is no /setup/snmp/snmpd.conf on the devices, transfer the configuration file snmpd.conf to the devices to `/setup/snmp/snmpd.conf` by using the Scout feature advanced_files Files configured for transfer

   Or:

   Modify the `terminal.ini` file by using the Adding individual file entries feature of Scout. Example:

File	`/setup/terminal.ini`
Section	SNMPD
Entry	rocommunity
Value	secret

3. Optionally, to define further SNMPD and SNMP Configuration Directives, use the ▢ feature and modify the `terminal.ini` file under SNMPD. Examples:

   `syscontact=contact@sampletec.com` `syslocation=testcenter` `doDebugging=1`

   For further information on SNMPD Configuration Directives, see http://www.net-snmp.org.

The section SNMPD of the terminal.ini file is evaluated by the client, and the file /setup/snmp/snmpd.local.conf is created, overwriting any existing /setup/snmp/snmpd.conf.

If the configuration file does not exist, the file `/setup/snmp/snmpd.local.conf` is created with default values.

NOTES ON CONFIGURING SNMP V3

- When you define users (**createUser**), set a password with at least 8 characters.
- For the authentication method, define `authPriv` or `authNoPriv`.

> **Note:**
>
> For SNMP v2, you can use `noAuthNoPriv` as the authentication method.

SNMPD and SNMP Configuration Directives

The following table refers to the eLux® software package **snmp**.

For further information on using SNMP with eLux, see SNMP

For further information on SNMP commands, see http://www.net-snmp.org.

Additional SNMP Configuration Directives:

APPLICATION	COMMAND
authtrapenable	1 | 2 (1 = enable, 2 = disable)
trapsink	host [community] [port]
trap2sink	host [community] [port]
informsink	host [community] [port]
trapsess	[snmpcmdargs] host
trapcommunity	community-string
agentuser	agentuser
agentgroup	groupid
agentaddress	SNMP bind address
syslocation	location

APPLICATION	COMMAND
syscontact	contact-name
sysservices	NUMBER
interface	name type speed
com2sec	name source community
group	name v1|v2c|usm security
access	name context model level prefix read write notify
view	name type subtree [mask]
rwcommunity	community [default|hostname|network/bits] [oid]
rocommunity	community [default|hostname|network/bits] [oid]
rwuser	user [noauth|auth|priv] [oid]
rouser	user [noauth|auth|priv] [oid]
swap	min-avail
proc	process-name [max-num] [min-num]
procfix	process-name program [arguments...]
pass	miboid command
pass_persist	miboid program
disk	path [minspace | minpercent%]
load	max1 [max5] [max15]
exec	[miboid] name program arguments
sh	[miboid] name program-or-script arguments

APPLICATION	COMMAND
execfix	exec-or-sh-name program [arguments...]
file	file [maxsize]
dlmod	module-name module-path
proxy	[snmpcmd args] host oid [remoteoid]
createUser	username (MD5|SHA) passphrase [DES] [passphrase]
master	specify 'agentx' for AgentX support
engineID	string
engineIDType	num
engineIDNic	string

SNMP CONFIGURATION DIRECTIVES

APPLICATION	COMMAND
doDebugging	(1|0)
debugTokens	token[,token...]
logTimestamp	(1|yes|true|0|no|false)
mibdirs	[mib-dirs|+mib-dirs]
mibs	[mib-tokens|+mib-tokens]
mibfile	mibfile-to-read
showMibErrors	(1|yes|true|0|no|false)
strictCommentTerm	(1|yes|true|0|no|false)
mibAllowUnderline	(1|yes|true|0|no|false)
mibWarningLevel	integerValue

APPLICATION	COMMAND
mibReplaceWithLatest	(1|yes|true|0|no|false)
printNumericEnums	(1|yes|true|0|no|false)
printNumericOids	(1|yes|true|0|no|false)
escapeQuotes	(1|yes|true|0|no|false)
dontBreakdownOids	(1|yes|true|0|no|false)
quickPrinting	(1|yes|true|0|no|false)
numericTimeticks	(1|yes|true|0|no|false)
suffixPrinting	integerValue
extendedIndex	(1|yes|true|0|no|false)
printHexText	(1|yes|true|0|no|false)
dumpPacket	(1|yes|true|0|no|false)
reverseEncodeBER	(1|yes|true|0|no|false)
defaultPort	integerValue
defCommunity	string
noTokenWarnings	(1|yes|true|0|no|false)
noRangeCheck	(1|yes|true|0|no|false)
defSecurityName	string
defContext	string
defPassphrase	string
defAuthPassphrase	string
defPrivPassphrase	string

APPLICATION	COMMAND
defVersion	(1|2c|3)
defAuthType	(MD5|SHA)
defPrivType	DES (currently the only possible value)
defSecurityLevel	(noAuthNoPriv|authNoPriv|authPriv)

Scout Command Interface

The Scout Command Interface (SCMD) initiates Scout client management tasks from the command line.

Overview

You can use the Scout Command Interface SCMD to initiate various Scout client management tasks from the command line, for example:

- Add devices or OUs
- Delete devices or OUs
- Add, modify or delete device configurations
- Add, modify or delete application definitions
- Query device data
- Perform commands

Some functions can be performed directly via a command in the command line:Execute commands directly.

Other functions, such as adding configurations, require structured text files that contain the information to be processed: Execute file processing commands.

The program file is included in the installation of the and, after the installation, is located in the program directory of the .

Running the Scout Command Interface

Running SCMD for direct execution of commands

REQUIRES
Some commands require a file containing the data to be processed. This file must be located in the program directory or contain a fully-qualified path.

1. Open the Windows command line and change to the program directory.
2. Enter the following command: [scmd -u -p [-d] [] []]

Running for execution of file processing commands

REQUIRES
The structured files must be prepared and located in the program directory or specified with a fully-qualified path.

1. Open the Windows command line and change to the program directory.
2. Enter the following command: [scmd -u -p [-d] <> <> <>] You can specify any number of files.

   To use pass-through authentication, skip the user specification.

The program uses the database access data entered during the installation and a dynamically created user DSN to connect to the database.

For the file processing commands, the files are processed in the order they have been entered. If the file names contain wildcards, they are put in alphabetical order.

Showing the version of the Scout Command Interface

1. Open the Windows command line and change to the program directory.
2. Enter the following command:

Showing help

1. Open the Windows command line and change to the program directory.
2. Enter the command without parameters.

The command syntax and optional parameters are displayed.

Execute commands directly

Use the Scout Command Interface to directly execute various types of commands such as

- Scout commands for device remote management
- Console communication commands
- Import devices
- Create, rename and delete OUs
- Delete devices

Further functions such as Adding/Modifying configurations are available via the **File processing commands**.

> **Note:**

> To show the command syntax and optional parameters, run the program without parameters.

> For all commands, specify the user name, password and domain for logging on to the server, so that a command call is initiated with:

```
scmd -u[ser] Username -p[assword] Password [-d[omain] Domain]
```

To use pass-through authentication, skip the user specification.

> **Note:**

> Remember that text files to which you refer must be located in the program directory or specified with a fully-qualified path.

Importing devices (device profiles)

Devices can be assigned to OUs with their name and MAC address even before they connect to the for the first time.

[REQUIRES]
The devices to be created are listed in a text file in the required structure.

1. Create a text file with the following properties:

   - Each line contains one device with the three properties , , .
   - Use commas or semicolons as separators between the properties.

   Example: 00199985F675,TC-MMI-S920,KA2 00E0C5422A2E,TC-PRO-t63,FR1

2. Run the program with the following syntax:

   scmd -u Username -p Password [-d Domain] -import ( ) | File

OPTION	DESCRIPTION
-user	Required for logon to server
-password	
-domain	
-dontcreate	A device is only imported if it already exists.
-dontrename	If a device already exists with a different name, the device will not be renamed.
-dontmove	If a device already exists in another OU, the device will not be moved.
-dontcreateou	A device is only imported if the specified OU exists.
-createdcgonly	The devices are only created as Dynamic Device Groups.
File	Fully-qualified file name and path of your text file

Delete and Create commands

Deleting devices

REQUIRES
The devices to be deleted are listed in a text file [File].

REQUIRES
Each line of [File] contains one device specified by the MAC address or host name (optionally including the OU). Example
[Device1]
[Europe/Germany/Karlsruhe/Device2]
[0018077C413]

- Run the program with the following syntax: :

[scmd -u -p [-d] --deletedevices]

Alternatively, use the following syntax:

[scmd -u -p [-d] --delete -devicelist]

Deleting OUs

REQUIRES
The OUs to be deleted are listed in a text file [File].
Each line of [File] contains one OU ID. Example
[10]
[11]
[12]

- Run the program with the following syntax:

[scmd -u -p [-d] --delete -oulist]

Renaming OUs

REQUIRES
The OUs to be renamed are listed in a text file [File].

REQUIRES
Each line of [File] contains an OU ID, a comma as separator and the new OU name. Example: [302],[KA_2]

- Run the program with the following syntax:

[scmd -u -p [-d] --rename -oulist]

Creating OUs

REQUIRES
The OUs to be created are listed in a text file [File].
Each line of [File] contains an OU ID, a comma as separator and an OU name with path. Example
[1000],[Europe/Germany/MA]
[0],[Europe/Germany/FR] (If OU-ID=[0], the new ID is created by the system.)

- Run the program with the following syntax:

[scmd -u -p [-d] --create -oulist]

To create devices in their infrastructure, import a device list provided in a text file.For further information, see [Importing devices (/en-us/unicon-elux-scout/2603-ltsr-cu1/admin_guides/scmd/scmd_commands/scmd_import_devices.html).

Commands for device management

commands for device management are applied to an OU, an individual device or a list of devices.They can be applied at once or scheduled to be run once or periodically.

General options for commands

SCHEDULING OPTIONS		
SchedulerSettings/	-now	Command is executed at once
	-once -at DateTime	Command is executed at once at the time specified

SCHEDULING OPTIONS		
	-every {-day Day, -sun,-mon, -tue, -wed, -thu, -fri, -sat} -at Time	Command is executed periodically in the time period specified
	The following date and time formats are allowed	
	Date	dd.mm.[yy]yy, mm-dd-[yy]yy
	Time	hh:mm[:ss]
	Day	Example: With 15, the 15th of the month is meant.

Example of a periodically recurring command:

The message is sent every Monday and at the 1st and 15th of every month at 9:00.

DESTINATION OPTIONS	
-to [FQName]	For [FQName], the following options are available
	• Fully-qualified name of an OU or device
	Example: [DE/Karlsruhe/KA1]
	• MAC address of one device
	Example: [00E0C5422A2E]
	Specify also the [-mac] parameter!
	• Text file containing multiple MAC addresses (one MAC address per line)
	Example: [devices_xy.txt]

DESTINATION OPTIONS	
	Specify also the [-list] parameter!
-subous	If you specify an OU as destination ([-to]), the subordinated OUs are included.
-mac	Use this parameter if you specify a MAC address as destination ([-to]).
-list	Use this parameter if you specify a device list (text file) as destination ([-to]).

Send messages

- Run the program with the following syntax:

scmd -u Username -p Password [-d Domain] -sendmsg -msg Message -to FQName SchedulerSettings

OPTION	DESCRIPTION
-user	Required for logon to server
-password	
-domain	
-sendmsg	Send message command
-title	Text for the message title
-msg	Message text
-to	OU/device, MAC address or text file with multiple MAC addresses (see above)
-visible	Specify a timeout (in seconds) after that the message will be closed.
-deviceswait	Delay (in milliseconds) after execution if more than one device is affected

Restart devices

- Run the program with the following syntax:

scmd -u Username -p Password [-d Domain] -reboot -to FQName [-informuser] SchedulerSettings

OPTION	DESCRIPTION
-reboot	Restart device command
-to	OU/device, MAC address or text file with multiple MAC addresses (see above)
-deviceswait	Delay (in milliseconds) after execution if more than one device is affected
-cancancel	Users are allowed to cancel the command.

Switch off devices

- Run the program with the following syntax:

scmd -u Username -p Password [-d Domain] -switchoff -to FQName [-informuser] SchedulerSettings

OPTION	DESCRIPTION
-switchoff	Switch off command
-to	OU/device, MAC address or text file with multiple MAC addresses (see above)
-informuser	Inform users by message before execution [for the specified time period]
-cancancel	Users are allowed to cancel the command.

Switch on devices[1[from Scout 15 2209]]

- Run the program with the following syntax:

scmd -u Username -p Password [-d Domain] -switchon -to FQName SchedulerSettings

OPTION	DESCRIPTION
-switchon	Switch on command
-to	OU/device, MAC address or text file with multiple MAC addresses (see above)

Perform firmware updates

- Run the program with the following syntax:

scmd -u Username -p Password [-d Domain] -update -to FQName [-informuser] (-now | -once at Date Time)

OPTION	DESCRIPTION
-update	Update command
-to	OU/device, MAC address or text file with multiple MAC addresses (see above)
-format	Before installation, the system partition of the devices will be formatted.
-informuser	Inform users by message before execution [for the specified time period]
-cancancel	Users are allowed to cancel the command.
-now , -once at	Specify the time of execution. A periodic execution is not possible.

Deliver software

Run the program with the following syntax:

scmd -u Username -p Password [-d Domain] -delivery -to FQName [-informuser] (-now | -once at Date Time)

OPTION	DESCRIPTION
-delivery	Delivery command

OPTION	DESCRIPTION
-to	OU/device, MAC address or text file with multiple MAC addresses (see above)
-clean	Before delivery, the update partition of the devices will be formatted.
-informuser	Inform users by message before execution [for the specified time period]
-cancancel	Users are allowed to cancel the command.
-now, -once at	Specify the time of execution. A periodic execution is not possible.

Perform a configuration run

Run the program with the following syntax:

scmd -u Username -p Password [-d Domain] -configrun -to FQNameSchedulerSettings

OPTION	DESCRIPTION
-configrun	Configuration command
-to	OU/device, MAC address or text file with multiple MAC addresses (see above)

Refresh the device status

Run the program with the following syntax:

scmd -u Username -p Password [-d Domain] -refreshdevicestatus -to FQName SchedulerSettings

OPTION	DESCRIPTION
-refreshdevicestatus	Refresh device status command
-to	OU/device, MAC address or text file with multiple MAC addresses (see above)

Reset to factory status

Run the program with the following syntax:

scmd -u Username -p Password [-d Domain] -factoryreset -to FQName [-informuser] (-now | -once at Date Time)

OPTION	DESCRIPTION
-factoryreset	Remote factory reset command
-to	OU/device, MAC address or text file with multiple MAC addresses (see above)
-retainlocalconfig	Retain local user-defined device configuration in unlocked fields
-removescoutaddress	Delete Scout Server address on the device
-removelicenses	Additionally delete client-side stored license information
-informuser	Inform users by message before execution [for the specified time period]
-cancancel	Users are allowed to cancel the command.
-now , -once at	Specify the time of execution. A periodic execution is not possible.

Commands for console communication

Closing consoles

Run the program with the following syntax:

scmd -u Username -p Password [-d Domain] -icc-closeconsole

OPTION	DESCRIPTION
-user	Required for logon to server
-password	
-domain	

OPTION	DESCRIPTION
-displaytime	Inform the console user through a message before the console is closed. If you specify a value >0, the message will be closed after the specified time period.
-cancancel	The console user is allowed to cancel the command.
-timeout	The closing of the console is delayed by the specified value.
-toall	All active consoles are closed.
-to	List of console users in the format DomainUser

Sending messages

Run the program with the following syntax:

```
scmd -u[ser] Username -p[assword] Password [-d[omain] Domain-icc-sendmsg
[-validfrom Date Time] [-validto Date Time][-displaytime Seconds] [-
cancancel] -msg Message(-toall | -to DomainUser DomainUser)
```

OPTION	DESCRIPTION
-validfrom	
-validto	Limits the validity period of the command. Allowed date formats: dd.mm.yy[yy] or mm-dd-yy[yy]. Allowed time format: hh:mm:ss
-displaytime	The message shown in the destination consoles will be closed after the specified time period.
-cancancel	The console user is allowed to cancel the command.
-msg	Message text. To create a line break, enter n.
-to	List of console users in the format DomainUser

Synchronizing console instances against AD

Run the program with the following syntax:

scmd -u Username -p Password [-d Domain] -icc-checkadusers (-deactivate | -delete)

OPTION	DESCRIPTION
-deactivate	All console instances, which have not been found in the Active Directory, will be deactivated.
-delete	All console instances, which have not been found in the Active Directory, will be deleted from the database.

Execute file processing commands

The following features require structured text files that follow the format of files in Windows.The following applies to the text files:

- Each file must contain a section named [] where the type of action is defined.
- The information to be processed for this action is specified in a second section named [].
- One file defines one action.Multiple actions require multiple files.

Preparing structured text files

1. Create a text file and insert the headers of the sections [] and [].
2. Under [], insert a line with the string

   `# represents a number between 0 and 13 and specifies the type of action.For further information, see [Type of action (/en-us/unicon-elux-scout/2603-ltsr-cu1/admin_guides/scmd/scmd_fileproc/scmd_fileinfo.html).`
3. Under [], insert a line for each mandatory parameter of the selected action.Insert further lines for optional parameters you want to process.Mandatory and optional parameters can be found in the topic describing the relevant action (type).

Example for a structured text file:

inientries.ini

```
[FileInfo]
Type=13
[Setup]
DeviceName=M625q
IniEntryFile1=/setup/terminal.ini
IniEntrySection1=Security
IniEntryKey1 = TestKey
IniEntryValue1 = TestVal
IniEntryFile2=/setup/terminal.ini
IniEntrySection2=Security2
IniEntryKey2 = TestKey2
IniEntryValue2 = TestVal2
IniEntryFile3=/setup/ica/module.ini
IniEntrySection3=Security
IniEntryKey3 = TestKey
IniEntryValue3 = TestVal
```

- Run the program with the following syntax:

[scmd -u -p [-d] <> <> <>]

Specify the text files with their file name and file name extension.If a text file is not located in the program directory, also specify the fully-qualified path name.

Type of action (FileInfo section)

Each structured test file must contain a section named in which the type of action is defined. The type of action defines which information is processed in the section.

Section

ENTRY	VALUE/EXAMPLE	DESCRIPTION	MAND.
Type	#typenumber	The type number controls the type of action.	✓
	0	Add or modify a device	
	1	Delete a device	
	2	Add or modify an OU	
	3	Delete an OU	

ENTRY	VALUE/EXAMPLE	DESCRIPTION	MAND.
	4	Add or modify configuration data	
	5	Delete configuration data	
	6	Add or modify an application definition	
	7	Delete an application definition	
	8	Add or modify software defaults	
	9	Delete software defaults	
	10	Set Advanced options	
	11	Rename an application	
	12	Query device data	
	13	Import advanced file entries	

Add or modify a device (Type=0)

Adding or modifying a device requires

- Section [] with entry
- Section [] with the relevant entries

Section

ENTRY	VALUE/EXAMPLE	DESCRIPTION	MAND.
FQOUName	/OU1/OU2	Fully-qualified name of the OU to which the device will be added	✓
ForceDBWrite	0,1	Determines whether the imported configuration is written to the database even if no changes are visible	
Name	Client1	Name of the device	✓
MAC	001122334455	MAC address of the device. This address is used to determine whether a device is already registered in the Scout infrastructure. If so, the existing configuration data is changed.	✓
Info1		Information on the device	
Info2		Information on the device	
Info3		Information on the device	
UseParentAppDefaults	0,1	The software defaults defined for this OU are applied.	
UseParentApps	0,1	Use applications defined in parent OUs	

ENTRY	VALUE/EXAMPLE	DESCRIPTION	MAND.
UseParentScreenSettings	0,1	Determines whether the screen settings of the parent OU are used. If 0, specify the values for Prop_Resolution, Prop_Frequency and Prop_ColorDepth	
Prop_Resolution	1024x768	Screen resolution (property)	
Prop_Frequency	60	Screen refresh rate (property)	
Prop_ColorDepth	16	Screen color depth (property)	
IniEntryFile#		Set advanced file entries. To define one advanced file entry, four entries with the same number are required. # represents a number between 1 and 255. If no entries are defined, existing entries will be removed.	
IniEntrySection#			
IniEntryKey#			
IniEntryValue#			
FileSrc#		Configured file transfer. For one file, both entries with the same number are	

ENTRY	VALUE/EXAMPLE	DESCRIPTION	MAND.
		required. # represents a number between 1 and 255. If no entries are defined, existing entries will be removed.	
FileDst#			
UseParent	0,1	Determines whether the device configuration is inherited from the parent instance or an independent device configuration is created. For UseParent=0, you can set all configuration data subsequently.	
IpAddress		IP address of the device	
DefaultPrinter			
DefaultPrinterType			
DriverNames			
LastApplicationAction	#ON, #OFF, #REBOOT, #RESTARTX, #MSG, #CUSTOMCOMMAND, #UPDATE, #DELIVERY, #FACTORYRESET	Specifies the action after the last client application has been closed	

ENTRY	VALUE/EXAMPLE	DESCRIPTION	MAND.
LastApplicationActionDelay	0	Delay before the action is performed	
UseParentPeerIP	0,1	Determines whether the peer IP settings of the parent OU are used.	
UsePeerIP	0	Determines whether the alternative (peer) IP address of the device is used.	
UseParentPartitions	0,1	Determines whether the partition settings of the parent OU are used.	
RootPartitionSize	128		
HomePartitionSize	0		
SwapPartitionSize	2		
WOLServer		WakeOnLan server	
UseParentMousKBSettings	0,1	The mouse/keyboard settings of the parent OU are used.	
Prop_Emulate3Buttons	empty or Emulate3DButtons	Determines whether three mouse buttons are emulated	
Prop_multiClickTime		Double-click speed	
Prop_Buttons	auto, none, 2, 3, 5	Auto-detection, No mouse, Two-button mouse, Three-button mouse, Wheel mouse	

ENTRY	VALUE/EXAMPLE	DESCRIPTION	MAND.
Prop_PointerSpeed		Mouse pointer speed	
Prop_XkbLayout		Keyboard language	
Prop_Numlock		Initial Num Lock state	
Prop_DontVTSwitch		Allows switching consoles with Ctrl-Alt-F1...F12	
Prop_DeadKeys		Dead keys	
Prop_KeyboardDelay		Keyboard delay	
Prop_KeyboardSpeed		Key repetition rate	

Note:

Values that are not available in the Setup section are not processed. Example: If the value Info1 is not available, the value is not changed and not deleted. To delete the value, Info1 must be defined with an empty value field: `Info1=`.

You can add as many printers as you want. For each printer, a `!PRINTER!#` section is required with the following properties:

- The printer sections (#) must be numbered consecutively starting with 1.
- All printers defined here are assigned to the device.
- If no printer sections are available, no printers are registered and existing printers are deleted.

Delete a device (Type=1)

Deleting device requires

- Section with entry
- Section [] with entry or If both entries are specified, the entry has precedence.

Section

ENTRY	VALUE/EXAMPLE	DESCRIPTION	MAND.
MAC	001122334455	MAC address of the device. The device with this MAC address is deleted from the Scout infrastructure.	✓
			Or
Name	Client1	Host name of the device. The device with this name is deleted from the Scout infrastructure.	✓

Add or modify an OU (Type=2)

Adding an OU requires

- Section with entry Type=2
- Section with the relevant entries

Section

ENTRY	VALUE/EXAMPLE	DESCRIPTION	MAND.
FQOUName	/OU1/OU2	Fully-qualified name of the OU to which the OU will be added. To create the OU on the top level, set a slash: /	✓
ForceDBWrite	0,1	Determines whether the imported configuration is	

ENTRY	VALUE/EXAMPLE	DESCRIPTION	MAND.
		written to the database even if no changes are visible	
Name	OU3	Name of the OU	✓
Info1		Information on the OU	
Info2		Information on the OU	
Info3		Information on the OU	
UseParentAppDefaults	0,1	The software defaults defined for this OU are applied.	
UseParentApps	0,1	Use applications defined in parent OUs	
UseParentScreenSettings	0,1	Determines whether the screen settings of the parent OU are used. If 0, specify the values for Prop_Resolution, Prop_Frequency and Prop_ColorDepth	
Prop_Resolution	1024x768	Screen resolution (property)	
Prop_Frequency	60	Screen refresh rate (property)	
Prop_ColorDepth	16	Screen color depth (property)	

ENTRY	VALUE/EXAMPLE	DESCRIPTION	MAND.
IniEntryFile#		Set advanced file entries. To define one advanced file entry, four entries with the same number are required. # represents a number between 1 and 255. If no entries are defined, existing entries will be removed.	
IniEntrySection#			
IniEntryKey#			
IniEntryValue#			
FileSrc#		Configured file transfer. For one file, both entries with the same number are required. # represents a number between 1 and 255. If no entries are defined, existing entries will be removed.	
FileDst#			
DefaultPrinter			
DefaultPrinterType			
DriverNames			
LastApplicationAction	0	Specifies the action after the last client	

ENTRY	VALUE/EXAMPLE	DESCRIPTION	MAND.
		application has been closed	
LastApplicationActionDelay	0	Delay before the action is performed	
UseParentPeerIP	0,1	Determines whether the peer IP settings of the parent OU are used.	
UsePeerIP	0	Determines whether the alternative (peer) IP address of the device is used.	
UseParentPartitions	0,1	Determines whether the partition settings of the parent OU are used.	
RootPartitionSize	128		
HomePartitionSize	0		
SwapPartitionSize	2		
WOLServer		WakeOnLan server	
UseParentMousKBSettings	0,1	The mouse/keyboard settings of the parent OU are used.	
Prop_Emulate3Buttons	empty or Emulate3DButtons	Determines whether three mouse buttons are emulated	
Prop_multiClickTime		Double-click speed	
Prop_Buttons	auto, none, 2, 3, 5	Auto-detection, No mouse, Two-button mouse, Three-button	

ENTRY	VALUE/EXAMPLE	DESCRIPTION	MAND.
		mouse, Wheel mouse	
Prop_PointerSpeed		Mouse pointer speed	
Prop_XkbLayout		Keyboard language	
Prop_Numlock		Initial Num Lock state	
Prop_DontVTSwitch		Allows switching between the consoles with Ctrl-Alt-F1...F12	
Prop_DeadKeys		Dead keys	
Prop_KeyboardDelay		Keyboard delay	
Prop_KeyboardSpeed		Key repetition rate	

Delete an OU (Type=3)

Deleting an OU requires

- Section with entry `Type=3`
- Section with entries and .

Section

ENTRY	VALUE/EXAMPLE	DESCRIPTION	MAND.
FQOUName	/OU1/OU2	Fully-qualified name of the OU that contains the OU to be deleted	✓
Name	OU3	Name of the OU you want to delete	✓

Add or modify configuration data (Type=4)

This action has the following effect for the OU FQOUName: The value **Use parent device configuration** is deleted and an independent device configuration is created. For the new device configuration, the configuration data of the next higher instance are copied and their values entered.

Adding or modifying configuration data of an OU requires

- Section with entry `Type=4`
- Section with the relevant entries

Section

ENTRY	VALUE/EXAMPLE	DESCRIPTION	MAND.
FQOUNAME	/OU1/OU2	FULLY-QUALIFIED NAME OF THE OU FOR WHICH CONFIGURATION DATA ARE TO BE MODIFIED. TO MODIFY VALUES OF THE BASE CONFIGURATION (TOP LEVEL), SET A SLASH: /	✓
NETWORK			
TIMEOUT	<#SEC>	TIMEOUT FOR ESTABLISHING A CONNECTION TO THE MANAGER	
IDLETIMEOUT	<#SEC>	TIMEOUT FOR EXISTING CONNECTION AFTER A SPECIFIED PERIOD OF IDLE TIME	
HOSTNAME (ONLY INDIVIDUAL DEVICES)	MYCLIENT	HOST NAME OF THE DEVICE	
NETWORKOPTIONS	1,0	SEND KEEPALIVE PACKETS	
HOSTADDR1	192.168.10.22	PART 1 OF A VALUE PAIR FOR A HOST TABLE	
HOSTNAME1	MYCLIENT22	PART 2 OF A VALUE PAIR FOR A HOST TABLE	
[NETWORKPROFILE !#]			
NETWORKTYPE	4, 5, 6	4 ETHERNET, 5 WLAN, 6 WWAN (MOBILE INTERNET)	

ENTRY	VALUE/EXAMPLE	DESCRIPTION	MAND.
FQOUNAME	/OU1/OU2	FULLY-QUALIFIED NAME OF THE OU FOR WHICH CONFIGURATION DATA ARE TO BE MODIFIED. TO MODIFY VALUES OF THE BASE CONFIGURATION (TOP LEVEL), SET A SLASH: /	✓
NETWORK			
TIMEOUT	<#SEC>	TIMEOUT FOR ESTABLISHING A CONNECTION TO THE MANAGER	
IDLETIMEOUT	<#SEC>	TIMEOUT FOR EXISTING CONNECTION AFTER A SPECIFIED PERIOD OF IDLE TIME	
HOSTNAME (ONLY INDIVIDUAL DEVICES)	MYCLIENT	HOST NAME OF THE DEVICE	
NETWORKOPTIONS	1,0	SEND KEEPALIVE PACKETS	
HOSTADDR1	192.168.10.22	PART 1 OF A VALUE PAIR FOR A HOST TABLE	
HOSTNAME1	MYCLIENT22	PART 2 OF A VALUE PAIR FOR A HOST TABLE	
[NETWORKPROFILE !#]			
NETWORKTYPE	4, 5, 6	4 ETHERNET, 5 WLAN, 6 WWAN (MOBILE INTERNET)	

ENTRY	VALUE/EXAMPLE	DESCRIPTION	MAND.
FQOUNAME	/OU1/OU2	FULLY-QUALIFIED NAME OF THE OU FOR WHICH CONFIGURATION DATA ARE TO BE MODIFIED. TO MODIFY VALUES OF THE BASE CONFIGURATION (TOP LEVEL), SET A SLASH: /	✓
NETWORK			
TIMEOUT	<#SEC>	TIMEOUT FOR ESTABLISHING A CONNECTION TO THE MANAGER	
IDLETIMEOUT	<#SEC>	TIMEOUT FOR EXISTING CONNECTION AFTER A SPECIFIED PERIOD OF IDLE TIME	
HOSTNAME (ONLY INDIVIDUAL DEVICES)	MYCLIENT	HOST NAME OF THE DEVICE	
NETWORKOPTIONS	1,0	SEND KEEPALIVE PACKETS	
HOSTADDR1	192.168.10.22	PART 1 OF A VALUE PAIR FOR A HOST TABLE	
HOSTNAME1	MYCLIENT22	PART 2 OF A VALUE PAIR FOR A HOST TABLE	
[NETWORKPROFILE !#]			
NETWORKTYPE	4, 5, 6	4 ETHERNET, 5 WLAN, 6 WWAN (MOBILE INTERNET)	

ENTRY	VALUE/EXAMPLE	DESCRIPTION	MAND.
FQOUNAME	/OU1/OU2	FULLY-QUALIFIED NAME OF THE OU FOR WHICH CONFIGURATION DATA ARE TO BE MODIFIED. TO MODIFY VALUES OF THE BASE CONFIGURATION (TOP LEVEL), SET A SLASH: /	✓
NETWORK			
TIMEOUT	<#SEC>	TIMEOUT FOR ESTABLISHING A CONNECTION TO THE MANAGER	
IDLETIMEOUT	<#SEC>	TIMEOUT FOR EXISTING CONNECTION AFTER A SPECIFIED PERIOD OF IDLE TIME	
HOSTNAME (ONLY INDIVIDUAL DEVICES)	MYCLIENT	HOST NAME OF THE DEVICE	
NETWORKOPTIONS	1,0	SEND KEEPALIVE PACKETS	
HOSTADDR1	192.168.10.22	PART 1 OF A VALUE PAIR FOR A HOST TABLE	
HOSTNAME1	MYCLIENT22	PART 2 OF A VALUE PAIR FOR A HOST TABLE	
[NETWORKPROFILE !#]			
NETWORKTYPE	4, 5, 6	4 ETHERNET, 5 WLAN, 6 WWAN (MOBILE INTERNET)	

		for diagnostic files	

Delete configuration data (Type=5)

This action has the following effect for the OU `FQOUName`: The value **Use parent device configuration** is set and the independent device configuration is deleted. The configuration data of the parent OU are inherited.

Deleting configuration data of an OU requires

- Section with entry `Type=5`
- Section] with the entry `FQOUName`

Section

ENTRY	VALUE/EXAMPLE	DESCRIPTION	MAND.
FQOUName	/OU1/OU2	Fully-qualified name of the OU for which the configuration data are to be deleted	✓
		The configuration data of the base configuration cannot be deleted.	

Add or modify an application definition (Type=6)

Adding or modifying an application definition requires

- Section with entry `Type=6`
- Section with the relevant entries

Section

ENTRY	VALUE/EXAMPLE	DESCRIPTION	MAND.
FQOUName	/OU1/OU2	Fully-qualified name of the OU for which an application definition is to be	✓

ENTRY	VALUE/EXAMPLE	DESCRIPTION	MAND.
		added or modified. To add or modify an application on the top level, set a slash: /	
Name	MyApp	Name of the application	✓
AutoStart	true or false	Determines whether the application is started automatically	
AutostartDelay	<#sec>	Delay for automatic application start	
TM	true or false	Determines whether the application runs permanently (terminal mode)	
Roaming	true or false	Enables roaming when a card reader is used	
AppOnDesktop	0 or 1	Determines whether a desktop icon is created on the device	
Type	0, 3, 4, 5, 7, 10, 13, 15, 17	Emulation, Browser, ICA/Citrix, Local shell, SAP, RDP, PNAgent, Virtual desktop, StoreFront	✓
ICAServer		Name of the Citrix® server	
ICAUser		Name of the ICA® user	

ENTRY	VALUE/EXAMPLE	DESCRIPTION	MAND.
ICAPass		Password of the ICA user	
ICADomain			
ICAApplication	appl1, #pubappl1	Application to be started. For a published application, the application name must be specified with a leading #.	
ICAWorkDir		Working directory for the application	
ICAResolutionDefault	0 or 1	Use of the general Citrix settings for the resolution	
ICAResolution	0, 1, 2, 3, 4, 5, 6, 7, 8	Resolution of the ICA session: 640x480, 800x600, 1024x768, 1280x1024, 1600x1200, User-defined size1, User-defined percentage2, Full screen, Seamless	
ICAWidth	1280	Width of the ICA session	
ICAHeight	1024	Height of the ICA session	
ICAPercent	75	Percent of screen resolution	

ENTRY	VALUE/EXAMPLE	DESCRIPTION	MAND.
ICAColorsDefault	0 or 1	Use of the general Citrix settings for color depth	
ICAColors	0,1,2, or 3	Color depth: 0: 16 colors, 1: 256 colors, 2: 16 bit colors, 3: 24 bit colors	
ICAColorMapDefault	0 or 1	Use of the general Citrix settings for color assignment with 256 colors	
ICAColorMap	0 or 1	Defines how the color table is assigned for 256 colors: 0: common (approximate colors), 1: private (precise colors)	
ICASound			
ICACrypt	0,1,2, 3, or 4	0: Basic, 1: RC5 (128 bit - login only), 2: RC5 (40 bit), 3: RC5 (56 bit), 4: RC5 (128 bit)	
ICAAutologin	true or false	Automatic logon with higher encryption depth	
ICACompress	0 or 1	Compression of the data to be transmitted	
ICABitmapCache	0 or 1	Creating a bitmap cache	

ENTRY	VALUE/EXAMPLE	DESCRIPTION	MAND.
ZLMouseMode	0,1, or 2	Values for the latency reduction: 0: off, 1: on, 2: auto	
ZLKeyboardMode	0,1, or 2	Values for the latency reduction: 0: off, 1: on, 2: auto	
ICAServerLocationDefault	0 or 1	Use of the general Citrix settings for the server locations	
BrowserProtocol	0,1, or 2	Browser protocol via which the ICA client searches the server / published applications: 0: TCP/IP + HTTP, 1: TCP/IP, 2: SSL + HTTPS	
BrowserAddress1	mybrowser1	Browser addresses	
BrowserAddress2	mybrowser2		
BrowserAddress3	mybrowser3		
BrowserAddress4	mybrowser4		
BrowserAddress5	mybrowser5		
DisableCtrlAltDel	true or false	Determines whether the Citrix logon dialog asks for a PIN (smart card) or user/password. For smart card use.	

Delete an application definition (Type=7)

Deleting an application definition requires

- Section with entry `Type=7`

- Section with the relevant entries `FQOUName` and `Name`

Section

ENTRY	VALUE/EXAMPLE	DESCRIPTION	MAND.
FQOUName	/OU1/OU2	Fully-qualified name of the OU for which an application definition is to be deleted	✓
Name	MyApp	Name of the application. To delete all application definitions of an OU, set an asterisk: *	✓

Add or modify software defaults (Type=8)

Adding or modifying software defaults requires

- Section with entry `Type=8`
- Section with the relevant entries

Section

ENTRY	VALUE/EXAMPLE	DESCRIPTION	MAND.
FQOUName	/OU1/OU2	Fully-qualified name of the OU for which the software defaults are defined	✓
ClientNameTemplate	$ICAUSER	Is used to set the device name. Allowed variables: $ICAUSER, $ICADOMAIN, $ICAAPPLICATION	
BrowserProtocol	0,1, or 2	Browser protocol via which the ICA client	

ENTRY	VALUE/EXAMPLE	DESCRIPTION	MAND.
		searches the server / published applications: 0: TCP/IP + HTTP, 1: TCP/IP, 2: SSL + HTTPS	
HttpBrowserAddress		Browser addresses	
HttpBrowserAddress 2			
HttpBrowserAddress 3			
...			
HttpBrowserAddress 15			
TcpBrowserAddress			
TcpBrowserAddress2			
TcpBrowserAddress3			
...			
TcpBrowserAddress15			
DisableSound	0 or 1	Determines whether the Citrix® client plays sound	
ClientManagement	on or off	Allows/denies automatic updating of the devices	
ClientDrive	on or off	Assignment of a user drive	

ENTRY	VALUE/EXAMPLE	DESCRIPTION	MAND.
DrivePathA	/misc/floppy	Name of the directory to be assigned	
DriveEnabledA	0 or 1	0: Drive is not assigned, 1: Drive is assigned	
DriveReadAccessA	0,1, or 2	0: Read access allowed, 1: Read access denied, 2: Confirm read access in first session	
DriveWriteAccessA	0,1, or 2	0: Write access allowed, 1: Write access denied, 2: Confirm write access in first session	
DrivePathB		Name of the directory to be assigned	
DriveEnabledB	0 or 1		
DriveReadAccessB	0,1, or 2		
DriveWriteAccessB	0,1, or 2		
DrivePathC	/misc/cdrom		
DriveEnabledC	0 or 1		
DriveReadAccessC	0,1, or 2		
DriveWriteAccessC	0,1, or 2		
DrivePathD	misc/usb0		
DriveEnabledD	0 or 1		

ENTRY	VALUE/EXAMPLE	DESCRIPTION	MAND.
DriveReadAccessD	0,1, or 2		
DriveWriteAccessD	0,1, or 2		
...			
DrivePathZ			
DriveEnabledZ	0 or 1		
DriveReadAccessZ	0,1, or 2		
DriveWriteAccessZ	0,1, or 2		
COM1	dev/ttyS0	Assign a device to COM port	
COM2		Assign a device to COM port	
COM3		Assign a device to COM port	
COM4		Assign a device to COM port	
COM5		Assign a device to COM port	
TransportReconnect Enabled	true or false	Determines whether the connection is re-established after a connection termination	
TransportReconnect Retries	#retries	Number of connection retries	
TransportReconnect Delay	#delay	Seconds until a connection is retried	

ENTRY	VALUE/EXAMPLE	DESCRIPTION	MAND.
Hotkey1Char		Keys to switch applications	
Hotkey2Char			
Hotkey3Char			
...			
Hotkey15Char			
Hotkey1Shift			
Hotkey2Shift			
Hotkey3Shift			
...			
Hotkey15Shift			
DesiredColor	0,1, 2, 4, or 8	1: 16 colors, 2: 256 colors, 4: 16 bit colors, 8: 24 bit colors	
DesiredHRES		Horizontal resolution	
DesiredVRES		Vertical resolution	
ScreenPercent		Resolution percentage of the screen	
ApproximateColors	Yes or No	Color assignment for 256 colors: Yes: common (approximate colors), No: private (precise colors)	

ENTRY	VALUE/EXAMPLE	DESCRIPTION	MAND.
TransparentKeyPassthrough	Local	Pass local key entries to the server: Local: Keyboard entries are also evaluated by the local system, FullScreenOnly: Keyboard entries are only evaluated by the server for full-screen sessions, Remote: Keyboard entries are only evaluated by the server	
KeyPassthroughEscapeChar	Shift	Keyboard shortcut with which the evaluation of the keyboard can be stopped by the server only	
KeyPassthroughEscapeShift	F1		
PersistentCachePath	/tmp	Path for cache data	
PersistentCacheSize	0	Maximum cache size in KB	
PersistentCachePercent	20	Maximum cache size in percent	
PersistentCacheMinBitmap	2048	Minimum bitmap size in bytes to be saved	
TcpGroupName1	Hauptguppe	Name of the primary server group	
TcpGroupName2	sicher1	Name of the server group Backup list 1	

ENTRY	VALUE/EXAMPLE	DESCRIPTION	MAND.
TcpGroupName3	sicher2	Name of the server group Backup list 2	
UseAlternateAddress	0 or 1	Determines whether an alternative address is used for firewall connections	
ProxyHost	proxy:8080	Name and port of the proxy for secured connections	
ProxyType	None, Socks, Secure, or Auto	No proxy, SOCKS, Secure (HTTPS), Automatic	
AllowAudioInput	false or true	Determines whether audio input data is evaluated	

Delete software defaults (Type=9)

Deleting software defaults requires

- Section with entry `Type=9`
- Section with entry `FQOUName`

Section

ENTRY	VALUE/EXAMPLE	DESCRIPTION	MAND.
FQOUName	/OU1/OU2	Fully-qualified name of the OU for which the software defaults are deleted	✓

Set Advanced options (Type=10)

Adding or modifying Advanced options requires

- Section with entry `Type=10`

- Section with the relevant entries

Section

ENTRY	VALUE/EXAMPLE	DESCRIPTION	MAND.
DiscoverPingTime	10	Ping time to identify devices	
DiscoverCollectTime	30		
UpdateMaxAtOnce	10		
UpdateConnectTimeout	2		
ManageOnlyLockedFields	0		
DoSmartSrv	0		
SmartSrvTimeout	30		
ScoutSrvGroup	0		
RenameExistingDevice	1		
AllowDynamicHostnames	0		
DeactivateNewDevices	0		
WOLWithUDP	0		
WOLWithIpAddress	0		
DebugLevel	1		
AllowDynamicGroupID	1		
CheckNetCrossing	0		

ENTRY	VALUE/EXAMPLE	DESCRIPTION	MAND.
DiscoverOnStart	0		
PrinterImportTime	0		
UseClassicLogin	1		
Password			
RecoverURL			
WolServer			
LastApplicationAction	0		
LastApplicationActionDelay	0		
UsePeerIP	0		
FirstContactAction	0		
RootPartitionSize	128		
HomePartitionSize	0		
SwapPartitionSize	2		
ReceiveTimeout	5		
UpdateResponseTimeout	10		
RecoverUsePartitionData	0		
RecoverAskUser	1		
RecoverProxy			
RecoverPort	0		

Rename an application (Type=11)

Renaming an application requires

- Section with entry `Type=11`
- Section with the entries `FQOUName`, `Name`, `NewName`

Section

ENTRY	VALUE/EXAMPLE	DESCRIPTION	MANDATORY
FQOUName	/OU1/OU2	Fully-qualified name of the OU which contains the application	✓
Name	MyApp	Name of the application	✓
NewName	YourApp	New name of the application	✓

Query device data (Type=12)

Querying device data requires

- Section with entry `Type=12`
- Section with entry `AttributeKey`

Section

ENTRY	VALUE/EXAMPLE	DESCRIPTION	MAND.
AttributeKey	MAC_address	Key to search for in the database	✓
AttributeValueString	001122334455	Text value to which the content of the key is compared	
AttributeValueInt	001122334455	Numerical value to which the content of the key is compared	

Import advanced file entries (Type=13)

The import of advanced file entries can be done via SCMD for an individual device. It requires

- Section with entry `Type=13`
- Section with the relevant entries

Section

ENTRY	VALUE/EXAMPLE	DESCRIPTION	MAND.
DeviceName	M625q	Host name of the device for which the entry is set	✓
IniEntryFile1	/setup/terminal.ini	File to be written to	✓
IniEntrySection1	Security	Section	✓
IniEntryKey1	TestKey	Entry/Key	✓
IniEntryValue1	TestVal	Value	✓

For further information, see Adding individual file entries in the `Scout` guide.

To import additional file entries for the device, increment the number for the entry. Further file entries can be defined for the same file or for other files.

Example: The text file inientries.ini that is shown in the figure on the left produces the file entries shown on the right on the device after it has been processed via SCMD.

Accelerated batch mode

If you execute many SCMD commands in a batch file one after the other, you can accelerate the execution of the batch file by starting the SCMD program in batch mode.

Starting the batch mode

- Enter the following command:

The call starts an instance of the program in the background. All subsequent calls are then passed to this instance and processed by it. Initializations such as the connection to the database then only need to be run once, and execution is faster.

To close the background instance, exit the batch mode.

Terminating the batch mode

- Enter the following command:

Example

[scmd -startbatch scmd -u -p -d -sendmsg -msg -to -once -at 12.12.2019 12:34 scmd -u -p -d -sendmsg -msg -to -once -at 13.12.2019 12:34 scmd -u -p -d -sendmsg -msg -to -once -at 14.12.2019 12:34 scmd -stopbatch]

> **Note:**
>
> - The logon data must remain the same in all calls, otherwise the command is not executed.
> - Parallel execution of multiple batch files is not possible.

Scout Cloud Gateway

Scout Cloud Gateway is a solution for managing devices that are on the Internet and have to be connected to their Scout Server via VPN.

Overview

Unicon's Scout Cloud Gateway is a convenient solution for managing devices that are on the Internet and have to be connected to their Scout Server via VPN. Home office devices or mobile devices in changing environments can thus be conveniently and securely connected to the existing infrastructure without the need for a VPN backend.

The Scout Cloud Gateway (SCG) comes with a **VPN infrastructure** - the VPN connection is established and provided automatically. The first time an eLux® device connects, users are required to authenticate. After successful authentication, the SCG transmits the VPN configuration

and Scout Server address to the device. This communication uses the secure management protocol.

eLux RP 6 devices have the VPN software already on board.

For each device, the system creates a certificate for client authentication which the administrator can view, revoke or renew at any time.

The Scout Cloud Gateway is tailored to eLux device management via the Scout Enterprise Management Suite and supports environments with multiple Scout Servers. One gateway instance can connect around 1024 devices from the Internet. To connect a larger number of devices, you can install **multiple gateway instances** in a cluster. The instances exchange necessary information with each other. Clustering makes the Scout Cloud Gateway **scalable**: Thousands of devices in the home office can be connected smoothly.

A further advantage is that the devices can be used **dynamically both from the Internet and the Intranet**. With the appropriate network configuration, the devices can be easily integrated and operated in their environment via the Internet or Intranet. The system then automatically recognizes whether it is on the Internet or the Intranet. Only when used outside the Intranet for the first time, users must enter their logon data for the Scout Cloud Gateway.

This convenient option for dynamic use can also be used for preparing and rolling out new devices.

The Scout Cloud Gateway is offered as a Debian installer and as a virtual machine with an image using Ubuntu 20.04 (from SCG 1 2107).For further information, see System requirements.

To configure and manage your gateway, the SCG WebAdmin interface is provided.

Access control

To connect new devices, the Scout Cloud Gateway (SCG) needs to ensure that the user is authenticated. To do this, a connection to Active Directory or to the identity management system Keycloak can be set up. Keycloak allows user authentication strictly separated from the SCG. As a third variant, you can define tokens for access directly at the SCG.

After a user has been successfully authenticated, a certificate is issued for the corresponding device, which is checked against at the next logon. As long as the certificate is valid, no new user logon to the SCG is required. The RSA certificates have a key length of 4k. (from SCG 1 2209) Administrators may configure the renewal of certificates and get a quick overview of expiring

certificates in the Devices view and the Dashboard 2 (from SCG 1 2209) of the WebAdmin interface.

User authentication via Active Directory

Active Directory handles the authentication and the assignment of the device to its OU. To control the OU assignments, the AD environment must be configured in advance.

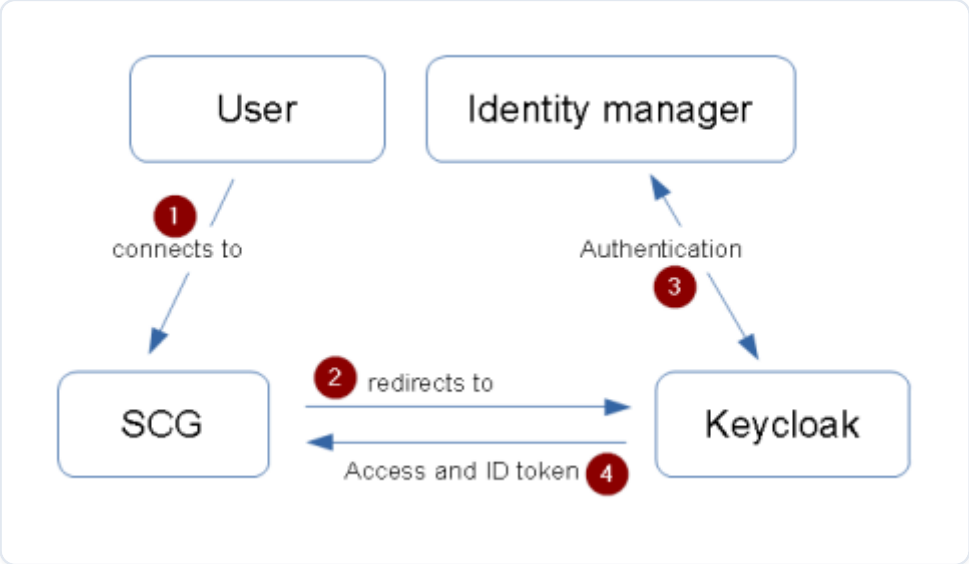
Authorized AD users then log on to the Scout Cloud Gateway once with their device. The AD logon data are evaluated by the AD server. Next to user authentication the devices are registered on the Scout Server in the OU specified by the user assignments.

In the **Devices** view of the WebAdmin interface, usernames are displayed with their domain.

User authentication via Keycloak

Keycloak is an implementation of the OpenID Connect protocol and can be used perfectly for connecting devices via the Scout Cloud Gateway. The user authentication is processed completely separated from the SCG and thus protects the logon data. Also, the Keycloak server must use signed certificates for TLS communication.

Keycloak also acts as an identity broker and supports user accounts from other identity and access management systems, including Active Directory.



The users are forwarded via the Scout Cloud Gateway (1) to Keycloak for logon (2). In the logon dialog of the identity manager defined in Keycloak, they enter their logon data, which are then checked by the defined identity manager (3). After successful authentication, Keycloak returns an access and ID token for the user to the SCG (4). This procedure completely isolates the SCG from the users' logon data.

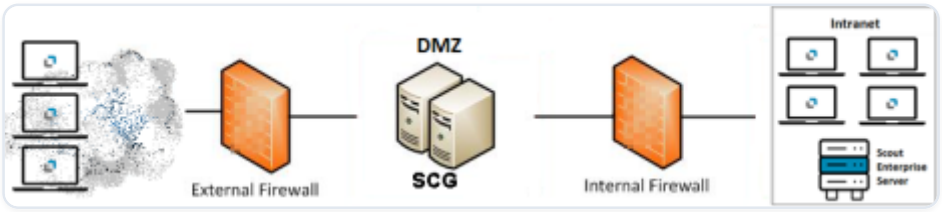
Users who are authenticated via Keycloak are always assigned to the standard OU defined in the Scout Console. In the **Devices** view of the WebAdmin interface, their usernames are displayed without a domain.

User authentication directly via Scout Cloud Gateway

Users receive a token from the administrator to register their device. The tokens are defined after installation in the SCG WebAdmin interface. To each token, you assign an OU in which the user's device is to be registered.

Scout Cloud Gateway in a DMZ

For the setup of the Scout Cloud Gateway in your network, we recommend the operation in a security-controlled 'Demilitarized Zone',



The following routes must be set up for configuration:

- From the eLux® devices to the DMZ

  You define these routes in the SCG WebAdmin interface under **Configuration > VPN device settings > Routes**.

- From the Scout Cloud Gateway to the Intranet

  For further information, see Static IP routes.

- From the Scout Server, AD server / identity manager and other required (application) servers back to the Scout Cloud Gateway in the DMZ

  For further information, see General network settings.

Use dedicated firewall rules to control access from the Internet to the DMZ and from the DMZ to the Intranet.

> **Note:**

> To use the Scout Cloud Gateway as a router between DMZ and Intranet, you can use a second network card. In the VM installation, two network cards are already considered and can be configured.

> Note the following

- If you intend to use multiple SCG instances in a cluster, they must also be able to communicate with each other.
- If you define a gateway and DNS only for the internal network, make sure that the names of the Scout Server and AD server / identity manager can be resolved.

Example for configuring routes

Premises:

(A) eLux devices are supposed to access resources on the Intranet (network 192.168.50.0/24).

(B) The SCG is located in a DMZ with network 192.168.100.0/24 and connects eLux devices via tunnel network 10.8.0.0/24.

(C) Both networks are connected via a router/gateway, with address 192.168.100.1 in the DMZ and 192.168.50.1 in the Intranet.

Required routes:

A) Network: 192.168.50.0, subnet mask: 255.255.255.0 (gateway is the VPN server of the SCG)

B) Network: 192.168.50.0, subnet mask: 255.255.255.0, gateway from DMZ to intranet: 192.168.100.1

C) Network: 10.8.0.0, subnet mask: 255.255.255.0, gateway from the intranet into the DMZ: 192.168.50.1

For further information, see General configuration.

System requirements

General requirements

- System resources
  - 4 CPUs

  - 8 GB RAM
  - 24 GB hard disk space

- Network
  - One IP address in the VPN address range for each connected eLux® device (from SCG 1 2209)
  - Internal HTTPS access to the SCG WebAdmin interface (Interface for configuration and management)
  - Bi-directional connection (via network card or firewall rules) of eLux devices to the Scout Server and of the Scout Cloud Gateway to the Identity provider
- Scalability by means of clustering

  If required, multiple machines can be installed as SCG servers. 1024 devices can be connected per server.

You can set up your instance with the help of a Debian Installer under a Linux system. Alternatively, a ready configured virtual appliance is available. In both cases, the network must be configured before the SCG is started to enable routing from the DMZ to the internal network. The virtual machine assists with the setup.

	DEBIAN PACKAGE	VIRTUAL MACHINE
Download	Debian package (.deb)	Virtual appliance (.ova)
		The virtual machine is installed using an .ova template (Open Virtual Appliance).
Required system	Configured Debian Linux system	Virtualization platform such as VMware
	We recommend Ubuntu 20.04.	
Specifics		Wizard supports system configuration
		Additional functions in the WebAdmin interface

	DEBIAN PACKAGE	VIRTUAL MACHINE
		• Upload public keys for SSH access to the instance
		• Upload individual HTTPS certificates for the WebAdmin interface connection (to replace the automatically created self-signed certificate)

Availability from the Internet

- **Management protocol** between eLux devices and Scout Server

For connecting the devices, the SCG server should be reachable from the Internet as FQDN. The default port for communication is **22125**.

If you are running multiple SCG instances in a cluster, we recommend that you distribute the requests to the different instances via router.

- **VPN port for connecting devices**

For cluster operation: Each SCG instance requires a unique combination of FQDN and port for its VPN server.

The administrator performs the configuration in the SCG WebAdmin interface, see Gateways.

Internal availability

- **HTTPS Port 443** for connecting the WebAdmin interface (configuration, clustering, management)
- If Active Directory or Keycloak is used, the servers must be reachable
- Only virtual machines (.ova): **port 22 for access via SSH** to the instance for administrative access (optional)

- For cluster operation: **MongoDB port 27017** for internal communication of the databases across the SCG instances

> **Note:**

> Make sure that your Linux system is always up to date. To do so, use the standard commands **apt-get update** and **apt-get upgrade**.

> For support periods and the compatibility matrix see the Whitepaper **Releases, Lifecycles and Compatibility**.

Installation

The Scout Cloud Gateway (SCG) may either be installed and run as a Debian package on Linux or operated on a virtual machine (VM).

For the **VM** installation, you need a virtualization platform. Unicon™ provides an OVF template (.ova) that you use to install your VM. The MongoDB database and web server used by the Scout Cloud Gateway are installed and preconfigured with it.

The **Debian** variant is installed with a Debian package. You install the MongoDB database yourself, the configuration for database and web server is supplied.

To configure your SCG instance, connect to it via a web browser and use the SCG WebAdmin interface.

To give users and their devices access to the corresponding OUs, define AD groups with a defined prefix and the OU ID in your Active Directory system. Then assign the users to them. Alternatively, configure Keycloak for authentication or create SCG accounts. For further information, see General configuration.

The devices will be integrated via the eLux® First Configuration Wizard.

Scout Cloud Gateway as a Debian machine

The Scout Cloud Gateway may be operated in a Linux environment. A Debian package (.deb) (tested with Ubuntu 20.04) is available for this purpose.

The Scout Cloud Gateway works with a **MongoDB** database. As a web server, you can use **NGINX** or **Apache**.

> **Note:**
>
> Beginning with SCG 1 2209, clusters are supported also for a Debian installation.
>
> The connection between WebAdmin interface and web server is always made via HTTPS (from SCG 1 2209 - Earlier versions do not support HTTPS). The certificate for this is automatically created during installation.

Networking requirements

For the routing to work, please make sure that IP forwarding is enabled. The value for `net.ipv4.ip_forward` needs to be set to `1`, using the following command.

```
sysctl net.ipv4.ip_forward=1 -w
```

To ensure the IP forwarding setting persists after a reboot, create the file `99-scg-ip-forwarding.conf` in the folder `/etc/sysctl.d/` with the following content:

```
net.ipv4.ip_forward=1
```

Debian package dependencies

A Debian package is available for installation in a Linux environment.

- **MongoDB** version 4.4

  The MongoDB database is preconfigured (from SCG 1 2209). For versions < SCG 1 2209, a MongoDB user for authentication, MongoDB database settings and a replica set with minimum one member must be configured).
- Web server **NGINX** or **Apache 2.x** Configuration data for both variants is supplied and must be activated after installing the SCG Debian package. In the example, we will use NGINX.
- **Node.js** major version 16
- **OpenVPN** for the VPN tunnel to the eLux® devices
- **jq** json tool, **moreutils**, **logrotate** and **zip** to install the Debian package (2-for versions < SCG 1 2209 also debconf-utils).

Installing the tools and database

> **Important:**

> The SCG machine's hostname may not be changed after the MongoDB database has been installed and configured. If required, change the hostname before.

> The following instructions use Ubuntu 20.04 and NGINX as a web server.

1. Add the package sources for MongoDB.

```
echo "deb [ arch=amd64 ] https://repo.mongodb.org/apt/ubuntu focal/
mongodb-org/4.4 multiverse" | tee /etc/apt/sources.list.d/mongodb.list

wget -O /tmp/mongo.asc https://www.mongodb.org/static/pgp/
server-4.4.asc

apt-key add /tmp/mongo.asc
```

2. Add the package sources for Node.js.

```
wget -O /tmp/nodeSetup.sh https://deb.nodesource.com/setup_16.x

chmod a+x /tmp/nodeSetup.sh

/tmp/nodeSetup.sh
```

3. Install MongoDB, Node.js and further dependencies.

```
apt-get update

apt-get install mongodb-org nodejs nginx openvpn jq moreutils zip
logrotate
```

4. Check for the MongoDB version. The version must be **4.4**.

```
mongo --version
```

5. Check for the Node.js version. The major version must be **16**.

```
node -v
```

Note:

Only earlier versions (1 from < SCG 1 2209): Configure the database in the next step, see Configuring MongoDB for SCG. For later versions, install the Debian package now.

Installing the Debian package and configuring the web server

REQUIRES
Node.js and further tools are installed. The MongoDB database is installed and configured.

1. From our myelux.com portal, under **Downloads > Scout Cloud Gateway**, download the SCG Debian package `scg_< version>_amd64.deb`.
2. Install the Debian package by using the following command:

   ```
   dpkg -i scg_< Version>_amd64.deb
   ```

   a. Start the SCG service.

   ```
   systemctl start scg-controller
   ```

   b. Optionally, configure the components for automatic system start-up.

   ```
   systemctl enable  'nginx' 'cron' 'scg-controller'
   ```

3. Configure the NGINX web server:

   By default, the SCG installs a functional web server configuration under the path `/etc/nginx/conf.d/scg.conf`. Here, self-signed certificates are linked for initial communication.

   a. Optional: Replace the self-signed certificates with your own. To do so, edit the file `/etc/nginx/conf.d/scg.conf` and replace the certificate files

```
ssl_certificate /opt/unicon/scg/certs/http/certificate.crt;

ssl_certificate_key /opt/unicon/scg/certs/http/certificate.key;
```

b. Make sure that the SCG configuration is not overwritten by the NGINX default configuration.

```
rm /etc/nginx/sites-enabled/default
```

c. To customize the NGINX user, edit the file `/etc/nginx/nginx.conf` (from SCG 1 2209)

```
user scg-web scg-web;
```

d. Restart the service.

```
systemctl restart nginx
```

The installation is complete. In the next step, use a browser to log on to the SCG WebAdmin interface and configure your SCG instance. For further information, see Gateway configuration in the SCG WebAdmin interface.

- After you have configured your SCG in the SCG WebAdmin Interface, to check whether all components are running, use the following command:

```
systemctl status --no-pager --lines=0 'scg-mongod' 'nginx' 'cron' 'scg-openvpn' 'scg-backend' 'scg-controller'
```

Uninstalling the Debian package

To uninstall the Debian package, use the following command:

```
apt-get remove scg
```

To additionally delete the configuration data of the last installation, append the `--purge` parameter.

Scout Cloud Gateway as a VM

The Scout Cloud Gateway is provided as a virtual machine. Based on the OVF template (.ova), you install a virtual machine that you can vary the configuration of.

The following WebAdmin interface features are supported in VM installations:(but not in a Debian installation)

- Upload public keys for SSH access to the instance
- Upload individual HTTPS certificates for the WebAdmin interface connection (to replace the automatically created self-signed certificate)

Installing the VM

REQUIRES
Virtualization platform and computing resources (4 CPUs, 8 GB RAM, 24 GB hard disk) for installing virtual machines based on an [.ova] template (Open Virtual Appliance)

1. From our myelux.com portal, under **Downloads**> **Scout Cloud Gateway**, download the current `.ova` template.
2. Within your virtualization platform, install a virtual machine from the template `scg-<version>.ova`, for example by selecting **Deploy OVF Template**.
3. Optionally, assign a host name for the new MAC address via DHCP. Otherwise, specify a name in the next step on the machine along with first network settings.
4. Power on your machine.

In the next step you can set initial network settings.

Setting first configuration settings on VM

Configure the root password, host name and network cards directly on the machine.

To operate the in a DMZ, you may configure two network interfaces for internal and external LAN. For further information, see Scout Cloud Gateway in a DMZ.

1. Start a remote console of the VM.

   > **Note:**
   >
   > The English keyboard language is active.

2. Use the following keys for navigation:

KEY	DESCRIPTION
`Tab`	Jump to next element
`Space`	Enable or disable selected option
`Return`	Trigger selected button

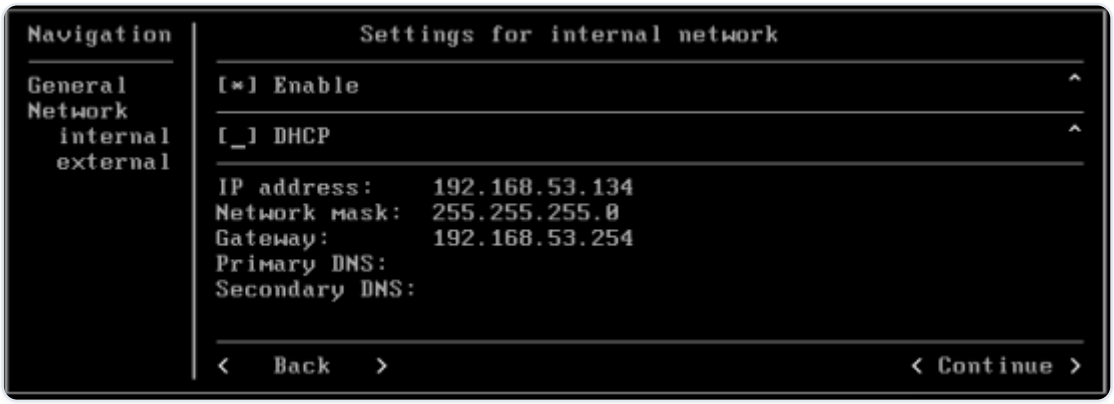
3. Under **General**, set and confirm an individual root password.
4. Under **Network**, optionally change the host name of your SCG machine.

   If you enter the host name as FQDN, avoid using `.local` as the domain.

   > **Important**
   >
   > The host name must not be changed later on.

5. Under **Network**, specify your NTP server.
6. Configure the internal and external network settings. Under **internal** and **external**, define whether the respective network card is used and whether DHCP is to be used. If you disable DHCP, you can configure the IP address and other data yourself.

   The first network card is defined as **internal**, the second as **external**.

   For further information on two network interfaces in the DMZ, see Scout Cloud Gateway in a DMZ.

Then, use a browser to log on to the SCG WebAdmin interface. (Up to SCG 1 2201, the system needed two to three minutes to complete the configuration.)

Connecting and logging on to the SCG instance

To manage your Scout Cloud Gateway, the WebAdmin interface is provided.

> **Note:**
>
> The system uses a secure HTTPS connection to the web server, for which a self-signed certificate was created during installation (from SCG 1 2101). We recommend that you replace the self-signed certificate with your own certificate.

1. In your browser, connect to your machine via HTTPS. For the URL, you can use the host name or the IP address. Example: `<https://192.168.54.24>`
2. Log on to the SCG WebAdmin interface with the username admin and initial password elux.
3. Enter a new password.
4. In the next step, in the **Configuration** view, configure your gateway.

The SCG WebAdmin interface is initially displayed in the language configured in the browser settings (German or English). For further information, see Logon and language.

In addition to the **Configuration** view, the WebAdmin interface offers the **Dashboard**, (from SCG 1 2209), **Gateways** and **Devices** views.

Upgrading to later versions

> **Note:**
>
> We recommend that you make a backup before you upgrade.

Via installation file

- for Debian and .ova installations -

> **Note:**
>
> This procedure works for upgrades from SCG 1 2201 to SCG 1 2209 and for later versions.

1. From our myelux.com portal, under **Downloads > Scout Cloud Gateway**, download the current SCG Debian package `scg_< version>_amd64.deb`.
2. Copy the file to your SCG instance and connect to it via SSH.
3. Enter the following command:

```
dpkg -i scg_< Version>_amd64.deb
```

> **Note:**
>
> During installation, the package manager checks for configuration changes. This is the case for the NGINX web server configuration. Apply the maintainer version and add any custom changes manually.

4. Only Debian: Check the web server configuration and restart the web server. For further information, see Installing the Debian package and configuring the web server.
5. If you have multiple instances, update all instances one by one using the above procedure.
6. Connect to an instance via WebAdmin interface and check whether the gateway configuration has been correctly applied.
7. Also, reset the browser cache to update outdated data. For example, use the key combination CTRL+F5 (Firefox).

Via cluster process

(for .ova installations < SCG 1 2201)

Initial situation: You have an existing Scout Cloud Gateway installation with minimum one instance in use. The registered devices are connected to the SCG server via a DNS entry. When updating to a later SCG version, you want the devices to remain on-boarded and configured as they were before.

To upgrade the version of an SCG installation (A) to a later version (B), create a new instance with version B and then transfer the configuration from A to B using the cluster function. To do so, perform the following steps.

1. Install an additional virtual machine (.ova) with the new Scout Cloud Gateway version B, see Installing the VM.

2. On the machine, configure the hostname and networks, see Setting first configuration settings on VM.
3. Log on to the WebAdmin interface of the new instance B via a browser, see Connecting and logging on to the SCG instance.

   > **Note:**
   >
   > Please do not perform any configuration of instance B. The configuration is automatically applied through the synchronization of the cluster. Instance-specific configuration is done after clustering.
   >
   > Browse to **Configuration > Cluster**.

4. At the same time, log on to the WebAdmin interface of instance A and open **Configuration > Cluster**. Copy the following values from instance A and paste them into the configuration of instance B:

   `Database password`

   `Cluster key`

   These two values ensure that instance B is added to the existing "cluster" and thus receives the configuration of instance A. Instance B connects to the existing MongoDB database.

5. Save the configuration of instance B and give it at least 10 minutes.

   > **Important:**
   >
   > Do not refresh the browser view of instance B and keep both VMs powered on.
   >
   > *After the system has saved the new configuration, both instances are visible in the **Gateways** view of both WebAdmin interfaces. Instance B has applied the configuration of instance A except for the gateway configuration.*

6. Configure the gateway settings for instance B. To do so, select instance B from the list box under **Configuration > Gateways** and configure the VPN connections. For further information, see Gateways.

*Both gateway instances are now shown with their properties in the **Gateways** view.*

7. In the **Gateways** view, deactivate instance A, see Gateways view.
8. Shut down the virtual machine A.
9. Set your DNS entry to the IP/host name of instance B.
10. Verify whether instance B allows for the client connections. To do so, you may use the following options:
    - The **Gateways** view must show the public host name and port of instance B.
    - In the WebAdmin interface of instance B, perform a health check. All components must be up and running.
    - On-board a new device or restart a connected device. The device must be assigned to gateway B in the **Gateways** view.
11. In the Gateways view, delete instance A.

Instance B has now adopted the entire configuration of instance A, including the devices. As soon as a device restarts, it connects to the new instance B.

> **Note:**
>
> Also use this procedure if you want to change the host name of a gateway instance (without updating to a later version).

General configuration

Configuring your Scout Cloud Gateway consists of various tasks such as preparing user authentication or setting general network settings.

Subsequently, configure your SCG instance in the SCG WebAdmin interface.

Configuring Active Directory environment

All users whose devices are to be connected to their Scout Server via the Scout Cloud Gateway must be authorized via AD groups and assigned to an OU.

1. In your AD system, create a group for each OU of a Scout Server that you want users to access through the gateway.

The group names must have the following format:

Prefix that is defined in the SCG WebAdmin interface (`SCG_Group_`) and the ID number of the OU (`42`)

Example: `SCG_Group_42`

The prefix identifies groups for the Scout Cloud Gateway.The devices used by the members of the created group to log on are automatically assigned to the defined OU ID. The prefix is predefined as SCG_Group_ and can be changed in the gateway configuration of the SCG WebAdmin interface.

The ID of an OU is shown in the Scout Console in the **Properties** window.

2. Add the relevant AD users to the defined groups.

> **Note:**
>
> Avoid assigning a user to multiple groups with this prefix, since a device can only be assigned to one OU at a time.

Users are authorized to access their Scout Server via the Scout Cloud Gateway. They can register with their devices as soon as the gateway is configured. The devices will be assigned to an OU via the AD users and their membership in a specific AD group.

Configuring Keycloak

> **Note:**
>
> Keycloak support is deprecated. Please do not use it anymore.

> Keycloak is an open source solution for identity and access management and is based on the OpenID Connect protocol. In order to use Keycloak for user logon, Keycloak must be configured on the server side and a Keycloak client must be installed.

All users whose devices are to be connected to their Scout Server via the Scout Cloud Gateway must be authorized via Keycloak roles.

1. Set up a Keycloak server.

> **Note:**
>
> To protect sensitive data, only HTTPS configuration is supported. For this you need a CA certificate.

2. In the Keycloak administration console, create a realm for your environment.
3. Create a Keycloak client for your application. For the Keycloak client, edit any options labeled on the figure below.

OPTION	DESCRIPTION
Settings > Client ID	Name of your client / application
Settings > Valid Redirect URIs	One or more SCG addresses to which Keycloak forwards users after successful authentication

4. Within your client, create roles and users. Assign a role to each user.

Users are authorized to access their Scout Server via the Scout Cloud Gateway and can register their devices once the gateway is configured. The devices are assigned to the default OU defined in the Scout Console.

General network settings

Routes in the network

In the relevant environment, configure the destination IP address and address range of the following connections to the eLux® devices you want to route via the secure communication of the Scout Cloud Gateway:

- Scout Server
- Scout Console (mirroring)
- Web server (images and eLux software packages)
- DNS server

Example for one instance: `route -p add 10.8.0.0/24 192.168.16.5`

For clusters, you need more routes accordingly.

Dynamic use of devices in the Internet and intranet

If you want to connect mobile devices both via Internet and intranet, configure the same FQDN for the Scout Server and the Scout Cloud Gateway. As a result, the FQDN in the intranet is resolved directly to the Scout Server. If the eLux device is used in the Internet, the FQDN will be resolved to the Scout Cloud Gateway. For further information, see DHCP configuration in the **Scout** guide.

On each restart, the device checks whether the IP address of the resolved FQDN is a Scout Server or a Scout Cloud Gateway.

> **Notes:**
>
> - If you use different FQDNs for the Scout Server and Scout Cloud Gateway, note that each time the operating environment changes between the intranet and Internet, a factory reset of the eLux device is required.
> - For VPN connections, a separate web server for firmware updates can be configured in the Scout device configuration.

Integrated VPN

For the Scout Cloud Gateway, OpenVPN is operated in routing mode (TUN device). This operating mode is recommended as the more efficient method: The VPN operates in layer 3 carrying only IP packets, which reduces bandwidth load.

For each connected device, one IP address is required (subnet topology) (from SCG 1 2209).

For a given number of devices to be managed, determine the required subnet mask that filters your IP address (VPN range). In **Configuration > Gateways**, for your instance, enter the subnet mask in dotted decimal notation.

Examples:

SUBNET MASK/CIDR	SUBNET MASK/DOT-DECIMAL	MAX. NUMBER IP ADDRESSES
`/27`	`255.255.255.224`	`30`
`/26`	`255.255.255.192`	`62`

SUBNET MASK/CIDR	SUBNET MASK/DOT-DECIMAL	MAX. NUMBER IP ADDRESSES
`/25`	`255.255.255.128`	`126`
`/24`	`255.255.255.0`	`254`
`/23`	`255.255.254.0`	`510`
`/22`	`255.255.252.0`	`1022`

The maximum number of available IP addresses corresponds to the maximum number of devices that can be connected.

The device manages the leases itself and acts like a miniature DHCP server. One lease is valid for 10 minutes.

In earlier versions, (< SCG 1 2209) the **net30** OpenVPN topology is used and the IP address range must be large enough to provide each device with four IP addresses. Each device receives its own network with four IP addresses (network, broadcast, virtual IP and client IP) and with a `/30` subnet mask.

> **Note:**
>
> In current eLux versions, OpenVPN in version 2.4.7 is used.

Static IP routes

To ensure the accessibility of your resources such as Scout Server, Active Directory, NTP and DNS, add static IP routes to the SCG machine. Static routes can be set for the external and internal network interface on the instance in the relevant configuration file provided:

`/etc/systemd/network/00-external.network`

`/etc/systemd/network/00-internal.network`

The syntax to be used can be found in the freedesktop documentation for the **systemd-networkd** service `<https://www.freedesktop.org/software/systemd/man/systemd.network.html>`

under Section Options.

Adding static routes

1. Connect to the SCG instance via SSH.
2. Edit the configuration file for the relevant network card and insert a section for each route according to the following example:

```
[Route]

Gateway=192.168.0.40

Destination=40.0.0.0/8

GatewayOnlink=yes
```

> **Note:**
>
> GatewayOnlink is required when you use DHCP and ensures that the route is set after the DHCP settings.

3. Restart the machine. Alternatively, restart the **systemd-networkd** service using the following command:

`systemctl restart systemd-networkd`

To check the result, use the command ip route. Example:

`root@xxx-scg-1:~# ip route`

For the example above, it provides the following information : `40.0.0.0/8 via 192.168.0.40 dev internal proto static onlink`

Gateway configuration in the SCG WebAdmin interface

The configuration of your Scout Cloud Gateway includes several areas. The settings for authentication and connection of devices are required for successful on-boarding of new devices. Other settings, for example for forming clusters or accessing the instance itself, are optional.

- Log on to the SCG WebAdmin interface of your SCG instance and switch to the **Configuration** view.

The options are grouped in multiple sections.

> **Important**
>
> Your changes will not be saved until you click **Save**.

User authentication / Device certificates

(from SCG 1 2201)

For each device, after the onboarding process, the system creates a client authentication certificate so that users do not need to perform an additional logon for the SCG. For further information, see Device certificates.

Configure the validity of the device certificates and the behavior when the validity expires. Expiring certificates can be renewed automatically. Alternatively or additionally, you may manually renew certificates for selected devices in the **Devices** view.

1. Validity period for a device certificate in days.

   The minimum value must be 30 days and the maximum value must be 5 years (1825 days).
2. Device certificates are automatically renewed before their validity expires.(Once users re-connect with their device)

   This option is deselected by default.
3. Certificates that have already expired are also renewed automatically.

For further information, see Device certificates.

User authentication / Identity provider

Select and configure the identity manager you use for user authentication.

Active Directory

	OPTION	EXAMPLE	DESCRIPTION
(A)(The numbering (A)-(E) refers to the graphical representation of the connection process,	Server	`ldaps://dc.int.sampletec-01.com:636`	LDAP protocol and FQDN, or the IP address of the AD server (Domain Controller)

	OPTION	EXAMPLE	DESCRIPTION
see Integrating new devices)			
			For LDAPS, you need a Base64-encoded CA certificate, see below.
	Base	`DC=int,DC=sample tec-01,DC=com`	LDAP search base of the Active Directory domain
	Group template	`SCG_Group_`	Prefix used for the newly created AD groups
			The prefix is used to authenticate users and assign devices to OUs of the Scout Server.
			Only users who are members of an AD group with this configured prefix can log on to the Scout Cloud Gateway.
	CA certificates		Base64-encoded CA certificate (.pem for secure LDAPS communication (LDAP over SSL)
			To convert from .der to .pem, use openssl
			`openssl x509 -inform der -in`

	OPTION	EXAMPLE	DESCRIPTION
			`cert.cer -out cert.pem`

- To check whether you can connect to the AD service with the specified AD data (without a new route), click **Test connection**. Enter the username and password of a test user and click **Connect**.

> **Note:**
>
> The username must be specified as a user principal name (UPN). Example: mailto:mmi@int.sampetec-01.com

OpenID Connect / Keycloak

> **Note:**
>
> Keycloak support is deprecated. Please do not use it anymore.

> View the configuration data for your Keycloak client in the Keycloak administration console under **Installation > Format Option: Keycloak OIDC JSON**

For further information on configuring Keycloak, see Configuring Keycloak.

	OPTION	EXAMPLE	DESCRIPTION
	Keycloak realm	`ScoutEnterpriseCloudGateway`	Name of the realm you have created in the Keycloak administration console for your environment
	Keycloak server auth URL	`https://keycloak.int.sampletec-01.com:8443/auth`	Server authentication URL as specified in your Keycloak client

	OPTION	EXAMPLE	DESCRIPTION
			on the **Installation** tab
			Note: For authentication via Keycloak, only HTTPS is supported. For this you need a CA certificate.
	Keycloak client ID	`SCG`	Name of your Keycloak client you have specified in Keycloak under **Settings > Client ID**
	Keycloak client secret	`u14n936i-2642-43fc-o911-8n5de21a0e76`	Key generated by Keycloak for your Client ID, in Keycloak under **Credentials > Secret**

Scout Cloud Gateway authentication

The Scout Cloud Gateway supports a token-based logon for the registration of new devices. You may create as many tokens as you like and give them to users to register their device.

You can have a token created automatically or create it manually.

- To create a new token, click **Add token**. Assign an OU to the token.

1. Select Identity provider **Scout Cloud Gateway**.
2. Enter a new token (option A) A token must have a minimum of 16 and a maximum of 50 characters. Allowed characters:
   - alphanumeric characters `(a-z, A-Z, 0-9)`
   - special characters `(!?#%&()<=>@\^~*+,./:;_-)`

3. Generate a new token (option B)(from SCG 1 2209. In previous versions, for each new entry a token is generated automatically which you may replace with individual names (16 characters in length))
4. ID for the OU in which the user's device is to be registered during on-boarding
5. Delete token / entry
6. Create a new entry

User/password authentication

Up to eLux RP 6.10, the definition of SCG users with password was provided. For compatibility reasons, this method is still supported.

- Select **Support user/password authentication** and add new user entries.

Server device settings and VPN device settings

Server device settings

	OPTION	EXAMPLE	DESCRIPTION
(D)(1-The numbering (B)-(E) refers to the graphical representation of the connection process, see Integrating new devices.)	Scout Servers	`scout.int.sampletec-01.com`	FQDN or IP address of the Scout Server to which the devices need to be connected
			Multiple entries are allowed if there are several servers with access to the same Scout database.
	Gateway certificate		Certificate and key file for the communication between devices and Scout Cloud Gateway

	OPTION	EXAMPLE	DESCRIPTION
			Only required if you use an individual CA certificate to encrypt the management protocol
			For further information, see Certificate-based management protocol in the **Installation** short guide.
			The certificate must be issued for the gateway server.

VPN device settings

	OPTION	EXAMPLE	DESCRIPTION
(B)	Name servers (DNS)	`192.168.17.2`	IP address of the name server (DNS) that transmits the Scout Cloud Gateway to the eLux devices, enabling them to resolve the FQDN of the Scout Server name
			Multiple entries are allowed.
(C)	Routes	`192.168.17.0`	IP address range for which eLux will use the secure communication via the Scout Cloud Gateway

	OPTION	EXAMPLE	DESCRIPTION
		`255.255.255.0`	Use this option for additional Scout Consoles
			Multiple entries are allowed.
(E)	VPN IP address range	`10.8.0.0`	Is configured in the **Gateways** view
		`255.255.255.0`	

SCG cluster

(only versions < (SCG 1 2209)

Multiple gateway instances in the same network can form a cluster. Communication between the instances is secured by a cluster key:

> **Important:**

> Before you change a cluster key, make sure that at least one other instance uses the new cluster key.

> By default, new instances are not clustered. When a new gateway instance is installed as a VM (.ova), a unique database password and a unique cluster key are generated. Both should be changed only to form a cluster.

To use multiple gateway instances as a cluster, see Defining a new cluster instance.

> **Note:**

> Clustering of instances is not supported in the Debian variant of the SCG (< SCG 1 2209).

SCG instance access

(only for VM installation)

For administrative access to your Scout Cloud Gateway instance via SSH, the public key of a generated key pair is required.

- To generate the key, use a suitable tool (such as PuTTY) and then copy the key content into the **Public key** field.

Multiple entries are allowed.

Gateways

(from (SCG 1 2101)

Configure each new instance after installation in the **Configuration** view under **Gateways**.

VPN connections of a gateway instance

Each new SCG instance must be reachable for the target devices via VPN.

1. Select your new instance.
2. Variants for multiple gateway instances: Same host name and different ports or different host names and same port.
3. VPN port
    - Use a free port. The system does not check for existing assignments.
    - Open the port in your firewall.
4. Range of IP addresses that will be assigned by the Scout Cloud Gateway for secure communication with eLux® Example: `10.0.10.0`
5. Subnet mask for the IP address range Example: `255.255.255.0`

Web server configuration

During installation (from SCG 1 2209 also for the Debian installation) the system automatically creates a self-signed certificate that secures the connection between the web server and the of the SCG instance (from SCG 1 2101). This allows the administrator to perform further configuration using the WebAdmin interface - and an HTTPS connection - immediately after installing an SCG instance. The self-signed certificate is valid for 365 days and is renewed automatically.

If you have an individual certificate, replace the self-signed certificate. To do so, click **Select file** (only for VMs (.ova installation)).

For multiple gateway instances, you can either use a separate certificate for each instance and assign it in the respective WebAdmin interface. Alternatively, use a wildcard certificate for all instances, if this is applicable within your structure.

SCG administrators

(from SCG 1 2103)

The Scout Cloud Gateway comes with an **admin** account that has full access. Optionally, you can define additional administrator accounts. To each administrator, assign one of the following roles:

ROLE	RIGHTS
Admin	Unrestricted access
Operator	Access is restricted to
	the Devices view including certificate revocation
	the Info dialog with download link for the diagnostic files
	the User profile dialog

Defining new administrator account

1. Under **SCG administrators**, click **Add user**
2. Enter a username and a password for it. Repeat the password.
3. Select either the **Admin** or **Operator** role.
4. Confirm with **Save**.

The password is stored as a hash value in the database. The new administrator can log on immediately with his credentials.

1	Define a new account

2	Specify its username and password
3	Select a role
4	Delete this account

All accounts defined by you are listed with their username and role

In the **User profile** dialog the logged on administrator can see with which account and role he is logged on and since when.

Logging

1	Choose extent of diagnostic files
	In addition to SCG software components, the log level affects the OpenVPN logs (from SCG 1 2103)
2	Configure thresholds for log rotation
	To download the diagnostic files, use the Info dialog.

Log level

(up to SCG 1 2101 under **SCG cluster**)

- Choose the extent to which log and configuration files are retrieved and downloaded when you request the **Diagnostic files**.

Error	Only error messages
Info	Error messages and additional information (default)
Debug	Log files with extended content

The log files are stored under `/opt/unicon/scg/logs`

If you are running a cluster of multiple SCG instances, you only need to set the log level on one instance: the setting is automatically applied to the other instances. A restart for activation is not required.

Log rotation

(from SCG 1.2103)

Especially for log level Debug the amount of data increases quickly. Rotating the log files keeps the storage demand low. For each log file, the number of backup files and the size is limited. When the threshold values are reached, the files are rotated: A new file is created and the oldest backup file is deleted.

Log rotation is active by default and is automatically executed once per day.

The threshold values can be configured as follows:

OPTION	DEFAULT	VALUE RANGE
Maximum log file size in MB	10	10-100
Maximum number of backup files	5	5-10

The configured values are saved to the file logrotate.conf. This file and all rotated files in /opt/unicon/scg/logs are part of the **Diagnostic files**.

For further information, see Downloading diagnostic files.

Monitoring

(from SCG 1 2107)

The health check service allows administrators to check and monitor the availability and performance of an SCG instance and its internal components (backend, controller, Mongo controller, OpenVPN...) A health check request provides the overall status and the status of individual services as a result.

By default, the service is not active, but must first be started under **Monitoring > Health check**.

The health check is initially secured via an automatically generated token.

1	Activate health check
2	Optional: Define individual password
3	First save configuration, then click for a test request
4	Output: Overall status and status of each component

Upon request, the system generates status information in format on all relevant components.

- Status `0` means: No errors.
- If the overall status is `0`, all components are running.
- The time stamp specifies from when a component is up / down.

Calling the health check service

For a call, an HTTP GET is executed on the health check URL with the configured token given with it.

Example with curl:

```
curl 'https://<SCG URL>/healthcheck' -H 'Connection: keep-alive' -H
'Authorization: Bearer <token>' -H 'Accept: application/json, text/
plain' --insecure
```

The following parameters are used:

PARAMETER	DESCRIPTION
`<SCG URL>`	URL of your Scout Cloud Gateway
`<token>`	Token or password as defined in the SCG WebAdmin Interface under **Configuration** > **Monitoring**
`--insecure`	Needs to be specified if you use a Self-Signed certificate

For example, if Open VPN is not running, the result is as follows:


```
{
  "status": 1,
  "backend": {
    "status": 0,
    "since": "Tue 2021-10-19 12:21:24 UTC; 1 weeks 2 days ago",
    "message": "OK"
  },
  "controller": {
    "status": 0,
    "since": "Tue 2021-10-19 13:30:21 UTC; 1 weeks 2 days ago",
    "message": "OK"
  },
  "mongod": {
    "status": 0,
    "since": "Tue 2021-10-19 12:20:58 UTC; 1 weeks 2 days ago",
    "message": "OK"
  },
  "nginx": {
    "status": 0,
    "since": "Tue 2021-10-19 12:20:56 UTC; 1 weeks 2 days ago",
    "message": "OK"
```

```
  },
  "openvpn": {
    "status": 3,
    "since": "Wed 2021-10-20 07:08:52 UTC; 1 weeks 1 days ago",
    "message": "ERROR"
  }
}
```


Multiple gateway instances as a cluster

Note:

Clustering of instances is supported also for Debian installations from SCG 1 2209.

By using clusters you achieve higher reliability and availability. If one Scout Cloud Gateway instance fails, another takes over. The second advantage is load balancing. Load balancing is controlled via the open VPN connections: When a device logs on, it is always forwarded to the gateway instance with the lowest load, meaning the lowest number of connections.

Note:

For reliable failover and fault tolerance, at least three instances are required in a cluster. This means that three databases (connected by clustering) are available on independent SCG instances in a so-called replica set. If you need more than three instances in a cluster, always use an uneven number.

Per instance 1024 devices can be connected.

To use the Scout Cloud Gateway to connect devices from the Internet, there are several possible scenarios:

- Scout Cloud Gateway on one host as a single instance
- Multiple gateway instances as a cluster
- Operate multiple gateway instances in different networks separately
- Group multiple gateway instances into several clusters

Once you have multiple gateway instances on the same network, they can form a cluster. Communication between the instances is secured by a cluster key. In previous versions, this cluster key has to be added for a new instance to be part of the cluster.(see Defining a new cluster instance (earlier versions)) From SCG 1 2209 onwards, this is done in the background:

To add a newly installed gateway instance to an existing cluster (or instance), just use the WebAdmin interface of the existing cluster instance with the **Add gateway** function of the Gateways view (from SCG 1 2209). There you have to authenticate with the administrator password of the new instance. Once the new instance is visible, you may specify the public access data for VPN under **Configuration > Gateways**. For further information, see Adding a new gateway instance.

> **Important:**

> When you add a new instance in the **Gateways** view, its configuration is overwritten with the cluster configuration. Normally, this is an advantage because you do not need to configure newly installed instances when they are to be clustered.

> The configured public port must be opened in the firewall. All gateway addresses must be accessible from the outside.

The required network services are included.

> **Note:**

> When upgrading to a later SCG version, always upgrade all SCG instances of a cluster.

Adding a new gateway instance

(from SCG 1 2209)

Newly installed instances can be easily added in the WebAdmin interface of an existing (cluster) instance and thereby clustered with it.

> **Note:**
>
> The new instance must be installed ready and located in the same domain.

1. In the WebAdmin interface of the cluster instance, click **Add gateway**.
2. Specify the host name (without domain) or IP address of the new instance.
3. Leave the **Use initial logon data** option active, if you have not yet changed the password of the administrator account on the new instance.

   Otherwise, to authenticate, specify the credentials for the new instance.

   > **Note:**
   >
   > Any previous configuration data of the new instance are overwritten. As soon as the new instance is added to the cluster, the configuration of the cluster is active, including administrator accounts.

4. Confirm with **Yes**.
5. Switch to **Configuration > Gateways** and specify the public access data via VPN to the new instance. For further information, see Gateways.

The new instance inherits all the rest of the configuration of the cluster instance.

Defining a new cluster instance (earlier versions)

(for SCG 1 2201 and earlier versions)

For versions earlier than SCG 1 2209, new instances must be explicitly defined as a cluster instance in the configuration of the WebAdmin interface: When you add a new instance, no automatic clustering takes place - new random values are generated for each newly installed instance and the administrator must apply cluster key and database password. For information on the procedure from SCG 1 2209, see Adding a new gateway instance.

Note:

Before you change a cluster key, make sure that at least one other instance uses the new cluster key.

1. Install another gateway instance and start the machine.For further information, see Installation.
2. Allow the system two to three minutes before you log on to the SCG WebAdmin interface.
3. Switch to the **Configuration > SCG cluster** view and edit the following options:

OPTION	DESCRIPTION
`Database password`	Copy and paste the database password from another instance.
`Cluster key`	Copy and paste the cluster key from this other instance.

   Clustering is only done if both instances have the same hash value. The hash value is built from the database password and the cluster key.
4. Leave the WebAdmin interface of your new instance.

*The new instance is added to the cluster. Once the instance is running with the new configuration, it is shown in the WebAdmin interface of the main instance, in the **Gateways** view.*

When a gateway instance is started, a broadcast message is sent, other instances with the same hash value can react. The instances communicate with each other via TCP port 27017 within their network.

Note:

Make sure that the instances can reach the Scout Server via appropriate network routes.

Do not change database password and cluster key after you have created a cluster. Changing any of the values may result in the instance becoming inoperable.

Client management and onboarding

New devices are configured to the gateway via the First Configuration Wizard.

The devices can be enabled to connect via intranet and via Internet,depending on where the user is located. For this, we recommend that you set up the network configuration in a way that the Scout Server name is resolved internally to the Scout Server and externally to the Scout Cloud Gateway. Users then only have to enter their logon data for the gateway the first time they use their devices outside the intranet. For further information, see General network settings.

Once successfully connected via the Scout Cloud Gateway, the devices remain connected to the gateway and retain their SCG configuration even in the following cases:

- A factory reset for the device is performed The option **Delete Scout Server address** must not be selected.
- Firmware update The option **Format system partition before update** may be selected.

> **Note:**
>
> If a device is connected via SCG, this information may be displayed in the Scout Console in the **Properties** window, as well as the SCG address (from Scout 15 2209 and eLux® RP 6 2209). These fields can also be used for creating records.

On-boarding new devices / standard

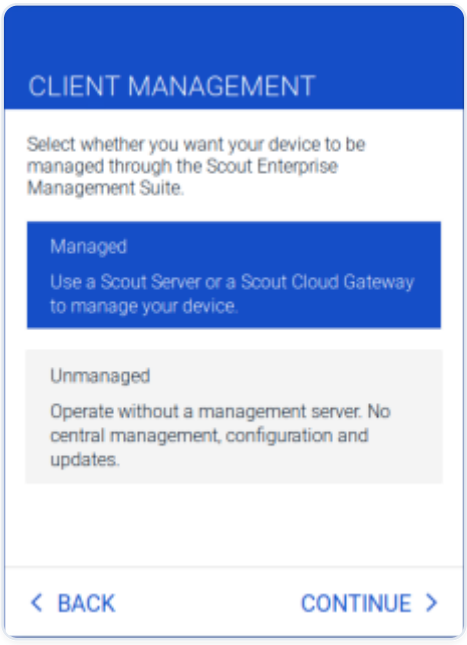
New devices that you want to connect via the Scout Cloud Gateway to your Scout Server need to be configured to the gateway (except when using uPilot) once via the First Configuration Wizard.

REQUIRES
The Scout Cloud Gateway is configured and users are authorized via one of the supported authentication methods.

1. Switch on your device. If the device is not in initial state, reset the device configuration. To do so, on the system bar, open the **Command panel** and click **Factory reset**.

   The First Configuration Wizard starts.
2. Select the display language and keyboard language.
3. Read and accept the license terms.

4. To configure the device's management via Scout Cloud Gateway, select **Managed**.



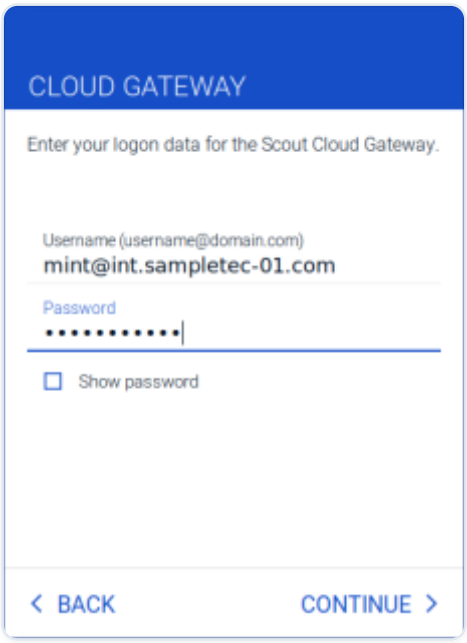
5. Enter the address of the Scout Cloud Gateway as FQDN or IP address.

> **Note:**
>
> If multiple gateway instances are operated in a cluster, it does not matter which of the gateway addresses is entered.

6. If Keycloak is used for authentication, you will be redirected to the Keycloak logon page. Specify and confirm your logon data.

   After authentication, you will return to the First Configuration Wizard.

7. If Active Directory is used for authentication, enter the AD user logon data. The username must be specified as a user principal name (UPN). Example: `mint@int.sampletec-01.com`



The relevant connection data are transferred via the Scout Cloud Gateway to the AD server. The AD user is checked, the Scout Server and OU are identified.

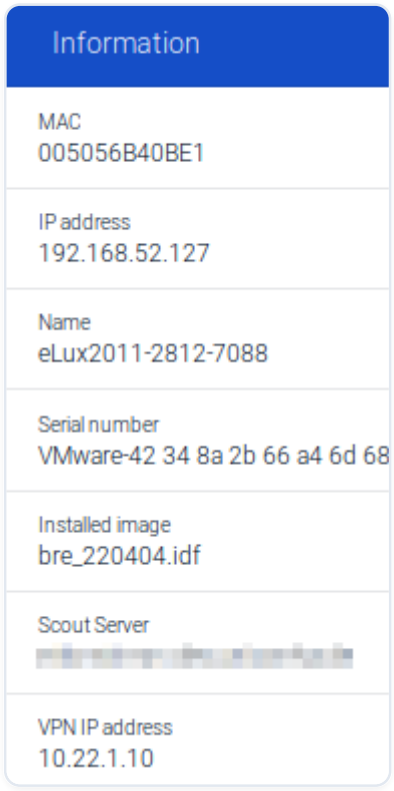
8. If the Scout Cloud Gateway authentication is used, enter the 16-digit token you have received from your administrator.

   The relevant connection data are transferred to the Scout Cloud Gateway and verified.

9. Click **Finish**.

*If the OU ID of the AD group or of the account definition for SCG authentication is invalid, the device enters the default OU defined in the Scout Console, under **Advanced options**. This is also true if authentication is done via Keycloak or token.*

The gateway address, username, Scout Server, and assigned OU are displayed in a summary.



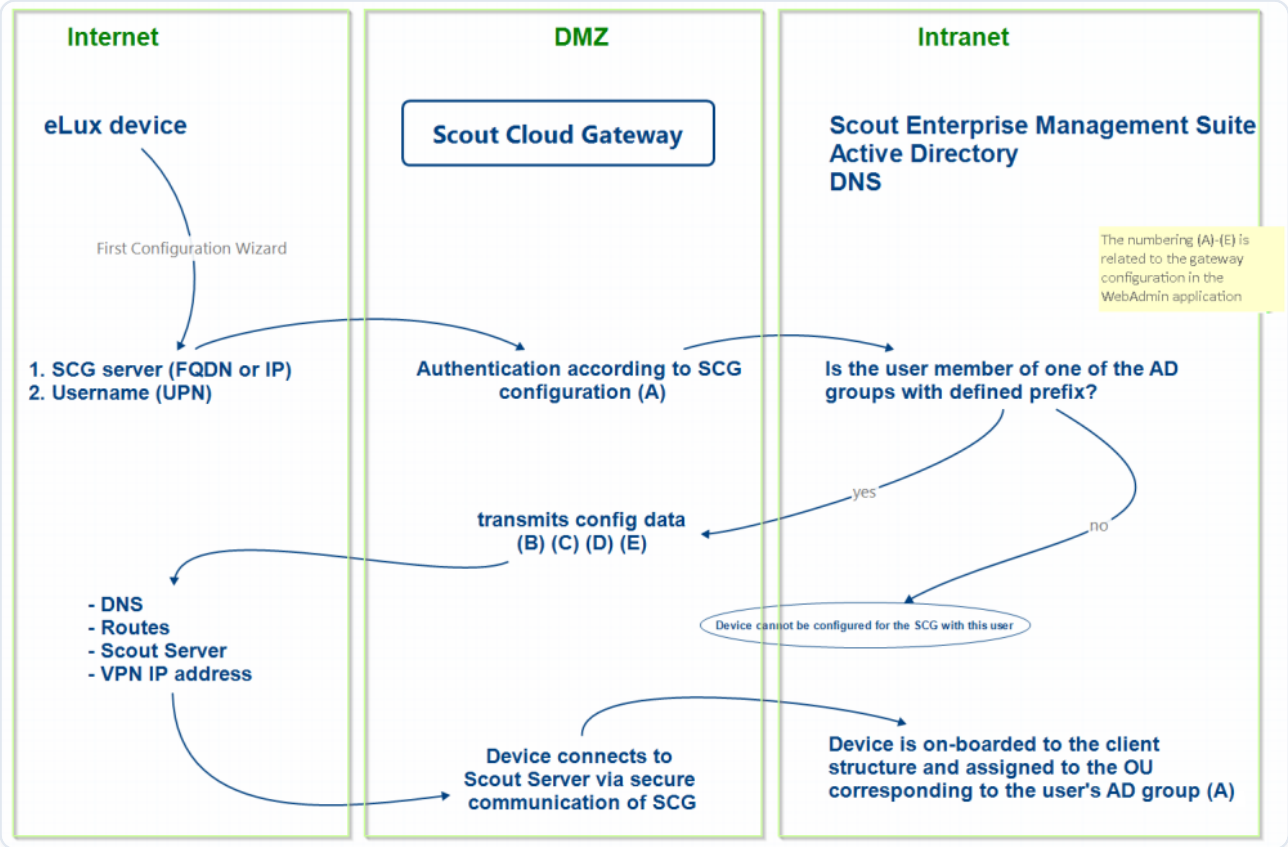
The configuration data are transferred from the Scout Cloud Gateway to the device and the device is restarted.

The device is connected to its Scout Server via the gateway. In addition to its local IP address, the device receives a VPN IP address.

The figure on the right shows the Configuration panel of the device after the First Configuration Wizard has been completed.

The device is shown in the WebAdmin interface in the Devices view with status connected (after short delay).

The flow chart below shows the communication of the onboarding process of a device via the Scout Cloud Gateway and via AD.



upilot Automated onboarding

(from eLux® RP 6 2302)

uPilot is a web service that simplifies onboarding of new eLux devices, for example for home devices, into a Scout infrastructure. The configuration of a new device is automated in a way that end users are able to on-board their device in a minimum of time, without having to know the address of their Scout Cloud Gateway. Especially when large numbers of devices need to be rolled out configured in short time, uPilot offers great support.

With uPilot, the assignment of the devices to their Scout Cloud Gateway is already defined before the actual roll-out:

uPilot is fed with a list of all devices that are to be connected. Customers specify the devices with a unique ID and a mapping to the URL of a Scout Cloud Gateway server.

As soon as a device with appropriate prerequisites starts the First Configuration Wizard, the following will happen:

The device sends its ID to the uPilot web service (1) which returns the URL of the associated Scout Cloud Gateway server (2). From the user's point of view, the only actions to be performed in the

First Configuration Wizard are to select the display and keyboard language and to enter the logon data (3).

Requirements

- Scout Cloud Gateway is configured and the user is authorized via one of the supported authentication methods.
- Software on the devices:
  - eLux RP 6 2302 or later In the eLux package **eLux Desktop Extensions**, the included feature package uPilot must be installed on the devices.

Customers provide the Unicon GmbH with a list of devices to be supported by uPilot and the associated Scout Cloud Gateway URLs. To do so, please contact the Unicon Support: myelux Support Case Form.

On-boarding a new device via uPilot

The following step-by-step instructions are from an end-user perspective.

1. Switch on your device. If the device is not in initial state, reset the device configuration. To do so, on the system bar, open the **Command panel** and click **Factory reset**.

   The First Configuration Wizard starts.
2. Select the display language and keyboard language.
3. Enter your user logon data. If Active Directory is used for authentication, the username must be specified as a user principal name (UPN). Example: `mint@int.sampletec-01.com`

   The relevant connection data are transferred via the Scout Cloud Gateway to the AD server. The AD user is checked, the Scout Server and OU are identified.
4. Click **Finish**.

The device is restarted and then connected to its Scout Server via the Scout Cloud Gateway. The assignment to the appropriate Scout Cloud Gateway was made in the background by uPilot.

SCG WebAdmin interface

In addition to configuration options, the WebAdmin interface of the Scout Cloud Gateway allows you to manage the connected devices and gateway instances.

The menu bar offers the following functions:

1. Switch between the views **Dashboard**, (from SCG 1 2209) **Gateways**, **Devices** and **Configuration**.
2. Info dialog with version information, documentation link and download link for diagnostic files.
3. **User profile** dialog with information on and functions for the logged-on user
    - Shows username, role and start of logon
    - Allows to switch the language (German/English)
    - Allows to **Change password**

For further information on the **Configuration** view, see Gateway configuration in the SCG WebAdmin interface.

Logon and language

For the initial logon to the WebAdmin interface the user account admin with an initial password is available, see Connecting and logging on to the SCG instance.

The admin account with global access rights can still be used with your own password. In addition, further users can be defined with the role **admin** or **operator** (from SCG 1 2103). For further information, see SCG administrators.

Failed logons

(from SCG 1 2103)

An account is temporarily locked for security reasons if too many incorrect password entries are made. By default, the account is locked for 10 minutes after three incorrect entries.

The number of incorrect entries and the lock time can be configured individually. To do so, edit the file `/opt/unicon/scg/config/config.json` and the following parameters:

```
[backend.lockUser.numFailingLogons]

[backend.lockUser.lockOutTimeInMinutes]
```

The new values are applied directly; a restart is not required.

Changing password

1. On the menu bar, click the **User profile** icon .
2. In the **User profile** dialog, click **Change password**.

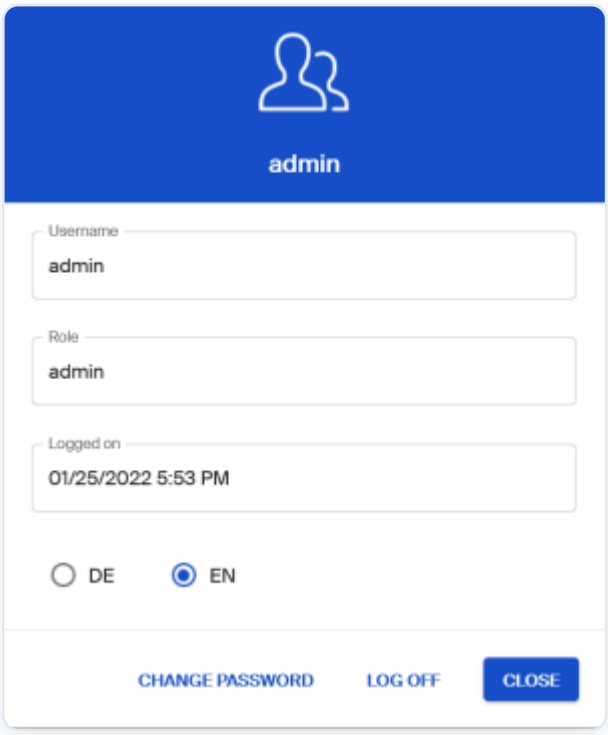
Changing language

(from SCG 1 2103)

1. On the menu bar, click the **User profile** icon .
2. In the **User profile** dialog, choose between English and German.

The display language is immediately changed.

> **Note:**
>
> The language can be set individually for each user and is stored in the user profile.



General list functions (Devices and Gateways)

In the **Devices** and **Gateways** views, you can display the columns in any order and sort devices or gateways as you wish. You can also filter the display of devices, see Devices view.

All settings are saved in the browser used beyond the session (from SCG 1 2209). For this, local data storage must be allowed in the browser.

	DESCRIPTION
1	To sort by a field, click the column header. To change the sort order, click again.
2	To set the sorting order (or a filter) via menu click
3	To change a column width, drag the column header border to the desired position.
4	To change the order of columns, drag a column header to the new position.

Dashboard

(from SCG 1 2209)

In the SCG Dashboard, various widgets show the most important key figures and key performance indicators so that you can get an overview of the entire infrastructure at any time.

Devices

1	Total number of all connected devices
2	Currently connected and disconnected devices
3	Number of connected and disconnected devices
4	Devices that have not been connected for at least one month / at least one year
5	Devices whose device certificate has expired / will soon expire

Load Balancing

1	Total number of all connected devices with current distribution to instances
2	Number of gateway instances that are currently online, offline, or not configured
	Unconfigured gateway instances can be freshly installed instances or disabled instances.
3	Number of VPN servers on the gateway instances that are currently online, offline, or not configured
	If a VPN server is not configured, **Configuration > Gateways > Gateway-specific configuration** is missing.
4	Total number of possible device connections
	with display of instances that have only a few free IP addresses left

Performance

Average CPU and RAM usage, as well as memory across instances.

The instance with the highest values is displayed.

Configuration

1	Name of the current [Scout Server]. CA certificate or self-signed certificate that ensures secure communication between SCG and devices see **Configuration > Server device settings**
2	Selected Identity provider. For SCG authentication additionally number of token see **Configuration > User authentication**

3	Health check service active or inactive.Web server configuration see **Configuration > Monitoring** and **Configuration > Gateways**
4	Latest SCG version in the cluster
	Number of instances that are on earlier version
	Number of SCG administrators with roles see **Configuration > SCG administrators**

Administrator activities

Various activities are logged, including configuration value changes and adding/changing/deleting gateway instances.

The old values are crossed out, the new ones are underlined.

Gateways view

The configured gateway instances are shown with their properties. A newly installed SCG instance is configured under **Configuration > Gateways**. For further information, see Gateways.

NUMBER	DESCRIPTION
1	The role of a gateway instance, but also a state before a role is applied,(from SCG 1 2209)are indicated by means of letters
	`P` Primary instance
	`S` Secondary instance
	`I` The instance is being initialized
	`E` The instance has an error or is not reachable

NUMBER	DESCRIPTION
	0 The instance synchronizes data (for example, when two instances are added at the same time)
	The status of a gateway instance is shown as a colored icon
	▢ Up and running
	▢ Switched on, but deactivated: Only the existing connections remain active
	▢ Switched off / not ready for operation
2	Number of connected devices (from SCG 1 2209)
	will be updated once per minute
3	Public host name
	The VPN hostname and port are specified under Configuration > Gateways. (from SCG 1 2101)
4	IP address range for the VPN connections
	The VPN IP address and subnet mask are specified under Configuration > Gateways. (from SCG 1 2101)
5	Deactivate or delete a gateway instance (from SCG 1 2209 in one column)
6	Add a new gateway instance to the cluster (from SCG 1 2209)

Deactivating a gateway instance

(from SCG 1.2103)

Within a cluster, you can deactivate a gateway instance for maintenance purposes while existing device connections remain active. Only after the next restart the devices connect to another instance. New devices are also connected via another instance.

> **Note:**
>
> An instance is deactivated by removing its gateway-specific VPN configuration.

1. For the relevant gateway instance, on the very right under **Deactivate**, click **Deactivate gateway instance**.
2. Confirm with **Yes**.

The deactivated gateway instance is displayed without its VPN configuration.

- To re-activate the gateway instance after maintenance or after an update, under **Configuration > Gateways**, re-enter the public access information for VPN.

Deleting a gateway instance

With **Delete Gateway**, a deactivated instance is removed from the internal database. A deleted instance no longer is connected to the cluster.(For versions < SCG 1 2209, the cluster information is retained and when the physical instance is restarted at a later time, a new clustering is triggered in the network)

REQUIRES
The cluster has more than one instance.

1. Make sure that the gateway instance you want to delete is deactivated.(For versions < SCG 1 2209, the instance must be powered off)
2. For the relevant gateway instance, under **Delete gateway**, click **Delete**.
3. Confirm with **OK**.

> **Note:**
>
> After confirmation, the system even removes the instance you are currently logged on to (from SCG 1 2209).

> You can continue to use a deleted instance separate from the cluster. Note that re-clustering a standalone instance (via the **Add Gateway** button) would also require re-onboarding the devices connected to the instance.

Adding a gateway instance

From SCG 1 2209, you easily add a newly installed instance to an existing instance or cluster so that the new instance inherits the entire configuration of the cluster. For further information, see Adding a new gateway instance.

Devices view

All integrated devices are shown with their properties. The **Devices** view gives you an overview of all devices in your infrastructure. You can filter the devices on one or more properties. For all or selected devices, you can perform actions and, for example, renew certificates.

For further information, see Device certificates.

NUMBER	DESCRIPTION
1	Number of all devices integrated in this SCG instance, number of connected devices, number of disconnected devices (from SCG 1 2107)
	If a filter is active, the numbers refer to the filtered values.
2	Select all devices
3	Select individual device
4	Status of a device: connected or disconnected
5	To sort by a property (ascendent, descendent, undo sorting), click the relevant column header
6	Point to a column header to display filtering and sorting functions.
7	Remove all filters

NUMBER	DESCRIPTION
	As soon as a filter is active, the icon is displayed in color.
8	Renew or revoke certificates for selected devices (from SCG 1 2201)
9	Remove selected devices (from SCG 1 2107)

Virtual IP address

(from SCG 1 2201)

The virtual IP address is the internal IP address assigned by the VPN for the duration of a connection. It is displayed once a device is connected and the VPN tunnel is established. (Displayed only for devices that are connected to an SCG instance from SCG 1 2201)

> **Note:**

> For clusters: When you upgrade to a later SCG version, always upgrade all SCG instances in a cluster. Otherwise, the virtual IP address is displayed only for devices that are connected to an SCG instance with version SCG 1 2201 or later.

> The virtual IP addresses also help you keep track of the boundaries of your VPN address ranges.

Filtering devices

Filter rules allow you to filter the devices on any fields (text and date). Multiple filter rules can be combined.(Some functionality is only available from SCG 1 2201.)

- A filter rule refers to one field and is displayed in one line.
- Additional filter rules referring to other fields may be added and must be linked by an operator (`And` / `Or`).
- The filter rules are dynamically applied to the data.

	DESCRIPTION
1	Field by which is filtered

	DESCRIPTION
2	Operator for date or text fields
3	Filter value
4	For date fields, either select or enter the date.
5	Operator for linking another filter rule (`And` / `Or`)
6	Delete individual filter rule
7	Add more filter rules

1. On a column header, click the ⋮ and then select **Filter**.

   A filter definition is displayed in the left section of the table, which is still incomplete. The selected column is already displayed.

2. Check that the desired field is selected under **Column** (1). If necessary, select the field you want to filter by from the drop-down list.
3. Select the **Operator** (2) using the drop-down list. Different operators are available for date fields (example: is before) than for text fields (example: contains).
4. Specify the **Value** (3) you want to filter on.
5. To filter by another field, click **Add Filter** (6) and specify whether the two filter rules should be linked by `And` or `Or` (5).

*The data is immediately and dynamically filtered by the defined filter rules. The **Remove filter** button becomes active.*

> **Note:**
>
> Defined filters remain persistent for longer than the current session. You can tell whether a filter is active by the blue color of the button .

- To delete all filter rules, click the **Remove filter** button on the right above the table.

Removing devices from the SCG integration

(from SCG 1 2107)

The devices may be connected or disconnected.

1. In the selection column on the far left, select one or more devices.
2. In the upper right corner, click **Remove devices**.

The devices are no longer connected via the SCG, only the currently existing connections remain unaffected. Certificates that are still valid are automatically revoked. To reconnect, the devices must go through an onboarding process.

Device certificates

For each device, after the onboarding process, the system creates a client authentication certificate so that users do not need to perform an additional logon for the SCG once they have registered their device. As long as the certificate is valid, users devices automatically connect to their Scout infrastructure via the SCG. Users only log on to Scout with their usual account, for example an AD account.

These certificates are called **Device certificates**. You can preconfigure their validity and expiration behavior for your SCG instance (from SCG 1 2201). For further information, see User authentication / Device certificates.

Validity period

The validity period of the device certificates is set to three years by default. It must be 30 days minimum and 5 years (1825 days) maximum. Within this range, you can freely define the validity period. The expiration date (**Validity**) of each device certificate is derived from the defined validity period. The validity date is set in the following situations:

- When a device is on-boarded
- On automatic renewal of a device certificate (if configured)
- On manual renewal of a device

You can view the validity date of a device certificate at any time in the certificate details under **Properties**. To do so, click the certificate icon. If a certificate is registered for renewal, you will see the new validity date under **Renewal**.

> **Note:**

> Defining a limited validity period without automatic renewal allows you, for example, to make the devices reboot after a certain time.

Automatic renewal

The device certificates can be renewed automatically before they expire (from SCG 1 2201). This function is not active by default. You can configure it in two variants: With the option **Automatically renew before expiration**, the system attempts to renew the device certificate seven days before expiration. If the user does not connect with their device within these seven days, the certificate expires. Only if the option **Allow renewal of expired certificates** is also active, the expired certificate will be renewed automatically on the next device start and contact to the SCG.

> **Note:**

> Automatic renewal (including expired certificates) allows you to avoid user interaction and the need for on-boarding a second time.

> With automatic renewal configured, the following situations result for a device within the four stages shown:

TIME	---------->	---------->	---------->	---------->
Stage	Onboarding	30 days before exp.	7 days before exp.	Expiration
Icon	▢	▢	▢	▢
Authentication for config 1	✓	✓	✓ Certificate is renewed	X
Authentication for config 2	✓	✓	✓ Certificate is renewed	✓ Certificate is renewed

Config 1: Option **Automatically renew before expiration** is active. Config 2: Option **Allow renewal of expired certificates** is additionally active.

> **Note:**
>
> After you revoke certain certificates, they will be excluded from automatic renewal.

Manual renewal

By default, device certificates expire automatically. Of course, you can also avoid re-onboarding known devices by renewing the device certificates manually in the **Devices** view. Again, the certificate icons are displayed in color and allow you to see at a glance whether action is required. Note the following:

- When you set a manual renewal, the validity of the relevant certificates is extended by the validity period defined in **Configuration > User authentification**. The new date is displayed in the certificate details under **Renewal**. The **Devices** view still shows the original date, but the color of the certificate icon changes to blue.
- After you set a renewal, the certificates of the relevant devices will be renewed on their next contact to the SCG. After that, the new validity date is displayed in the **Devices** view.
- A device that users log on with after the manually defined expiration date (for example because the mobile/home device has not been used for a long time) is denied. Either the device needs to be on-boarded again or the admin needs to renew the certificate manually.

Manage device certificates

In the **Devices** view, administrators may view, revoke or renew the certificates for their devices at any time. The certificates are displayed differently depending on their validity:

Certificate is valid more than 30 days

Certificate will expire in less than 30 days

Certificate has expired

Certificate is not available or has been revoked

The same colors are used for the button to revoke a certificate.

1	Expiration date for the validity of a device certificate
2	View details of a device certificate
3	Revoke certificate of a device
4	Renew or revoke certificates of multiple devices (from SCG 1 2201)
	The devices must be selected beforehand.

- To select either individual or all devices, use the selection column on the left side of the table.

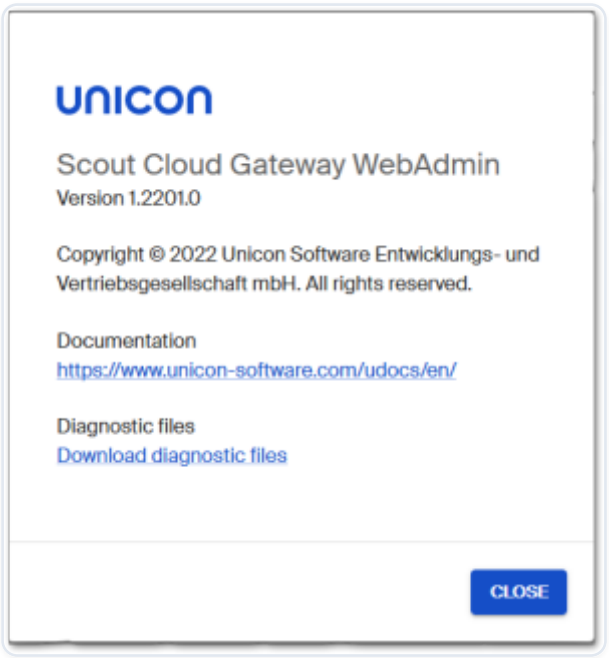
For further information, see Devices view.

Downloading diagnostic files

In the WebAdmin interface of the Scout Cloud Gateway, you can download all relevant diagnostic files and have a look into the communication between the Scout Cloud Gateway and the device. The diagnostic files provide support staff with helpful information and help identify missing routes or not properly working services.

The scope of the diagnostic files (**log level**) can be defined in the gateway configuration under **Logging**. For further information, see Logging.

1. On the right of the menu bar, click .
2. In the **Information** dialog, click the **Download diagnostic files** link.



A set of log files is compiled and downloaded in a .zip file.

3. Download the .zip file.

Access diagnostic files without WebAdmin interface

If the WebAdmin interface is not available, you need to access the instance directly to run the diagnostic script. Provided the public key for SSH access is installed, you can access the machine with `ssh <IP or host name>`. Then, in the relevant directory, run the script that creates the diagnostic files and retrieve the created .zip file:

```
cd /opt/unicon/scg/scripts

./createDiags.sh



scp root@<IP or hostname>:/opt/unicon/scg/scripts/scg-
diags_<timestamp>.zip
```

If the public key for SSH access is not installed, connect to your machine locally or through vCenter.

For the Debian installation, the script is located in the same directory. However, it creates a smaller .zip file, as system files are not retrieved.

eLux® 7

eLux 7 is a hardware-independent operating system designed for cloud-computing environments.

Overview and general information

About eLux 7

Keyboard shortcuts

Touchpad gestures for mobile devices

About eLux® 7

eLux® 7 is a hardware-independent operating system designed for cloud-computing environments. It can be run by mobile and stationary thin clients, common PCs, Laptops - any endpoints with x86 architecture. eLux is based on Linux and provides a write-protected file system which makes it secure against computer viruses and other malware.

eLux 7 comes with a new desktop that is specially tailored to hybrid work scenarios and a new partition layout that offers more space for installing apps on the end device.

You can find information on migrating existing end devices with eLux RP in the Shortguide migration of eLux RP 6 to eLux 7.

This guide supports the system administrator in setup, maintenance and operation of devices and PCs running eLux 7, hereafter referred to as **eLux** devices. The documentation refers to the current version eLux 7.

This guide assumes knowledge of:

- Installation, maintenance and operation of computer networks and peripherals
- Operating system skills of the server machines in use

> **Note:**

> eLux 7 devices can be centrally managed through the Scout Enterprise Management Suite version 15.x.

> The management protocol for communication between Scout Server and eLux devices offers end-to-end encryption via TLS 1.2. TLS-encrypted communication is done via port **22125**.

For support periods, see the Whitepaper **Releases, Lifecycles and Compatibility**.

Keyboard shortcuts

SHORTCUT	FUNCTION
CTRL+ALT+↓	Switch between open applications to the left.
CTRL+ALT+↑	Switch between open applications to the right.
CTRL+WIN	Switch from other applications to eLux: The eLux taskbar/system bar with open applications is shown.
WIN+ALT+I	Open the device information.
CTRL+ALT+Home	Unlock the **Configuration panel**: Requests the local device password.
CTRL+ALT+END	Lock the device's screen. If user authentication is active, the user password is required for unlocking.
CTRL+ALT+Function key	Switch between the consoles, if the **Console switch** option is enabled.
	The following consoles are available
	F1: eLux® desktop
	F4: Message console

Touchpad gestures for mobile devices

Mobile devices often need to be operated via a touchpad and do not always have hardware mouse buttons.

By default, mouse actions are mapped to the following touchpad gestures:

MOUSE ACTION	TOUCHPAD GESTURE
Click with left mouse button	Tap one finger on the touchpad (single-tap)
Right-click (right mouse button)	Tap two fingers on the touchpad (single-tap)

MOUSE ACTION	TOUCHPAD GESTURE
Middle-click (wheel button)	Tap three fingers on the touchpad (single-tap)
Move cursor	Drag your finger on the touchpad
Scroll	Slide two fingers at the same time

Installation

eLux® can be installed directly on a device's flash memory or on a hard disk. The installation procedure is a kind of recovery installation and can be performed in two ways:

- from USB stick: For all supported operating system versions, we provide an **eLux USB Stick** image, available for download on our myelux.com portal and suited to create a stick for installation.
- via PXE recovery: For large environments, PXE-capable devices can be installed through the network if the eLux software container and Scout Enterprise Management Suite® are already installed.

Both procedures are described in detail in our short guide **eLux Recovery procedures**.

Installation in a VM

eLux can be installed in a VM (Virtual Machine). Depending on the virtualization solution, specific settings need to be activated.

> **Note:**

> It is recommended to use at least 16 GB of storage. The eLux ISO image needs to be mapped as a drive for the installation.

XenServer®

SETTING	VALUE
Guest OS	Ubuntu Jammy Jellyfish 22.04 (Ubuntu)
Boot Mode	UEFI Boot

> **Note:**

> Please also see eLux 7 2503 VM on XenServer as a reference

VMware

Types: vSphere, Fusion, Workstation

SETTING	VALUE
Guest OS	Other (64-bit)
Boot Option - Firmware	(U)EFI

Virtual Box

SETTING	VALUE
Type	Linux
Subtype	Ubuntu
Version	Ubuntu (64-bit)
Extended feature	Enable I/O APIC
Extended feature	Enable EFI

Further information

System requirements

First boot procedure

First configuration

Device password

Self-administration on the device

System requirements

Hardware requirements eLux® 7

	MINIMUM REQUIREMENTS
Processor	x86, 64-bit capable and powerful (2 GHz quad-core or more recommended, less could affect system performance depending on the use case)
RAM	4 GB (at least 3.5 GB available for the operating system)
HDD	16 GB (8 GB are also supported, but only with limited functionalities)
GPU (Graphics processing unit)	AMD or Intel chipset
Network	Ethernet or WLAN
I/O ports	USB 3.0 or USB 2.0, USB boot support
Resolution	1920 x 1080 (Full HD)
Boot mode	UEFI

Certified Hardware

The supported hardware models (Hardware Compatibility List) for each operating system version are published on our technical myelux.com portal within the current eLux container under **Certified Hardware**. (even for previous versions)

For support periods and the compatibility matrix, see the Whitepaper **Releases, Lifecycles and Compatibility**.

Minimum Scout, ELIAS, and Scout Cloud Gateway Versions for eLux 7

Please use Scout versions:

- 15 2302 LTSR CU1 and later
- 15 2309 and later

and ELIAS versions

- 18 2403 and later

and Scout Cloud Gateway versions:

- 1 2209 and later

for the management of eLux 7 devices via Scout Console, Scout Board, ELIAS and Scout Cloud Gateway.

First boot procedure

The first boot procedure for an eLux® device in initial state, after a factory reset or after a Recovery installation, is processed as follows:

a. Scan UEFI

b. Make a DHCP server request

> **Note:**
>
> To enable the device to connect to the Scout Server, either DHCP or DNS must be configured. For further information, see Self-registration of devices in the **Scout** guide.

c. Start the eLux operating system

If either DHCP or DNS has been configured for the Scout Server, the device is automatically registered in the Scout infrastructure and receives a new configuration.

If a device cannot retrieve the IP address of the Scout Server, the First configuration opens and leads you through the first configuration.

First configuration

During the first boot procedure, a Wizard is launched which helps you through the first configuration. The First Configuration Wizard is also started when you reset the device to factory state.

The First Configuration Wizard offers the following options:

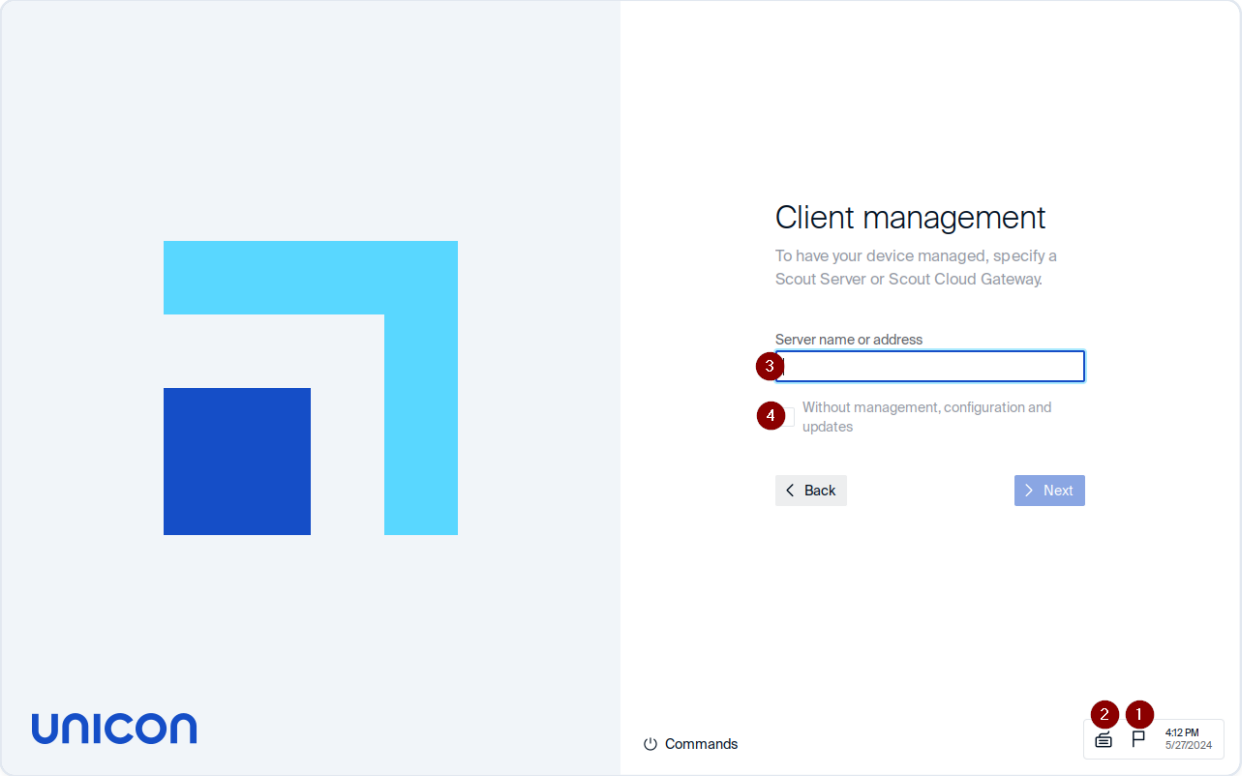
- Manage the device through the Scout Console The configuration data are transferred from the Scout Server.

- Connect the device through the Scout Cloud Gateway and manage it via the Scout Console
  The configuration data are transferred from the Scout Server.
- Configure eLux® manually, which means locally on the device

Going through the first configuration and connecting to a Scout Server

1. Select the display language **(1)** and keyboard language **(2)**.

   The following languages are supported: English, German, French and Spanish



2. Enter the address of the Scout Server as FQDN or IP address *(3)*.

   If your device is to be connected via Scout Cloud Gateway, enter the address of the Scout Cloud Gateway.

3. Select the destination OU for the device in the Scout Console.
4. Confirm with **Finish**.

The device is registered in the Scout infrastructure, added to its destination OU, and is restarted. The device contacts the Scout Server and downloads the configuration and application data of the destination OU.

If a profile for this device has already been created in the Scout Console, the device is assigned the configuration of the existing profile.

For further information on connecting via the Scout Cloud Gateway, see Integrating new devices in the Scout Cloud Gateway guide. For further information on managing devices with Scout, see the **Scout** guide.

Device password

All devices managed by a Scout Server receive the same device password. There is only one device password for all devices of the same infrastructure which is defined in the base device configuration.

The device password is used for unique assignment and authentication to the Scout Server, so that no other Scout Server can manage this device.

In the initial state, the device password is `elux`.

> **Note:**
>
> The device password has nothing to do with a user password, which is used for example for AD user authentication.
>
> For further information on passwords, see also Passwords in the **Scout** guide.

Self-administration on the device

With administrator rights, users can change the configuration locally on the device or completely disconnect from the Scout infrastructure. To prevent abuse, we recommend that you change the initial device password and do not release it. For further information, see Local device password in the **Scout** guide.

Logging on with full access on the device

1. Open the Configuration panel.
2. Press `Ctrl+Alt+Home`.
3. Enter the device password.

You are provided with full access rights to the device configuration and application definitions on the device.

eLux® 7 user interface

> **Note:**
>
> eLux 7 devices can be centrally managed through the Scout Enterprise Management Suite version 15.x. For further information, see Compatibility client platform and Scout Enterprise Management Suite in our **Releases** Whitepaper.

> The eLux 7 user interface provides the following elements:

	LEGEND TO NUMBERS
1	**Desktop** with applications.
2	**Systembar** with **Start menu**, running applications and system icons.
3	**Start menu** with application search, system commands and **Device configuration**.
4	**Device configuration**.

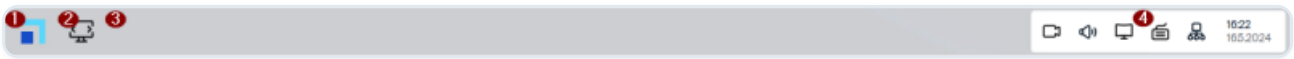
System bar

Device configuration

Applications in the eLux 7 interface

System bar

The **System bar** provides various functions and can be configured.



	DESCRIPTION
1	Show **start menu** with application selection and system commands (see below)
2	Show desktop
	Minimizes all open windows.

	DESCRIPTION
3	Taskbar for displaying applications with an open window.
4	System icons to easily change the settings for hybrid work scenarios (e.g. video, audio, monitor layout, network). The device configuration can be opened from the system icon dialogs via the gear icon. Tooltips display status information for the system icons as soon as the mouse pointer is positioned over the icon.
	To display information about active third-party software (e.g. Zoom, Cisco Secure Client), click on the up arrow -icon

Start menu

Notifications

Start menu

eLux® commands such as **Shut down** and **Restart** are located in the **Start menu**, in the **System commands**.

Which commands are displayed depends on whether the devices are managed by Scout, whether user authentication is configured, and the user rights the administrator has defined.

- To show the Command panel, on the left of the system bar, click the eLux icon .

AVAILABLE ELUX COMMANDS

1	All predefined applications are searchable here and listed via an icon.
	Enumerated applications from a Citrix Store are also searchable and can be displayed by left-clicking on the Citrix Store icon.

2	The device configuration provides the settings for the eLux Client. You can find more information on this under Device configuration.
3	Managed devices: Start firmware update. The system checks whether a firmware update is required. If the image definition file on the ELIAS 18 / web server is newer than the one on the device, the user can start the update process.
4	Managed devices: Synchronize device configuration. The current device configuration and the current application definitions for this device or OU are loaded from the Scout server and the device is restarted. Local configurations are overwritten if they are not protected.
5	Managed devices: Reset device to default state. The **device configuration** of the client is reset to the default state. Local application definitions and configurations are deleted. The firmware image with software packages is retained.
6	For AD authentication: Changing the AD password.
7	With AD authentication: The logged-in user is logged out and the login dialog is displayed.
8	With AD authentication: The screen is locked on the software side and can only be unlocked by entering the password.
9	The device is set to standby mode (Suspend to RAM).
10	The device is shut down and restarted immediately.

11	The device is shut down and switched off.

For further information, see eLux commands.

> **Note:**
>
> Before a command is executed, the user receives a confirmation message (except for the **Lock** command).

Notifications

System notifications inform users about a changed network status or the connection or removal of peripheral devices, for example. Notifications are shown in a small message window at the bottom right of the screen and are removed after a few seconds.

To make it easier for users to read several messages, especially if they arrive at short intervals, an additional icon allows you to show all notifications in a separate panel. This panel displays the notifications one under the other and remains active until you close it with a click.

1	To open the Notification Center, click on the temporarily displayed icon.
2	Notifications are displayed in the notification center according to their display duration.
	System notifications from hardware detection disappear automatically after a few seconds.
	The display duration for messages from the Scout administration can be configured in the Scout console.
	Notifications are listed one below the other if several notifications are waiting. To close a notification, click on the Close field.

	The panel with further notifications remains open.

Device configuration

The **Device configuration** can be opened via the **Start menu**.

Showing the Device Configuration

REQUIRES
The option **Show Device configuration** in **Start menu** is enabled.

- In the **start menu** , click the Device configuration button. Alternatively, click the time or one of the live information icons.

or

- On the keyboard, press WINDOWS+ALT+C.

  *The **Device configuration** with the Configuration dialogs is shown.*

  The recently used Configuration dialog is opened. If you click the time or a live information icon, the corresponding dialog is opened.

The dialogs are displayed alphabetically or sorted by content. Sorting can be defined in the Scout Console.

	DESCRIPTION
1	Switch to **Applications** tab
2	To open a dialog, click the dialog title. You can open only one dialog at a time.
3	Open a drop-down list or context menu
4	Enable or disable an option

Using the configuration and application dialogs

- To view all dialog titles or all dialog options, you might have to scroll down the frame.
- Many options are applied directly after you have set them. Some dialogs require you to click buttons such as **Apply** or **Cancel** before they are closed.
- When you enable an option, further entries might be required.
- Mandatory fields are characterized by an asterisk* on the right. If you have missed to fill in a mandatory field, it is displayed in red.

For further information about the content of the dialogs, see

Device configuration and

Defining applications

Applications in the eLux® 7 interface

Starting applications

- Click on an application icon on the desktop or in the application selection in the **Start menu**.

Disconnecting applications

- On the system bar, right-click the open application and then click **Close**.

> **Note:**

Searching for applications

- Enter the search term in the search field of the application search in the Start menu.

Only the relevant applications are displayed in the application selection.

- To deactivate the search filter, click on the **X** in the search field.

Switching between sessions/applications and the eLux Desktop interface

- Press the key combination CTRL+ALT+D

Showing the desktop and minimizing application windows

- On the system bar, click the icon.

For further information, see Defining applications .

Device configuration

> **Important:**

> If the device is managed by Scout, configuration is normally done centrally in the Scout Console. With inheritance enabled, local configuration changes on the device will be overwritten as soon as the device connects to Scout. For further information, see Device configuration in the **Scout** guide.

> The local device configuration is located in the **Configuration panel** that you can show on the right of the screen.

For further information on viewing and operating the Configuration panel, see Configuration panel.

Applications

Audio

Date and time

Desktop

Diagnostics

Display

Drives

Firmware

Hardware

Information

Keyboard

Mirror

Mouse

Network

Power management

Printer

Scout

Security

User authentication

Applications

Definition of **applications** for connecting to a backend system (e.g. VDI, DaaS, cloud environments) and local applications (e.g. Zoom Client for Linux).

The corresponding software packages must be installed on the end device (e.g. Citrix Workspace app, web browser, VMware Horizon Client, eLuxRPD Client).

Audio

Settings for the output and input of the connected audio devices.

Date and time

Setting the date and time, as well as the time zone and NTP (Network Time Protocol) usage.

Desktop

Desktop settings for the language, font and background color, keyboard shortcuts and the system bar.

Diagnostics

Setting the log level and the output of the diagnostic files.

Display

Configuration of the monitor layout and screen resolution.

Drives

Configuration of CIFS drives for accessing files and directories in the network.

Firmware

Configuration of the image file and the firmware update procedure.

Hardware

Settings for the use of USB mass storage devices and smart cards via USB.

Information

Display of device information, such as MAC address and serial number, as well as display of license and version information.

Keyboard

Setting the keyboard language and configuring keyboard functions.

Setting additional keyboard (XKB) options

Some keyboard layouts use a comma (,) as the decimal separator on the numeric keypad. To make it always produce a period (.) instead, or to apply other XKB options, add the option to the keyboard layout in the eLux® keyboard database.

> **Note:**
>
> Setting `XkbOptions=` under `[Screen]` in `terminal.ini` has no effect in eLux 7 and should not be used.

1. Edit the keyboard database file `/setup/elux/kbd.ini`.
2. In the section for the keyboard layout used on the device, set the `options=` value. The following example sets the numeric keypad decimal key to a period for the German layout (`[de]`):

   ```
   [de]

   options=kpdl:dot
   ```

   You can specify multiple options separated by commas, for example `options=kpdl:dot,compose:ralt`. Set the option on the section that matches the layout the device actually uses.
3. Deploy the modified `/setup/elux/kbd.ini` to the device through Scout (see Files configured for transfer), then restart the device so the file is in place at startup.

To confirm the option is active, run the following command on the device:

```
setxkbmap -query
```

The output lists the layout and its options, for example `options: kpdl:dot`.

> **Note:**
>
> If you later change the device's keyboard layout, add the option to the new layout's section in `kbd.ini` as well. An option set on one layout section does not apply to another.

Mirror

Configuration of remote access (mirroring) by a Scout administrator to the end device.

Mouse

Settings for using the mouse and touchpads.

Network

Configuration of the network interfaces and display of the network information for the active connection. The available network interfaces are displayed according to the hardware equipment and the installed software packages of the end device.

Power management

Settings for saving energy, especially for mobile devices in battery mode.

Printer

Configuration of locally connected and network-connected printers.

Scout

Display of the OU path (hierarchy structure of the organization units) and the info fields, as well as adjustment of the OU path (if this is permitted according to the Scout configuration).

Security

Setting user rights for access to system functions and changing the device password.

User authentication

Activation and configuration of user authentication (e.g. Active Directory).

eLux® commands

The following eLux commands are provided to the user depending on the configured user rights.

Open the Start menu and click the relevant command.

Updating the firmware

Synchronizing configuration

Resetting a device to factory state

eLux Command Scheduler

Updating the firmware

You can check anytime whether the current software status of a device does match with the available IDF on the server and, if required, initiate a firmware update on-demand.

1. Check whether the firmware settings of the device configuration are configured correctly. For further information, see Configuring firmware update.
2. On the system bar, show the Start menu.
3. Click the **Update** button.

   The device's firmware is compared to the specified image on the web server. A message will inform you, if the image on the web server contains updated packages and hence requires a firmware update.

   > **Note:**
   >
   > Before starting the update, click **Details** to view the components that require an update.

4. To perform the firmware update, click **Yes**.

The firmware update is executed. After the software is installed, the device will be restarted.

Synchronizing configuration

After having modified the device configuration or application definitions locally on the device, you can reset the configuration data to the server-side defined settings anytime.

1. Open the Start menu of the System bar, and then, click the **Device configuration** button.
2. Confirm with **Yes**.

The current device configuration and application definitions for the device or OU are loaded from the Scout Server and are available on the next device restart. Local configuration settings are overridden, unless they are protected.

Resetting a device to factory state

> **Important:**

> A factory reset causes the system to reset local configuration data.

> Resetting a device to factory state can be useful for troubleshooting, for example, if the locally defined device configuration does not work correctly.

1. Open the Start menu of the System bar, and then, click the Factory reset button.
2. Confirm with **Yes**.

The device configuration is set back to initial state. Local application definitions and locally stored configuration data are deleted. The connection data to the Scout Server are deleted.

The installed image with all software packages (firmware) is retained.

On the next restart, the device acts like in initial operation. If it is part of a Scout infrastructure, the device can be reconnected to a Scout Server via the following methods:

- Local First Configuration Wizard on the device
- DNS alias `ScoutSrv`
- DHCP options `222` and `223`
- Searching for the device by using the **Discovery** feature of the Scout Console

The device subsequently receives the configuration of the OU to which it is assigned.

> **Note:**

> Devices managed via Scout may be reset remotely by the administrator. In this case, the connection to Scout with server address and OU ID can be maintained.

eLux® Command Scheduler

The eLux Command Scheduler can schedule and execute recurring time-based commands. In contrast to the Scout commands initiated on the server side, the commands are executed according to the local time zone of the devices.

Commands to be scheduled must first be defined through an `.ini` file by the administrator. For further information, see eLux Command Scheduler in the **Scout** guide.

Appendix

eLux 7 partitions

IP ports

SNMP

eLux® 7 partitions

The partition layout of eLux 7 offers more space for applications on devices that have more than the minimum hardware requirements of 16 GB flash memory. The capacity of the system partition for applications is determined by the size of the flash memory. Many more applications can be used on a device with a flash size of 128 GB or 256 GB than on a device with a flash size of 16 GB.

The flash memory of a client is divided into four or five partitions during the eLux installation. Each partition is reserved for a dedicated purpose and is only called up when executing tasks related to the relevant partition. All partitions are recreated during a recovery installation.

> **Note:**
>
> You can find the minimum requirements in the article System requirements.

Partitions overview

PARTITION	PURPOSE	RECREATED WITH	OTHER
System	Reserved for the firmware (software packages)	Scout Update command with	–
Boot	Boot section	–	–

PARTITION	PURPOSE	RECREATED WITH	OTHER
Setup	Device configuration Local application definitions	Factory reset command	Does not affect the system partition with installed firmware
Update	Software delivery in advance (before firmware update) via Scout command or notification. Signature check for eLux software packages. Devices with update partition can be used as Dynamic Proxy (Provider) for firmware updates.	Scout Delivery command with option Clean up update partition before delivery	–
Custom	Optionally usable by applications with separate storage space requirements.	–	Unencrypted partition without a specific mount point. No custom partition is created on devices with less than 16 GB of flash memory.

Network requirements

All contents relating to ports, protocols and network requirements are covered by the **network requirements** guide. For further information, see Network requirements guide.

SNMP

SNMP (Simple Network Management Protocol) is a network protocol for monitoring and controlling network devices.

For eLux® RP 6 and newer, version SNMPv3 is used.

> **Note:**

> The command line program **snmpget** is not included in the software package. To query SNMP status information, please use third party software.

> Configuring SNMP

SNMPD and SNMP Configuration Directives

Configuring SNMP

1. From our myelux.com portal, under **eLux Software Packages**, for your eLux version, under **Add-On**, download the package SNMP Environment and deploy it to the devices.
2. If there is no `/setup/snmp/snmpd.conf` on the devices, transfer the configuration file `snmpd.conf` to the devices to `/setup/snmp/snmpd.conf` by using the Scout feature Files configured for transfer

   Or:

   Modify the terminal.ini file by using the Adding individual file entries feature of Scout. Example:

File	/setup/terminal.ini
Section	SNMPD
Entry	rocommunity
Value	secret

3. Optionally, to define further SNMPD and SNMP Configuration Directives feature and modify the `terminal.ini file` under SNMPD. Examples:

   `syscontact=

   syslocation=testcenter doDebugging=1`

   For further information on SNMPD Configuration Directives, see http://www.net-snmp.org.

The section SNMPD of the `terminal.ini` file is evaluated by the client and the `file /setup/snmp/snmpd.local.conf` is created. An existing `/setup/snmp/snmpd.conf` will be overwritten.

If the configuration file does not exist, the file `/setup/snmp/snmpd.local.conf` is created with default values.

NOTES ON CONFIGURING SNMP V3

- When you define users (**createUser**), set a password with at least 8 characters.
- For the authentication method, define `authPriv` or `authNoPriv`.

> **Note:**
>
> For SNMP v2, you can use `noAuthNoPriv` as the authentication method.

SNMPD and SNMP Configuration Directives

The following table refers to the eLux® software package snmp.

For further information on using SNMP with eLux, see SNMP

For further information on SNMP commands, see http://www.net-snmp.org.

APPLICATION	COMMAND
authtrapenable	1 | 2 (1 = enable, 2 = disable)
trapsink	host [community] [port]
trap2sink	host [community] [port]
informsink	host [community] [port]
trapsess	[snmpcmdargs] host
trapcommunity	community-string
agentuser	agentuser
agentgroup	groupid
agentaddress	SNMP bind address
syslocation	location
syscontact	contact-name
sysservices	NUMBER

APPLICATION	COMMAND
interface	name type speed
com2sec	name source community
group	name v1|v2c|usm security
access	name context model level prefix read write notify
view	name type subtree [mask]
rwcommunity	community [default|hostname|network/bits] [oid]
rocommunity	community [default|hostname|network/bits] [oid]
rwuser	user [noauth|auth|priv] [oid]
rouser	user [noauth|auth|priv] [oid]
swap	min-avail
proc	process-name [max-num] [min-num]
procfix	process-name program [arguments...]
pass	miboid command
pass_persist	miboid program
disk	path [minspace | minpercent%]
load	max1 [max5] [max15]
exec	[miboid] name program arguments
sh	[miboid] name program-or-script arguments
execfix	exec-or-sh-name program [arguments...]
file	file [maxsize]

APPLICATION	COMMAND
dlmod	module-name module-path
proxy	[snmpcmd args] host oid [remoteoid]
createUser	username (MD5|SHA) passphrase [DES] [passphrase]
master	specify 'agentx' for AgentX support
engineID	string
engineIDType	num
engineIDNic	string

SNMP CONFIGURATION DIRECTIVES

APPLICATION	COMMAND
doDebugging	(1|0)
debugTokens	token[,token...]
logTimestamp	(1|yes|true|0|no|false)
mibdirs	[mib-dirs|+mib-dirs]
mibs	[mib-tokens|+mib-tokens]
mibfile	mibfile-to-read
showMibErrors	(1|yes|true|0|no|false)
strictCommentTerm	(1|yes|true|0|no|false)
mibAllowUnderline	(1|yes|true|0|no|false)
mibWarningLevel	integerValue
mibReplaceWithLatest	(1|yes|true|0|no|false)
printNumericEnums	(1|yes|true|0|no|false)

APPLICATION	COMMAND
printNumericOids	(1|yes|true|0|no|false)
escapeQuotes	(1|yes|true|0|no|false)
dontBreakdownOids	(1|yes|true|0|no|false)
quickPrinting	(1|yes|true|0|no|false)
numericTimeticks	(1|yes|true|0|no|false)
suffixPrinting	integerValue
extendedIndex	(1|yes|true|0|no|false)
printHexText	(1|yes|true|0|no|false)
dumpPacket	(1|yes|true|0|no|false)
reverseEncodeBER	(1|yes|true|0|no|false)
defaultPort	integerValue
defCommunity	string
noTokenWarnings	(1|yes|true|0|no|false)
noRangeCheck	(1|yes|true|0|no|false)
defSecurityName	string
defContext	string
defPassphrase	string
defAuthPassphrase	string
defPrivPassphrase	string
defVersion	(1|2c|3)
defAuthType	(MD5|SHA)

APPLICATION	COMMAND
defPrivType	(DES)
defSecurityLevel	(noAuthNoPriv|authNoPriv|authPriv)

ELIAS 18

ELIAS 18 is the platform-independent, web-based front end of the eLux® Image Administration Service (ELIAS).

Overview

Our new web-based ELIAS 18 front-end is platform-independent and easy to use. It helps administrators keep their firmware images up-to-date with the latest software packages, always with regard to compatibility between all packages and components.

Our eLux® Image Administration Service (ELIAS) allows you to easily manage the firmware of devices running eLux. Firmware images contain the operating system and applications and can be modified by using ELIAS. ELIAS helps you create individual image definition files (IDF). An IDF is a structured package list defining all software packages for the operating system and applications that will be installed on the devices' flash memory. Once created, an IDF can be used as target for a firmware recovery or a firmware update. After a successful recovery installation or update to an individual IDF, the firmware image installed on the device is consistent with the image defined in the IDF.

For reasons of simplicity, in the following, we shall simply call an **IDF image**.

ELIAS ensures that all software packages required for an operational image are included, checks software dependencies and avoids version mismatches. Required packages are automatically identified and requested. You can even choose whether ELIAS should always provide the latest version of a specific software package and, at the same time, you need not worry about compatibility with other packages.

Starting ELIAS 18

REQUIRES
ELIAS has been installed successfully. For further information, see Installation: ELIAS 18 in the **Installation** guide.

ELIAS is started with a URL pointing to your installed ELIAS web service.

Without IIS:

- In the web browser, type the following URL: `http://< host name>:< port number>` or `https://< host name>:< port number>`

  `< host name>` refers to the computer name or IP address of the computer ELIAS is installed on.

  `< port number>` is the port you have specified for the ELIAS web service

With IIS redirection:

- In the web browser, type the following URL: `http://< host name>/path` or `https://< host name>/path`

  `< host name>` refers to the computer name or IP address of the computer ELIAS is installed on / your web server.

  `< path>` is the specified ELIAS path name under your web site (elias in the example above)

> **Note:**

> The computer ELIAS is installed on provides an ELIAS desktop icon.

> For access from outside the network, see Installation: ELIAS 18 in the **Installation** guide.

Access management and logon

When you start ELIAS, you will be prompted to log on.

ELIAS 18 offers users three different ways to log on:

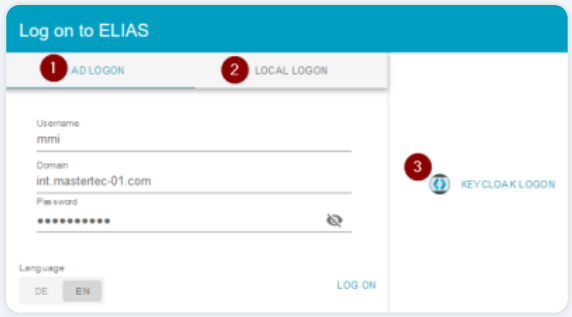
- Active Directory user accounts
- Local admin account
- Keycloak user role (identity management software / OpenID Connect)

By default, the logon types are available side by side and you can switch between them. The user accounts for Active Directory and Keycloak must be prepared for this purpose.

If needed, you can deactivate the local logon option.

Logging on to ELIAS 18

1. To choose the logon type, click the required tab or the link for Keycloak logon:



1 AD logon: The AD user account must be authorized for ELIAS.	
2 Local logon with username **admin**	
3 Logon via Keycloak: Redirection to the Keycloak logon page	

2. Type your username and password.
3. Confirm with **Log on** or press the Enter key.[1]

*The user you are logged on as is shown on the ELIAS menu bar next to the **User profile** icon.*

ELIAS role concept

Regardless of whether users are authenticated via Active Directory or via Keycloak, they obtain their access rights by being assigned to one of the ELIAS roles **Info**, **User** or **Admin**. The rights of the three roles build on each other.

Info: View the content of images, no changes possible

User: Create and edit templates and images within a container

Admin: Manage container, delete software packages, import / edit certificates within a container

On top of this is the **Global admin**, who manages access rights, can manage multiple containers and change ELIAS settings. The **Global admin** is defined by an ELIAS access right with **global access**. Initially - directly after the installation - the local admin account is provided for this purpose. With the local admin account, you initially define the ELIAS access rights. An ELIAS access right contains the following:

- Relevant users (AD or Keycloak)
- ELIAS role with rights
- Container to which the ELIAS role rights refer
- Option for global access

You can see that the user and access management depends on the ELIAS access right. After you have defined an ELIAS access right with global access (**Global admin**), you are free to disable the local admin account.

FEATURE	ADMIN	USER	INFO
Templates and images			
• create	✓	✓	X
• save (as)	✓	✓	X
• edit	✓	✓	X
• rename	✓	✓	X
• solve	✓	✓	X
• delete	✓	✓	X
• sign	✓	✓	X
• lock	✓	✓	X
• export	✓	✓	X

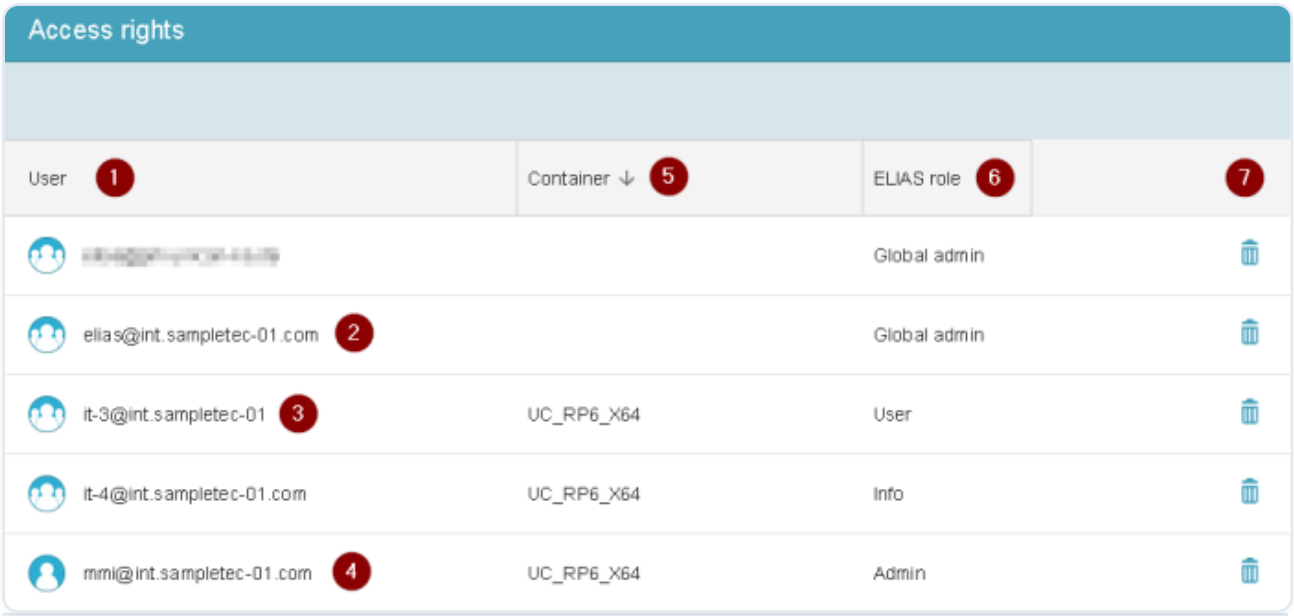
FEATURE	ADMIN	USER	INFO
Container			
• export	✓	✓	X
• create	✓	X	X
• import	✓	X	X
• rename	✓	X	X
• clean up	✓	X	X
• delete	✓	X	X
Container information			
• view	✓	✓	X
Software packages			
• delete	✓	X	X
Change log			
• view	✓	✓	X
• export	✓	✓	X
Notification history			

FEATURE	ADMIN	USER	INFO
• view	✓	✓	X
Certificates			
• view	✓	✓	X
Certificates for package validation			
• import	✓	X	X
Certificates for image signing			
• edit	✓	X	X

ELIAS access rights

ELIAS access rights assign users to a container and an ELIAS role, which in turn controls the individual rights within the container. To cover multiple combinations, define as many ELIAS access rights as you need.

The figure below shows the **Access rights** dialog after access rights for AD users and groups have been defined:

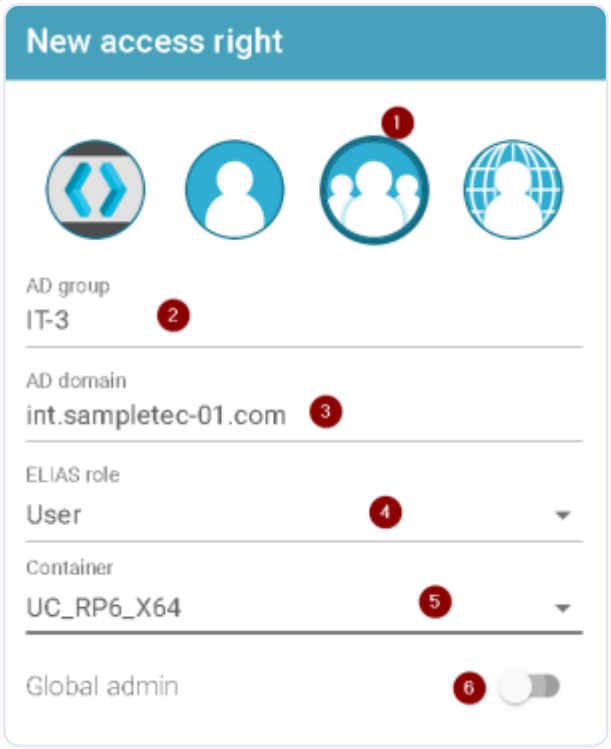


1	Users / User groups originating from Active Directory or Keycloak
2	**ELIAS** AD group as access right with global access (created by installation routine)
3	AD group as basis for ELIAS access right (defined by global administrator)
4	AD user as basis for ELIAS access right (defined by global administrator)
5	Container that users are allowed to access
	The global administrator is allowed to access all containers
6	Defines the allowed functions, see ELIAS role concept
7	Delete access right
	Note: The access right is deleted without confirmation.

Defining access rights

1. Prepare the relevant users in your Active Directory or Keycloak system. For further information, see Access management via AD andAccess management via Keycloak .

2. Log on as **Gobal admin**. To do so, you can use the user account of the local administrator (Local logon).[1[the local admin password has been defined during installation]]
3. On the ELIAS menu bar, click **Access rights**.
4. At the bottom right, click **Add**. For your new ELIAS access right, specify the properties. The figure shows the definition of an access right for an AD group.



1	Origin of the user accounts
	`Keycloak role`: All members of the Keycloak role to be specified under (2)
	`AD username`: AD username to be specified under (2)
	`AD group`: All members of the AD group to be specified under (2)
	`AD domain`: All members of the AD domain to be specified under (2)
2	Name of the Keycloak role, AD user, AD group or AD domain

3	AD domain – only for AD users and AD groups
4	ELIAS role that will be assigned to the users selected above
5	Container the selected users are allowed to access
6	Global access for the selected user(s)
	No assignment to a role or a container is made.

> **Important:**
>
> Users who receive **global access** via their ELIAS access right are authorized for rights management, container administration and ELIAS configuration.

5. Save the ELIAS access right.
6. Create further access rights, if required.

> **Note:**
>
> Once ELIAS access rights have been defined, they cannot be edited. Delete and recreate an access right if required.

Access management via AD

Provided, you have activated the **AD logon** during installation, AD users can log on to ELIAS 18 with their accounts if they are either members of the **ELIAS** AD group or are authorized via an access right defined in ELIAS.

AD group ELIAS

Members of the **ELIAS** AD group, which has been set up in your AD system for this purpose, can log on directly with their AD account to ELIAS 18.

Prerequisite:

- In your Active directory system, create an AD group named **ELIAS** and add the relevant users to this group.

After installation, you will find a group named `elias@< AD domain>` in ELIAS under **Access rights**. `< AD domain>` corresponds to the domain specified during installation. Example: `elias@int.sampletec-01.com`

> **Note:**
>
> All members of the **ELIAS** AD group have the **Global Admin** role and thus access to all functions and all containers.
>
> After the ELIAS installation, you can change the AD group name:

- Edit the file Program `FilesUniconScoutELIASserver.json` and modify the value of the adGroup parameter. Subsequently, restart the ELIAS service in `Services.msc`

ELIAS access rights

The users are granted an access right configured in ELIAS. This may be a user or a group of any domain. The access rights are defined by an administrator with global access. Included in the access right is the ELIAS role that defines which functions the users have access to.

Users who are authorized to log on via an ELIAS access right also log on with their AD account. However, authentication is independent of any predefined group or domain membership.

Prerequisite:

- In ELIAS 18, in the **Globaler Admin** role, create access rights based on your AD system.

For further information, see ELIAS access rights.

Access management via Keycloak

Keycloak is an open source solution for identity and access management. Keycloak also acts as an identity broker and supports user accounts from social networks. In order to use Keycloak for user logon, Keycloak must be configured on the server side and a Keycloak client must be installed.

Keycloak uses role-based access rights. In Keycloak, you define and assign roles and users. The Keycloak roles are then assigned to the ELIAS roles **Info**, **User** and **Admin**.

Configuring Keycloak

1. In the Keycloak administration console, create a realm for your environment.
2. Create your Keycloak client.
3. Within your client, create roles and users. Assign a role to each user.

> **Note:**
>
> Only Keycloak **roles** (and no Keycloak users) can be assigned in ELIAS. To make distinctions in access rights or the assigned container, define a separate Keycloak role for each combination.

4. From your Keycloak client, on the **Installation** tab, download the Keycloak configuration file as **Keycloak OIDC JSON**.
5. Add the Keycloak configuration file `keycloak.json` to your ELIAS installation directory (Example: `C:Program FilesUniconScoutELIAS`). Alternatively, save the file in a different directory that you specify during the ELIAS installation. The Keycloak configuration file will then be copied into the installation directory as `keycloak.json`.

Configuring ELIAS for Keycloak after installation

1. Copy the Keycloak configuration file `keycloak.json` manually into the ELIAS installation directory.
2. In the **ELIAS settings > Authentication**, select the option **Use Keycloak logon**.

Installing certificates for HTTPS connection

REQUIRES
To allow Keycloak logon via a secured Keycloak server, web server certificates are required.

1. In the ELIAS installation directory, create a new directory named .
2. Copy your web server certificates (intermediate and root) into the new directory. The certificate files can be of type or
3. Additionally copy the certificates into your browser's certificate store.

Logon as local admin

For local logon, use the **admin** user account. During the installation, you are requested to define an initial password for it. The admin password will be hashed and saved to the file Program `FilesUniconScoutELIASserver.json` under the **adminPassword** parameter.

The local **admin** account is equipped with global access and is primarily intended for initial configuration independent of access management via AD or Keycloak. Use the local **admin** account to define ELIAS access rights. Once you have defined an access right with global access, you may deactivate the local account.

Changing password for local admin account

1. On the ELIAS menu bar, click **Logon**.
2. In the logon dialog, click **Change password**.

Disabling local admin account

> **Note:**

> First make sure that there is another admin account with global access.

1. Edit the file Program `FilesUniconScoutELIASserver.json`
2. Set the **adminPassword** parameter to an empty string ("").
3. Restart the ELIAS service in `Services.msc`.

Switching between the accounts

1. On the ELIAS menu bar, click **User profile**.
2. Click **Log off**.
3. In the **Logon** dialog, click the required logon type. Then type your username and password.

First steps

Importing software packages

Quick run to create an image

Importing software packages

Before creating an image, you need to make the relevant eLux® software packages available in ELIAS, in a container.

The first container is created by the installation routine, but is still empty. One container is sufficient if you use only one major eLux version such as eLux RP 6. For further information on creating additional containers, see Creating a container.

You will need to fill your container with the required software packages. To do so, import software packages downloaded as `.zip` files from our portal into the container. To be on the safe side, we recommend that you download the `eLuxRP-X.X.X_AllPackages.zip` for the relevant eLux version. This archive contains all available software packages of an eLux version.

Of course, you can download and import individual packages at any time to obtain up-to-date software packages.

> **Important:**

> Before importing `.zip` files, do **not** unpack them. This is done by ELIAS.

1. To download software packages, sign in on our myelux.com portal. Then, under **Downloads > eLux Software Packages**, click the required eLux version and download the relevant `.zip` file or files.
2. In ELIAS 18, open the container you want to fill. Switch to the **Package Library**.

   *The name of the open container is shown on the ELIAS title bar. The **Package Library** of new containers is initially empty.*
3. In the **Package Library**, at the bottom, click **Import**.
4. Select the relevant `.zip` file from the file system and click **Open**.

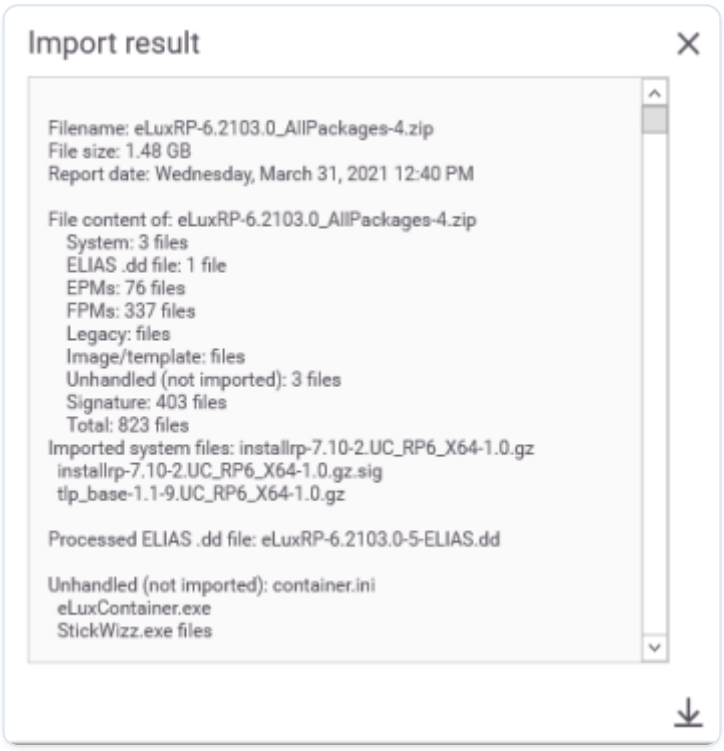
> **Note:**

> Importing the `AllPackages.zip` may take some minutes.

> During the import process ELIAS performs the following tasks:

- The file or files are uploaded and extracted.
- System files, software packages and, if available, images and templates are imported into the current container.
- The signatures of the software packages are verified.
- The solver is generated to resolve the dependencies.

The import result is displayed containing information on all imported files and any error messages. Use the download icon to download and save it as a text file:



After they have been imported, the packages are shown in the **Package Library**, grouped by categories. If you have imported images or templates, they are shown in the **Image Library**.

For further information, see Package Library.

Further information on package import

- In addition to software packages (`.zip`), you can import images (`.jidf`) and templates (`.idt`).
- Files to be imported may have a maximum size of 10 GB[1] (2 GB via API)
- You can import multiple files in one step. To do so, select them while pressing CTRL.
- Several import processes may be triggered and executed in parallel. (from ELIAS 18 2104)

Quick run to create an image

Before taking a closer look at detailed functions, let's first do a quick run to create an image.

REQUIRES
Container with all software packages of an eLux® version
For further information, see Importing software packages.

1. To create a new template, in the **Image Library**, at the bottom, click **Create**. Type a name for the template and press **Enter**.

   Your new image template is created but still empty.

2. To add software packages, in the **Package Library**, open a category. Then use drag-and-drop operations to add at least one software package to your template. Click **Save**.

   Your template is saved.

3. Click **Solve** to create an image out of the template.

   ELIAS replaces unversioned packages by the latest version. Also, ELIAS checks for additionally required packages and adds missing packages. These are displayed as colorless icons, unless you click to pin them.

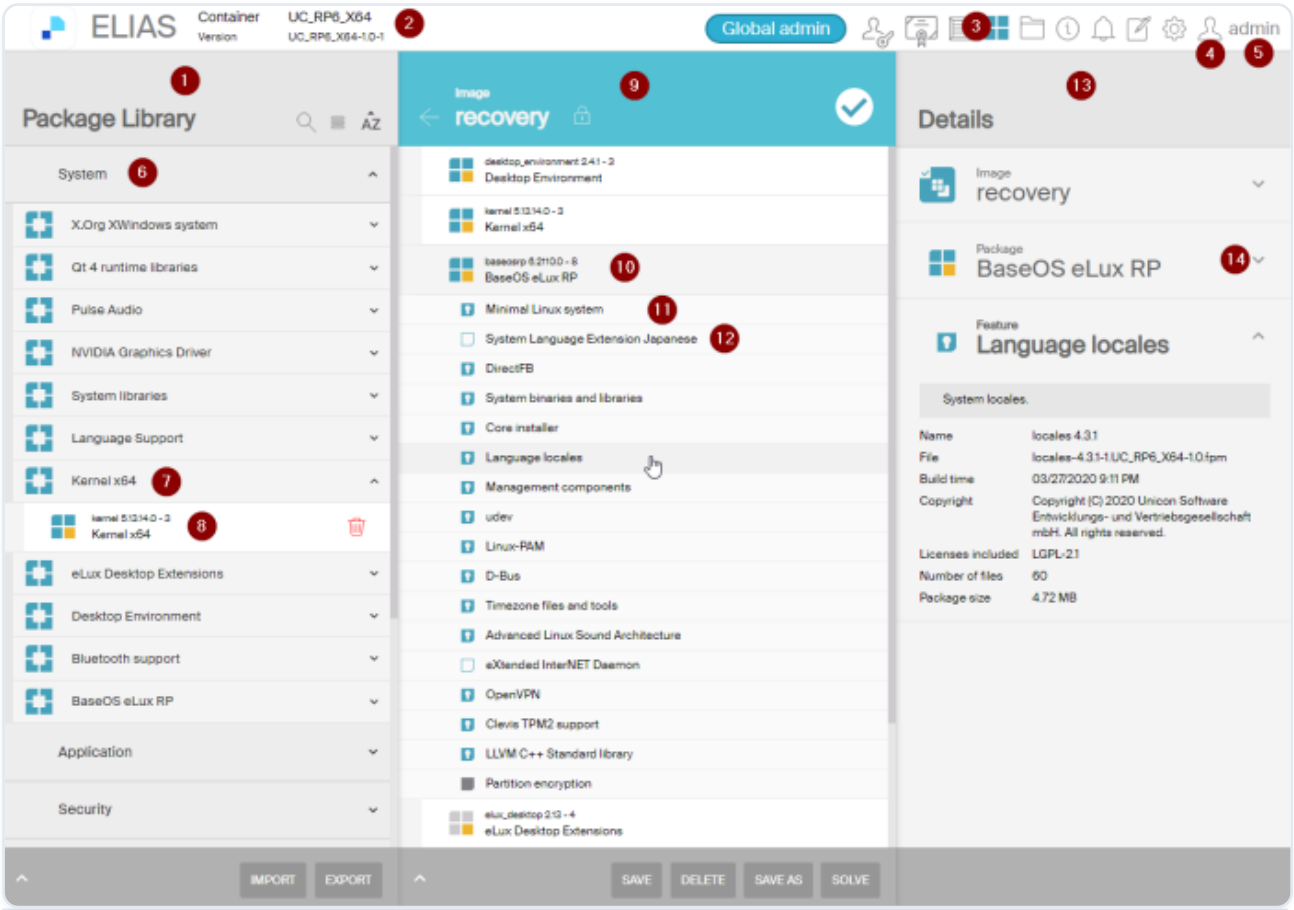
4. Click **Save** to save the image.

Interface

After you have installed ELIAS, you can change its look to something similar to below. To do so:

- Fill your container by importing software packages.
- Create an image template.
- Solve the template to an executable image.

When you use ELIAS, you will mostly use the **Package Library** view. The ELIAS interface then is divided into two sections. The figure shows a container with imported software packages on the left (**Package Library**) and an open image to the right of it (**Image Library**). On the very right, you see details of the image and selected package and feature.



1	**Package Library** of the open container
2	Name and version of the open containers
3	Switch from **Package Library** to **Certificate Library** or **Container Library**
4	Open user profile
5	User **admin** logged on
6	One of the categories the software packages are grouped by
7	eLux® software package (unversioned)
8	eLux software package in current version
9	**Image Library** with open image (IDF) related to the open container, is currently being edited and is marked as complete

10	eLux software package defined in the image
11	Subordinate feature, mandatory
12	Subordinate feature, optional
13	Details of the image and selected package and feature
14	Hide/Show details

In the **Image Library**, you can alternatively open an image template or show all available images and image templates.

ELIAS menu bar

The icons on the menu bar depend on the users logged on and the ELIAS role assigned to them. Members of the **Info** role have only few functions available, the **Global Admin** has all functions, see figure above.

☐	ELIAS role of the logged-on user or Global Admin
☐	ELIAS access rights
	only visible for Global Admin
☐	Certificate Library
☐	Container Library
☐	Package Library
☐	Container Explorer[1[[ ]from ELIAS 18 2104]]
☐	Information on the software version
	API and documentation links

☐	View all notifications
☐	Change log for images and templates
☐	ELIAS settings
	only visible for Global Admin
☐	User profile
	• Change language and date format
	• Change password
	• Log off

ELIAS menu bar

The icons on the menu bar depend on the users logged on and the ELIAS role assigned to them. Members of the **Info** role have only few functions available, the **Global Admin** has all functions, see figure above.

☐	ELIAS role of the logged-on user or Global Admin
☐	ELIAS access rights
	only visible for Global Admin
☐	Certificate Library
☐	Container Library
☐	Package Library

	Container Explorer[1[[ ]from ELIAS 18 2104]]
	Information on the software version
	API and documentation links
	View all notifications
	Change log for images and templates
	ELIAS settings
	only visible for Global Admin
	User profile
	• Change language and date format
	• Change password
	• Log off

Library icons

Image Library (Overview) icons

ICON	DESCRIPTION
	Image template
	Serves as a basis for all images (IDFs)
	Incomplete image
	Image definition file (IDF) that is not yet operational
	Complete image

ICON	DESCRIPTION
	Image definition file (IDF) that is operational
	Imported image that has defined packages/ features that are not contained in the container
	To make the image operational, replace or delete the packages/features shown in red.
	Signed image

Package Library icons

ICON	DESCRIPTION
	eLux software package
	Package placeholder (unversioned package) in the Package Library
	Under a package placeholder, one or more specific versions can be found
	eLux software package in a specific version
	Signature is verified
	eLux software package in a specific version
	Signature is not verified

Package and feature icons used in open images or templates (Image Library)

ICON	DESCRIPTION
	eLux software package
	Can act as a package placeholder (unversioned package) in a template but is transformed in a versioned package when you create an image (with same icon)

ICON	DESCRIPTION
	eLux software package in a specific version, unpinned
	• Automatically added to an image as required package when you solve a template that lacks these packages.
	• Can be created by you: After dragging packages with a specific version into your image or template, click their icon to unpin them.
	Unpinned software packages are replaced by the latest version when you solve an image or template.
	The yellow corner indicates that the signature has been verified by ELIAS.
	eLux software package in a specific version, pinned
	• When you drag software packages with a specific version into your image or template, they are automatically pinned.
	• You can pin automatically added packages to anchor the current package version.
	The package version of pinned software packages is saved in an image/template and remains anchored when you solve the image or template.

ICON	DESCRIPTION
	The yellow corner indicates that the signature has been verified by ELIAS.
	Mandatory feature of a package that you cannot deactivate
	Optional feature, excluded
	Optional feature, included

Package Library

In the **Package Library**, the container you have selected in the **Container Library** is displayed.

Once you have imported software packages into the container, they are shown in the **Package Library**. The figure below shows the packages of an imported `eLuxRP-6.xxxx.x_AllPackages.zip` file.

> **Note:**
>
> The term **Packages** refers to eLux® software packages (EPM, eLux package module). They are stored in `.epm` files and normally have subordinate features (FPM, Feature package module) which are stored in `.fpm` files.
>
> The **Package Library** allows you to do the following:

- Import and export packages. For further information, see Importing software packages .
- Select and define packages for an image or image template. For further information, see Creating a template.

1	Search for package names and descriptions
	For further information, see Searching for packages, images and templates

2	Switch to list view: Shows packages in alphabetical order
3	Toggle alphabetical sort order (ascending / descending)
4	Category: Related packages are grouped in categories.
5	Package placeholder (unversioned package)
6	Packages in specific versions
7	Delete package ([Admin] role required)
8	Show/Hide subordinate entries

Note:

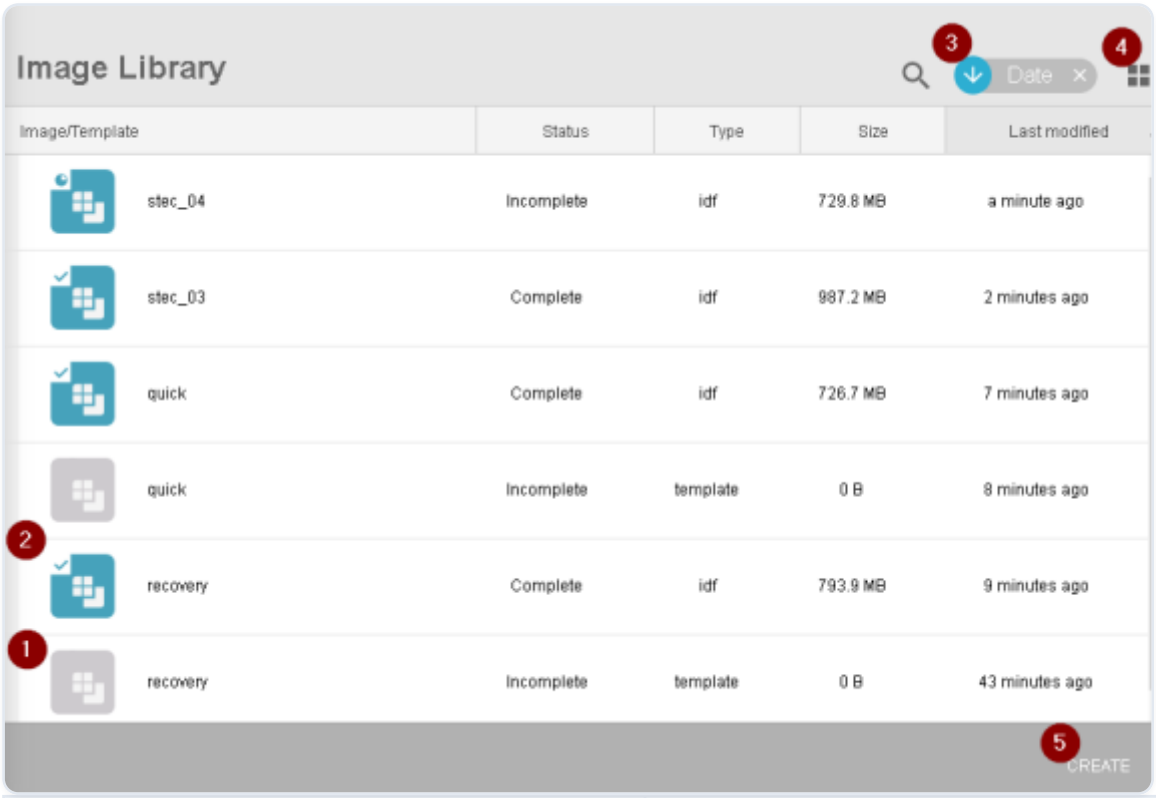
You will probably mostly work with the **Package Library** and **Image Library**. If required, to change the view, click one of the buttons **Certificate Library** or **Container Library** on the ELIAS title bar.

Image Library

The **Image Library** contains all images and templates created so far. As soon as you open an image or template, you will see their details.

The **Image Library** allows you to do the following:

- From the overview (/en-us/unicon-elux-scout/2603-ltsr-cu1/admin_guides/elias_18/elias18_idf/elias18_template_create.html).
- From an open image/template: Solve templates or images to create operational images. For further information, see Creating an image.
- From an open image/template: Edit, sign, export, delete and save templates and images under a new name.



1	Image template, click to open
2	Image (IDF), click to open
	operational/complete
3	Change sorting
4	Switch to icon view
5	Create new template

The icon color indicates whether it is a template or an image. If a template or image is complete in the sense of operational, it is displayed with a check mark.

> **Note:**
>
> All images are based on templates. Templates do not need to be complete but images do to be operational.

Container Library

> **Note:**

> All container functions require the **Admin** role.

- To view the **Container Library**, on the ELIAS title bar, click the **Container Library** icon.

1	Name and version of the currently open container (dark highlighting)
2	A container version is undefined until the first software package or image/template is imported.
3	Clean up container
	All packages that are not used by any image are deleted. Sub-ordinated features are deleted unless they are used by an active package.
4	Delete container
	After confirmation, the container and its contents are deleted.

You can manage multiple containers, for example one for each operating system version. The first container is created by the installation routine. To create further containers, at the bottom, click the **Create** button. For further information on creating and renaming, see Creating a container.

Note that initially, the containers are empty. In the **Package Library**, to fill them, import software packages as `.zip` files. For further information, see Importing software packages .

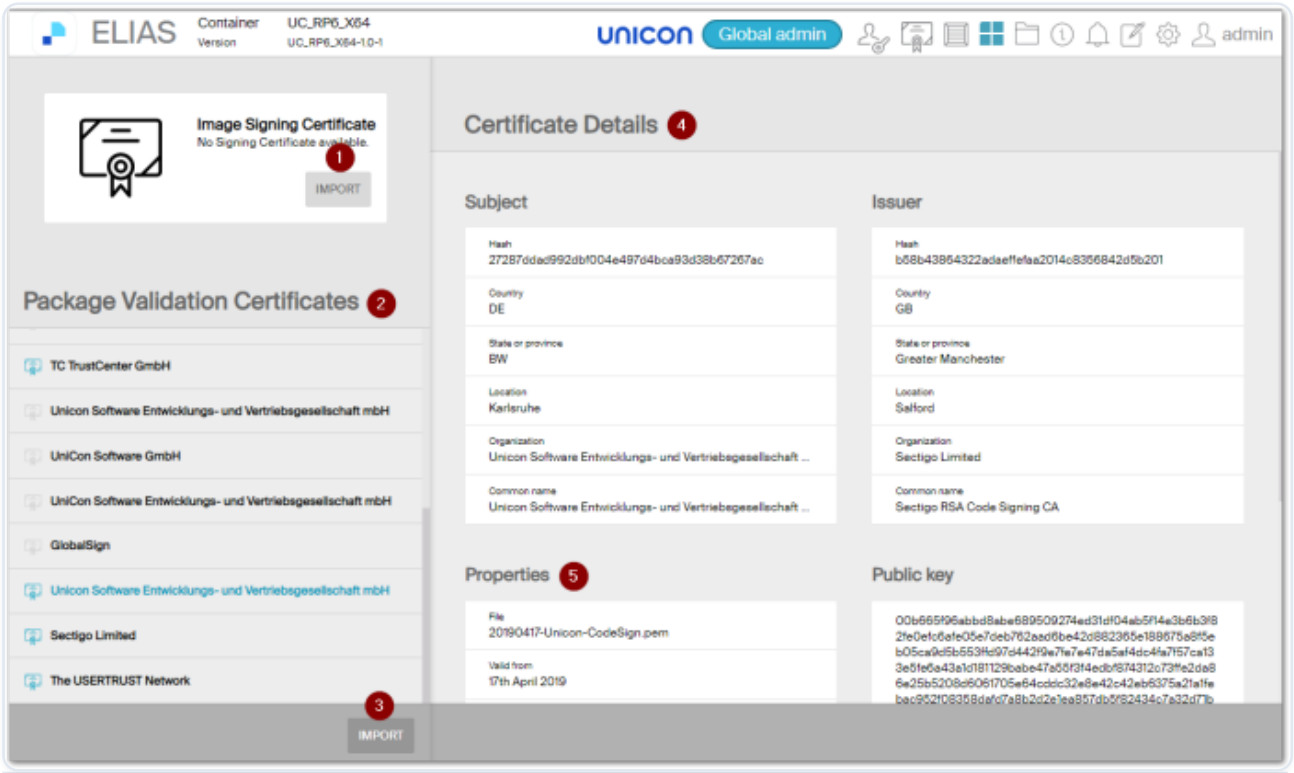
The **Container Library** is only used to create, delete, clean up and select containers.

> **Note:**

> To show the content of a container, in the **Container Library**, click the relevant container name or icon. It will be displayed in the **Package Library**.

Certificate Library

With the **Certificate Library**, you can manage package validation certificates and image signing certificates.



1	Each container may include one private signing certificate for signing images and templates.
	If no image signing certificate has been imported yet, the button is named [Import].
2	Certificate store for package validation certificates and general certificates
	When you install ELIAS, the current package validation certificates are also installed.
3	Import further or newer certificates
4	Details of the selected certificate

5	File name and validity period of the selected certificate

For further information, see Managing certificates.

Container Explorer

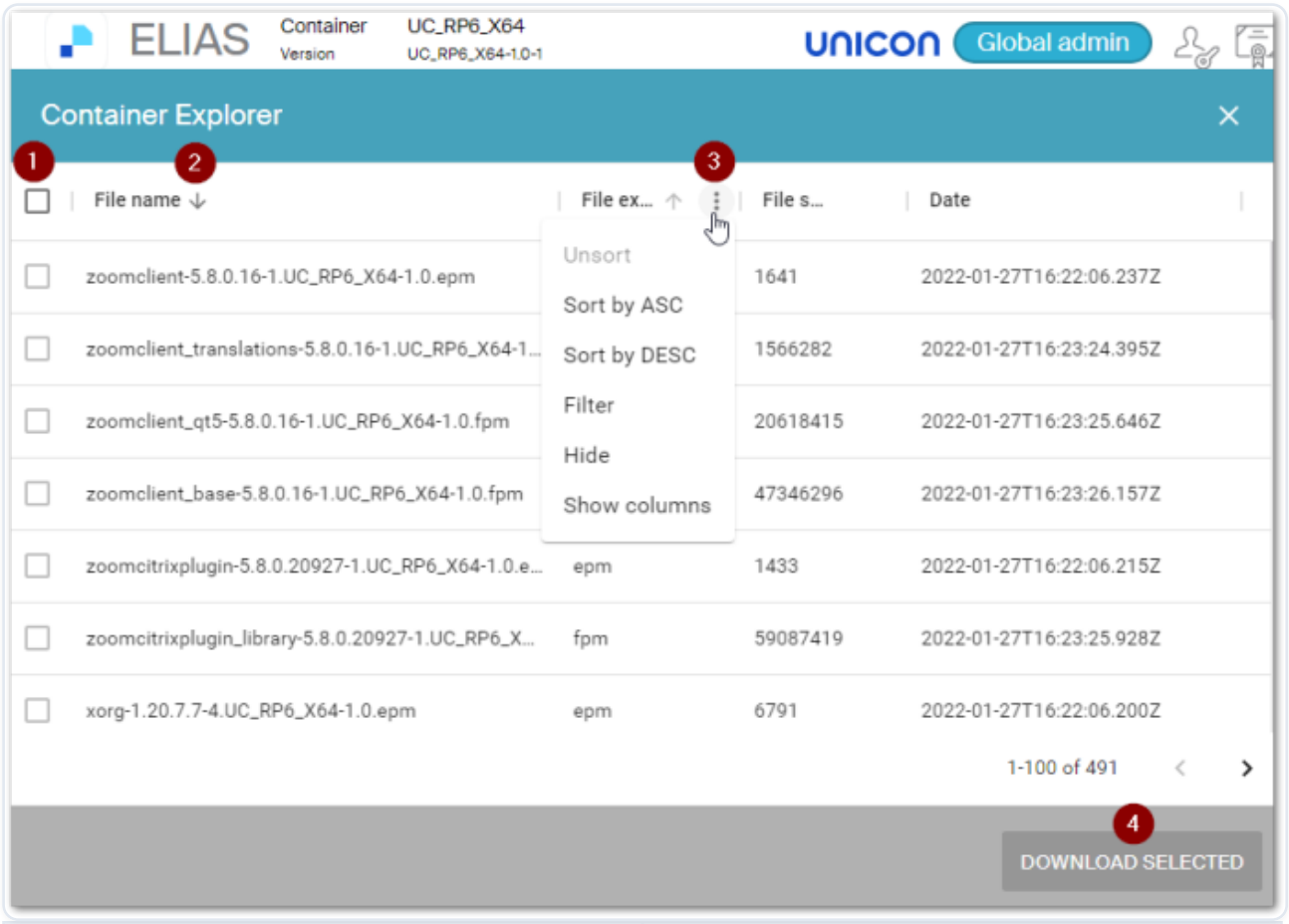
-- from --

REQUIRES
ELIAS access right with global access (Global admin)

In the **Package library**, all **packages** of a container are displayed. To view all **files** of a container in list form, the **Container Explorer** is provided. eLux® software packages (`.epm`), their child features (`.fpm`), archive files, signature files, recovery system files and .dd files are displayed.

The **Container Explorer** allows downloading files that are not part of a package. Export of all packages of a container can be done from the **Package library**, export of an image from the **Image library**. In the **Container Explorer**, the administrator now has the option to download other files such as archive files.

For the download, a `.zip` file is created with the selected files, which you can download via the browser function.



1	Select all files
2	Sort by file name
	Sorting can be done for any column.
3	For each column, sorting and filtering functions are available.
4	Download selected files
	For security reasons, **.epm**, **.fpm** and **.sig** files cannot be downloaded.

Files for UEFI updates

To perform UEFI updates in the same way as firmware updates, import the required UEFI file (`.udf`) and UEFI firmware archives (`.bin, .bup, .cab`) into your container. These files are only shown in the Container Explorer.1(From ELIAS 18-2201) For further information, see UEFI update in analogy to firmware update in the UEFI/BIOS update guide.

Changing language

ELIAS is started with the language you have chosen for the installation. You can change the display language at any time:

1. On the ELIAS menu bar, click the icon .
2. In the **Logon** dialog, select a language.

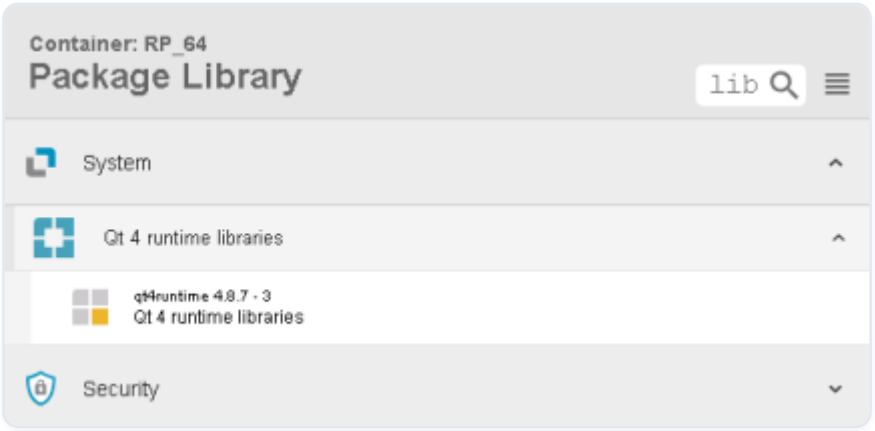
   The display language is immediately changed.
3. To close the dialog, click next to it.

> **Note:**
>
> In the **User profile** dialog, also the date format can be changed.

Searching for packages images and templates

In the **Package Library**, you can search for software packages.In the **Image Library**, you can search for images and image templates.

Your search is applied in both views, in the list view and group or icon view.



1. To open the **Search** box, click the magnifier icon.
2. Type the name of an element.You can type partial names.

After each typed character, ELIAS filters the list and only shows elements that match the search string.

Note the following:

- The search is not case-sensitive.

- The search finds all matches that contain the specified string on any position.The wild-card character * is not required and not supported.
- In the **Package Library**, the search is applied to names and descriptions of software packages and features and to their details.In the group view, open the relevant categories or groups to see the matches.Note that the search string might be used only in the details.Matches in Features and Details are found, but not necessarily displayed.Features can only be displayed in the image library, so they must be contained in an image or image template.Regardless, ELIAS will find such matches, but will only display the containing packages in the package library

For example, if you search for `editor` and have all packages of an eLux version in the package library, only the **Desktop Environment** package will be displayed.This package contains two features that use `editor` in the name.

> **Note:**
>
> To remove the filter and show all elements again, in the search box, remove the search string.

Saving views

When you are in one of the libraries, have opened a specific container, or are editing a specific image, the URL in the browser's address bar also indicates this:

- To save an individual view, save its URL as a bookmark.

Creating a container

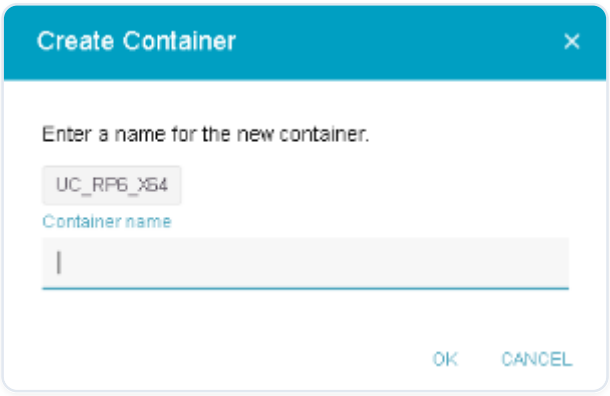
REQUIRES
Admin role

A container contains software packages of a major eLux® version. The container name can be freely chosen and can be changed later on.

A first container is created by the installer and is empty unless you import packages.

> **Note:**
>
> you can create and manage multiple containers in ELIAS 18. Within a major eLux version, it is advantageous to manage all packages and images in one container.

1. Switch to the **Container Library**.
2. At the bottom, click the **Create** button.
3. In the **Create Container** dialog, click one of the buttons to name the container according to the eLux version used. Alternatively, type a name into the text field. Special characters are allowed but no spaces.



4. Confirm with **OK**..

*The new container is created and shown in the **Container Library**. If you have selected the name via a button, the eLux version will be set and displayed immediately. If you have assigned a different name, the version remains undefined and is only set when you import the first software package or an image/template.*

- Click the new container to show it in the **Package Library** and import software packages.

> **Note:**
>
> If you use the container parameter for the firmware configuration of your devices, the container name must correspond to the given eLux version name. The container parameter allows you to operate devices with different eLux main versions with the same device configuration. For further information, see Different eLux versions in the **Scout** guide.

Renaming an existing container

1. Make sure that the container you want to rename is currently open and its name is shown on the ELIAS menu bar. It does not matter which library you are in.
2. Move the mouse pointer over the container name on the ELIAS title bar.

   An **Edit** icon is shown.
3. Click the icon and enter a new name for the container.

Creating images and templates

An image -more correct: an image definition file -defines all software packages of the device's firmware. It includes packages for the operating system and for applications.

In ELIAS 18, images are based on templates. An image template can roughly define the software packages required for an OU, while you can create slightly different images from the template. The procedure is very simple:

1. Create a template that contains the main software packages for your image.
2. Solve the template to create an image. ELIAS replaces unversioned packages by the latest package version. Also, ELIAS adds missing packages.
3. Modify the image as required and save it.

> **Note:**
>
> To enable eLux® devices to use the image for firmware updates, the file is converted internally to `.idf`. In the Scout Console, in the device configuration, the administrator specifies the path name defined during the ELIAS 18 installation and the image name with file extension `.idf` Example: `.../elias/UC_RP6_X64/recovery.idf` For further information, see Configuring firmware updates in the **Scout** guide.

> Creating a template

Creating an image

Defining and pinning packages

Creating a template

Templates are meant to save time and effort. Normally, templates cover all data that are required as a minimum. In addition, they also provide general information that is to be specified later on.

In ELIAS, use a template to provide the required software packages but, where applicable, leave the version number open. This helps you quickly create images with the latest software packages.

1. To create a new template, in the **Image Library**, at the bottom, click **Create**.
2. Type a name for the template and press **Enter**.
3. In the **Package Library**, open a category to view the contained software packages.
4. To add an unversioned software package that will act as a placeholder, select the relevant entry and drag it into your template.

   You only need to add packages that are important for you. ELIAS will add all required packages to make your image operational when you solve the template. You even need only one package - ELIAS will do the rest for you.

   > **Important:**
   >
   > If you add versioned packages, they will not automatically be updated to later versions later on. To avoid conflicts, we recommend that you always use the latest packages provided with an eLux® operating system version.

5. Optional: In the image template (Image library), show the features of a package and choose which features to include in all images that will be created based on this template. To show the features, click the package name (versioned or unversioned package).
6. Click **Save**.

*Your template is saved to the **Image Library** and is ready to be solved into an image.*

Creating an image

Images are based on image templates.

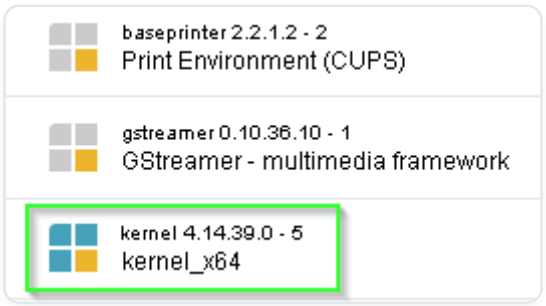
REQUIRES
An image template ready for use

1. In the **Image Library**, open the relevant image template.
2. Click **Solve** to create an image from the template.

   ELIAS replaces package placeholders (unversioned packages) by the latest package versions in the container (1).

   Also, ELIAS checks for additionally required packages and adds missing packages (2). These packages are displayed as almost colorless icons unless you click their icon.

   The image is marked as complete by showing a check mark (3).
3. To anchor or pin automatically added packages in your image, click the relevant icon.



   Pinned packages remain with their version when you solve the image the next time, even if later packages are available.

   The icon is shown with blue squares.

   Note:

   To avoid conflicts, we recommend that you always use the latest packages provided with an eLux® operating system version.
4. Optionally, to modify the image name, click it. If an image with that name already exists, you may choose a new name or replace the existing image.
5. Save the image with **Save**.

*Your image is saved to the **Image Library**.*

For further information, see Defining and pinning packages.

System-side solving versus manual image compilation

You can solve templates and images.The idea of solving is to create operational images at any time at the push of a button. If the template contains package placeholders, they are replaced by the latest package version. If the image contains unpinned packages, they are also replaced by the latest package version.

> **Note:**

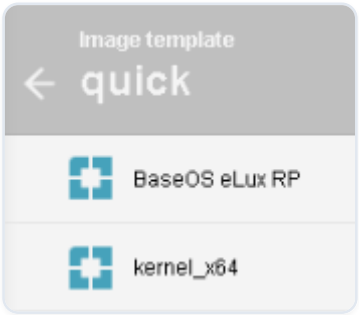
> Images can be solved multiple times to allow you to use the latest package versions each time you want to be sure to deliver an up-to-date image.

> If, however, you drag versioned packages into an image, conflicts between versions of dependent packages may occur. In such a case, the newly added package is displayed in red and the image cannot be resolved. Then check the package versions and use appropriate packages.

Defining and pinning packages

Images are based on templates. You can create images by solving a template or an existing image. To see what options are available for defining software packages for your image and what impact they have, view the following example.

Our example image is based on a very simple image template with only two package placeholders (unversioned packages):



For further information, see Creating a templateand Creating an image.

Once we solve the template and create an image from it, we receive an operational image. ELIAS then completes the image for us by:

- Adding all required but missing packages in the latest available version

- Solving the package placeholders defined by the template to packages in the latest available version

After the image has been created, it looks like this:

1	The image is complete
2	Defined packages
	The latest version has been selected automatically.
3	Automatically added packages that are required to make the image complete
4	Click the package name to expand the package and view its features.
5	These features are locked: You cannot deactivate them.
6	Click the package icon to pin or unpin the package.

You can save the image and use it as it is. Also, for most packages, you can select or deselect included features. Moreover, you can pin package versions to anchor the version to your image definition.

Selecting individual features of a defined package

1. In the **Image Library**, for the relevant package, click the icon to pin the package.

> **Note:**
>
> This step is necessary to ensure that the relevant features are included in the package. Otherwise, when you re-use and solve the image later on, an updated package version might lack the feature that you have explicitly selected.

2. To expand the package, click the package name.

   The features of the package are displayed.

3. Select or deselect features as required. Note that you cannot deactivate locked features .

> **Note:**

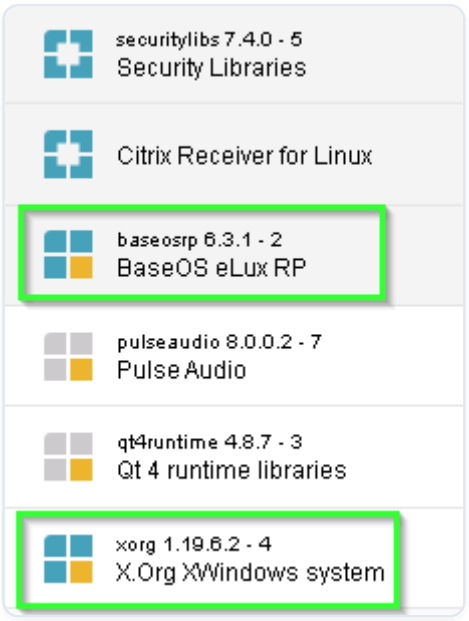
> You can already define in the image template which package features are to be included or excluded.

Pinning package versions

When solving a template or image, ELIAS defines the latest versions for automatically added packages and for package placeholders. To save a specific version with the package in the current image, pin the package:

- Click the relevant package icon.

The icon of a pinned package is shown with blue squares . The package version is saved in the image and remains anchored when you solve the image at a later time. In contrast, unpinned packages are replaced by a later version if available.



> **Note:**

> To avoid conflicts, we recommend that you always use the latest packages provided with an eLux® operating system version.

Managing images and templates

To work with an image or template, in the overview of the **Image Library**, click to open it. The operations described below can only be applied to open (and saved) images or templates.

1	Back to the overview of the **Image Library**
2	Rename image or template
	Click the name and enter a new name.
3	Image or template is unlocked
4	Image or template is complete
5	Preferences
6	Lock/unlock image or template
	To locked images or templates no changes can be made.
7	Sign image or template
	Requires a code signing certificate. For further information, see Signing an image.
8	Export an image with or without packages
	The **Export** button is only shown if the image or template is saved. As soon as you make a change, the **Save** button is shown instead.
	For further information, see Exporting an image.

9	Delete the image or template from the container
10	Save a copy of the image or template under a new name
11	Solve the image or template
	For further information, see Creating an image.

Locking an image

Once you have created a template which is perfectly suited as a basis for creating images, you have the option to lock it. This means that your template will be write-protected and no further changes can be made. The same applies to images: Once you have created an image ready for roll-out, you can lock it. No more changes can be made.

REQUIRES
Admin or **User** role

1. In the **Image Library**, open the relevant image or template. Make sure it is saved.
2. Click **Lock**.

*No more changes can be made. The only remaining actions are **Unlock**, **Export** and **Save as**. Note that the **Lock** icon next to the name has changed.*

Signing an image

Images can be provided with a signature that is verified when a device's firmware is updated. Templates can also be signed.

REQUIRES
• **Admin** or **User** role
• A private code signing certificate has been created and imported. For further information, see Organizing certificates for image signing.

1. In the **Image Library**, open the image you want to sign.
2. Make sure it is saved. Click **Sign**.
3. Depending on the format of the certificate, enter the path of the private key file.
4. Depending on the configuration of the certificate, enter the password.

*The generated signature file is saved in the container. A signed image is identified by a special icon next to the image name. In the overview of the **Image Library**, a signed image is shown by a yellow stripe in the icon.*

> **Note:**
>
> Any changes to a signed image in ELIAS will delete the signature. After having changed and saved a signed image, sign the image again.

> Any changes to a signed image in ELIAS will delete the signature. After having changed and saved a signed image, sign the image again.

The result of the signature verification is documented in the update log on the device. After the update is performed, the update log file is sent to the Scout Server and can be viewed for the selected device in the **Properties** window by double-clicking the **Update status** field.

Exporting an image

This feature helps you export an image so you can use it in another environment. Depending on your settings, the defined software packages will additionally be included in the target file.

REQUIRES
• **Admin** or **User** role
• To export an image with the **Container** or **StickWizz** option, it must be complete.

1. In the **Image Library**, open the relevant image.
2. To select the export format, at the bottom, click the **Preferences** button
3. Depending on your intention, select one of the following formats:

OPTION	DESCRIPTION
Image	Export image definition file as a `.jidf` file (JSON format)
	With this option, you can use your image with other ELIAS 18 instances.
Container	Export image definition file and herein defined packages in a `.zip` file (**Admin** role required)
	Use this option to manage an individual image and its variants in a dedicated container. To do so, create an additional container and import the `.zip` file.
	The `.zip` archive contains the image definition file as a `.jidf` file (in JSON format) and as a `.idf` file (in formerly used text format).
StickWizz	Export image definition file and packages as a `.stw` file and create a recovery system in a DD-like format
	For further information, see Image as a USB recovery system.

4. Click **Export**.

The image is exported as defined and made available as a download by your browser.

> **Note:**

> Templates can also be exported for use in other ELIAS 18 instances.

Exporting image definition file as an .idf file

Images created with ELIAS 18 can also be exported to the formerly used format (plain text). This option helps you use an ELIAS 18 image in the legacy ELIAS application for mixed environments, for example.

1. In the **Image Library**, open the relevant image.
2. Modify the URL shown in the browser's address bar as follows:
   - Replace the string `container` by `api`
   - Insert the string `export` / before the image name
   - Replace the file extension `.jidf` by `.idf`

   Example: `.../elias/container/UC_RP6_X64/recovery.jidf] changes to [.../elias/api/UC_RP6_X64/export/recovery.idf`
3. Call the URL (for example via `Return`).

From the current image, an `.idf` file is created and made available as a download by your browser.

Image as a USB recovery system

For all current eLux® versions we offer a ready-to-use eLux USB stick image for download on our technical portal. Alternatively, you have the option to use an individual firmware image as a basis for your USB recovery stick.

For a USB recovery stick to be bootable for eLux, it must contain the recovery system of the relevant eLux version. For an individual image, the relevant image file and all software packages defined in it are also required. The current recovery system is installed as a .dd file with the ELIAS 18 installation.[1[from ELIAS 18 2104]] Example: `eLuxRP-6.2103.0-5-ELIAS.dd`. The .dd file is located in the ELIAS program directory.

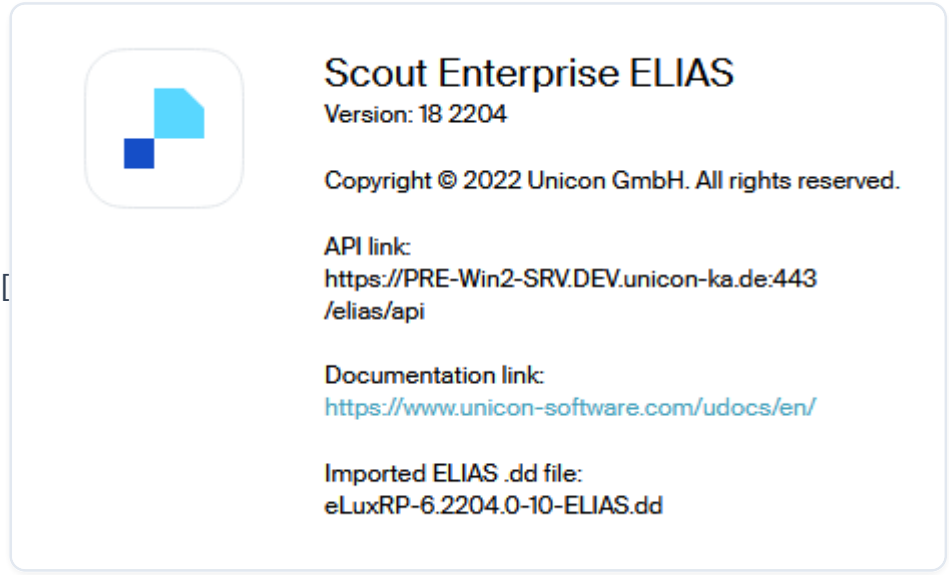
> **Note:**

> The .dd file is not shown in the Container Explorer.

> For each eLux version, Unicon additionally provides the recovery system on the technical portal.[2[ab eLux RP 6 2103]] To provide an existing ELIAS 18 container with an up-to-date

> recovery system, download the recovery system (example: `ELIAS USB Recovery system 6.2204.0-10`) from our technical portal and import it into your ELIAS 18 container.

Imported recovery systems are displayed with their version in the **Info** dialog:



When you export an image with the **Stickwizz** option, all the data you need is written to a `.stw` file, which you then download via your browser and save on a Windows PC. With the help of the **StickWizz** application, in the next step, you transfer the image from the `.stw` file to your stick.

Preparing individual USB recovery system

1. Check whether the recovery system in the ELIAS program directory corresponds to the eLux version (see BaseOS package) in your ELIAS container.
2. If required, download the appropriate recovery system from our portal and import it into your ELIAS container, see Importing software packages
3. In the image library, export your image with the **Stickwizz** option, see Exporting an image.
4. Download the `.stw` file and save it on a Windows system.

Your customized recovery system is now ready to be written to a stick.

For further information, see Recovery via USB and Writing an image to USB (with StickWizz) in the **Recovery** short guide.

Tracking and viewing changes

All changes made to images and templates are tracked and reported in a change history.

REQUIRES
Admin or **User** role

1. On the ELIAS menu bar, click .
2. Filter and sort the columns as required (see screenshot).
3. To download the change history, click **Export**.

The change history lists all actions executed on images and templates (3) with their user (1), relevant image or template (2), changed file (4) and time stamp (5).

Managing certificates

For firmware updates, you can specify that the signatures of the eLux software packages and/or the signature of the image will be verified.[1[see Device configuration in the Scout guide]] Both certificate types can be managed in ELIAS 18:

- Package validation certificates

  The current certificates for our software packages are automatically imported with the ELIAS installation. Newer certificates can be downloaded and imported from our portal.

- Image signing certificate

  To sign an image with a private signature certificate, you first need to create a code signing certificate. Then, you import the certificate into the container. You can sign images and templates.

Organizing certificates for package validation

Organizing certificates for image signing

Organizing certificates for package validation

> **Note:**
>
> Along with the ELIAS 18 installation, the current certificates for our software packages are automatically imported. Only if updated certificates are available on our portal, import them to replace the existing ones (see below).

1. Download the package validation certificates from our portal under **eLux® Software Packages**. Then extract the `.zip` archive.
2. To enable the devices to verify the software package signatures during a firmware update, transfer the certificates to the devices to `/setup/cacerts`. To do so, use the Scout feature **Files configured for transfer**. For further information, see https://author-p185304-e1959167.adobeaemcloud.com/en-us/unicon-elux-scout/2603-ltsr-cu1/admin_guides/scout/advanced_options/advanced_files.html in the **Scout** guide.
3. Make sure the signature verification for the is enabled. Signature verification is an option of the device configuration in **Device configuration > Firmware > Security**. For further information, see Firmware security through signature in the **Scout** guide.

Importing updated certificates

1. If newer package validation certificates are available on our portal, under **eLux Software Packages**, download them. Then extract the `.zip` archive into a directory that you can access with ELIAS.
2. In ELIAS 18, switch to the **Certifcate Library** and, in the bottom section, click the **Import** button.
3. Import all certificate files (Trusted Root, Trusted Intermediate, Trusted Issuer) into the ELIAS 18 certificate store, one after the other.

   ELIAS 18 gives feedback whether the certificates have been saved or already exist. The individual certificates are saved and displayed according to the certificate chain.

   For further information, see Certificate Library.

> **Note:**

If you create your own eLux packages and want to apply certificates to them, use the PKCS#6 standard with PEM encoding (file formats `.pem/.crt`). We recommend that you also import the certificates into ELIAS so that the packages are displayed as signed.

Organizing certificates for image signing

REQUIRES
Certificates must comply with the PKCS#12 standard and be DER-encoded. Certificate and private key are included in a `.pfx` file.

1. Create a private code signing certificate.

   Note:

   The **Organization** attribute of the certificate must be filled to use a certificate in ELIAS.

2. In ELIAS 18, switch to the **Certifcate Library** and import the certificate file of the signing certificate and the root certificate into the general certificate store. To do so, in the bottom section, click the **Import** button.
3. Import the signing certificate (such as `.pfx`) into the certificate store intended for image signature certificates. To do so, in the top section, click the **Import** button.

   A container can contain one image signing certificate at a time. This can be replaced at any time.
4. To sign an image, switch to the **Image Library** and open the image you want to sign. Make sure it is saved. Click **Sign**.

   The generated signature file is saved in the container. A signed image is identified by a special icon next to the image name.

   Note:

   In addition to images, you can also sign templates.

5. Transfer the signing certificate and root certificate to the devices to . To do so, use the Scout feature **Files configured for transfer**. For further information, see https://author-p185304-e1959167.adobeaemcloud.com/en-us/unicon-elux-scout/2603-ltsr-cu1/admin_guides/scout/advanced_options/advanced_files.html in the **Scout** guide.
6. Make sure the signature verification for the devices is enabled. Signature verification is an option of the device configuration in **Device configuration > Firmware > Security**. For further information, see Firmware security through signature in the **Scout** guide.

For further information, see Certificate Library.

ELIAS settings

REQUIRES
Administrator with **global access**.

- To change any ELIAS settings after it has been installed, on the ELIAS menubar, click **ELIAS settings**.

Active Directory

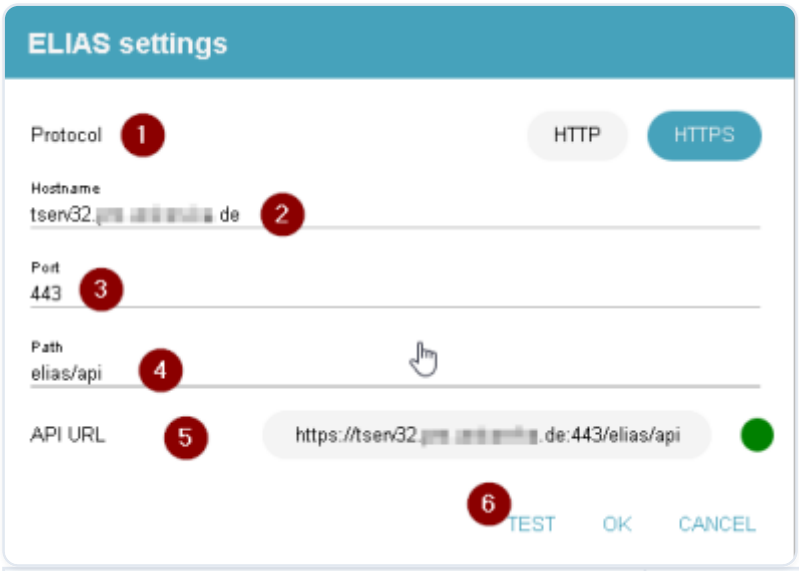
- AD protocol (**LDAP** or **LDAPS**)
- AD domain

Authentication

For user authentication, the logon types **Active Directory** and **Keycloak** can be enabled or disabled.

API address

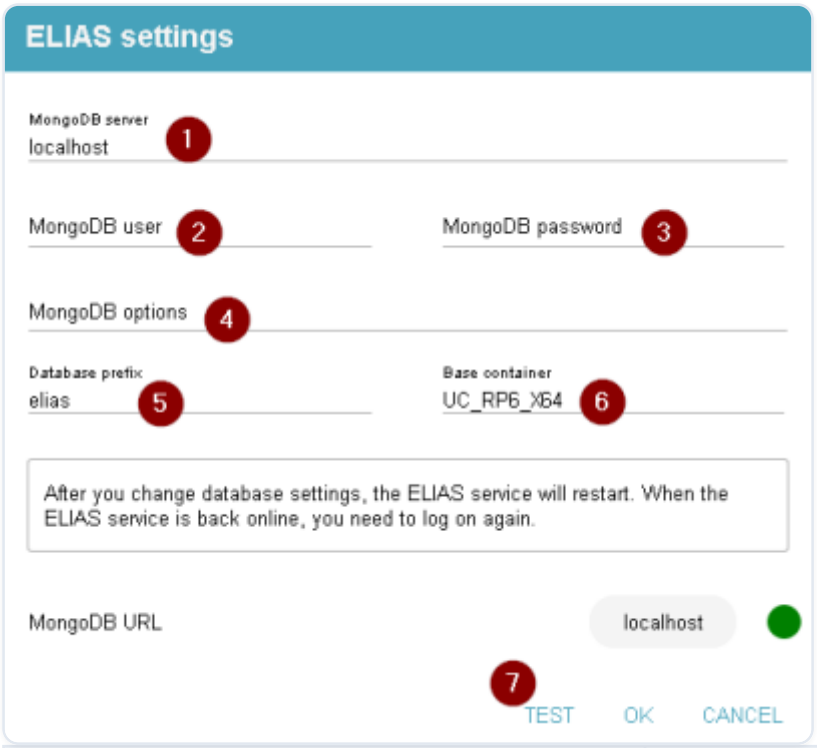
Server settings and API



1	Protocol used to access the ELIAS server
2	Host name of the ELIAS server
3	Port for the ELIAS service
4	API path
5	Resulting API URL
6	Check accessibility of the ELIAS API

MongoDB

MongoDB server and MongoDB database settings



1	Host name of [MongoDB] server
2	Username for [MongoDB] database
3	Password for [MongoDB] database user
4	Optional parameters for [MongoDB] server
5	Database prefix for ELIAS databases
6	Name of the base container
7	Check accessibility of the [MongoDB] server

Log level

Enable or disable enhanced logging (**Debug**)

By default, enhanced logging is disabled.

The log files are located in the following directory:

- Windows: `C:\Windows\System32\config\systemprofile\AppData\Roaming\ELIAS`
- Linux: `/var/log/ELIAS`

ELIAS

The eLux® Image Administration Service (ELIAS) allows you to manage the firmware of devices running eLux.

Overview

The eLux Image Administration Service (ELIAS) allows you to easily manage the firmware of devices running eLux. Default images such as `recovery.idf` contain the operating system and applications but can be modified by using ELIAS. ELIAS helps you create new image definition files (IDF) or modify the existing ones.

For an individual client image, you select exactly those software packages for your IDF you want to have installed on the devices' flash memory. An IDF is a structured package list defining all software packages for the operating system and applications.

Administrators can select all required software packages available in a container and define them in an IDF. Once created, an IDF can be used as target for a firmware recovery or a firmware update. After successful recovery or update to an individual IDF, the firmware image installed on a device is consistent with the image defined in the relevant IDF.

ELIAS is part of the Scout Enterprise Management Suite and may be installed as an optional component of it. For further information, see Installing Scout Enterprise Management Suite in the **Installation** guide.

ELIAS offers the following features:

- User-friendly graphical interface that allows you to easily create software images
- Easy management of device firmware by using the "container method"
- Software components can be activated/de-activated to optimize the use of available resources
- Image size bar to avoid exceeding flash size
- Automatic check of software dependencies
- Import feature to add the latest software versions
- Export feature to export a package along with its software dependency information
- Digital signatures to verify the originality of software packages and IDFs.
- Creating USB Recovery sticks or disk images containing an individual software image

Starting ELIAS

Keyboard shortcuts

Starting ELIAS

From the Scout Console:

- Click **View > ELIAS**.

From the Windows start menu:

- Click **Programs > Scout > ELIAS**.

If you have installed the Scout Enterprise Management Suite® by using the default settings, ELIAS automatically opens the defined container at start-up. Otherwise, you have to select the relevant container in a first step.

Keyboard shortcuts

KEYS	SELECTED ELEMENT	DESCRIPTION
↑ ↓	Any list entry	Moves selection
TAB	Anywhere	Screen navigation in the ELIAS window
CTRL+←	**Available packages** list entry	Adds selected package to the **Defined packages** list
DEL	**Available packages** list entry	Removes the selected package from the container
INSERT	**Available packages** list entry	Opens the **Import package** dialog
→ ←	**Defined packages** list entry	Shows or hides feature packages (FPM) of the selected package
CTRL+→	**Defined packages** list entry	Removes the selected package from the **Defined packages** list

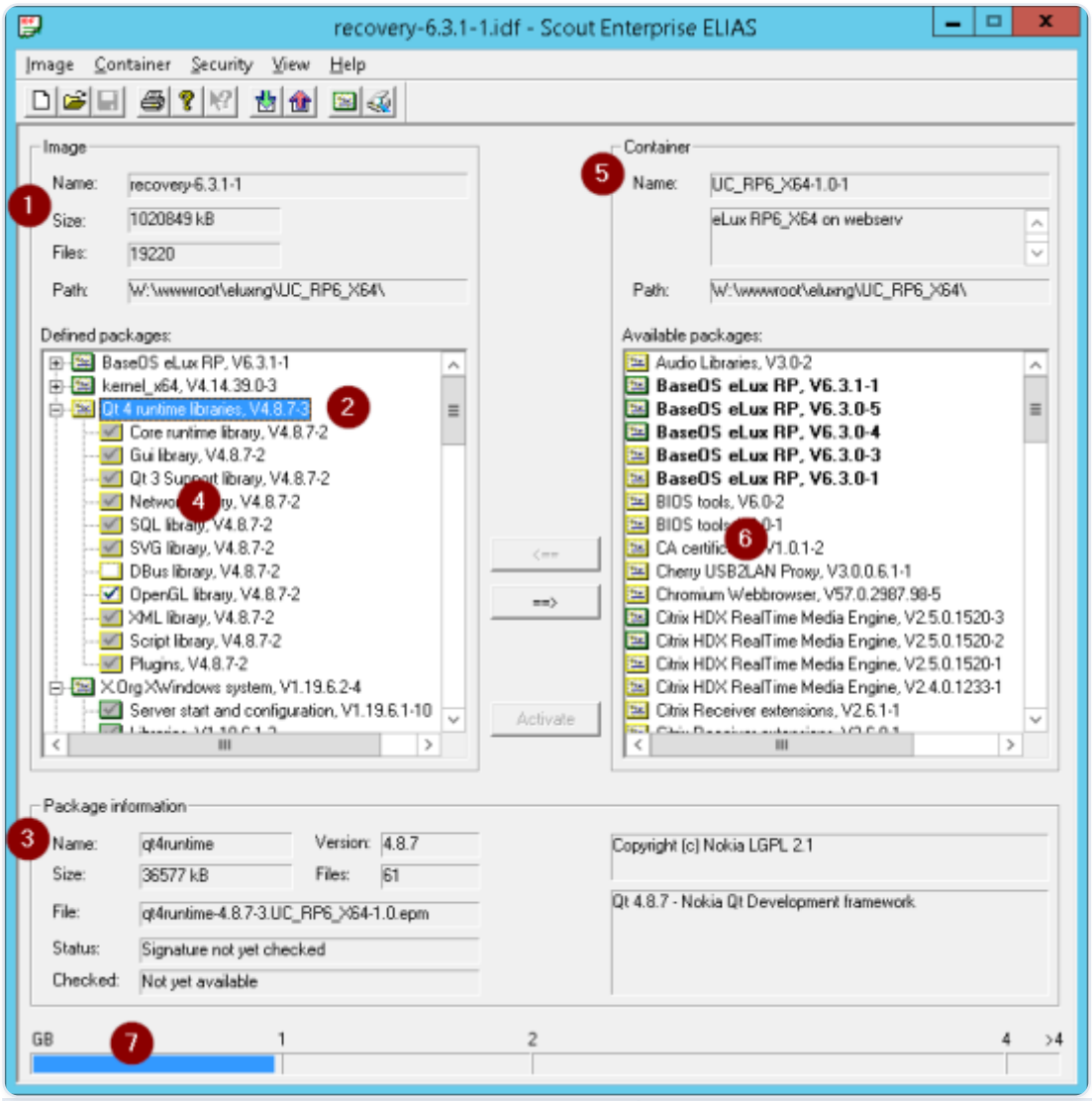
KEYS	SELECTED ELEMENT	DESCRIPTION
SPACE	**Defined packages** list entry	1. Expands/collapses the selected eLux package (EPM)
		`2`. Activates/deactivates the selected feature package (FPM)

Interface

The ELIAS application window is divided into the **Image** section on the left and the **Container** section on the right.

- To view the available packages on the container, select the relevant software container from your web server or FTP server.
- To view an existing image definition file (IDF), open the relevant `.idf` file from the selected container. If you create a new image definition file, you start with an empty list in the left section.

With both lists in front of you, you can easily define your image, adding software packages from the container, and if required, removing packages from the IDF.



	LEGEND TO NUMBERS
1	Image definition file (.idf) with name, total size of all packages selected so far, number of defined files and storage location
	The current image size is graphically represented on a status bar on the bottom of the window (7).
2	Software package defined in the IDF (eLux® package module, .epm)
3	Details for the selected software package (selected in either the left or the right list)
	Name, version, size, number of included files, signature status, signature proof date as well

	LEGEND TO NUMBERS
	as the copyright information and the description of the selected feature or package
4	Subordinate features of the software package (feature package module, .fpm)
5	Selected software container with name, description and storage location
6	Software packages provided in the software container (.epm)
7	Image size based on currently selected software packages

Installation: eLux® Container

> **Note:**

> If you use the web-based ELIAS 18, there is no need to install the eLux container. In ELIAS 18, you simply import the relevant software packages or the AllPackages bundle. For further information, see Importing software packages in the **ELIAS 18** guide.

> The eLux container is a compilation of software packages used to create the firmware (IDF) of a client. The administrator selects a subset of the software package pool to define an IDF (Image definition file) that can be installed on the devices.

The software packages can be provided via the container installation on a web or FTP server (legacy ELIAS) or via the ELIAS 18 web service in a MongoDB (locally or in combination with IIS).

For the main operating system versions , eLux RP 6 / 32-bit and eLux RP 5, dedicated containers are provided. Current eLux RP 6 / 64-bit versions are installed under `…eluxngUC_RP6_X64`.

Installing a container

Exporting packages and containers

Importing packages into a container

Deleting packages from a container

Installing a container

The installation is covered by the Installation guide. For further information, see Installing a container.

Exporting packages and containers

> **Important:**

> Use the ELIAS **Export** feature instead of copying files by using the Windows Explorer. This is to ensure that all sub-components and basic information on dependencies are included.

> From ELIAS, you can export all packages of an IDF or individual packages:

- By clicking **Image > Export...** you create a new container to which all the packages defined in the relevant IDF are copied. This is particularly useful if you want to update your devices from a USB mass storage device and all packages (optional: only these) should be included on the USB stick to which you want to update.
- By clicking **Container > Export Package...** you export individual packages, for example to provide them on the Scout Server without internet access. As for the import feature, all features of the software packages are automatically transferred and there is no need to copy anything manually.

Exporting all packages of an IDF into a new container

1. Click **Container > Select...** to open the existing container.
2. Click **Image > Open...** to open an existing IDF, or click **Image > New** to create a new image. See also Creating a new IDF. Alternatively, use the provided icons.
3. Click **Image > Export...**
4. Click **Browse** and select a target directory as the new container, and then type a name for the IDF. Confirm with **Save**.
5. Optional: If the target directory, for example on an eLux® Recovery USB stick, already contains packages but you only want the defined packages from the IDF, select the option **Deleting packages that are not defined in the IDF from the target directory**.
6. Confirm with **OK..**

The IDF and the defined packages including all features (components) are copied into the new container located in the target directory. The container includes both, the activated and the non-activated features.

Exporting individual packages

1. To open the relevant container, click **Container > Select...**.
2. On the right, select one of the packages of the container.
3. Click **Container > Export package...**.
4. Select the target directory and confirm with **Save**.

   The standard name of the package must not be changed. The package is saved to the selected directory.
5. Repeat these steps until all required packages have been exported.

Importing packages into a container

A container published on our technical portal includes **all** software packages which have been released for the relevant eLux® operating system version. In some cases, there is no need to download a complete container as only few packages suffice.

The **Import Packages** feature allows you to import either a new software package or to overwrite an already existing package in your container. All features and the software dependencies of a software package are kept.

Important:

- To import packages, use the ELIAS **Import** feature instead of copying files in the Windows Explorer. This is to ensure that all sub-components and basic information on dependencies are included.
- Do not unpack the `.zip` file. This is done by ELIAS.

1. To download individual software packages, sign in on our myelux.com portal.
2. Under **Download > eLux Software Packages**, select the container for the required eLux version and download the relevant package.
3. Save the package on your web server into the directory containing the relevant container. Do not save the package directly into the container!

The software package is saved as a `.zip` file.

4. Start ELIAS and open the required container by clicking **Container > Select...**
5. Click **Container > Import Package**.
6. Select the `.zip` file from the file system and click **Open**. You can open more than one `.zip` file in one step.

The software package is added to the container currently opened in ELIAS.

Deleting packages from a container

To remove out-of-date software from a container, delete the relevant software package.

> **Important:**
>
> Deleting packages cannot be undone. Before deleting, check if the relevant package is part of any image definition file `.idf`. To restore packages that have been deleted accidentally, import the package once again

1. In ELIAS, open the relevant container.
2. Select the software package you want to delete.
3. Press **DEL**.
4. Confirm the message with **Yes**.

IDF

An image definition file is a structured package list defining all software packages for the operating system and applications.

One of the basic ELIAS features is the built-in mechanism checking for conflicts and dependencies regarding other packages. Required packages are automatically identified and requested. They can easily be added with just one click. This prevents software incompatibilities in the image definition file.

Creating a new IDF

Updating existing IDFs

Saving an IDF

Signing an IDF

Exporting an IDF

Printing an IDF

Creating a new IDF

In an image definition file, you define those software packages you want to have installed on the devices' flash memory. This allows you to compose images to be installed according to individual requirements.

In ELIAS, the first step is to load the contents of a software container. These are then shown as available packages on the right. From the available packages, create a list of the packages you want to include in the IDF on the left.

> **Note:**
>
> To avoid conflicts, we recommend that you always use the latest packages provided with an eLux® operating system version.

> During the definition process, note that details on the software packages and the current IDF are shown on the bottom. For further information, see Interface.

Creating a new IDF

1. Make sure that the relevant container is shown on the right. If this is not the case, click **Container> Select...** and, in the file system, on the FTP or HTTP server, open the relevant container directory (Example: `UC_RP6`) with the `container.ini` file.

   *The available container packages are listed on the right under **Available packages**.*

2. Click **Image** > **New**.

   On the left, the list of **Defined packages** is empty when you start.

3. On the right, click a container package. You might begin with the relevant `BaseOS` package. Alternatively, you can begin with the required application, for example Citrix Workspace-App.

   For the selected package, the details are shown below.

4. Click the left arrow button **<==** to add the selected package to your IDF.

   The container package selected on the right is copied to the **Defined packages** list on the left.

5. If a package requests further packages or features for correct installation, confirm with **OK**.

   The requested packages are also copied.

   > **Note:**
   >
   > ELIAS continues requesting required software packages until the defined firmware is operational. You can only save an IDF that is operational. For example, the `BaseOS`, `XWindows` and `GNOME Desktop` packages represent a minimal eLux firmware. Depending on the device's hardware, additional driver packages might be required such as VIA or ATI graphics drivers.

6. To activate/deactivate individual features within a defined package, click the `+` icon of the package on the left, select the relevant feature and press the `Spacebar`. For further information, see below.
7. Add all relevant packages to the **Defined packages** list.

   All packages displayed on the left will be included in the IDF.

8. Click **Image** > **Save**, choose a descriptive name and save the new IDF to the container.

Activating/Deactivating features in defined packages

Each `.epm` package contains different components, the so-called features `.fpm`. To expand the package and show the features, click the `+` icon in front of the package symbol. The features have different symbols:

gray box	Mandatory component that cannot be deactivated
white box	Feature you can activate or deactivate

- Press the `spacebar` to activate or deactivate.

or

select the feature and click **Deactivate/Activate**.

Deleting packages from an IDF

1. Select a package from the **Defined packages list** on the left.

2. Click **==>**.

The package is removed from the Image Definition File.

Size restriction

While defining an IDF, there is no restriction regarding the maximum size. The total size of an image may be greater than the capacity of the flash memory as long as you are trying and testing. The final IDF, however, must not exceed the size of the flash memory. Check the total size before saving the IDF.

Updating existing IDFs

Updating the firmware of your devices can become necessary or reasonable when specific packages are available in a newer version, for example. In ELIAS, you can easily replace individual packages in your IDF.

> **Note:**
>
> - The installed BaseOS can be replaced quickly and safely by another BaseOS version. All packages required are also replaced by the respective versions, provided you confirm the replacement in the upcoming dialogs.
> - To avoid conflicts, we recommend that you always use the latest packages provided with an eLux® operating system version.

1. Make sure that the relevant container is shown on the right. If this is not the case, click **Container > Select...** and, in the file system, on the FTP or HTTP server, open the relevant container directory (Example: `UC_RP6`) with the `container.ini` file.

   *The available container packages are listed on the right under **Available packages**.*

2. Open the relevant IDF. To do so, click **Image > Open...** and load the relevant `.idf` file.

   *On the left, the list of **Defined packages** is shown.*

3. To add packages to your IDF, select them on the right and click the left arrow button **<==**
4. To activate/deactivate individual features within a defined package, click the `+` icon of the package on the left, select the relevant feature and press the `Spacebar`.
5. To delete a package from the IDF, in the **Defined packages list** on the left, select the package and click **==>**.

For further information, see Creating a new IDF.

6. Click **Image > Save**, or click the relevant icon to save the updated IDF to the container.

The devices will be updated to the modified IDF without further action if they are configured to check for new versions on start or shutdown.

> **Note:**

> If an IDF is unchanged, the devices will not update to it as Scout checks the file for modifications.

Saving an IDF

1. Check the total size of the image shown in the upper section under **Image**.The total size of an image must not exceed the capacity of the devices'flash memory.
2. Select **Image > Save** `or` **Image > Save as...**
3. Choose a descriptive file name without white spaces for your IDF.
4. Make sure the target directory is pointing to the correct container directory. Normally, the container is located in the root directory of the FTP or HTTP server.
5. Click **Save**.

The IDF is saved to the respective container.

Signing an IDF

ELIAS provides the possibility to sign image definition files (IDFs) by using private code-signing certificates. To do so, you need to import an appropriate certificate into the ELIAS certificate management.

1. Create and import a code signing certificate (`.pem/.cer Base64-encoded`) in the certificate management of ELIAS. To do so, select **Security > Manage certificates... > Import...**

> **Note:**

> The **Organization** attribute of the certificate must be filled to use a certificate in ELIAS.

2. To sign the IDF currently opened in ELIAS, ensure that it is saved, and then select **Security > Sign IDF...**

   Depending on the format of the certificate, the signing process might involve specifying the file path of the private key file.

   The generated signature file is saved to the directory (container) of the IDF. Once an IDF is signed, the string `signed` *is added to the IDF file name shown in the image title.*

3. Transfer the code signing certificate and root certificate to the devices to `/setup/cacerts`. To do so, use the Scout feature **Files configured for transfer**. For further information, see https://author-p185304-e1959167.adobeaemcloud.com/en-us/unicon-elux-scout/2603-ltsr-cu1/admin_guides/scout/advanced_options/advanced_files.html in the **Scout** guide.
4. Make sure the signature verification for the devices is enabled. Signature verification is an option of the device configuration in **Device configuration > Firmware > Security**. For further information, see Firmware security through signature.

> **Note:**

> Any changes to the signed IDF in ELIAS will delete the signature. After having changed and saved a signed IDF, repeat the signing procedure.

> The result of the signature verification is documented in the update log on the device. After the update is performed, the update log file is sent to the Scout Server and can be viewed for the selected device in the **Properties** window by double-clicking the **Update status** field.

Exporting an IDF

This feature helps you export the currently opened IDF and the herein defined software packages to another local directory, to a network file server or to a USB storage device.

1. Click **Image > Export...**.
2. Click **Browse** and select the required target directory.
3. If required, change the default file name **recovery.idf**.
4. Click **Save**.
5. To delete packages from the target directory that are not defined in the IDF, select the relevant option.
6. Confirm with **OK**..

The IDF is exported as defined.

Printing an IDF

For documentation purposes, it might be useful to print a list of the software packages of a particular IDF.

1. To choose layout and printer options, click **Image > Print Options...**
2. If required, click **Image > Print Preview**.
3. Click **Image > Print...**.

USB recovery stick

> **Note:**

> This feature is provided for eLux RP 5 -- UEFI devices included -- and eLux RP 6. Alternatively, for all eLux versions, we provide a predefined recovery image on our myelux.com in the **Recovery** guide.

> ELIAS helps you create a USB recovery stick that you can use to install individual eLux® images on your devices.

In addition, to make it distributable, you can use the generated USB recovery stick to create a disk image in `.zip` format.

The installation of eLux with an individual USB recovery stick is carried out in the same way as a USB recovery installation with a USB image from our portal. The installation process is started right after the system prompts you for confirmation of the eLux RP recovery. For further information, see Performing a USB recovery in the **Recovery** short guide..

Creating an individual USB recovery stick

REQUIRES
• PC running Windows 10 or later versions with the ELIAS tool installed
• Administrator rights for ELIAS

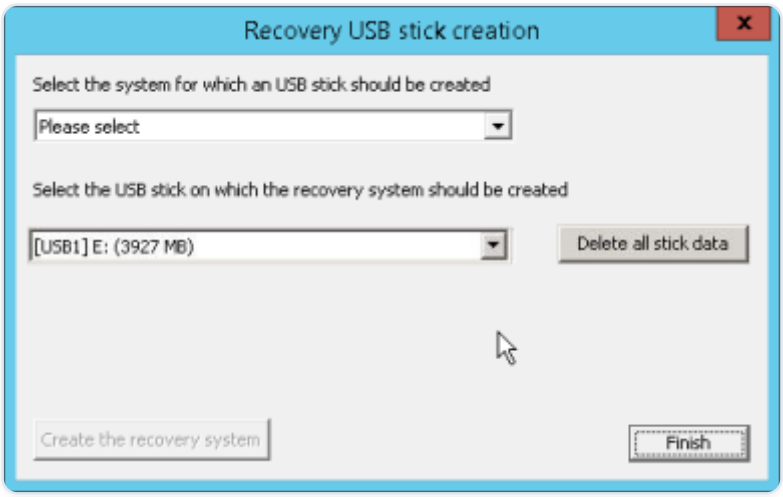
REQUIRES
• Write access to a local USB port
• Connected USB stick with 2 GB memory space

Creating a USB recovery stick

1. In ELIAS, open the relevant IDF (`.idf`).
2. Click **Image > Create USB stick ...**
3. In the **Recovery USB stick creation** dialog, in the first list field, select the operating system of the relevant devices for which you want to create the recovery flash drive:

eLux® RP 6 X64 Recovery	eLux RP 6 (64 Bit) with and without UEFI
eLux RP 5 Recovery	eLux RP 5
eLux RP 5 UEFI Recovery	eLux RP 5 with UEFI

   Note that, depending on the selected system, the image must contain the relevant eLux packages for the 64 bit kernel and UEFI.
4. In the second list field, select the appropriate USB drive. If more than one USB storage device is connected, select the relevant one.
5. To delete the contents on the selected USB stick or reset an existing recovery system, click the **Delete all stick data** button.
6. Click **Create recovery system**.



Creating a USB recovery stick involves two important steps:

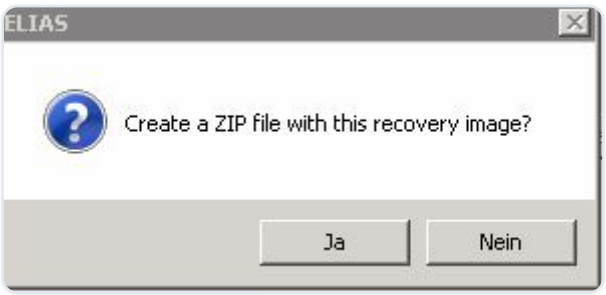
- Creating the recovery system on the USB storage device to turn the USB stick bootable
- Transferring the individual IDF containing the relevant eLux packages to the container directory on the USB stick.

A progress bar at the bottom indicates the status of the copying process. As soon as the USB stick is fully written, a message is displayed.

After having created your USB recovery stick, you can optionally create a disk image in terms of a `.zip` file to make it distributable.

Creating a disk image of the USB recovery stick

1. Immediately after the creation of the USB recovery stick, click **Yes**.



2. Select a storage location and name for the `.zip` file.

ELIAS creates a `.zip` file containing an image of the USB recovery stick as a DD file. In addition, the `.zip` file includes the application `StickWizz.exe` and a text file with a description .

The **StickWizz** tool helps you create a bootable USB stick out of the image. For further information, see Preparing a recovery image for USB in the **Recovery** guide.

> **Note:**

> Hint: Alternatively create an `stw` file from your image on a local or network drive. The **StickWizz** tool then helps you create a bootable USB stick out of the `stw` file. For further information, see (/en-us/unicon-elux-scout/2603-ltsr-cu1/admin_guides/elias/elias_usbrecovery/stickwizz.html) in the **Recovery** guide.

Creating an individual StickWizz file as an installation base

You might want to create a disk image of your image as an installation base on a local or network drive. **StickWizz** helps you create a bootable USB stick out of the `.stw` file.

REQUIRES
• PC running Windows 10 or later versions with the ELIAS tool installed
• Administrator rights for ELIAS

1. In ELIAS, open the relevant image (`.idf)`.
2. Click **Image > Create StickWizz image...**
3. In the **Create recovery image for StickWizz** dialog, in the first list-field, select the operating system of the relevant devices for which you want to create the recovery flash drive:

eLux® RP 6 X64 Recovery	eLux RP 6 (64 Bit) with and without UEFI
eLux RP 5 Recovery	eLux RP 5
eLux RP 5 UEFI Recovery	eLux RP 5 with UEFI

Note that, depending on the selected system, the image must contain the relevant eLux packages for the 64 bit kernel and UEFI.

4. In the second list-field, choose a name for your `.stw` file. To change the storage location, click **Browse**.

   By default, the `.stw` file is saved to the container selected in ELIAS.

5. Click **Create recovery image**.

ELIAS creates a recovery image as a disk image in a DD -like format.

By using **StickWizz**, you can create a bootable USB stick out of the file. For further information, see Preparing a recovery image for USB in the **Recovery** guide.

Writing an image to USB (with StickWizz)

REQUIRES
• Image file as a `.stw` or `.dd` file on a PC running Windows 10 or later versions
• **StickWizz** application on your PC (see Preparing a recovery image for USB)
• USB stick on local port with 4 GB memory space, write access

1. Plug the USB stick into the PC.

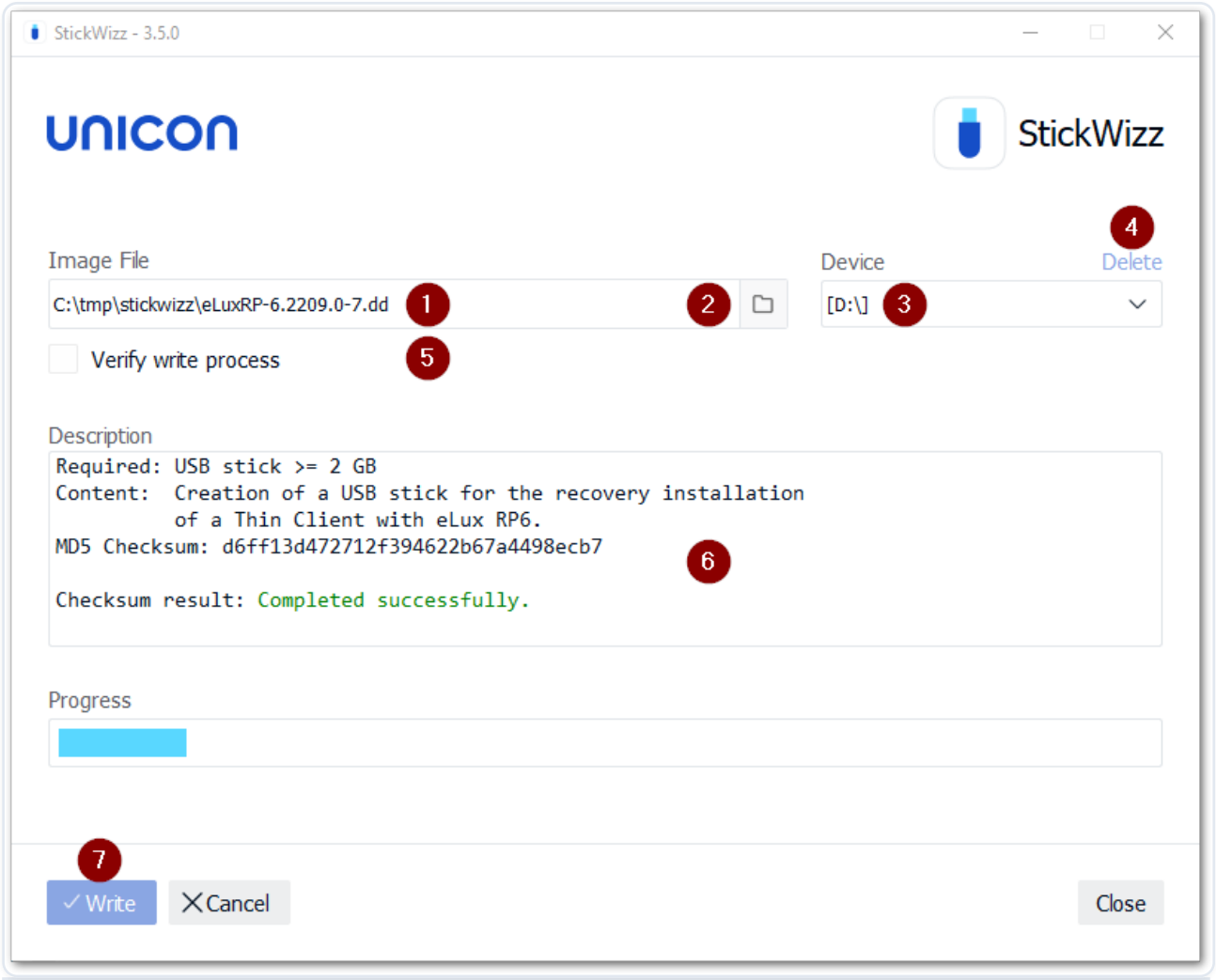
   Note that the Windows Explorer must not be open on the USB drive.

2. Run the `StickWizz.exe` program.

   The language depends on the Windows display language selected. The checksum is calculated and compared to the one provided.

3. Check whether the appropriate source under **Image file** (1) and the appropriate destination under **Device** (3) are selected.
4. Click **Write** (7).

The image is written to the USB stick. A message confirms the successful completion.



1	After StickWizz is started, it shows a .stw or .dd file located in the same directory.
2	Browse to select your image file from another directory, if required.
3	Device /drive, on which a USB stick is connected.Choose a different drive if required.
4	Delete all data from the selected device (before writing the image)

5	During the write process, each written data packet will be compared to the source.
6	As soon as StickWizz opens with the specified file, the checksum is calculated and displayed.
7	Start writing

Security menu

OPTION	DESCRIPTION
Checking for signature automatically during import	Signatures are automatically verified during import of the packages.
Check all packages in the container	Verifies the signatures of all packages in the container
Checking selected package	Verifies the signatures of a selected package and its components in the container or image window
Signature and certificate information...	Displays the security status of a package selected in a container or image list and of its components
Signing IDF...	Allows to sign image definition files (IDF) by using private certificate-based signatures (pem/p12)
	Requires the configuration of an appropriate certificate in the ELIAS certificate management
Manage certificates...	Enables the management of certificates in the "trusted list" of ELIAS
Settings...	An Online Certificate Status Protocol (OCSP) server is used to verify the status of a certificate ([valid], [revoked], [unknown]).
	Proxy server (optional): If you use a proxy server to connect to the Internet, enter the

OPTION	DESCRIPTION
	proxy server IP address (or name) and the port.
	OCSP server: In general, the OCSP server is automatically read from the certificate. If this information is not available, you can enter a default OCSP server in URL format and the port manually. Format: `http://< OCSP server URL>`
	Default: `http://ocsp.openvalidation.org` and port 80
	OCSP check when packaging: Before signature check, the validity of the certificate is automatically verified.

Scout Board

Scout Board is a modern and intuitive management solution for eLux® end points. The web-based interface allows you convenient secure access to your Scout infrastructure. Many features for managing eLux-devices as well as Windows-based devices that have installed the Scout Agent for Windows are already known from the Scout Console.

You continue to use your existing SQL Server database for Scout. Your entire device infrastructure with device configuration, defined applications and access rights structure can now be used via the new Scout Board interface.

> **Note:**
>
> **Scout Board** supports Microsoft SQL Server and SQL Server Express, but not SQL LocalDB.

> As part of the Scout Enterprise Management Suite® installation, the Scout Board is published for the first time with the software release of Scout 15 2209.

The functionality realized so far maps common use cases:

- A dashboard, which gives the administrator a quick overview of some important KPI's
- Access to devices via OU (organization unit) structure with organization units
- Access All devices
- Apply filters to device lists, sort the data, organize data columns
- Dynamic Device Groups (View only)
- Create custom views to manage dynamic device lists
- Search for navigation elements, OUs and devices
- View details for selected devices, OUs and schedules
- Schedule and execute commands on OUs and devices
- Mirror eLux sessions
- Rename, move, delete devices
- Download device diagnostics files
- View and edit device configurations and advanced device configurations for devices and OUs
- Manage settings and preferences
- Schedule device and OU actions
- View and edit device configurations and advanced device configurations for devices and OUs
- View and edit base and base advanced device configurations
- Manage applications
- Manage administrators, permissions and object rights
- Manage the command history
- Compare device configurations
- FollowMe Desktop
- Database cleanup
- Dynamic device configuration (from Scout version 16 2601)
- A REST-API for automating use-cases or integrating eLux management features in custom applications

> **Note:**

> The REST-API and its documentation is an integral part of Scout Board. In order to access the REST-API documentation, open your web browser and navigate to `https://host:port/html/api-docs.html`.

> This already provides administrators with a simple and intuitive interface to monitor OUs and devices and provide support to end users. The functionality will be continuously extended and expanded with the following versions.

> **Note:**

> Logon via Active Directory is supported.

Overview

Scout Board is a modern and intuitive management solution for eLux® end points. The web-based interface allows you convenient secure access to your Scout infrastructure.

You continue to use your existing SQL Server database for Scout. Your entire device infrastructure with device configuration, defined applications and access rights structure can now be used via the new Scout Board interface.

> **Note:**

> **Scout Board** supports Microsoft SQL Server and SQL Server Express, but not SQL LocalDB.

> As part of the Scout Enterprise Management Suite® installation, the Scout Board is published for the first time with the software release of Scout 15 2209.

The functionality realized so far maps common use cases:

- A dashboard, which gives the administrator a quick overview of some important KPI's
- Access to devices via OU (organization unit) structure with organization units
- Access All devices
- Apply filters to device lists, sort the data, organize data columns

- Dynamic Device Groups (View only)
- Create custom views to manage dynamic device lists
- Search for navigation elements, OUs and devices
- View details for selected devices, OUs and schedules
- Schedule and execute commands on OUs and devices
- Mirror eLux sessions
- Rename, move, delete devices
- Download device diagnostics files
- View and edit device configurations and advanced device configurations for devices and OUs
- Manage settings and preferences
- Schedule device and OU actions
- View and edit device configurations and advanced device configurations for devices and OUs
- View and edit base and base advanced device configurations
- Manage applications
- Manage administrators, permissions and object rights
- Manage the command history
- Compare device configurations
- FollowMe Desktop
- Database cleanup
- A REST-API for automating use-cases or integrating eLux management features in custom applications

Note:

The REST-API and its documentation is an integral part of Scout Board. In order to access the REST-API documentation, open your web browser and navigate to `https://host:port/html/api-docs.html`

This already provides administrators with a simple and intuitive interface to monitor OUs and devices and provide support to end users. The functionality will be continuously extended and expanded with the following versions.

> **Note:**
>
> Logon via Active Directory is supported.

Installing Scout Board

> **Note:**
>
> Scout Board is installed as a component of the Scout Enterprise Management Suite.

1. Download the latest version of the Scout Enterprise Management Suite and start the installation process. For further information, see Installing Scout Enterprise Management Suite.
2. Fresh installation: By default, the Scout Board is part of the installation. Upgrading a previous version without Scout Board: Activate the Scout Board component.

3. Then configure your Scout Board settings.

	INFORMATION
1	Number of the port on which the Scout Board service should run
2	Computer name (FQDN) of the machine on which the database layer is to run
3	With **HTTPS**, a secure connection to the interface is used.
	Initially, a self-signed certificate is created. To configure an individual SSL certificate instead, see below.

1. Follow the further instructions.

Configuring an individual HTTPS certificate for Scout Board

Self-signed certificates are not accepted as secure by the browser. To provide the browser with a valid SSL certificate for Scout Board perform the following steps:

1. Rename your certificate file to `certificate.crt` and the private key file to `private.key`
2. Copy both files into the Scout Board program directory. By default, this is `C:\Program Files\Unicon\Scout\Scout Board`
3. Still in the Scout Board program directory, edit the `.env` file and change the entry `CERTIFICATE=SELFSIGN` to `CERTIFICATE=SIGN`
4. Restart the Scout Board service.

For further information and detailed certificate configuration, see HTTPS Certificate Setup.

Configuring authentication options for Scout Board

During installation, you can choose which authentication method should be used for Scout Board.

The following authentication methods are supported:

- AD (Active Directory)
- Entra ID (SAML or OIDC)
- OKTA (SAML or OIDC)

After an authentication method has been configured during the installation, you can change or reconfigure it in the `.env` file. As a reference and an example, the template file `.env.example` can be used.

AUTHENTICATION METHOD	PARAMETER	VALUE
AD	`AUTH_TYPE`	`AD`
OIDC	`AUTH_TYPE`	`OKTA_OIDC` or `ENTRAID_OIDC`
	`OIDC_ISSUER`	`your-issuer-url`
	`OIDC_CLIENT_ID`	`your-client-id`
	`OIDC_CLIENT_SECRET`	`your-client-secret`

AUTHENTICATION METHOD	PARAMETER	VALUE
SAML	`AUTH_TYPE`	`OKTA_SAML` or `ENTRAID_SAML`
	`SAML_SSO_ENTRY`	`your-sso-entry-url`
	`SAML_LOGOUT_URL`	`your-logout-url`
	`SAML_SERVICE_PROVIDER_ISSUER`	`https://scoutboard.com/47d899fb-3002-4692-b741-8068b8f7fbb4` Only replace, if you are not using the Scout Board default issuer
	`SAML_CERT_PATH`	`your-certificate-path`
	`SAML_PRIVATE_KEY_PATH`	`your-provate-key-path`

Configuring AD domains for Scout Board

The AD domain for the logon process is being used from the provided data during the installation process.

In the `.env` file, the configuration is available. If the domain changes or additional domains need to be added, these can be added manually.

The defined domains will be selectable in the Scout Board Logon dialog.

Example:

DOMAIN OPTION	DOMAIN CONFIGURATION
One domain	`DOMAINS=["production.mydomain.com"]`
Multiple domains	`DOMAINS=["production.mydomain.com", "test.mydomain.com"]`

If you are using LDAPS,in the `.env` file, you can configure them: `LDAPS_SERVERS=["ldaps://my.ldaps.server", "ldaps://my.other.ldaps.server"]`

> **Note:**

> After modifying the .env file, the windows service **"Scout Board"** needs to be restarted.

Scout Board HTTPS Certificate Setup

Scout Board HTTPS Certificate Setup

> **Note:**

> This certificate setup guide applies to **Scout Board version 15.2603 and later**

> This guide explains how to configure the HTTPS certificate that Scout Board presents to administrators in the web browser.

Three options at a glance

OPTION	WHEN TO USE	BROWSER WARNING?
Self-signed (default)	Lab, proof of concept, internal testing	Yes
CA-signed certificate from a public CA (DigiCert, GlobalSign, Let's Encrypt, ...)	Production, internet-facing	No
CA-signed certificate from your internal/corporate CA	Production, internal-only	No, on machines that trust your corporate root

Option A: Self-signed certificate (default)

If you do nothing, Scout Board generates a new self-signed certificate the first time the service starts. Browsers will warn that the connection is not trusted, but Scout Board itself works.

This is suitable for evaluation only. Use a CA-signed certificate for production.

Option B: CA-signed certificate, selected during installation

The Scout Board installer in 26.03 introduces a **Certificate** dropdown on the Scout Board settings page. The dropdown lists every certificate in the Windows **LocalMachine --> Personal** store that has an accessible private key. When you pick a certificate, the installer exports it together with any

intermediate (and root) certificates it can find in the Windows certificate stores, and writes the result to `certificate.crt`.

STEPS

1. On the machine where you will install Scout Board, import your CA-signed certificate (with private key) into **LocalMachine --> Personal**. For PFX/PKCS#12 imports, use the Windows Certificate Manager (`certlm.msc`); the wizard places intermediate CA certificates into **Intermediate Certification Authorities** and any root into **Trusted Root Certification Authorities**. Give your leaf cert a meaningful Friendly Name so you can recognise it in the dropdown.
2. Run the Scout bundle installer.
3. On the Scout Board settings page, open the **Certificate** dropdown and pick your imported certificate. If you imported it after the wizard was already open, click the refresh button next to the dropdown.
4. Continue the installation as usual.

> **Note:**
>
> If after install you see a browser certificate warning or you are redirected to a license-switch page when signing in, the chain in `certificate.crt` is most likely incomplete. The next section tells you how to verify and fix.

IF THE CERTIFICATE CHAIN IS INCOMPLETE AFTER INSTALL

The installer can only export intermediate certificates that exist in the Windows certificate store at the moment you run it. If you imported a leaf certificate without its accompanying intermediate(s) - a common situation when the source PFX did not bundle them - the installer will write only what it found, which may be just the leaf.

To diagnose:

1. Run the count check from Verifying your setup below. A count of 1 with a CA-signed certificate means the chain is incomplete.
2. Check `backend.log` for `certificate.crt contains only a leaf certificate` or `unable to verify the first certificate`.

To fix:

1. Stop the **Scout Board** Windows service.
2. Append the missing intermediate certificate(s) (and, for an internal/private CA, the root) to `<InstallDir>\scoutboard\certificate.crt`. The next section shows how to build the chain.
3. Start the **Scout Board** Windows service.

On the next install or upgrade, the installer will detect that you have built a complete chain and will leave your `certificate.crt` alone, so the manual fix survives.

Option C: CA-signed certificate, configured after installation

Use this when you cannot pick the certificate at install time, when the certificate is being renewed, or when you upgrade Scout Board from a release older than 26.03.

1. Stop the **Scout Board** Windows service.
2. Place your full chain file at `<InstallDir>\scoutboard\certificate.crt`. The default install directory is `C:\Program Files\Unicon\Scout\Scout Board\`.
3. Place your unencrypted `PKCS#8` private key at `<InstallDir>\scoutboard\private.key`.
4. Open `<InstallDir>\scoutboard\.env` in a text editor and set `CERTIFICATE=CUSTOM`. The legacy value `CERTIFICATE=SIGN` is still accepted as an alias for backward compatibility.
5. Start the **Scout Board** Windows service.

Building the certificate chain file

`certificate.crt` is a single text file in PEM format. It can hold one or many certificates back-to-back. The order matters: leaf first, then each intermediate moving up to the root.

A correctly built file looks like this:

```
-----BEGIN CERTIFICATE-----

... your leaf certificate (the one issued for your Scout Board hostname)

-----END CERTIFICATE-----

-----BEGIN CERTIFICATE-----

... intermediate CA certificate (the one that signed your leaf)

-----END CERTIFICATE-----
```

```
-----BEGIN CERTIFICATE-----

... next intermediate, if your CA uses a multi-level chain

-----END CERTIFICATE-----

-----BEGIN CERTIFICATE-----

... root CA certificate — only required if you use an internal/private CA

-----END CERTIFICATE-----
```

What to include depends on your CA:

1. **Public CA** (DigiCert, GlobalSign, Let's Encrypt, etc.): leaf + intermediate(s). The root is optional - Scout Board already trusts public roots out of the box.
2. **Internal/corporate CA**: leaf + intermediate(s) + your private root. Without the root, Scout Board cannot validate the chain because it does not consult the Windows trust store by default.

Most public CAs publish a downloadable bundle file (often called "intermediate bundle" or "CA bundle") that you can simply append to your leaf certificate using any text editor or:

```
# Build certificate.crt = leaf + intermediate bundle

Get-Content leaf.crt, intermediate-bundle.crt | Set-Content
certificate.crt
```

Verifying your setup

After placing the files and (re)starting the Scout Board service, run these checks.

HOW MANY CERTIFICATES DOES `CERTIFICATE.CRT` CONTAIN?

```
`Select-String -Path "C:\Program Files\Unicon\Scout\Scout
Board\certificate.crt" -Pattern "BEGIN CERTIFICATE" | Measure-Object`
```

If the count is `1` and you are using a CA-signed certificate, the chain is incomplete. Append your intermediates and restart the service.

DOES WINDOWS AGREE THE CHAIN IS VALID?

```
certutil -verify -urlfetch "C:\Program Files\Unicon\Scout\Scout
Board\certificate.crt"
```

DOES THE SCOUT BOARD LOG SHOW ANY CHAIN WARNINGS?

Open `backend.log` in `C:\Users\Public\Documents\Unicon\Scout\ScoutBoard\` and look for either of these lines:

```
certificate.crt contains only a leaf certificate without the intermediate
CA chain

License validation error ... unable to verify the first certificate
```

Either of these means the chain is incomplete. The companion file `backend_error.log` in the same folder records error-level entries only - chain warnings appear in `backend.log`.

```
Select-String -Path "C:
\Users\Public\Documents\Unicon\Scout\ScoutBoard\backend.log" -Pattern
"leaf certificate|unable to verify"
```

OPEN SCOUT BOARD IN A BROWSER

The browser should display the padlock without warnings, and you should be able to log in without being redirected to a license-switch page.

Troubleshooting

SYMPTOM	MOST LIKELY CAUSE	FIX
Browser warns "Not Secure" / "Certificate not trusted"	You are using the default self-signed certificate, or your `certificate.crt` is missing intermediates	Switch to a CA-signed certificate, or append the missing intermediates and restart the service
After login you are redirected to a license-switch page	`certificate.crt` does not contain the full chain (typically the installer could not find your intermediates in the Windows store)	Append intermediates to `certificate.crt` using Building the certificate chain file above and restart the service

SYMPTOM	MOST LIKELY CAUSE	FIX
Internal/corporate CA: redirect persists even with intermediates included	Your private root is not included in `certificate.crt`	Append your corporate root certificate to `certificate.crt` and restart the service
Browser warning resolves but Scout Board still does not load	The intermediate bundle includes the wrong intermediate, or its order is wrong	Verify with `certutil -verify -urlfetch`; the leaf must be first, intermediates above it in issuance order

Permissions, ownership and restart

1. Both `certificate.crt` and `private.key` must be readable by the account running the **Scout Board** Windows service. By default this account is `LocalSystem`, which has access.
2. Replacing either file requires a service restart - Scout Board reads them once at startup.
3. The installer cleans up both files on a complete uninstall, but preserves them across upgrades. Your custom certificate survives a Scout Board upgrade.
4. If you have manually edited `certificate.crt` to append intermediates, the installer detects this on upgrade and leaves the file alone - your manual fix survives.

Frequently asked questions

CAN I USE A WILDCARD CERTIFICATE?

Yes. Wildcard certificates work the same way — the leaf is the wildcard certificate, and the intermediate(s) come from the CA that issued it.

CAN I USE THE SAME CERTIFICATE AS ANOTHER SERVICE ON THIS MACHINE (IIS, CITRIX GATEWAY, ETC.)?

Yes, provided the certificate covers the hostname customers use to reach Scout Board. Import it into `LocalMachine\Personal` once and it will appear in the installer dropdown.

WHAT HAPPENS AT CERTIFICATE RENEWAL TIME?

Stop the Scout Board service, replace `certificate.crt` and private.key with the renewed file (full chain), then start the service. No installer rerun is needed.

WHAT HAPPENS DURING A SCOUT BOARD UPGRADE?

Your existing `certificate.crt` and `private.key? are preserved across upgrades. You do not need to re-import or rebuild the chain file. If you previously appended intermediates manually, those are preserved too.

WHERE ARE THE SCOUT BOARD LOG FILES?

`C:\Users\Public\Documents\Unicon\Scout\ScoutBoard\backend.log` (info-level entries and above, including chain warnings) and `backend_error.log` (error-level only) in the same folder. There is also a Windows service wrapper log at `%APPDATA%\ScoutBoard\`, but for certificate issues you want `backend.log`.

WHY DOES SCOUT BOARD NEED THE INTERMEDIATES IN THE FILE RATHER THAN RELYING ON THE WINDOWS CERTIFICATE STORE?

The Scout Board service uses its own bundled trust list and does not read intermediate certificates from the Windows store at runtime. This keeps Scout Board behaviour consistent across servers regardless of what is installed in the local Windows trust store. The trade-off is that you must ensure `certificate.crt` is a complete chain.

Interface

The Scout Board main window after logon shows the **Dashboard** with status information and an overview on different aspects of the managed devices.

The language of the entire application depends on the language setting in the browser.

	INFORMATION
1	Navigation Panel
2	The section selected on the Navigation Panel is displayed.
3	Global search for data and navigation elements
4	Help, notifications, user profile
5	The Devices section provides three alternative views.

	INFORMATION
6	For the selected OU, the contained devices are displayed on the right.
7	For the selected device, the details are displayed on the right.
	Use the **Navigation Panel** on the left to choose the section you want to work with. Next to **Dashboard** and **Devices**, the **Navigation Panel** offers access to additional functions such as reports, advanced configuration and administrator management.

Note:

Close the **Navigation Panel** with to save space. Then only the section icons will be displayed.

Devices

The Devices section offers three different views to show your devices:

- **All devices**: All devices are displayed in a list regardless of their organizational unit. The list can be sorted and filtered as desired.
- **OU structure**: The hierarchical OU structure is displayed as a tree view. All organizational units (in the following OUs) can be expanded and displayed.
- **Dynamic Device Groups**: Defined Dynamic Device Groups are displayed with the devices currently matching.

For a selected OU or Dynamic Device Group, on the right under Devices, the assigned devices and schedules are shown. To the right, the details of an OU, a Dynamic Device Group or a device may be displayed.

Dashboard

The Scout Board Dashboard provides you with various widgets showing the current status of your infrastructure at a glance.

- Number of devices
  - total
  - online / offline / firmware update in progress
  - not seen in a month/ year
  - internally connected devices (via corporate network), externally connected devices (via VPN), not yet connected devices with reserved device profiles
  - with pre-defined image (compliant)
- License information (from Scout 15 2302)
  - Selected Subscription type
  - Number of used and available Scout Management licenses
  - License information for

  For further information, see Showing license information.
- Distribution of images

A firmware image is called compliant if it is an image that has been globally predefined in the **Advanced Device Configuration**. For further information, see Predefined IDFs and containers in the **Scout** guide.

Devices / All devices

In this view, all devices of your infrastructure are displayed as a list.

Use sort and filter functions to display relevant devices, see Filtering devices.

If required, adjust the number of rows (devices) displayed per page.

Devices / OU structure

The OU structure is organized hierarchically and may reflect the structure of your organization. OUs can contain devices at each level, but also other OUs. An OU can represent a site with its devices or group devices with the same details.

The advantage of having a hierarchical structure is that properties such as the device configuration and applications can be inherited from top to bottom.Actions can be applied to all devices in an

OU, optionally including all devices in subordinate OUs, in one step. This applies, for example, to a command like **Restart**.

OU structure view

1	Click to expand an OU
	An expanded OU is indicated by
2	For each OU, the number of included devices, defined applications and schedules are displayed.
3	For a selected OU, the devices contained are displayed on the right in the [device list].

The device list on the right shows the devices located in the OU selected on the left. Devices located in subordinate OUs are not displayed.

The device list can be edited, sorted and filtered in the same way as in the **All devices** view, see Display devices. The **Actions** context-menu for the devices is also available, for example to issue commands or mirror a device.

Additional functions of a device list within the OU structure

The following functions are not available in the **All Devices** view.

1	Show schedules
	For each OU, you may schedule commands which are displayed as schedules.
2	Manually add individual devices to an OU
3	Create a link for a selected device
	A device can be uniquely identified and shared via a device link.

	If no device is selected, the link is created for the OU.

Devices / Dynamic Device Groups

Dynamic Device Groups you already have in your database can be displayed and used in the Scout Board. For Dynamic Device Groups, the same Scout commands are provided as for OUs.

1	Show existing Dynamic Device Groups
2	Open **Actions** context-menu for a Dynamic Device Group
3	Select a Scout command

For the selected Dynamic Device Group, the currently matching devices are displayed on the right. On the far right, the Dynamic Device Group's details are displayed.

A Dynamic Device Group is only shown if the administrator is allowed to see at least one of the included devices. As soon as an administrator has access to only a part of the devices, he cannot issue commands on this Dynamic Device Group.

Searching for devices and other objects

The Scout Board allows you to search for devices and OUs from any view. In addition, you can search for navigation elements like the **All devices** list and just jump to selected item.

Showing license information

- On the Scout Board title bar, click and then **License information**.

Subscription type

Subscription licenses can be managed in the following ways:

LICENSE TYPE	DESCRIPTION
Enterprise (default)	The Subscription licenses of all devices are managed in one shared pool. The Subscription is equally applied to all devices from this pool. An end date for the Subscription validity is calculated on this basis for all devices and displayed below the type.
OU	As with the Enterprise type, Subscription pools are formed for all top-level OUs. Within an OU, the Subscription is equally applied to all devices.
Managed Service Providers	For use by MSPs, the username of the myelux account registered on our reporting portal and the last logon are displayed. The maximum number of managed devices for the current month is displayed and can be recalculated at any time. In addition, authentication to the reporting portal can be performed directly from here and then license reports can be viewed.

Scout Management licenses

- Total number of Scout Management licenses
- Number of Scout Management licenses currently used by devices
- Number of available (unused) Scout Management licenses that will be automatically assigned to new devices during onboarding

Application licenses

Application licenses are required to use licensed third-party applications on a device. Assigning these licenses to devices can be done manually or automatically.

For more information on license management, see also our whitepaper **Licensing and Subscription Scout 15 and eLux® RP 6** and the **License management** short guide.

Devices and OUs

In the Devices section of the Scout Board, you can map the structures of your organization and manage the devices within OUs (organizational units). The OU structure makes it easier for you to both configure and manage your devices.

- The **device configuration** depends on the OU.

Different device configurations can be represented well by OUs. By default, the device configuration is inherited from the base - at the top of the OU tree - down through all existing OU levels to the devices. At each level, you have the option to break the inheritance. Even a single device can have an independent device configuration. For further information, see Concept in the Scout guide.

- **Applications** are defined for OUs.

Application definitions are inherited from the top down, just like device configuration. At each level, you can decide whether you want to inherit applications from above -- or prefer to define applications dedicated to that level.

- Manage **schedules** and perform **commands**

Scout commands such as a device restart or a firmware update can be sent to all devices of an OU simultaneously. If commands are not to be executed immediately, they can be scheduled for a later time. The commands are then managed as schedules with an OU. You can also use schedules to define periodic execution of certain commands.

Alternatively, you can assign commands, schedules and other functions to selected devices in the **All devices** or **Dynamic Device Groups** views.

So, within the OU structure, inheritance applies by default: applications -- or more precisely application definitions -- are inherited by subordinate OUs. Device configurations are inherited down to the device level. When a new device is added to an OU, it automatically receives the application definitions and configuration of this OU.

For further information, see https://author-p185304-e1959167.adobeaemcloud.com/en-us/unicon-elux-scout/2603-ltsr-cu1/admin_guides/scout/setup/config_concept.htmlConcept in the Scout guide.

Display devices

In Scout Board, the devices are displayed in the form of a list in all views. You can sort, filter and search a device list by any field. If possible, limit the scope of the displayed devices already in the first step by selecting the view:

All devices

- In the **Navigation Panel**, choose **Devices** > **All devices**.

All devices of your Scout infrastructure are displayed as a list.

DEVICES OF A SPECIFIC OU

1. In the **Navigation Panel**, choose **Devices** > **OU structure**.
2. Expand higher level OUs until you can see the relevant OU. To do this, click in front of an OU name.
3. Click the OU.

All devices of the selected OU are displayed. The device list of an OU has two tabs: One for the devices themselves and next to it all schedules of this OU.

DYNAMIC DEVICE GROUPS

1. In the **Navigation Panel**, choose **Devices** > **Dynamic Device Groups**.
2. Click one of the defined Dynamic Device Groups.

All devices matching the filter criteria of the Dynamic Device Group are displayed.

Device list functions

The same basic functions for a device list are available in all views.

1. **Sort** by a field.
2. **Filter** by one or more fields.
3. : Define which of all available fields will be displayed as columns and choose the sequence.
4. Open the **Actions** context-menu of a device.
5. Show **details** of a device (or OU or Dynamic Device Group) on the right.
6. Selection column: Select multiple devices to apply an action on them. (from Scout 15 2302)

Especially when many devices are displayed, sorting and filter function are important tools.

List settings, filters and sorting are saved in the browser's local storage so that they are associated with a specific user and the browser used.

To re-use your settings beyond this and easily switch between settings , save them in individual views. For further information, see Individual device list views. (from Scout 15 2304)

Changing List settings

Specify which fields are displayed in the device list and in what order.

1. On the right, above the device list, click .

   All available fields are displayed in a list-field with scroll bar. The fields currently shown in the device list can be found at the top and are selected.
2. Use the scroll bar or search to view more fields in the list-field
3. To select specific fields for display, check the fields. To select all fields, click the icon in front of **Select all**.
4. To change the sequence, point to the left of a field name to . Drag the field to the desired position within the list-field.
5. Confirm with **Save**.

	[
1	Open list settings
2	Search for a field
3	Select a field to be displayed as a column
4	Drag a field to a new position (Sequence of columns)

> **Note:**

> If not all configured fields can be shown in the due to space constraints, a horizontal scroll bar will be displayed.

> To use your beyond the current session, save them in a view. For further information, see Individual device list views. (from Scout 15 2304)

Sorting devices

-- applies for device lists in all views --

> **Note:**

> By default, a device list is sorted by the **Name** field.

1. In a , move the mouse pointer on a column header.

   A sorting arrow is displayed to the right of the field name.
2. Click the column header to sort the by this field in the direction specified. By default, sorting is in alphabetical descending order.
3. To sort in reverse order, click the column header again.

Once you have set sorting, the sorting arrow is permanently displayed next to the column header.

To re-use sorting settings and apply them to any device lists, save them in an individual view.

> **Note:**
>
> To undo a sort order, use the **Actions** menu of the column header.

Filtering devices

-- applies for s in all views --

> **Note:**
>
> The filter function allows you to create multiple filter rules, which you can OR or AND.

1. Click the filter .
2. To define a filter rule (1), specify the relevant field (the column to filter by), an operator, and the value the field should have.
3. Click **Add rule** (2).

   The rule is added to the section below (4).
4. If you want to limit (AND) or extend (OR) the number of devices by a second filter rule, define another filter rule (1) and apply it via **Add rule** (2).

5. Make sure that the correct link type (OR/AND) is set (3).

   The link type applies to the linking of all filter rules and cannot be set differently for individual links.

6. Only if all the filter rules in the lower section (4) are correct, click **Apply filter**.

Only devices that match the filter rules are displayed. The **Filter** button is displayed in blue.

To re-use filter settings and apply them to any device lists, save them in an individual view.

> **Note:**
>
> To remove a filter, click next to the **Filter** button and select **Remove filter**.

FILTER OPERATORS

The following operators are available:

DATA FIELD TYPE	OPERATORS	EXAMPLE RULE
String (text)	`contains`, `is equal to`, `is not equal to`, `begins with`, `is empty`, `is not empty`	Name `contains` main
Numeric (numbers)	`=`, `>`, `>=`, `<`, `<=`	Flash size (GB) `<=` 4
Date	`on`, `is after`, `is before`	Last contact `is before` 05/02/2023
Logical values (Boolean)	`is`	Encrypted setup partition `is` No

Individual device list views

-- applies for s in all views -- (from Scout 15 2304)

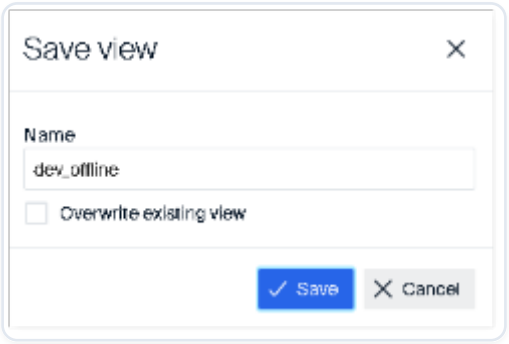
List settings such as your selected fields for display as well as sorting and filtering criteria you have applied to a device list can be saved in views for re-use. You can save and reload as many views as you like. Use descriptive names for them. Views are saved to your user profile in the database.

Once saved, a view is available in all device lists, including the OU structure.

1	Active view
2	Save, load, or edit views
3	Active filter rules
4	List settings

SAVING A VIEW

1. Apply the List settings, sorting or filter criteria as desired.
2. On the right above a device list, click ⋮
3. On the sub-menu, select **Save view**.



4. In the dialog, enter a name for the view and confirm with **Save***.

The new view's name is shown as the active view above the device list.

LOADING A VIEW

1. Click ⋮
2. On the sub-menu, select **Load view**.

   All views you have saved in this browser will be displayed.
3. Choose one of the displayed views.

The selected view is applied to the device list and will be shown as the active view.

EDITING A VIEW

-- Renaming or deleting saved views --

1. Click ⋮
2. On the sub-menu, select **Edit view**.

   All views you have saved in this browser will be displayed.
3. To rename a view, click ✎ to the right of it
4. To delete a view, click 🗑 to the right of it.

CHANGING A VIEW

1. Load the relevant view.
2. Change the List settings, sorting or filter criteria as required.
3. Save the view with the following options in the **Save View** dialog
   - Enter the name of the active view (identical spelling).
   - Select **Overwrite existing view**.

Adding devices

-- applies to device lists in the OU structure --

Within an OU, you may add individual devices manually to the device list.

1. Display the OU structure, see Display devices.
2. Click to select an OU.
3. Above the device list, click **Add Device**.

   For the device name of the new device, the string `DEVICE-` and an appended incremented digit string are used by default.
4. If desired, replace the suggested device name.
5. Specify the MAC address of the new device.

> **Important**

> Use the following format: `00:11:22:33:44:55`

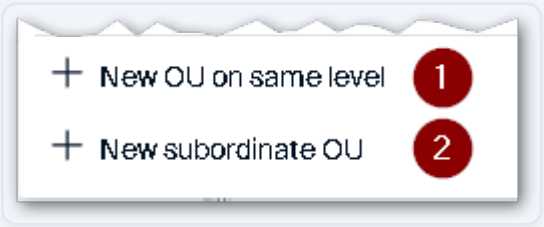
The new device is added to the current OU. It receives a unique Client Identifier.

Adding OUs

-- applies to device lists in the OU structure

You can add new organizational units (OUs) in the OU structure at any level.

1. Display the OU structure, see Display devices.
2. To the right of an OU, click ⋮. From the context-menu, choose one of the following actions:

	
1	The new OU is created "next to" the selected OU.
2	The new OU is created below the selected OU.

3. Enter a name for the new OU and optionally additional information (**Info 1**, **Info 2**, **Info 3**) which will be shown in the device details (if configured).

Actions for devices

-- applies to device lists in all views --

Via a device list you have access to your devices and can, for example, send commands. All available actions can be found in the **Actions** context-menu of a device.

Performing an action for a device

1. To the right of a device, click ⋮.
2. From the context-menu, choose an action.

1	Execute or schedule a [Scout] command on the device
2	Rename, delete or move the device

3	Mirror the device
4	Retrieve diagnostic data

Note:

You may also use drag-and-drop operations to move devices between OUs.

Performing actions on multiple devices

1. Use the selection column on the left to select multiple devices (1).

   Once you have selected multiple devices, a button (2) is displayed above the list.
2. Click (2), and then select a command or action from the context menu.

Note:

To remove the selection of devices, click the minus icon above the selection column.

Actions for OUs

-- applies to the OU structure --

Performing an action for an OU

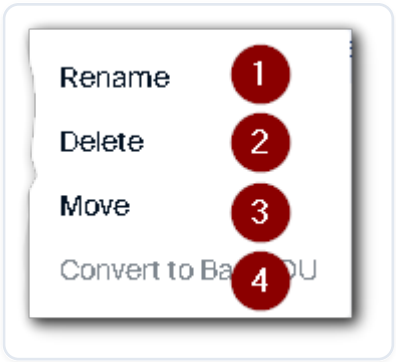
1. Display the OU structure, see Display devices.
2. To the right of an OU, click ⋮
3. From the context-menu, choose an action.

1	Execute or schedule a Scout command for all devices of the OU

2	Rename, delete or move the OU
3	Open the device configuration for this OU
4	Open the advanced device configuration for this OU
5	Add a new OU on the same level
6	Add a new OU below the current level

Renaming, deleting or moving an OU

1. To the right of an OU, click ⋮
2. From the **Actions** context-menu, choose **Edit**, and then one of the provided actions:



| | | |--|--- | | 1 |Overwrite the existing OU name | | 2| Delete the OU | | 3| In the dialog, browse the OU structure and select a new destination OU. | | 4| Make this OU the base OU when appropriate rights provided |

Before any action is performed, you will be asked to confirm.

Dynamic device configuration

(from Scout version 15 2601)

When an eLux endpoint connects to Scout, it will receive its configuration. By default, eLux endpoints are being managed within organization units.

Organization units are managed in a tree structure. While there is a base configuration for organization units and devices, with the tree structure, these device configurations are inherited down the tree structure.

Depending on the requirements, an administrator can break this inheritance and a different device configuration can be defined for an organization unit or for endpoints.

Changes made in an upper level in the tree structure will then not be inherited anymore.

Using a flag called "check independent configurations", the modified device configurations can be pushed down the tree structure, while manually choosing which configurations should be pushed.

When a device configuration has changed, there is a save option "Save and check independent configurations" for this feature to be applied.

For enabling a much more dynamic possibility to add specific configuration to devices, with the Scout Board version 15 2601, a rule and label based configuration approach has been added in addition to the default configuration mechanism.

Labels

A label is a small unit for defining specific configurations for a defined use-case. In a label, one or more sections from the device configuration can be added and the configuration can be defined.

This way, each label has a group of specific configurations, which can be applied to an endpoint.

One or more labels can be assigned to a defined rule.

The sequence of the labels also defines the priority or the configuration for an endpoint.

If the endpoint receives two labels, and both labels contain the same configuration section but with different configurations, the higher priority label will override the other during the merge of the configurations.

Rules

A rule contains a logical expression, with different parameters and values. When a device connects to Scout, and this logical expression is valid, then this rule will be applied.

Each rule will have one or more labels assigned to it.

The device will then get the configurations defined in the assigned labels.

If multiple rules apply for an endpoint, the higher priority rule and the labels assigned to the rule will override the other during the merge of the configurations.

The available parameters are the same like the ou filter rules: User defined OU filters

Priorities

Labels and rules have a priority. The higher priority wins and overrides the lower priority one.

When the configuration for a device is calculated, the different configurations will be merged by following a defined sequence of priority.

The priority is listed from low to high, while the higher will override the same configuration parameters.

- Base configuration --> lowest priority
- Inherited device configuration or Independent device configuration
- Advanced device configuration
- Label (Assigned directly to the OU)
- Label (Assigned directly to the Device)
- Independent configuration (Device)
- Rules --> highest priority

Commands and schedules

Commands can be issued to OUs, Dynamic Device Groups and individual devices.The following commands are available in the Technical Preview:

- Switch on
- Restart
- Switch off
- Send message
- Reset configuration
- Deliver software (from Scout 15 2304)
- Pre-defined command (from Scout 15 2304)

If a command is not executed immediately, it is saved as a schedule. A schedule can be defined for one-time execution or for periodic execution.

Schedules are displayed for each OU in the device list on a separate tab. The number of existing schedules is additionally shown in the for each OU, along with the number of existing devices and applications.

You may create commands or schedules for the following target devices:

- a device of a device list (all three views)
- the devices of an OU in the OU structure, optionally additionally on all devices in subordinate OUs
- the devices of a Dynamic Device Group

To create a command or schedule, open the **Actions** menu for the target devices. To do this, click on ⋮ to the right of the device, OU or Dynamic Device Group.

Creating a command

> **Note:**
>
> Means a Scout command that is executed immediately.

1. For the target devices, open the **Actions** menu, then choose **Commands**, and then the desired command.
2. Leave the setting for **Type** (time) on **Now**.

   *If you do not want to execute a command right away, set a one-time execution time or even a periodic execution, see **Creating a schedule**.*

3. In the **Command** dialog, edit the remaining fields.Common options are the following:

OPTION	DESCRIPTION
Inform users	The relevant users will receive a message before the command is executed.
Include subordinated OUs	The command is additionally applied to all devices in subordinate OUs.

   For further information, see Command options in the Scout guide.

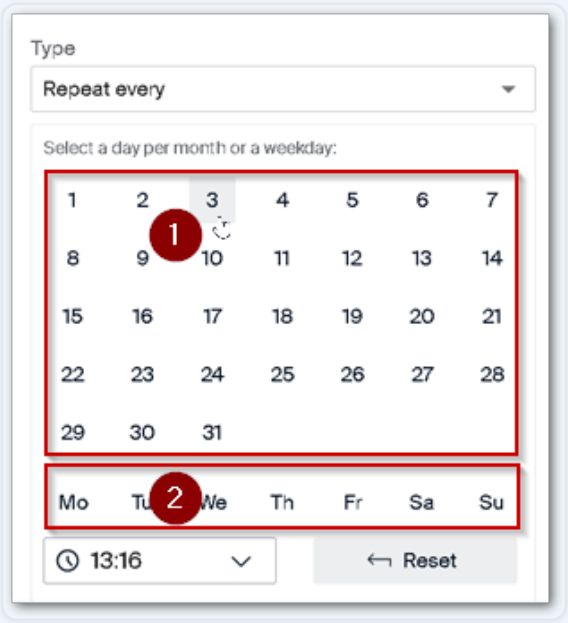
4. Click **Execute**.

The command is executed immediately.

Creating a schedule

> **Note:**
>
> Means a Scout command that is scheduled for later execution.

1. For the target devices, open the **Actions** menu, then choose **Commands**, and then the desired command.
2. Under **Type** (time), select either `Once` or `Repeat every`.
3. If you intend to plan a one-time execution, then select the date and time.
4. If you intend to plan a periodic execution, select one day per month one day of the week below it.



1	Option A: Choose a fixed day per month (for example, every 5th.)
2	Option B: Choose a fixed day per week (for example, every Tuesday)

Specify the time.

5. In the **Command** dialog, edit the remaining fields. For further information, see Command options in the **Scout** guide.
6. Click **Execute**.

A schedule is created for the planned one-time or periodic execution of the command.

Help-desk and remote maintenance

Helpdesk functions such as mirroring a device are available to assist users with problems.

You can refer to the following:

- Mirroring a device
- Retrieving diagnostic data

Mirroring a device

-- from eLux® RP 6 2302 --

Individual devices can be mirrored if they are switched on and the following requirements are met.

REQUIRES
The target device must have installed **eLux RP 6 2209** or later.
A VNC server must be installed on the target device. For this, the VNC Server extension feature package included in the eLux package **XOrg** must be installed.
Mirroring must be enabled and configured for the target device in **Device Configuration** > **Security** > **Mirror settings**.

1. Select a device on a device list (all three views).
2. Click ⋮ to open the device's **Actions** menu and choose **Mirror**.

   > **Note:**
   >
   > Your browser must allow pop-ups.
   >
   > The mirrored device is displayed in a new browser tab.

3. If configured in **Device Configuration** > **Security**, a password is required for mirroring. On the new tab, enter the password.

Depending on the device configuration, users must confirm the mirroring request within a defined time span before a session can be initiated.

For further information, see Configuring mirroring in the Scout guide.

1	New browser tab for the mirror session
2	IP address of the mirrored device
3	Options for the mirror session
4	Permanent system message

Role of the system message

Throughout a mirror session, a system message (4) will be displayed on the screens of the user (mirrored) and administrator (mirroring):

- The message remains permanently in the foreground and indicates that mirroring is active.
- The mirrored user can end the session at any time (**Quit** button).
- The mirrored user can specify that the administrator may only read, which means that he is not allowed to take control of the mirrored device (**Read only** option).

Configuring a remote session

During a mirror session, there are several options available.

- At top right, click **Configure remote session**:

1	Disconnect mirror session. The session window and tab are closed.
2	Send `Ctrl+Alt+Del` into the session, for example to make a logon
3	Show session window in full screen mode. To return to window mode, press `Esc`
4	Fit session window to browser window.Depending on the aspect ratio, the entire screen of the mirrored device may not be displayed.
5	Adjust session window in maximum size to the local situation (default)

The set options for the session window size are saved in the browser for the current user.

For further information, see Mirroring in the Scout guide.

Retrieving diagnostic data

> **Note:**

> We recommend that you temporarily switch on enhanced logging on the device before device diagnostics. This generates and makes available diagnostic files on a larger scale.

1. Select a device on a device list (all three views).
2. Click ⋮ to open the device's **Actions** menu and choose **Request diagnostic files**.

The scripts defined in the system template are executed. All log and configuration files are packed and made available for download in the browser. In the Scout Board, you will receive notifications about the progress and a link to the download.

For further information, see Device diagnostics in the **Scout** guide.

Preferences

On the Navigation Panel, on the bottom section,you will find the **Preferences** entry with preferences for various functions.

Devices

General device and filter settings, especially with regard to onboarding devices

1	OU new devices are assigned to, by default
2	Activates the OU filter for new devices. The OU filter has priority over other methods. The OU filter is configured using subnets and masks or using specific asset information of the devices.
3	Newly added devices are not synchronized with the server's device configuration
4	Allows dynamic assignment of devices via DHCP
5	The Scout Server accepts only devices with registered MAC addresses.

This section corresponds to the **Advanced Options** > **Devices** > **New devices** dialog in the Scout Console.

Logging

On the Scout Server machine, server activities of the Scout Server and the Scout Keep Alive service are logged.

1	Log level **Error**: Only error messages, **Info**: Error messages and additional information (default), **Debug**: Log files with extended content
2	Configure log file rotation / File size
3	Configure log file rotation / Number of files
4	Enable Scout Server logging (`eluxd.log`)
5	Enable Keepalive logging (`keepAlive.log`)

Predefined commands

User-defined commands can be centrally predefined and provided as ready-to-use commands for administrators managing their devices remotely. All active commands in this list can be applied by administrators to individual devices, to OUs, or to Dynamic Device Groups.

A predefined command is shared with all administrators by default, but can be restricted to specific groups. For the executing administrators, the name of the command is displayed instead of the command.

1	Add new entry
2	Delete selected entries
3	Name of a command
4	Command definition (syntax)
5	The command requires system rights

6	The command will be displayed in the list-field for predefined commands.
7	Restrict command to specific administrators or administrator groups
8	Edit the name or syntax of an entry

To create a predefined command, click **Add** (1), and then in the new entry, specify the **Name** (3), **Command** (4) and further options (5, 6, 7).

For further information, see Creating predefined commands in the Scout guide.

Firmware preferences

For **Device Configuration** > **Firmware**, you may predefine values that operational administrators are allowed to use. If you restrict the administrators' object rights so that only predefined entries can be selected, no custom values can be created. For further information, see Protecting firmware configuration in the **Scout** guide.

The following values can be predefined:

IMAGE TYPES	DESCRIPTION AND EXAMPLE
Image files	Names of images which may be used for firmware updates and upgrades
	Example: `recovery_x.idf`
Container for software packages and images	Container paths which may be used to select images
	Example: `elias/UC_RC6_X64`
UEFI files	Names of `.udf` files which may be used to update the devices UEFI firmware with matching binary data.
	Example: `uefi.udf`

To create a predefined firmware value, click **Add**, and then in the new entry, enter the exact name or path. The entry must correspond in spelling to the existing file and path names. Note the following:

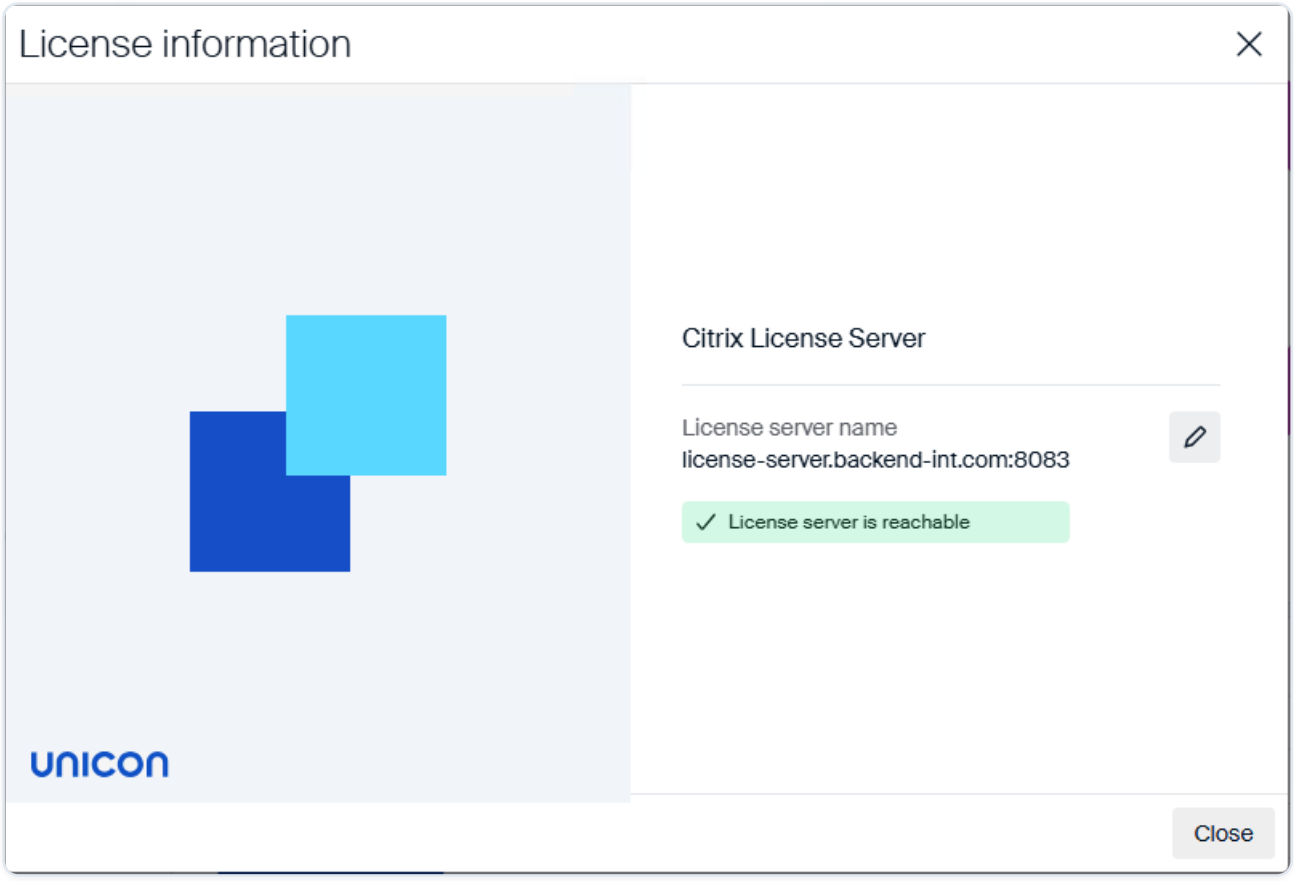
- File and path names are case-sensitive
- Do not use spaces.
- Specify file names with their extensions such as `.idf`.

The **Active** option allows you to specify for each defined entry whether it will be displayed or hidden in the **Firmware** dialog.

Licensing

The currently used license and the operation modes are shown in the dashboard and in the license information dialog

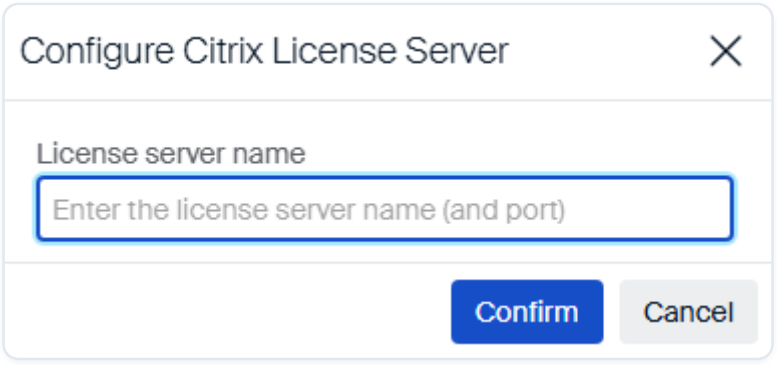
The license information dialog can be opened on the top-right corner in Scout Board, by clicking the help (question mark) icon.



This dialog will show the currently used operating mode and license information. If the Citrix License Server is being used, this dialog will show the `hostname:port` for the server, if it is reachable and additional information.

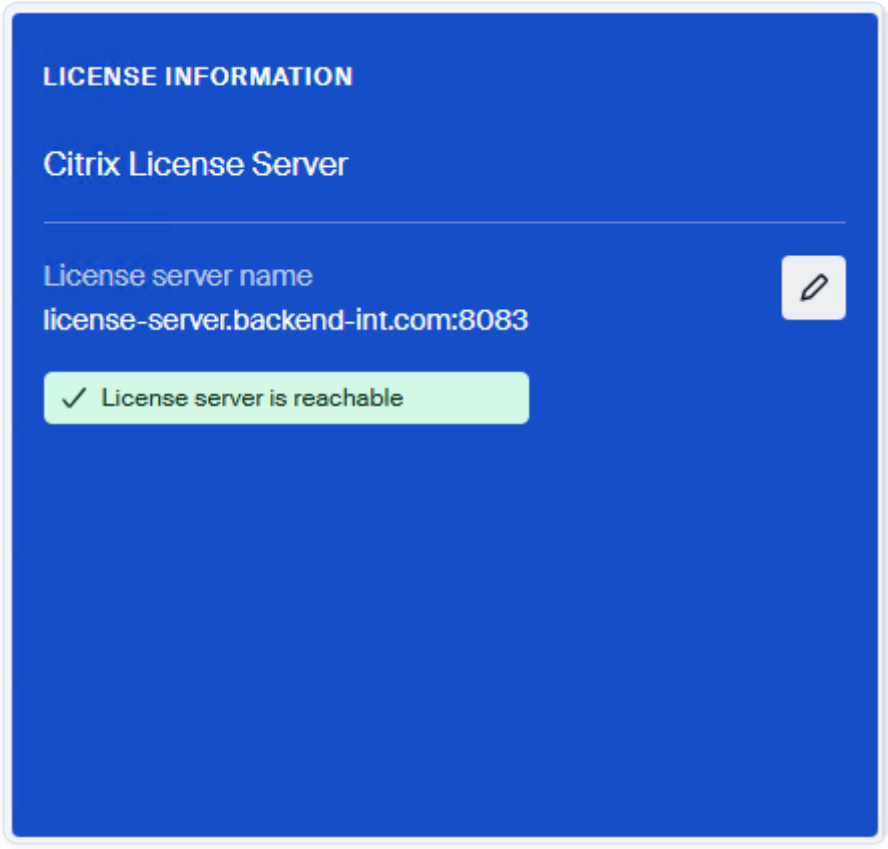
If the Citrix License Server is not reachable, Scout will go into a grace period. Then, a valid configuration needs to be provided in this period, in order for the Scout services to continue operation.

In order to set or change the Citrix License Server configuration, simply click the pen icon, provide the hostname and port and confirm.



The Citrix License Server configuration will be validated and information will be shown accordingly.

The dashboard widget for licensing will also provide all the information. There, you will also see the possibility to either change the Citrix License Server configuration or switch to the Citrix License Server from another license (i.e. Enterprise Subscription)



Switching between licenses

In Scout Board, depending on the current license, you can switch to another license mode.

With Scout Board versions 15.2511 and 15.2508.1, you can switch between the following licenses.

- Perpetual --> Enterprise Subscription
- Perpetual --> Citrix License Server
- Enterprise Subscription --> Citrix License Server

Please note, that the license switch is only one way. A switch back is not possible.

> **Note:**

> Switching from Perpetual --> Enterprise Subscription has been added with Scout Board versions 15.2511 and 15.2508.1. Before, only the other two methods were possible.

> Operation and grace period

Depending on the availability of the Citrix License Server and/or the entitlement for the Unicon Management Solution, the operation of the Scout Server can have different states.

CITRIX LICENSE SERVER	UNICON ENTITLEMENT	INFORMATION
Connected	Available	Scout operates normally. Devices can be managed.
Connected	NOT Available	Scout Server goes into a grace period of 30 days. After the grace period: Scout operates normally. Devices can't be managed.
NOT connected	NOT Available	If the Citrix License Server connection and a Unicon entitlement existed in the last 5 days: Scout Server goes into a grace period of 14 days. After the grace period: Scout Server stops. Devices can't be managed.
		If the Citrix License Server connection and a Unicon entitlement did not exist in the

CITRIX LICENSE SERVER	UNICON ENTITLEMENT	INFORMATION
		last 5 days: Scout Server goes into a grace period of 5 days. After the grace period: Scout Server stops. Devices can't be managed.

Additional information

To see more information about the Citrix License Server and the licensing process, please follow the following links:

- Citrix License Server
- Citrix License Manager

UniconOS Management Cloud

Overview

With UniconOS Management Cloud, Citrix offers a fully managed cloud infrastructure for the complete UniconOS endpoint management solution. The infrastructure is setup automatically, and the customer administrators get access to the endpoint management applications and can focus on the endpoint management.

With UniconOS (elux), Citrix provides a lean and secure operating system which runs on endpoints like thin clients, desktop PCs, and laptops.

With UniconOS Management (Scout) the provided components enable administrators to manage all endpoints from a central location.

The management components consist of Scout, Elias and Scout Cloud Gateway (SCG).

Scout is the central management component running on a Windows Server. Scout consists of multiple components, each running as a service: Scout, Scout Board, DBLayer, Keep Alive Service.

Elias enables administrators to create and manage software images and packages. An image makes up the complete UniconOS (eLux) system (operating system, drivers and applications), which runs on the endpoint.

Scout Cloud Gateway (SCG) is a VPN Gateway solution, which gives the possibility to securely connect UniconOS endpoints with UniconOS Management Cloud, so that these endpoints can be managed.

Running and maintaining the management components requires to setup the infrastructure, network and all the components. The required network and firewall rules need to be set and maintained. Components like SCG need to have a scalable infrastructure and high availability. Elias needs to be able to provide the images and packages to the endpoints and needs to be accessible.

When running on premises, the management components need to be set-up, maintained, updated, and managed by the customers' IT departments, while the endpoint administration will be performed by the same or dedicated administrators. This takes time and effort, and it is costly.

Nowadays, cloud offerings are common practice and software as a service is commonly used by many companies. The infrastructure is managed by the provider, and complex requirements like high availability, scalability, UpToDate software, backup and disaster recovery and support are part of the offering.

When using UniconOS Management Cloud, instead of managing the whole infrastructure on premises, you can work with a ready to use cloud environment and focus on the endpoints and the end user experience. While the endpoint management solution is part of the cloud environment, administrators can use the web-based application for performing endpoint management tasks.

Endpoint Management Components

In a UniconOS Management Cloud environment, the following components exist:

- Scout (UniconOS Management)
  - Scout Server
  - Scout Board
  - Scout Board DBLayer
  - Scout KeepAlive
- Elias 18 (Image and package management)
  - Elias Server
- Scout Cloud Gateway (SCG ~ Remote endpoint management)
  - SCG Services

- OpenVPN Server

Architecture

The UniconOS Management Cloud production environment facilitates the required resources and provides the optimal environment for production. The following features are key to the production environment.

- High availability
- Scalability
- Backup and disaster recovery (Full backups are being performed daily)
- Centralized logging (Logs are aggregated centrally for easy troubleshooting)
- Monitoring
- 24/7 Support
- Software updates and maintenance
- Image and package distribution via Blob Storage
- Multi-region

The database is based on SQL for endpoint management (Scout). For Elias and SCG, Document DB is being used.

The administrator can access the management applications using a web browser, and the communication is secured via HTTPS.

The communication between the endpoint and the management component is secured via HTTPS (Elias software packages) and TLS (bidirectional communication between eLux and Scout and eLux and SCG).

> **Info**

> Please contact your ATS for the next steps.

Implementation Guides

An Implementation Guide is a product manual that describes a dedicated topic or scenario that can be implemented by one or more of our products, including detailed instructions on how to proceed.

We provide Implementation Guides for our eLux operating systems, our management software Scout Enterprise Management Suite® and also for some additional features such as ELIAS, Scout Report Generator and others.

The documentation refers to the latest product versions.

Persistent User Configurations

This guide describes the Persistent Home feature in eLux. When enabled, user-specific configuration data is retained across logoff/logon cycles and device reboots.

By default, Persistent Home is disabled.

Overview

When persistent home is enabled, your personal settings will be saved and selected user configurations remain available across:

- User sign-out and subsequent sign-in
- Device restarts

Persistence is controlled explicitly via configuration (see Configuration).

Supported Applications and Data

Google Chrome

- Persisted data:
  - User profiles
  - Appearance settings (themes, fonts, color schemes)
  - other browser preferences

- Behavior:
  - Settings persist across sessions
  - Depending on backend configuration, settings may follow the user across devices

Firefox

- Persisted data:
  - Appearance settings (fonts, color schemes
  - Download location/path
  - other browser preferences
- Important:
- See Limitations regarding full profile file support.

ICA (Citrix Workspace App)

- Persisted data:
  - User-specific ICA settings (for example, session-related preferences)
- Continuity behavior:
  - Enables improved session continuity during temporary disruptions in the Citrix Delivery infrastructure.

GNOME Keyring (credentials/secrets)

This option is typically used in combination with browser or ICA persistence.

- Persisted data:
  - Stored credentials
  - Application secrets used by supported applications

Configuration

Persistent Home can be configured either locally or centrally.

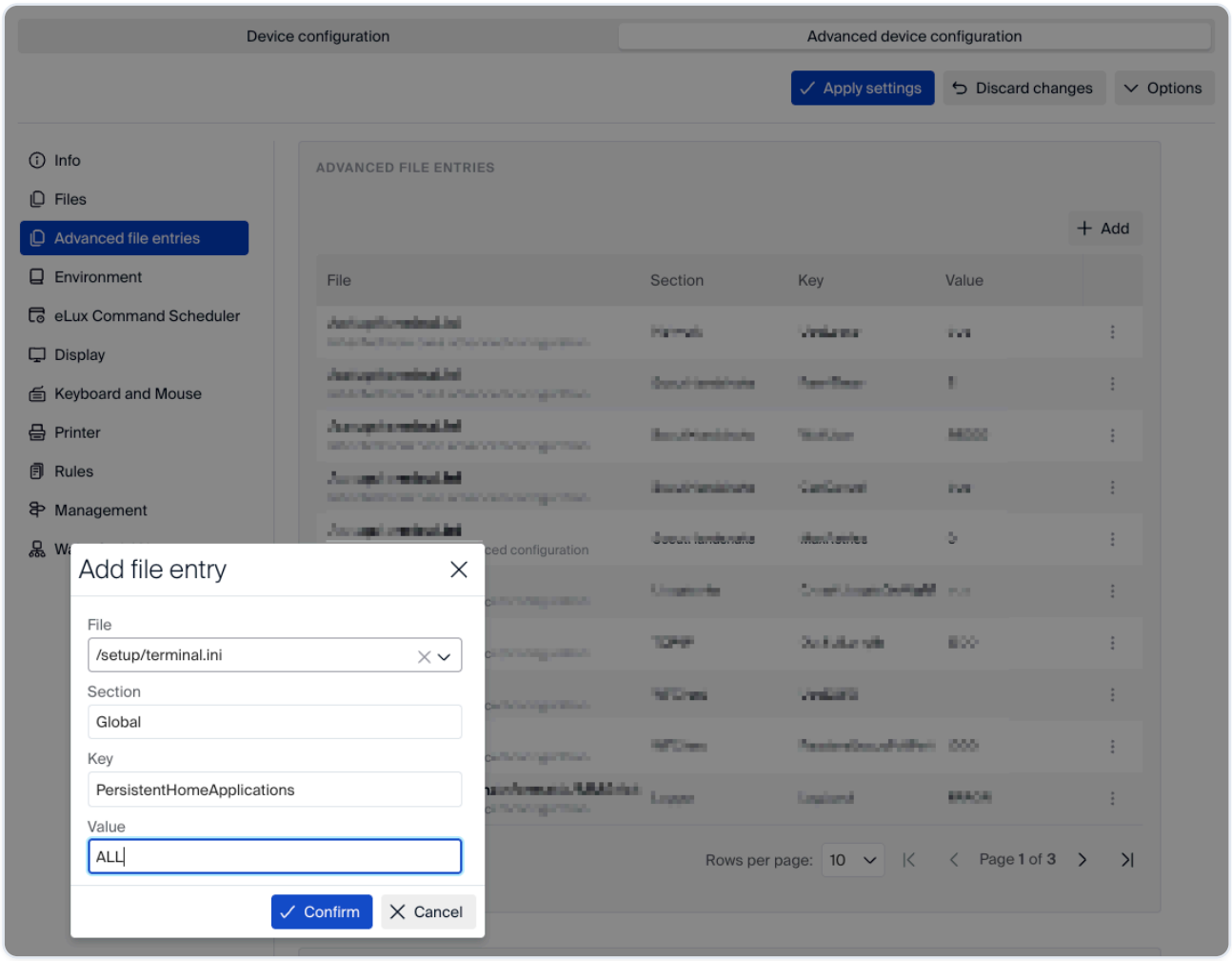
- Local file: `/setup/terminal.ini`
- Central management: Configurable via device profile in Scout

Configuration Parameter

- Use `PersistentHomeApplications` to select which items persist.
- Allowed values:
  - `google-chrome`
  - `firefox`
  - `ica`
  - `gnome-keyring`
  - `ALL`

Behavior

- Multiple values can be combined as required.
- ALL enables persistence for all supported applications available in the environment.
- gnome-keyring can be combined with:
  - google-chrome
  - ica
  - or both



Limitations

- Firefox:
  - Customized settings persist
  - Full profile file persistence is not supported in this release
- Persistence behavior may vary depending on:
  - Organizational security policies
  - Infrastructure configuration
  - Central management settings
- Data stored for Persistent Home will not be affected by Factory Reset - If data needs to be removed a full re-installation with partition format is required.

Administrators should validate persistence behavior in their specific deployment scenario.

Persistent User Configurations

This guide describes the Persistent Home feature in eLux. When enabled, user-specific configuration data is retained across logoff/logon cycles and device reboots.

By default, Persistent Home is disabled.

Overview

When persistent home is enabled, your personal settings will be saved and selected user configurations remain available across:

- User sign-out and subsequent sign-in
- Device restarts

Persistence is controlled explicitly via configuration (see Configuration).

Supported Applications and Data

Google Chrome

- Persisted data:
  - User profiles
  - Appearance settings (themes, fonts, color schemes)
  - other browser preferences
- Behavior:
  - Settings persist across sessions
  - Depending on backend configuration, settings may follow the user across devices

Firefox

- Persisted data:
  - Appearance settings (fonts, color schemes
  - Download location/path
  - other browser preferences
- Important:
- See Limitations regarding full profile file support.

ICA (Citrix Workspace App)

- Persisted data:
  - User-specific ICA settings (for example, session-related preferences)
- Continuity behavior:
  - Enables improved session continuity during temporary disruptions in the Citrix Delivery infrastructure.

GNOME Keyring (credentials/secrets)

This option is typically used in combination with browser or ICA persistence.

- Persisted data:
  - Stored credentials
  - Application secrets used by supported applications

Configuration

Persistent Home can be configured either locally or centrally.

- Local file: `/setup/terminal.ini`
- Central management: Configurable via device profile in Scout

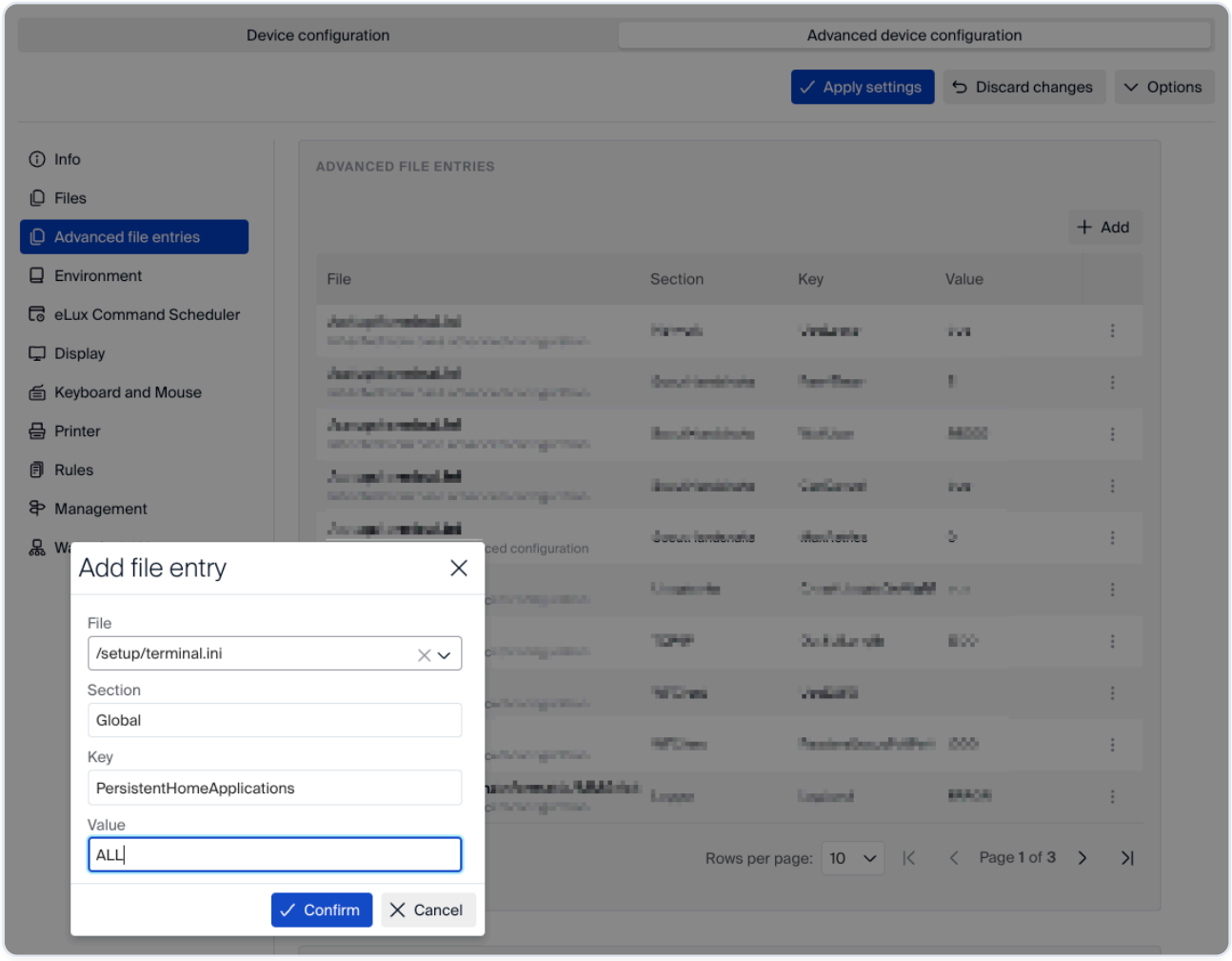
Configuration Parameter

- Use `PersistentHomeApplications` to select which items persist.
- Allowed values:
  - `google-chrome`
  - `firefox`
  - `ica`
  - `gnome-keyring`
  - `ALL`

Behavior

- Multiple values can be combined as required.
- ALL enables persistence for all supported applications available in the environment.
- gnome-keyring can be combined with:

- google-chrome
- ica
- or both



Limitations

- Firefox:
  - Customized settings persist
  - Full profile file persistence is not supported in this release
- Persistence behavior may vary depending on:
  - Organizational security policies
  - Infrastructure configuration
  - Central management settings
- Data stored for Persistent Home will not be affected by Factory Reset - If data needs to be removed a full re-installation with partition format is required.

Administrators should validate persistence behavior in their specific deployment scenario.

AD Credentials Pass-Through for Google Chrome

Overview

eLux supports **Active Directory (AD) credential pass-through for Google Chrome**, enabling seamless access to internal web resources without repeated authentication prompts.

Key Capabilities

- **Single sign-on (SSO)** After logging in to eLux with AD credentials, Google Chrome automatically reuses the same Kerberos-based authentication context (Kerberos TGT).
- **Policy-Controlled Experience** Chrome enforces centrally defined policies, ensuring transparent and compliant access to corporate web services.

Prerequisites

Ensure the following conditions are met before configuration:

- AD authentication is enabled and functional in Scout.
- The user successfully logs in to eLux with AD credentials.
- Google Chrome is installed on the eLux endpoint.
- You have sufficient permissions to modify the device profile in Scout.

Configuration (Scout)

1. Open the relevant device profile in **Scout**.
2. Navigate to **Applications** → **Google Chrome** → **Free parameters**.
3. Add the following parameter and save the configuration: `AllowSSOLists=example.com`
   Notes:
   - "example.com" is an example and may be exchanged with your domains.
   - The parameter accepts a comma separated list of domains.
   - The parameter accepts Google Chrome policy wildcards (e.g. *.example.com)
4. Save and deploy the profile to the target devices.

End-User Experience

1. The user logs in to eLux using their AD credentials.

2. Google Chrome is launched (manually or via autostart).
3. When accessing any domain listed in **AllowSSOLists**, Chrome performs automatic authentication without prompting for credentials.

Troubleshooting

Google Chrome still prompts for a password

- Verify the user logged in with an AD account.
- Confirm the destination site is included in **AllowSSOLists**.
- Redeploy the Scout profile if changes were made recently.

POLICY NOT APPLIED

- In Chrome, enter `chrome://policy` and check that **AuthNegotiateDelegateAllowlist** shows the expected entries.

Need Help?

Contact your system administrator or eLux support for further assistance.

Smart Card Actions Configuration

Goal

Configure Smart Card Actions on eLux devices to control system behavior when a smart card is inserted or removed. This includes session locking, logoff, user switching, and execution of custom scripts.

Prerequisites

Before you begin, ensure the following:

- You have access to **Scout** with permission to edit device profiles.
- The target device is assigned to the correct profile.
- Smart card hardware is installed and functioning.
- Any required Bash scripts are prepared and available for upload.

Configuration Overview

The Smart Card Actions feature provides seven configuration parameters. Configuration must be done via Advanced File Entries:

- **Scout → Device Profile → Advanced device configuration → Advanced file entries**

The relevant sections are:

```
[Hotkeys]

PauseAction



[SmartCard]

OnScInsert

OnScRemoval

ScInsertScript

ScRemoveScript

PauseActionOnDifferentCardInserted

PauseActionTimeoutSeconds
```

Configuration Details

The following parameters control system behavior when a smart card is inserted, paused or removed.

PauseAction

Defines the modifier key used to trigger the pause action.

- **Default value:** `<shift>`
- **Valid values:** `<shift>`, `<ctrl>`, `<alt>`

> This setting does not distinguish between left and right modifier keys.

OnScInsert

Defines the action performed when a smart card is inserted.

- **Default value:** `none`

- **Valid values:** `none`, `runscript`
- **Requirement:** Value must be lowercase.

VALUE	BEHAVIOR
`none`	No action is performed.
`runscript`	Executes the script defined in `ScInsertScript`.

OnScRemoval

Defines the action performed when a smart card is removed.

- **Default value:** `none`
- **Valid values:** `none`, `lock`, `logoff`, `runscript`
- **Requirement:** Value must be lowercase.

VALUE	BEHAVIOR
`none`	No action is performed.
`lock`	Locks the current eLux session.
`logoff`	Logs off the current eLux session.
`runscript`	Executes the script defined in `ScRemoveScript`.

ScInsertScript

Specifies the script executed when a smart card is inserted.

- **Default value:** `empty`
- **Valid values:** Full path to a Bash script on the device.

> The file must exist on the device and match the configured path exactly.

ScRemoveScript

Specifies the script executed when a smart card is removed.

- **Default value:** `empty`
- **Valid values:** Full path to a Bash script on the device.

> The file must exist on the device and match the configured path exactly.

PauseActionOnDifferentCardInserted

Defines system behavior when a different smart card is inserted during an active session.

- **Default value:** `switchuser`
- **Valid values:** `switchuser`, `ignore`
- **Requirement:** Value must be lowercase.

VALUE	BEHAVIOR
`switchuser`	Logs off the current session and displays the Smart Card login screen.
`ignore`	Continues the current session without interruption.

PauseActionTimeoutSeconds

Defines the timeout (in seconds) for the pause action.

- **Default value:** `30`
- **Valid values:** Numeric (5–300 seconds)

Behavior outside valid range:

- Values below `5` are automatically adjusted to `5`.
- Values above `300` are automatically adjusted to `300`.

Script Files

Smart Card Actions support custom **Bash scripts** that can be executed on card insertion or removal.

Supported Script Type

- Only **Bash (`.sh`) scripts** are supported.
- Scripts must be uploaded in **Scout** under:

Advanced device configuration → Files

File Location Requirements

The **destination path** configured during file upload must exactly match the value specified in:

- `ScInsertScript`
- `ScRemoveScript`

If the configured path and the uploaded file location do not match, the script will not execute.

> Ensure the script has the correct permissions and is accessible on the device.

Imprivata Guide

This documentation provides guidance on the integration of **Imprivata authentication** with **eLux**, enabling secure, streamlined, and user-friendly access to applications and desktops in data-sensitive environments, with a particular focus on healthcare.

Modern hospitals and care facilities rely on fast, reliable, and secure authentication workflows to reduce login delays and minimize administrative burden for clinicians. With this integration, eLux acts as the endpoint operating system for thin clients and workstations, while Imprivata serves as the **central authentication backend**. This combination delivers a strong, enterprise-grade solution for **Single Sign-On (SSO)** and **proximity card (Tap&Go)** access, tailored to the unique needs of healthcare IT.

Purpose of the Integration

The integration allows clinicians to:

- **Log in quickly** using Imprivata-supported methods such as username/password or proximity cards.
- **Automatically connect** to Citrix StoreFront sessions using Single Sign-On, reducing the need for repeated credentials input.
- **Maintain security and compliance** while ensuring resilience against login errors (e.g., incorrect PIN/password entries).

By linking Imprivata’s authentication backend with eLux, users benefit from:

- **Improved clinical workflow efficiency:** Caregivers spend less time on login screens and more time with patients.

- **Reduced support overhead:** Centralized authentication policies simplify IT management.
- **Compliance with healthcare security standards:** Multi-factor authentication and controlled session access safeguard sensitive patient data.
- **Resilience and continuity:** Robust error handling and session management designed for 24/7 healthcare operations.

Target Audience

This documentation is intended for:

- **IT administrators** managing hospital endpoints and Citrix environments.
- **System integrators** deploying Imprivata in healthcare institutions alongside eLux OS.

It assumes familiarity with:

- Citrix StoreFront / Virtual Apps and Desktops
- Imprivata ProveID appliance configuration

Workflows

eLux supports multiple authentication workflows, which can be applied individually or in combination depending on organizational requirements.

Authentication Only Provides secure authentication through the Imprivata appliance at device login.

After successful authentication, the user receives access to the standard eLux desktop with all configured applications.

Fast User Switching (FUS) Enables immediate access to a predefined desktop or application after login, reducing delays and supporting quick role changes in clinical environments.

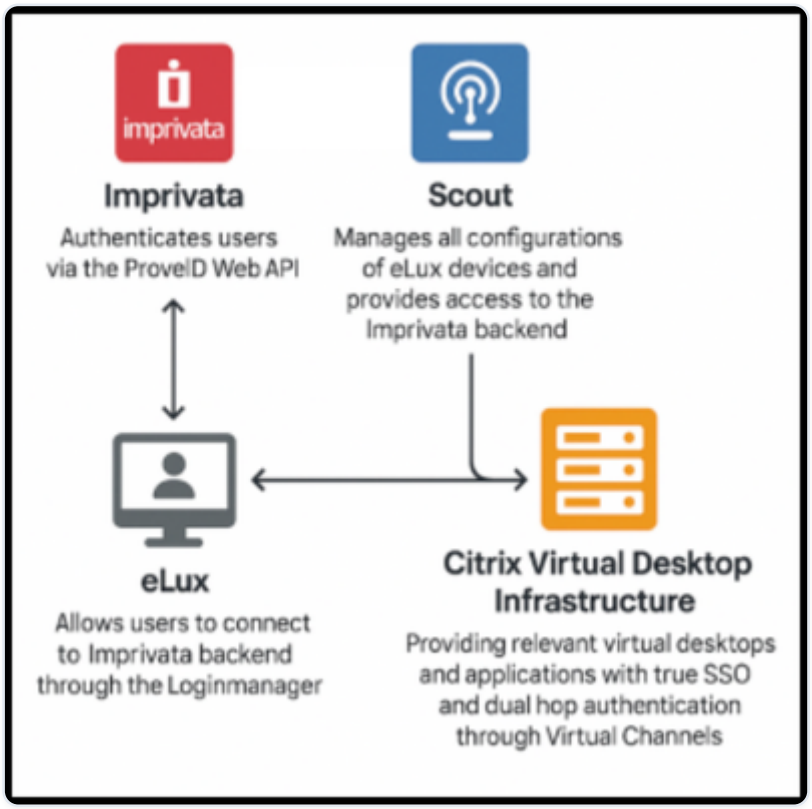
EPCS Reauthentication Supports clinical workflows that require reauthentication within target applications running on VDI, leveraging virtual channel integration.

Combined Workflows Workflows can also be combined. For example, **Kiosk Mode** can be paired with **Fast User Switching** to enable rapid handovers while maintaining direct access to a core application.

Architecture

Secure authentication with Imprivata in eLux is achieved through the tight integration of four system components:

- **Imprivata** – Authenticates users via the ProveID Web API.
- **Scout** – Manages eLux device configurations and provides connectivity to the Imprivata backend.
- **eLux** – Connects users to Imprivata through the Login Manager and serves as the fundamental operating system.
- **Citrix Virtual Desktop Infrastructure (VDI)** – Delivers desktops and applications with full SSO and dual-hop authentication via Virtual Channels.



eLux Usage

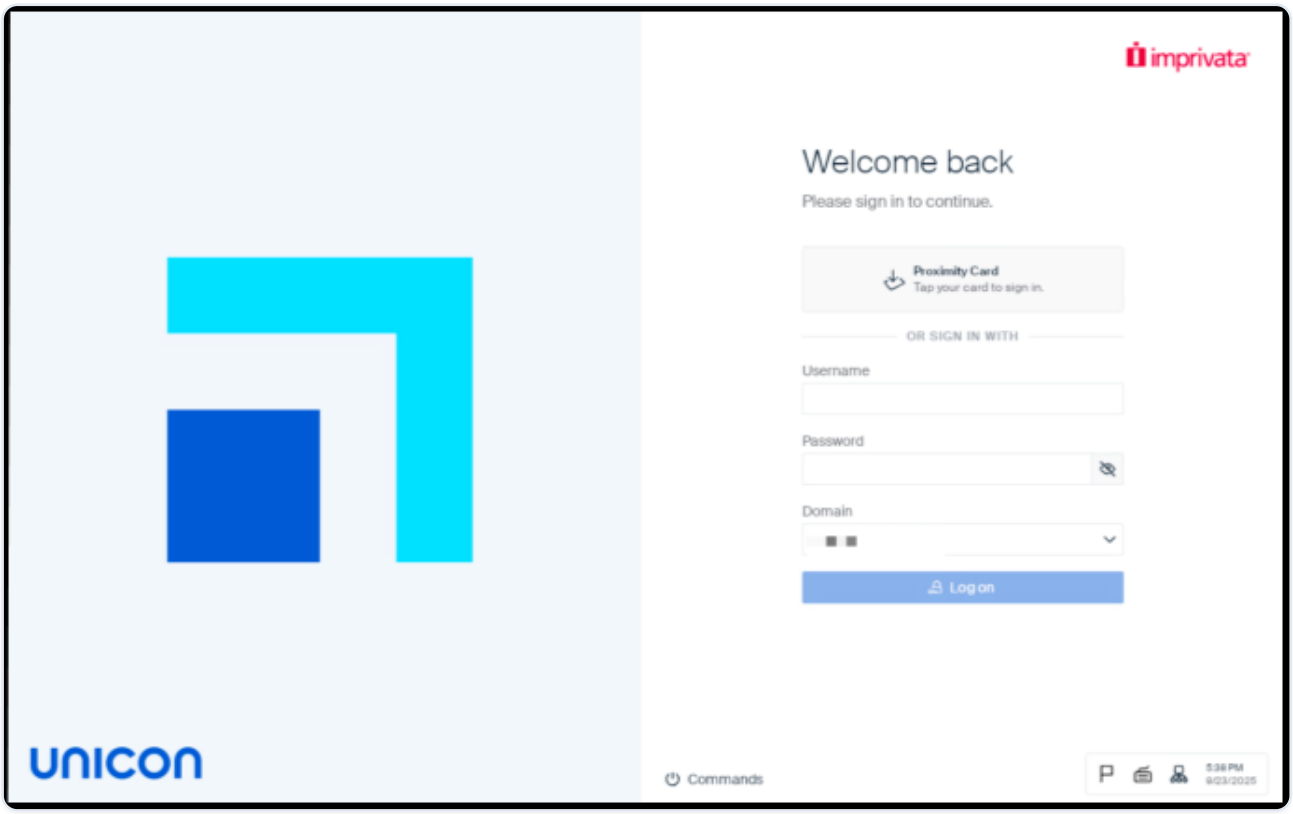
Login Manager

When an eLux device is configured to use **Imprivata** as the authentication backend, the Login Manager displays the **Imprivata logo** after startup.

If a supported card reader is connected, the interface will present a **Proximity Card field**, indicating that **Tap & Go** authentication is available.

The specific login methods offered to the user depend on the configuration defined in the Imprivata backend and may include:

- Username and Password
- Proximity Card with Password
- Proximity Card with PIN
- Proximity Card only



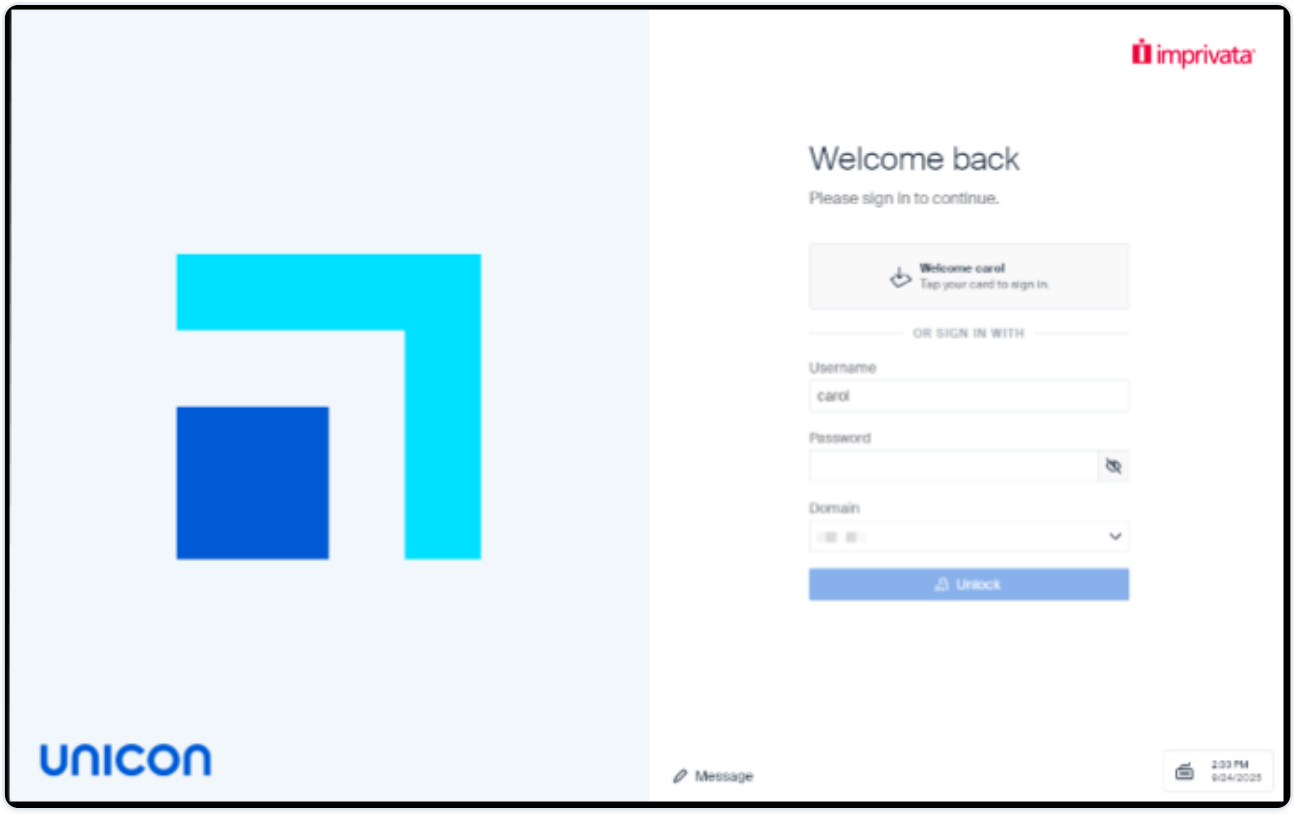
Lock Screen

An eLux device can be locked securely using different methods, depending on the system configuration:

- **Card tap** – Quickly lock the system by tapping the proximity card.
- **Start Menu** – Select *Lock* from the eLux Start Menu.
- **Keyboard shortcut** – Use the configured key combination.
- **Inactivity timeout** – The device locks automatically after a defined period of no user interaction.

Once the lock screen is active:

- The currently signed-in user can unlock the session using one of the configured authentication methods.
- Alternatively, another user may log in, which will end the active session of the first user.



Imprivata Card and PIN Enrollment

Goal

Enable users to enroll additional authentication modalities (smart card and PIN) directly during login on eLux devices integrated with Imprivata. This ensures that unknown cards can be registered and required PINs can be set without separate administrative steps.

Feature Overview

eLux supports dynamic enrollment of the following Imprivata authentication modalities during login:

- **Card Enrollment**
  - When a user taps a smart card that is not yet registered in the Imprivata backend, the card can be enrolled during the login process.
- **PIN Enrollment**
  - If the PIN modality is configured in the Imprivata backend but no PIN has been enrolled yet, the user is required to enroll a PIN during login.

No additional client-side configuration is required in eLux.

Prerequisites

Before using card or PIN enrollment, ensure the following:

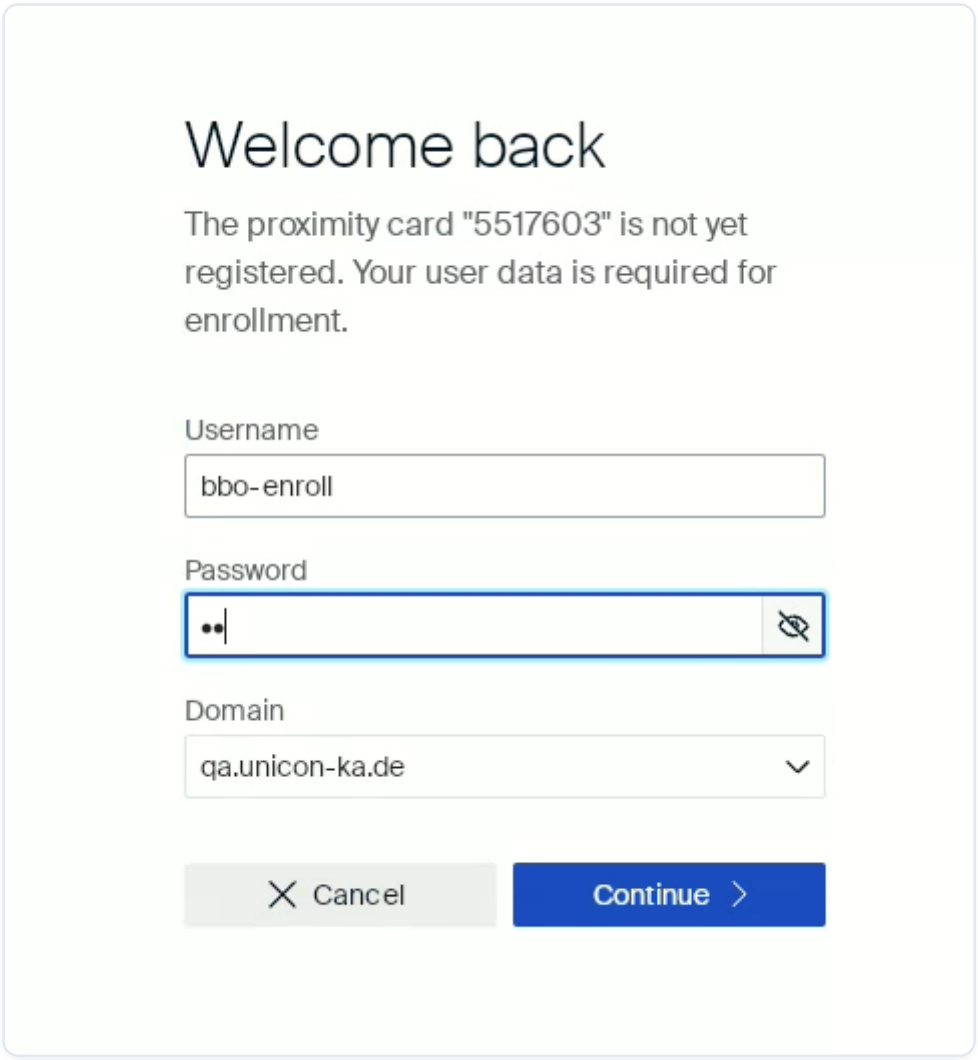
- eLux device is integrated with Imprivata.
- The Imprivata backend is reachable from the device.
- The user account exists in the Imprivata backend.
- For PIN enrollment:
  - The **PIN modality** is enabled for the user in the Imprivata backend.
- For card enrollment:
  - Card enrollment is allowed in the Imprivata backend configuration.

Authentication Flow with Enrollment

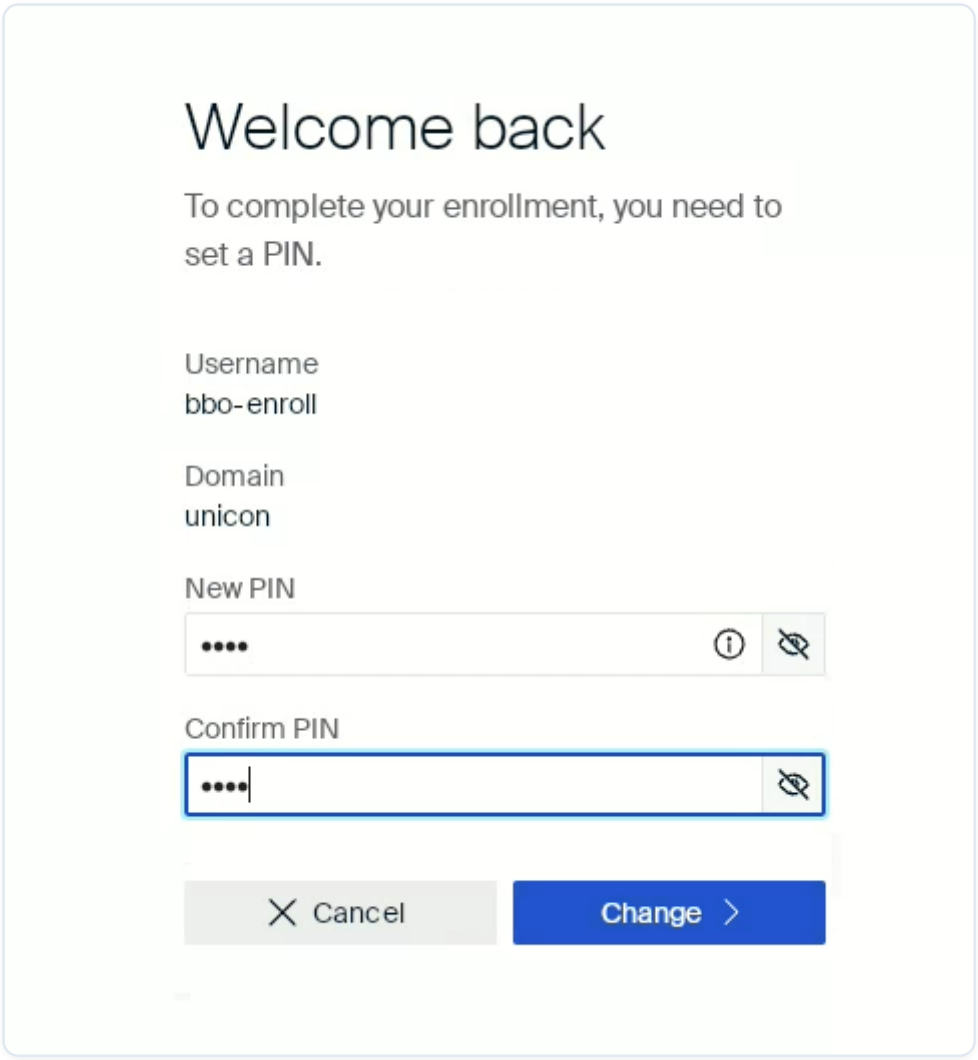
The following describes the authentication flow when both card and PIN enrollment are required:

1. The user taps an unknown smart card.
2. eLux detects that the card is not registered in the Imprivata backend.
3. The user is prompted to enroll the card.
4. The user provides:
   - Username
   - Password
   - Domain
5. If credentials are valid, the card is enrolled and associated with the user account.
6. If the backend requires PIN enrollment, the user is prompted to set a new PIN.
7. After successful PIN enrollment, the user is logged in.

Example card enrollment UI:



Example pin enrollment UI:



Configuration

This section describes relevant configuration aspects on both eLux and the Imprivata backend.

eLux Configuration

No additional configuration parameters are required. The enrollment functionality is enabled by default and cannot be disabled on the client side.

Imprivata Backend Configuration

CARD ENROLLMENT

- Card enrollment must be permitted in the Imprivata backend.
- If card enrollment is disabled:
  - The eLux client displays an error message when an unknown card is tapped.
  - Enrollment cannot proceed.

PIN ENROLLMENT

- The **PIN modality** must be configured for the user.
- If configured and no PIN is enrolled:
  - The user is required to complete PIN enrollment before login.
- If the PIN modality is not configured:
  - The user is not prompted for PIN enrollment.

Known Limitations

- PIN enrollment is triggered immediately if:
  - The PIN modality is configured in the Imprivata backend, and
  - No PIN has been enrolled yet.
- This applies even if the user logs in using username and password instead of a smart card.

> **Example:**

> A user logs in with username and password. If the PIN modality is configured but no PIN exists yet, the user must enroll a PIN before access is granted.

Imprivata Card and PIN Enrollment

Goal

Enable users to enroll additional authentication modalities (smart card and PIN) directly during login on eLux devices integrated with Imprivata. This ensures that unknown cards can be registered and required PINs can be set without separate administrative steps.

Feature Overview

eLux supports dynamic enrollment of the following Imprivata authentication modalities during login:

- **Card Enrollment**
  - When a user taps a smart card that is not yet registered in the Imprivata backend, the card can be enrolled during the login process.
- **PIN Enrollment**

    - If the PIN modality is configured in the Imprivata backend but no PIN has been enrolled yet, the user is required to enroll a PIN during login.

No additional client-side configuration is required in eLux.

Prerequisites

Before using card or PIN enrollment, ensure the following:

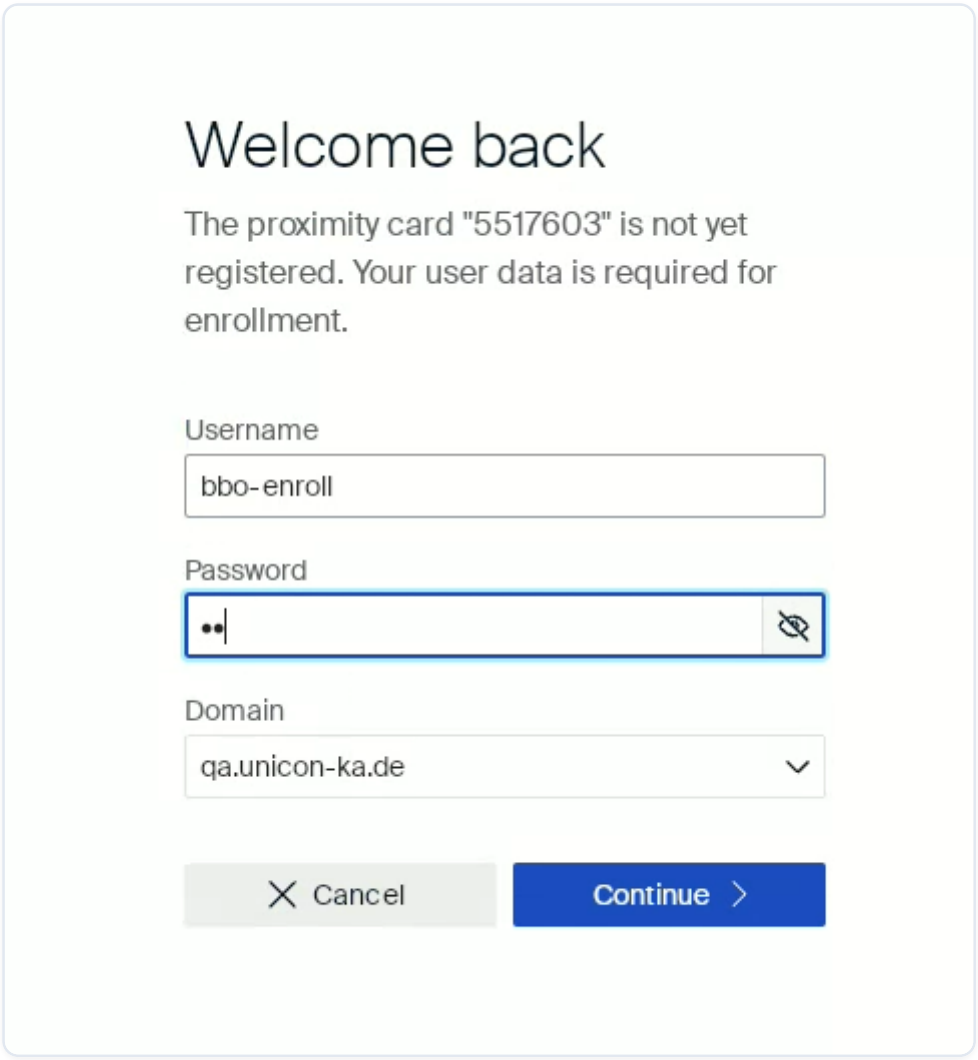
- eLux device is integrated with Imprivata.
- The Imprivata backend is reachable from the device.
- The user account exists in the Imprivata backend.
- For PIN enrollment:
    - The **PIN modality** is enabled for the user in the Imprivata backend.
- For card enrollment:
    - Card enrollment is allowed in the Imprivata backend configuration.

Authentication Flow with Enrollment

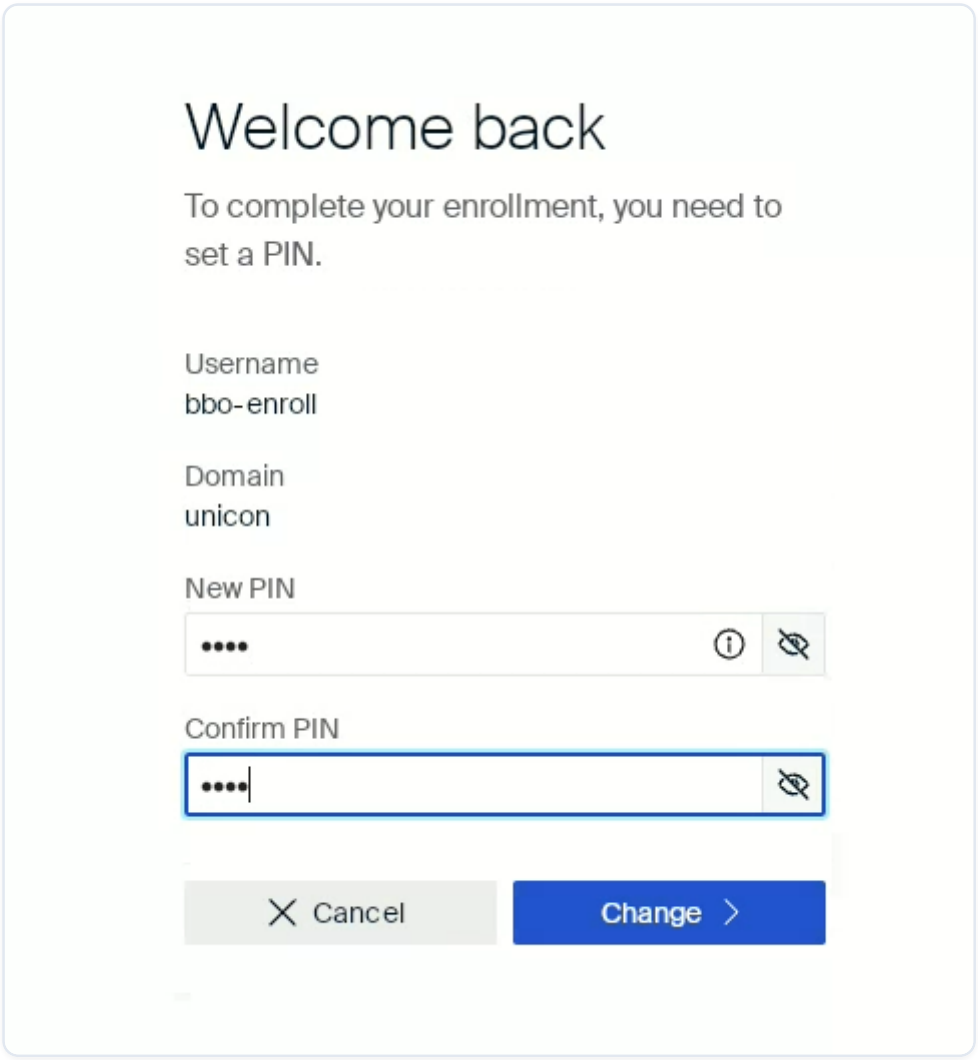
The following describes the authentication flow when both card and PIN enrollment are required:

1. The user taps an unknown smart card.
2. eLux detects that the card is not registered in the Imprivata backend.
3. The user is prompted to enroll the card.
4. The user provides:
    - Username
    - Password
    - Domain
5. If credentials are valid, the card is enrolled and associated with the user account.
6. If the backend requires PIN enrollment, the user is prompted to set a new PIN.
7. After successful PIN enrollment, the user is logged in.

Example card enrollment UI:



Example pin enrollment UI:



Configuration

This section describes relevant configuration aspects on both eLux and the Imprivata backend.

ELUX CONFIGURATION

No additional configuration parameters are required. The enrollment functionality is enabled by default and cannot be disabled on the client side.

IMPRIVATA BACKEND CONFIGURATION

Card Enrollment

- Card enrollment must be permitted in the Imprivata backend.
- If card enrollment is disabled:
  - The eLux client displays an error message when an unknown card is tapped.
  - Enrollment cannot proceed.

PIN Enrollment

- The **PIN modality** must be configured for the user.

- If configured and no PIN is enrolled:
  - The user is required to complete PIN enrollment before login.
- If the PIN modality is not configured:
  - The user is not prompted for PIN enrollment.

Known Limitations

- PIN enrollment is triggered immediately if:
  - The PIN modality is configured in the Imprivata backend, and
  - No PIN has been enrolled yet.
- This applies even if the user logs in using username and password instead of a smart card.

> **Example:**

> A user logs in with username and password. If the PIN modality is configured but no PIN exists yet, the user must enroll a PIN before access is granted.

Configuration

The following steps are required to enable authentication with Imprivata. Configuration must be performed both on the Imprivata appliance and within Scout.

System Requirements

Enable ProveID Web API Access in Imprivata

Scout Settings

Citrix Configuration

System Requirements

Before beginning the configuration, verify that all required software versions are installed and prerequisites are fulfilled.

Supported Versions

- **Imprivata EAM:** 26.1

- **Scout:** 15.2507 or later
- **eLux:** 7.2509 or later

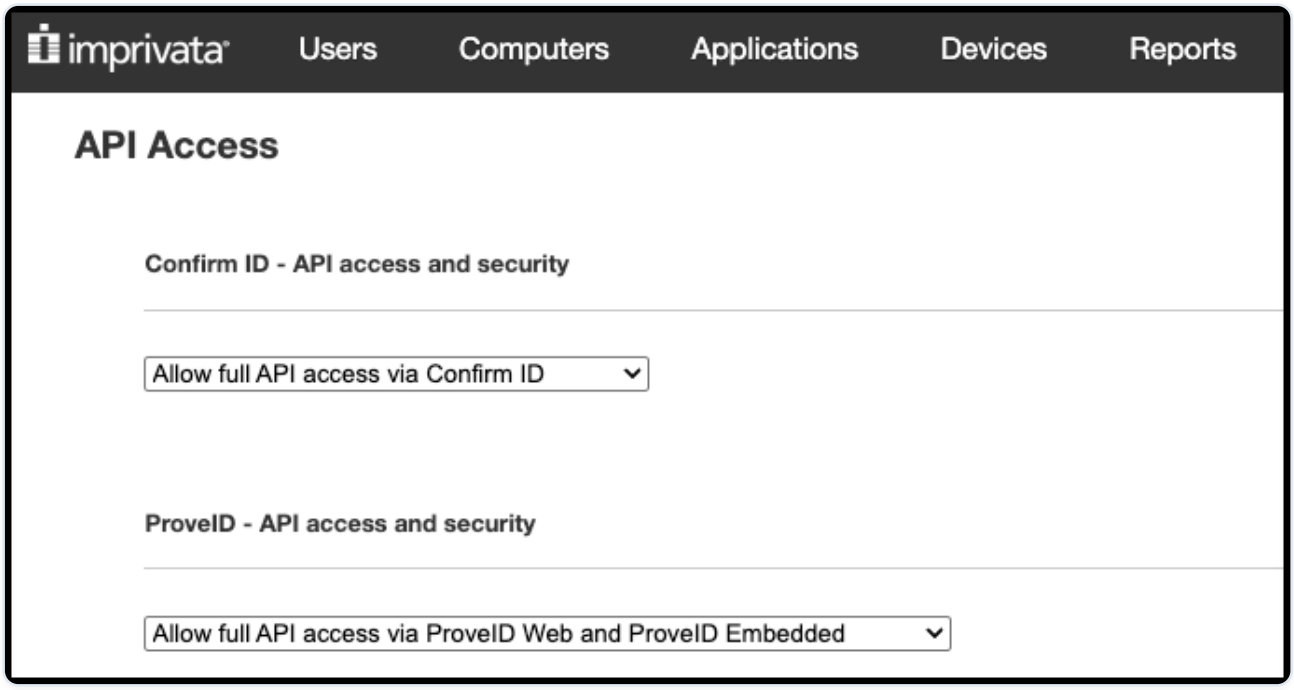
Installation Prerequisites

- The eLux device image must include the **Imprivata Access Management EPM package**.
- For full support of all Imprivata-related features, ensure the following optional FPM packages are also enabled:
  - Proximity Reader support
  - Fast User Switching support
  - Imprivata test D-Bus client/server (debugging purposes only)
- For support of pass-through logon, the **Dialog Extension** FPM of the **Citrix extensions EPM package** must be included in the eLux device image.

Enabling ProveID Web API Access in Imprivata

To allow eLux to integrate with Imprivata as an authentication backend, API access must first be enabled on the Imprivata appliance:

1. Log in to the **Imprivata Admin Console** with an administrator account.
2. In the upper-right corner of the console, click the **gear icon** and select **API Access**.
3. Under **ProveID – API Access and Security**, enable the option **Allow full API access via ProveID Web API and ProveID Embedded**.
4. Select the **Citrix eLux** checkbox to complete the configuration (Future 1 in older versions of the Imprivata EAM).

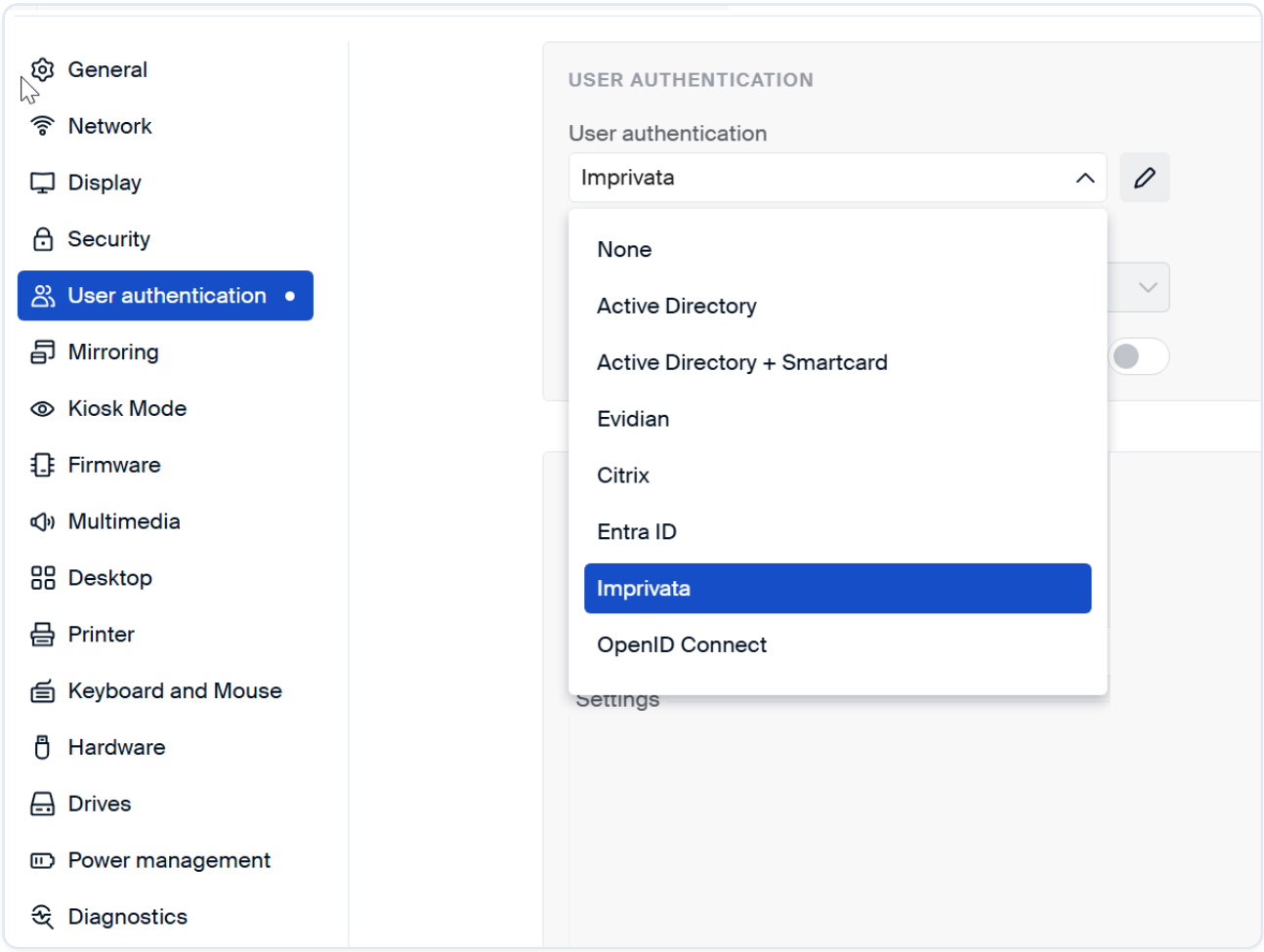


Scout

Enabling Imprivata as Authentication Type

To configure Imprivata as the authentication method, proceed as follows:

1. Open **Scout Board** and navigate to **Device Configuration**.
2. In the **User Authentication** section, click **Edit**.
3. From the drop-down menu, select **Imprivata**.
4. Click the **Edit** button below and enter the **Bootstrap URL**, which is the address of the Imprivata appliance.
5. The **timeout setting** (default: 60 seconds) defines how long the system waits for a password or PIN entry after a proximity card tap. If the entry is not completed within this period, the system automatically returns to the login screen.
6. Click **Apply/OK** to confirm the changes.



Deploying Imprivata Appliance Certificates

A certificate in **Base64 X.509 format (.crt)** must be installed on eLux devices to enable communication with the Imprivata appliance.

- By default, each Imprivata appliance is shipped with a **self-signed certificate** generated by the appliance's built-in root CA.
- This certificate is sufficient for initial testing or smaller deployments.
- However, self-signed certificates must still be signed by a **root-CA**, and the corresponding **root-CA certificate** must also be available on the client.

Many organizations prefer to replace the default certificate with one signed by a trusted authority. This can be:

- An **internal enterprise CA** (e.g., Microsoft Active Directory Domain Enterprise Root CA), or
- A **publicly trusted CA** such as DigiCert.

In such cases, the **complete certificate chain** (root CA, intermediate/subordinate CA) must be exported in Base64 X.509 format and deployed to all eLux devices via Scout.

OPTION 1: USING THE IMPRIVATA SELF-SIGNED ROOT CERTIFICATE

1. Log in to the **Imprivata Appliance Console** and open the **Security** tab.
2. If the following message is displayed: “The SSL certificate for this appliance has been self-signed by the certification authority (CA) on this appliance. Download the certificate of this CA.”*
3. Click **Download the certificate**.
4. The file will be saved as a **.crt file**.

This root certificate must then be deployed to your eLux devices via **Scout** to establish a trusted connection between clients and the appliance.

☞ For guidance on how to deploy certificates via Scout, please refer to Certificates.

OPTION 2: USING A TRUSTED THIRD-PARTY OR ENTERPRISE CA

If the Imprivata appliance has been configured with a certificate signed by a corporate or public CA, the root and intermediate certificates cannot be downloaded from the appliance directly.

In this scenario:

1. Use a supported browser (e.g., **Microsoft Edge** or **Google Chrome**) to access the **Imprivata Appliance Console**.
2. Export the complete certificate chain (root CA, intermediates, and appliance certificate).
3. Ensure the chain is saved in **Base64-encoded X.509 format (.crt)**.
4. Deploy the certificate chain to your eLux devices via **Scout**.

> ***Note:*** Please make sure that the time and date settings on your eLux devices are accurate. Incorrect time settings can lead to certificate validation failures, preventing successful communication with the Imprivata appliance.

Citrix Configuration

The configuration of Citrix VDI sessions is managed through **Scout**.

For a detailed walkthrough, please refer to Connecting to a Citrix® farm.

When configuring Citrix sessions in the Imprivata context, it is important to distinguish between **private user sessions** and **persistent sessions**:

Private User Session

A private Citrix virtual desktop or application session provides each user with access to their own set of applications and desktops.

- The session is terminated upon logoff.
- Best suited for **classic, user-centric workloads** such as office and communication applications.

CONFIGURATION OF PRIVATE USER SESSIONS

In order to configure private user sessions in Scout, proceed as usual but ensure the following settings are applied in the configuration of your Storefront application:

- In advanced settings:
  - Timed logoff: **Logoff after** is not set (disabled).
  - Manual logoff: **Logoff server and disconnect sessions** is enabled.

Persistent Session

For workflows requiring **Fast User Switching (FUS)** and immediate access to shared clinical applications (e.g., Electronic Health Records), persistent Citrix session is required.

- These sessions remain active even after a user logs off.
- The application is instantly available to the next user logging in.
- The Imprivata Agent running inside this Citrix session is informed of changes in regard to the user session (i.e. logon/logoff/locking/unlocking).
- Each eLux client configured for a persistent session requires a **dedicated service account**.

CONFIGURATION OF A PERSISTENT SESSION

In order to configure the persistent session in Scout, proceed as usual but ensure the following settings are applied in the configuration of your Storefront application:

- Username and password fields are set to the dedicated service account credentials for this eLux client (see tip below).
- In advanced settings:
  - Timed logoff: **Logoff after** is not set (disabled).
  - Application Reconnection is set to **Do not reconnect**.
  - Manual logoff: is set to **Logoff only server**.
- In *Free parameters* add the following entries:
  - *Variable*: Persistent; *Value*: true
  - *Variable*: Hidden; *Value*: true

TIP:

The %HOSTNAME% macro can be used in the username field to automatically append the eLux client hostname to the service account name. This allows the use of a single service account prefix for all eLux clients (e.g., "citrixsvc_%HOSTNAME%").

Note:

When using version 2508 of the Citrix CWA EPM package, please also add the following entry under advanced file entries in Scout: *file*: /setup/sessions.ini *Section*: ICADefaults *Key*: StorebrowseIPCDisabled *Value*: true

> **Important:**

> When using persistent and private sessions in parallel, please ensure that different Storefront URIs are used to avoid conflicts.

Imprivata Failover & Recovery

Goal

Ensure uninterrupted authentication by automatically:

- Switching to a **failover site** when all configured home sites are unavailable.
- Periodically checking whether a home site has recovered.
- Switching back to a working home site without user interaction.
- Recovering automatically from offline mode when any site becomes reachable.

Prerequisites

Before you begin, ensure the following:

- eLux **7.2601.0** or later is installed.
- Imprivata is configured and operational.
- A valid **bootstrap URL** is defined.
- Network access to home and failover sites is properly configured.
- Administrative access to run network test commands (for validation scenarios).

How Failover Recovery Works

NORMAL OPERATION

- The device connects to a configured **home site**.
- Authentication works as expected.

HOME SITE UNAVAILABLE

- If the home site becomes unreachable, the device automatically switches to a configured **failover site**.
- Users can continue logging in without interruption.
- While connected to the failover site, the device periodically checks whether a home site is available again.

AUTOMATIC SWITCH BACK

- Every `failoverSiteTimeoutSeconds` (default: 5 minutes), the device attempts to contact the home sites.
- As soon as a home site responds, the device automatically switches back.
- No user interaction is required.

OFFLINE MODE

If both home and failover sites are unreachable:

- The device enters **offline mode**.
- Authentication is not possible.
- Every `offlineTimeoutSeconds` (default: 5 minutes), the device retries all configured sites.
- As soon as any site becomes reachable, the system reconnects automatically.

Configuration

Failover behavior can be configured either locally or centrally.

- Local file: `/setup/terminal.ini`
- Central management: Configurable via device profile in Scout

Failover behavior is controlled under the **"Imprivata"** section.

PARAMETER	DESCRIPTION	DEFAULT
`failoverSiteTimeoutSeconds`	How often the system checks for a recovered home site while connected to a failover site.	**300 s** (5 min)
`offlineTimeoutSeconds`	How often the system retries connecting when all sites are unreachable.	**300 s** (5 min)

Tip:

Lower values mean faster recovery but slightly more network traffic. The default of 5 minutes is a good balance for most environments.

License Activation Service

License Activation Service (LAS) is the next generation technology for product activations across the suite of Citrix products.

The Citrix License Server integration has been released with the Scout version 15 2508 LTSR.

Customers updating from a previous version can update as usual. Then, in Scout Board (web based management UI), the administrator can switch the license mode to use the Citrix License Server. The switch can be performed in the License information dialog. Customers performing a fresh installation of the Scout version 15 2508 LTSR can configure the Citrix License Server during installation. The administrator can reconfigure it any time later, if the Citrix License Server configuration changes.

To see more information about the Citrix License Server and the licensing process, please follow the following links:

- Citrix License Server
- Citrix License Manager

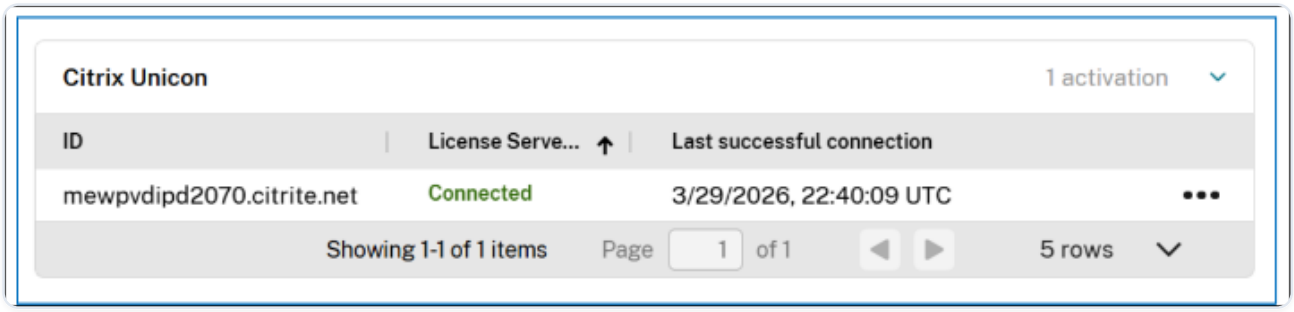
> **Note:**

> File based licensing system (including perpetual eLux and Scout licences and the Enterprise Subscription operating mode), traditionally used for activating various on-premises components, will be End of Life (EOL) on April 15, 2026. License Activation Service (LAS) is the next generation technology for product activations across the suite of Citrix products. LAS will be the only way to activate and license Citrix on-premises components after April 15, 2026. To remain supported, your Scout deployment must be on a LAS compatible version. The minimum required Scout version that is LAS compatible is Scout 15 2508 LTSR. Should you have questions or concerns, contact Customer Care. Citrix might limit or suspend your Citrix Maintenance for non-compliance with these requirements without liability in addition to any other remedies Citrix might have at law or equity. These requirements don't apply where prohibited by law or regulation.

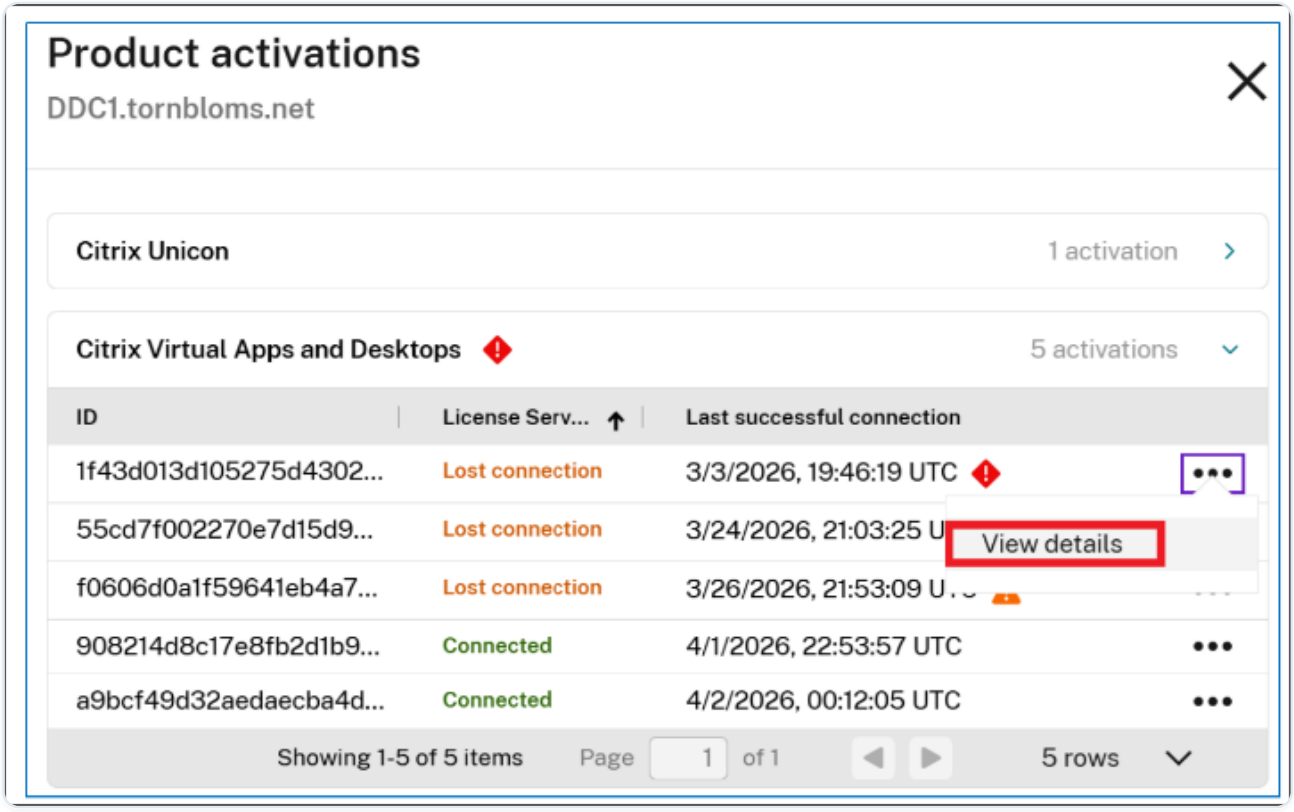
Troubleshooting

In the Citrix Cloud, navigate to **Licensing -> licensed deployments -> license Servers**

Click the expand button **>** across Citrix Unicon to see the list of Unicon Scout that were activated.



Click **View details** across the activations to see the granular details.



DeploymentId: The deployment ID

InstanceId: The Scout Server instance id

Hostname: The Scout Server host name

Scout Server Telemetry information

- Scout Server Telemetry data

Scout Server telemetry data elements

Headers

KEY	DESCRIPTION	EXAMPLE
`ver`	The version of the telemetry event.	`1.0`
`id`	A unique identifier of the event.	`0924268E-F1B9-4FCA-942E-45E63DE050ED`
`type`	The event type used to identify feature usage events.	`Scout.Device.Count`
`prod`	Identify the originating product and deployment of the usage event.	`UniconScoutServer`
`prodVer`	Scout Server product version.	`25.08.0`
`st`	The timestamp when the event is generated by the Scout Server.	`2025-08-16T09:05:00Z`

Payload

FIELD	DESCRIPTION	EXAMPLE
`scoutServer`	The ID of the server sending the event.	`3E693C53-B290-4765-9B24-88F927E91459`
`deviceCount`	Count of devices in the Scout database	`42`
`serverCount`	Count of Scout Servers in the Scout database	`1`
`timestamp`	ISO formatted time of event	`2025-08-16T09:05:00Z`

Security Considerations and Best Practices for running Citrix Unicon Scout

Security Considerations and Best Practices

Introduction

Threat Analysis

Modern IT systems face a range of security threats that often target communication paths between users and services. One major threat category includes man-in-the-middle attacks, in which an attacker secretly intercepts or manipulates communication between two parties. Such attacks can be carried out through techniques like network spoofing, ARP spoofing, the exploitation of unsecured networks, or by leveraging malware already present on a device.

Another significant threat involves backdoor and rootkit attacks, where attackers exploit hidden or poorly secured access points within a system. Once a backdoor is found or planted, a rootkit can be installed to gain persistent, unauthorized control over the system, making detection and removal difficult.

Closely related are viruses and Trojan horses, which typically rely on backdoors to infiltrate systems. Trojans disguise themselves as legitimate software, tricking users into installing them, while viruses spread across systems and networks. Social engineering, especially through deceptive emails, remains a common method for delivering these malicious programs to unsuspecting users.

Additionally, malware and ransomware pose a substantial risk. These programs are often installed intentionally by users or administrators who believe they are deploying legitimate tools. Hidden within them, however, is harmful code capable of encrypting data, disrupting operations, or opening additional attack pathways.

A further consequence of these attack types is data theft, a frequent outcome once a system is compromised. Stolen data may include sensitive corporate information, intellectual property, or personal data, and its exposure can significantly harm a company’s operations, reputation, and legal standing.

Security with eLux and Scout

Unicon's secure operating system eLux and the corresponding management software Scout are designed to minimize the potential threats and their impact. The security architecture includes the following features:

1 eLux does not store any personal information or sensitive information locally on a device. The system is kept read-only and only digitally signed software can be installed. This removes the risks for malware and ransomware, it also minimizes the possibility that a rootkit can be installed. 2 Enterprise communication offered by eLux uses state-of-the-art security measures to allow certificate based IEEE 802.1X network authorization. Certificates are stored in TPM chips on the eLux device. All internal network communication between Scout and eLux is secured by TLS. Users are warned when they connect to insecure networks. This mitigates the risk for man-in-the-middle attacks accordingly. 3 Rights management allows the Scout administrator to granularly adjust individual rights for the users of eLux devices. This can also include USB device handling. This reduces the chances of data theft by insider threats. 4 Applications and the eLux system are sandboxed and shielded to minimize the risk to security breaches and backdoor attacks. Additional sandbox features are available for all user applications and browser content. 5 eLux is a secure Linux system that adheres to Linux security standards. CVEs are patched and packages are updated accordingly when necessary as well as in regular updates. 6 Further security measures are also supported by eLux like smartcard authentication, running browser applications in an encapsulated kiosk mode and more.

Layers of Protection

Security in eLux is based on several layers of protection. Even if one defense is breached, the multitude of layers allow to protect the system.

SECURITY LAYER	ADVANTAGES
Custom Linux kernel build with own kernel configuration	Only necessary drivers and components are enabled
Custom Linux kernel build with own kernel configuration	Unneeded code is not integrated, thus minimizing a possible attack vector
Secure network communication	Support of modern network security, IEEE 802.1X
Secure network communication	Encrypted communication with Scout to avoid man-in-the-middle attacks

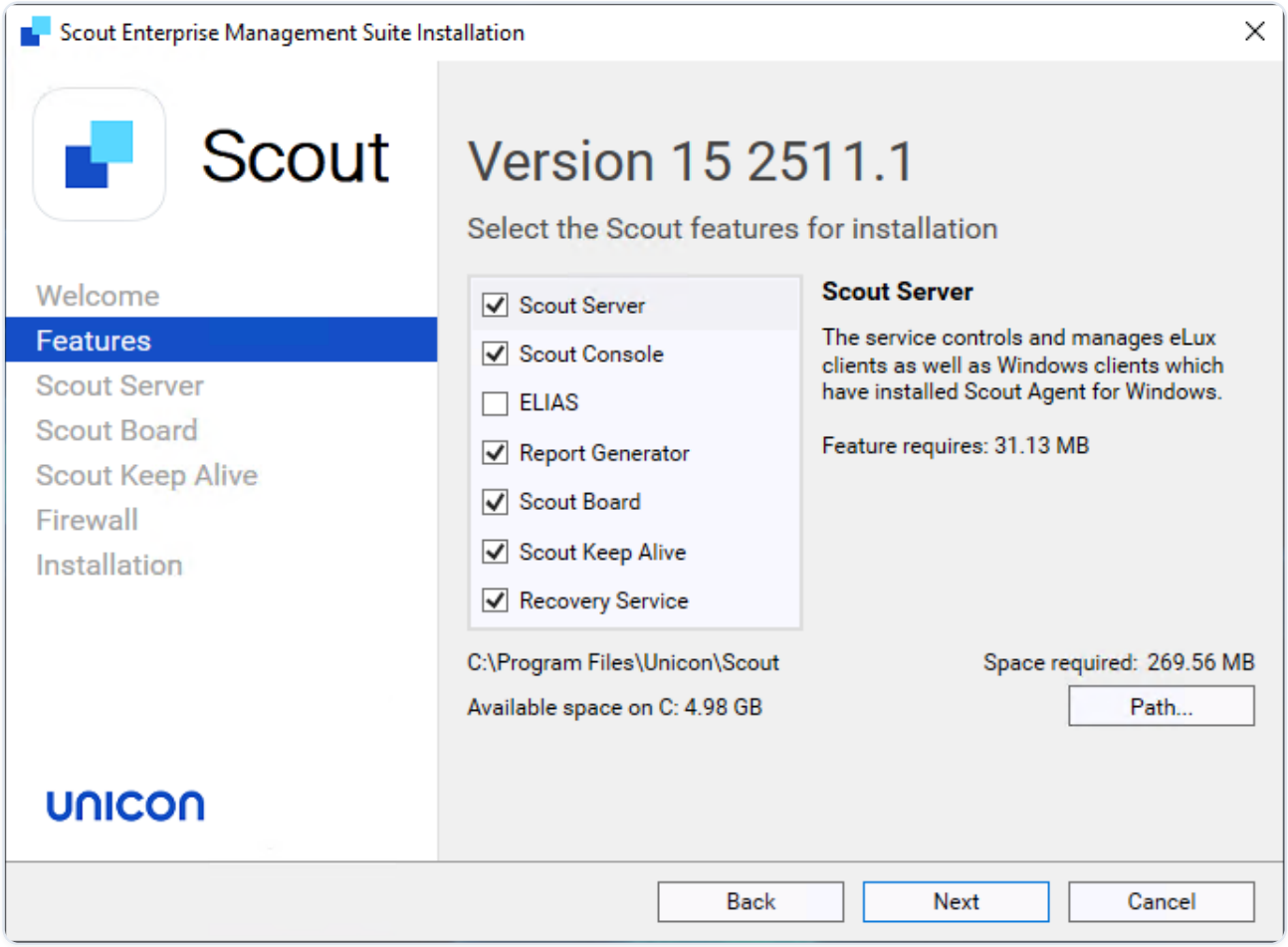
SECURITY LAYER	ADVANTAGES
Separation of privileged system access and unprivileged access	By default code will be executed unprivileged
Only a selected number of default components are installed	Components are hand selected from Ubuntu including Ubuntu backports of upstream security fixes
Only a selected number of default components are installed	Minimizing the number of packages, thus minimizing a possible attack vector
Read-only filesystem	Programs and processes cannot change, write or delete files
Read-only filesystem	Avoids unauthorized and unwanted changes
Encrypted filesystem	Content of filesystem cannot be retrieved with external tools
Encrypted filesystem	Key is secured in TPM chip that recognizes tampering with the system
Individual customer software images	Allows administrators to fine tune installed packages
Individual customer software images	Minimizing possible attack vectors to only keep the essential files
Installation of only signed packages	Possibility to only allow software packages from trusted sources
Installation of only signed packages	Avoid man-in-the-middle attacks
Regular security updates for the current release and long term support	secure packages, where security fixes were published
Additional security packages and features available	For scalable security, additional security components are available in the form of firewall, network access control, sandboxes and more
Secure login procedures include Active Directory, smart card, identity providers	Protection against unauthorized access

SECURITY LAYER	ADVANTAGES
Granular user rights granted by administrators	Administrators can control the rights of users very detailed to mitigate insider threats
No personal data is stored	Personal data and login data is never stored on the device
No personal data is stored	No data can be lost or compromised by the theft or loss of a device

Installation and Setup

Installation Scope

During the installation select *User defined* as installation scope. Only select the Scout features needed for your setup, unselect any features that are not necessary.



MS SQL Setup

Data at Rest

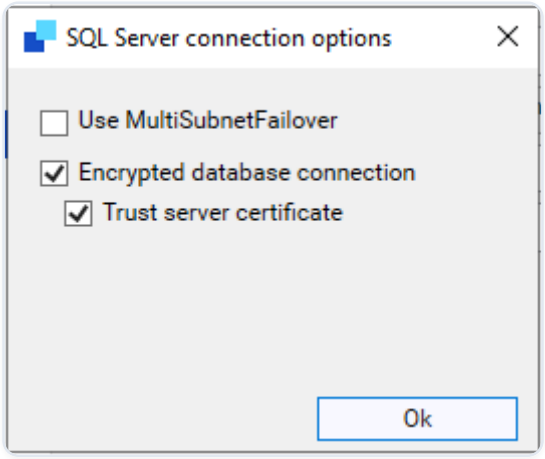
On the MS SQL setup ensure the encryption of the data at rest through Transparent Data Encryption (TDE). The performance impact through TDE should be neglectable for average and peak Scout database communication.

Transparent Data Encryption (TDE) encrypts the physical database files (MDF, NDF) and transaction logs (LDF) using AES or 3DES at the storage layer. This means the encryption happens on the server itself and is transparent to applications: data is encrypted before being written to disk and decrypted automatically when loaded into memory. Additionally, TDE protects backups, preventing offline attacks against copied database files or backup media. TDE uses a hierarchy of keys: a Database Encryption Key (DEK) stored in the database boot record, protected by either a server certificate or an asymmetric key stored in an external key store such as an HSM. This prevents an attacker who obtains database files from reconstructing the plaintext content.

Data in Transit

MS SQL Server protects data in transit using industry standard TLS (Transport Layer Security). When TLS is enabled for a SQL Server instance, all communication between clients and the server —queries, results, login credentials—is encrypted before transmission across the network. This prevents network level attackers from capturing or modifying data exchanged between endpoints. TLS requires installing a trusted certificate on the SQL Server host and configuring SQL Server Network Configuration to enforce encryption.

In the Scout installation select *"Connection options..."* and select *"Encrypted database connection"*.



Scout and eLux

Data at Rest

The eLux system partitions are already mounted read-only. By default disk encryption should always be turned on. This is done by selecting the *"Encryption FPM"* in the BaseOS when configuring an eLux image. The key for the encryption will be stored safely in the TPM 2.0 chip. This ensures that all information stored on the device remains unreadable if the hardware is lost, stolen, or tampered with. TPM 2.0 provides a hardware-based root of trust, meaning the encryption keys are securely bound to the device and cannot be extracted or used elsewhere, which significantly raises the barrier for attackers.

Data in Transit

By default the data transfer between Scout and eLux is encrypted using at least TLS 1.2. To add additional trust the certificate should be verified when used. To ensure this, set the TLSVerifyOption according to the Scout Installation documentation, configure the correct trust level and certificates. TLS-encrypted communication is useful because it protects management traffic and device communication from interception, tampering, and impersonation. In the eLux and Scout environment, TLS ensures that configuration updates, commands, and status data exchanged between endpoints and the management server cannot be read or altered by unauthorized parties.

Connecting devices outside the corporate network should be done via a dedicated VPN setup to connect eLux with Scout or using the Scout Cloud Gateway (SCG). SCG establishes a built-in, certificate-based VPN tunnel for each endpoint, removing the need for traditional VPN solutions while still ensuring fully encrypted communication back to the Scout server. This allows remote, home-office, and mobile users to receive the same configurations, updates, and policies as on-premises devices with minimal user interaction. SCG also streamlines onboarding through automated authentication and certificate issuance.

Device Configuration

Network

Network settings rely on the given infrastructure. We recommend to set up eLux in an IEEE 802.1X network environment.

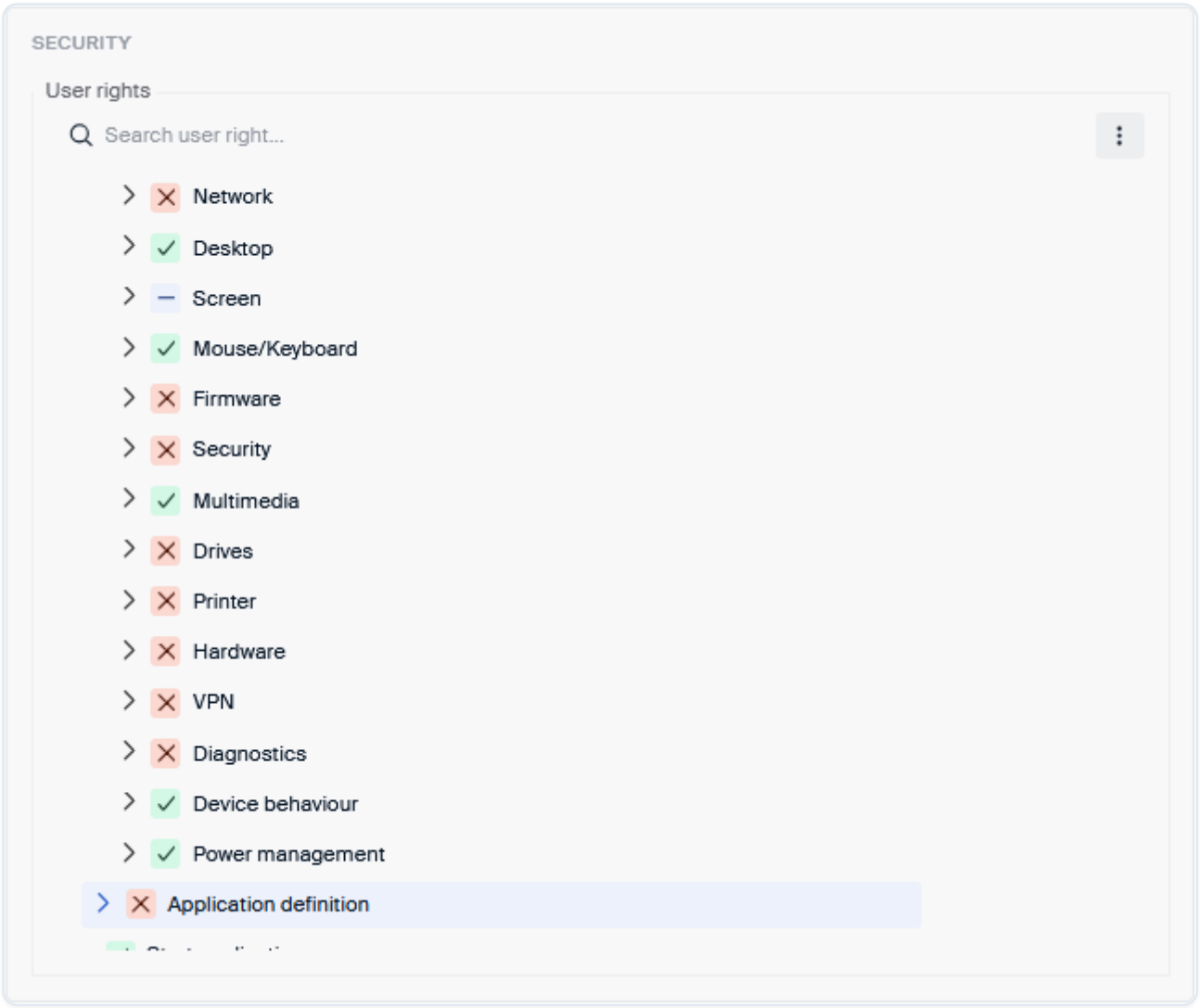
802.1X strengthens network security by ensuring that only authenticated and trusted devices can connect. Every connection attempt is validated before access is granted, which prevents unauthorized devices from joining the network and reduces the risk of internal threats. The protocol enables identity-based access control, enhances protection against malware and rogue devices,

and supports a modern Zero Trust security approach. This results in a more secure, well-regulated, and resilient network environment across both wired and wireless infrastructures.

Restrict User Access

When configuring the device, in the *Security* tab the user rights need to be set. Restrict as much as possible to ensure that users do not tamper with any internal system settings. We recommend restricting at least the following settings, these settings need to be set by the administrator on Scout:

- Network
- Screen Advanced
- Firmware
- Security
- Drives
- Printer
- Hardware
- VPN
- Diagnostics
- Application definition



User Authentication

Users should authenticate when using an eLux device. eLux supports a range of authentications, however, also a web-based or VPN-based authentication for a customer setup could be sufficient. This depends on the use-case and the customer environment.

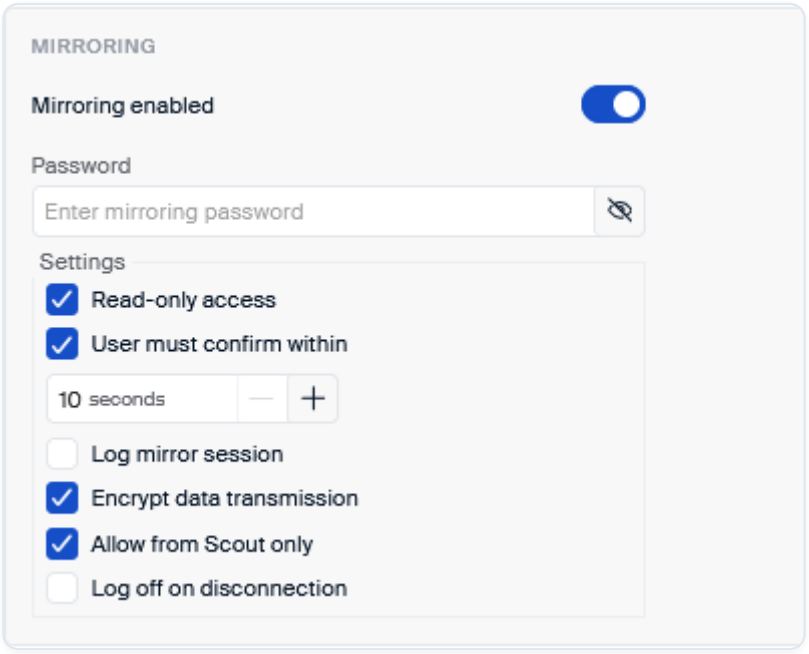
We recommend turning off *"Show last user"* when using one of the supported authentications on eLux. Showing the username of the last logged-in user is a security risk, especially on shared or publicly accessible systems. When eLux displays the last signed-in username, it gives an attacker half of the credentials needed to breach an account. This enables faster brute-force or targeted password-guessing attacks because the attacker no longer needs to guess the username.

Mirroring

Mirroring is a powerful tool in any helpdesk scenario, allowing the administrator to have direct access to the eLux device. To reduce any risks from man-in-the-middle attacks and inside threats we recommend to:

- Only allow the administrator read access to the target machine
- Ask the user to confirm with the mirroring of his device
- Allow access only initiated by this Scout server

It is also highly recommended to use mirroring only from Scout Board instead of the Scout Console. This ensures the most secure, fully encrypted WebRTC connection. When using the Scout Console instead, ensure to set the password and encryption options additionally. Note that due to protocol restrictions, only the first 8 characters of the password are used, thus only granting weak further protection.



Kiosk Mode

When setting up a kiosk mode for eLux, e.g. when used in a public setting, ensure to turn off *"Show last user"* in the settings. Use the automatic log-off feature to ensure no open session can be hijacked easily.

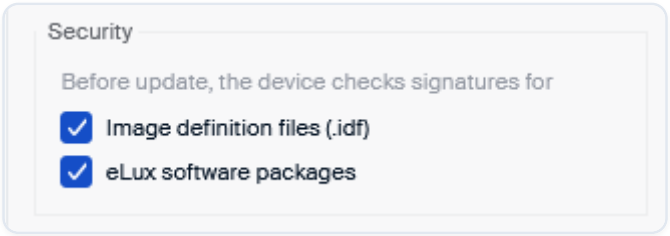
For a browser kiosk mode, customize the browser to remove the navigation bar and print menu if possible.

Firmware

For firmware updates, use secure web servers with HTTPS configured and username and password set for additional security.

ELIAS allows the selection of feature packages (FPM) on a very granular level. This should be used to minimize the image size to reduce the possible attack surface. For each package check the FPM selection and only select FPMs that are needed by the user.

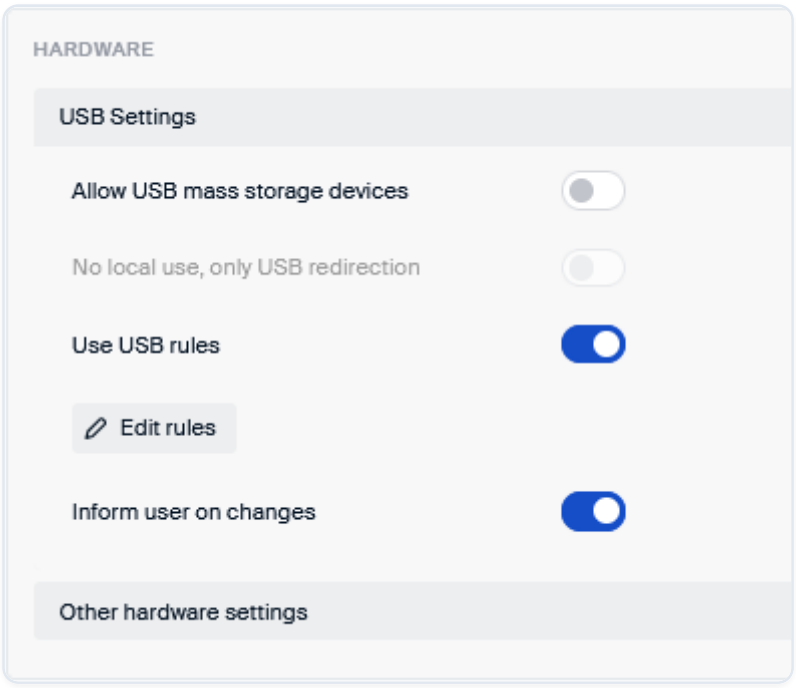
Do not install any packages from unknown origin. All packages by Unicon and Citrix are signed with the official private keys and can be checked with the included public certificates in ELIAS. This should be enforced by checking both the image file (IDF) and the packages for valid signatures.



Hardware

Restricting or disabling USB device access on eLux is a strong security measure because it prevents untrusted removable media from introducing malware, leaking data, or bypassing the system's controlled and centrally managed environment. This makes it possible to enforce USB blocking or read-only policies, which reduces exposure to malware, data exfiltration, and unauthorized device use.

We recommend disabling the use of USB mass storage devices. Instead, define USB rules to allow or deny access to specific USB device classes.



Diagnostics

Diagnostics should be used for trouble-shooting issues on eLux systems. On production systems it should be turned off to avoid accumulating log and system data in the temporary folders.

Sandboxing Applications

With the additional Firejail package local eLux applications can be run in an additional sandbox. This can be highly useful on eLux because it adds an additional, configurable isolation layer that strengthens the system’s architecture. Firejail allows applications, especially browsers and other internet-facing tools, to run inside tightly restricted namespaces and profiles, limiting their access to the file system, network, and system resources. This containment helps prevent malicious web content or compromised applications from affecting the rest of the device, complements built-in browser sandboxing, and supports flexible security scenarios such as switching profiles when devices move between networks.

We recommend using Firejail for scenarios where the Built-in Browser is used in a mobile use-case often switching between safe and unsafe networks.

Administrator Permissions

Hardening administrative access in Scout is essential because the management server has complete control over all registered eLux endpoints, their firmware, configurations, certificates, policies, and operational behavior. A compromise of Scout administrator access would immediately translate into a compromise of the entire endpoint fleet. For that reason, administrator hardening must be approached with strict least-privilege design, identity assurance, and configuration governance.

Scout supports multi-admin deployments and granular permissions, which must be used to separate operational responsibilities. Only a minimal number of individuals should receive super-admin privileges, as these roles can modify global configuration, certificate handling, onboarding flows, and security-critical settings. Delegated administrators should instead be restricted to narrow functional scopes such as device group configuration, OU-specific adjustments, or image deployment. It is recommended to give delegated administrators only access to predefined firmware configurations, predefined commands and command templates.

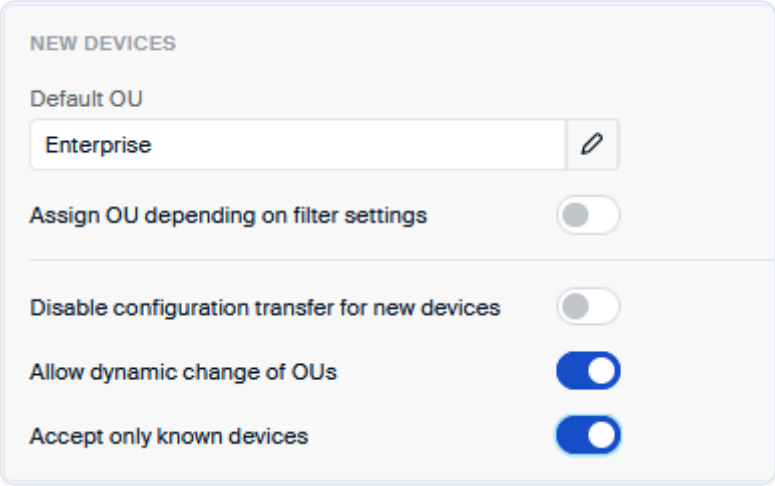
Scout supports authentication via identity providers like Entra ID and Okta using OIDC or SAML. These must be used in place of local accounts, enabling MFA enforcement, conditional access, centralized password policy, and identity lifecycle management. Relying on enterprise identity providers ensures that administrator access remains tied to corporate identity systems rather than isolated credentials that may not be governed by enterprise policies.

Scout Preferences

Devices

Setting a secure device password is important. The device password is used as a last line of defense to prevent man-in-the-middle attacks. It also is used to secure command execution with system rights on eLux. Therefore, only the master administrator should know the device password, other administrators should only use predefined commands if they need to run commands with system rights.

New devices should be sent to a quarantine OU with limited settings, e.g. no certificates, no applications. From there they can be moved to the right OU by the administrator. Only accepting only known devices further prevents malicious attacks where rogue eLux devices are connecting to Scout.



Predefined firmware configuration

Firmware updates for eLux and UEFI BIOS should be prepared by the master administrator. All other administrators should only be able to select from the list of predefined firmware updates. This prevents administrators from installing untested or potentially dangerous images, even if they have no malicious intent. This prevents administrators from deviating from hardened configurations, which could otherwise introduce vulnerabilities or weaken compliance. Firmware-level control also ensures that only validated OS versions, UEFI update processes, and secure package sets are used, reducing risk from untested images or misconfigurations; this aligns with the roadmap focus on secure firmware handling, logging improvements, and controlled image/firmware update workflows in Scout and Elias. Together, these restrictions ensure device integrity, prevent unauthorized or accidental downgrades or insecure configurations, and maintain a robust, enforceable security posture.

Predefined commands

Predefined commands as discussed above allow the master administrator to define a set of commands that other administrators can send to the devices even with system access. This prevents unauthorized, risky, or harmful actions from being executed on managed eLux endpoints. By limiting which remote commands an admin can send organizations minimize the risk of accidental misconfiguration, privilege misuse, or insider threats.

Command Templates

Command templates in Scout enhance security by tightly controlling how device-level actions can be executed, ensuring that administrators are limited to predefined, validated command structures rather than entering arbitrary or potentially harmful input. By providing centrally created command templates organizations ensure that only safe, vetted commands can be executed on devices, reducing the attack surface and ensuring consistent behavior across large deployments. This safeguards the environment from malformed parameters, enforces compliance with operational policies, and prevents delegated admins from performing unauthorized or risky low-level operations across the endpoint fleet.

Recovery Settings

We do not recommend to use PXE boot in general as it operates before security controls are active, uses historically insecure protocols (DHCP/TFTP), and is susceptible to rogue DHCP/TFTP servers, man-in-the-middle image manipulation, remote code execution vulnerabilities in UEFI PXE stacks and unauthorized image capture or injection. Use PXE boot only on trusted networks with strong security controls on the infrastructure level. However, when needed, PXE must only be used on trusted, isolated networks with strong security controls at the infrastructure level.

Short guides

A short guide provides information on specific procedures for our products that goes beyond the administrator's guide. Usually, short guides can be followed without having to read the full administrator's guide.

Short guide example topics are the migration between two versions of our operating system eLux® and recovery procedures for eLux devices.

All contents from the uDocs online documentation are also available for download in the PDF downloads section.

Introduction to Installation

System requirements

> **Note:**

> The following information refers to Scout Enterprise Management Suite 15.x.

Requirements for Scout Server and Scout Console

- Hard disk space 600 MB (only Scout Enterprise Management Suite®, the software container requires additional space)
- Scout Server and Scout Console: Microsoft Windows Server 2019, 2022 or 2025
- Scout Console only: Microsoft Windows 10 or 11

each including the relevant software updates provided by Microsoft at the time of installation

> **Note:**

> We recommend operating the Scout Enterprise Management Suite on a Windows Server system. To run the Scout Console, a Windows workstation is sufficient.

- Microsoft .NET Framework version 3.5 and Microsoft .NET Framework version 4.5.1 or later
- Suitable ODBC driver

Note

Please use the latest GA version of the Microsoft ODBC driver, as more recent versions might need the latest drivers. Scout looks for the ODBC driver in this sequence:

- ODBC Driver 18 for SQL Server (since 15.2309, 2023)
- ODBC Driver 17 for SQL Server (since 15.9.1000, 2020)
- ODBC Driver 13 for SQL Server (since 15.9.1000, 2020)
- ODBC Driver 11 for SQL Server (since 2015)
- SQL Server Native

One of the drivers must be installed.

Requirements for the database system

- Microsoft SQL Server 2019, 2022
- or for small installations: MS SQL Server Express

Requirements for the eLux® container

– applies only if you use the legacy ELIAS, user-defined installation –

- `FTP` or `HTTP` server, locally installed or via network
- The required space depends on the number of provided operating system versions. For the current eLuxContainer installation, for example, we recommend an available disk space of 5 GB minimum.

Generally, the minimum requirement can be obtained from the size specified for the AllPackages archive on our myelux.com portal.

Requirements for ELIAS 18 and MongoDB

– applies only if you use ELIAS 18, to be installed separately –

- Hard disk space 30 GB (depending on container installations)
- For each eLux release or container, 5 GB disk space should be precalculated
- RAM 8 GB minimum
- Microsoft Windows Server 2019 or later, 64-bit version

Microsoft Internet Information Service (IIS) 8.0 or later versions including WebSocket Protocol for automatic page refresh

Important note

With the Elias 18 Version 2511, a MongoDB Update has been introduced:

- When updating from a previous ELIAS 18 version, the MongoDB database must be updated using the official MongoDB migration guides.
- Supported MongoDB versions for ELIAS 18 2511 are versions 6, 7, or 8.
- The required migration steps depend on the MongoDB version selected.
- Ensure a complete database backup is available before performing the update.
- For fresh installations, MongoDB version 7 is installed by default.

Requirements for the web applications Scout Board, ELIAS 18 and Scout Cloud Gateway

See also system requirements for Scout Cloud Gateway in the SCG guide

- Web browser / minimum version
  - Mozilla Firefox ⇒ 136
  - Google Chrome ⇒ 137
  - Microsoft Edge ⇒ 127

> **Note:**
>
> Some functions use pop-up windows. Make sure that pop-up windows are not blocked by your browser. This function can usually be found in the browser settings under **Privacy** or **Security**.

- Screen resolution Full HD

For support periods and the compatibility matrix, see the whitepaper **Releases, Lifecycles and Compatibility**.

System limitations

There are no system restrictions known for any component of the Scout Enterprise Management Suite.

You can run additional services such as Citrix XenApp on the same system.

Database support

Scout requires database software such as Microsoft SQL Server.

Overview

Authentication to SQL Server

SQL Server users and application roles

Scout Server cluster

Number of ODBC connections

Overview

Microsoft SQL Server

You can use any Microsoft SQL Server version with available product support as an SQL database. We recommend that you create the required databases before installing Scout Enterprise Management Suite.

A minimum of one **Scout database** is required to manage the following data:

- Device configuration
- Application definitions
- Asset data (static)
- Server settings
- Administrator management
- Console management
- License information
- Transaction logging

A Scout database requires about 50 MB free disk space per 1.000 devices.

For Scout versions up to Scout 15 2204, the Scout Statistics Service was included as an optional component of the Scout Enterprise Management Suite® which required a dedicated statistics database. From Scout 15 2209, the Scout Statistics Service has been replaced by the Scout Keep Alive service. The keep alive data are now stored in the Scout database.

New databases can alternatively be created in Microsoft SQL Server during the installation process of the Scout Enterprise Management Suite, provided the required permissions are available.

Multiple database connections

By using the database connection editor, you can define various database connections for the Scout Console. You then can select one or more of the defined connections when starting the console. From your console, you can use multiple connections to different databases at the same time.

The database connection editor is provided on the Windows Start menu. For further information, see Database connections.

Database cleanup

Outdated data can be deleted using the **Database cleanup** feature. For further information, see Database cleanup.

Authentication to SQL Server

When you select the **Microsoft SQL Server** database type during installation, you can choose between two authentication methods for the database engine, **Windows authentication** and **SQL Server authentication**.

The authentication requires either an SQL user or a Windows user. Each of them must be a member of certain database roles in SQL Server. For further information, see SQL Server users and application roles.

METHOD	DESCRIPTION
Windows authentication	'Trusted connection': The user identity is confirmed by Windows.
	The Scout service must be run with a user account equipped with the relevant permissions in SQL Server (For Scout Enterprise Management Suite 15.5 and later

METHOD	DESCRIPTION
	versions, gMSA (Group Managed Service Accounts) are also supported).
	In the Scout installation dialog, specify the account name in the form `DOMAIN\username` (no case-sensitivity) Example: `INT\mmi`
	Note: For the Scout installation, either a user with the relevant permissions in SQL Server or the technical user account (Log on as a service) to be used must be logged on to the Windows server system.
SQL Server authentication	The username and password must refer to an SQL Server user account with the relevant user rights in SQL Server.
	In the Scout installation dialog, specify the SQL username and password.

If you choose to install the Scout Enterprise Management Suite® completely, three databases are required. You can configure access to the three databases in the following ways:

- If the same authentication method is used for all databases, you can configure different users for each database.
- If the same user is used for all databases, you can configure different authentication methods for accessing each database.

SQL Server users and application roles

Below description gives a rough overview of the permissions required in SQL Server, as well as how to use an application role. An application role increases security by assigning operational administrators only the necessary permissions for the duration of a session.

Create databases

We recommend that you create the required databases in SQL Server before installing the Scout Enterprise Management Suite.

Please note the following:

- To create databases in SQL Server, you need the SQL Server role **dbcreator** as a minimum.

- The database names may be freely chosen.
- The tables within the databases are created by the installation routine of the Scout Enterprise Management Suite.

Important:

Do not delete the original database when backing up and restoring! The unique database ID must be preserved for the initialization of the license database. For further information, see Troubleshooting.

Permissions for users of the Scout Enterprise Management Suite

SQL SERVER ROLE

For the use of the Scout Enterprise Management Suite, the server role **public** in general is sufficient. Only if users are to perform additional tasks in SQL Server, extended permissions are necessary. For example, restoring databases requires the **dbcreator** server role.

SQL DATABASE ROLE

On database level, for console-only users in standard environments, the database roles **db_datareader** and **db_datawriter** are sufficient. **db_owner** is needed, for example, for database activities related to updating to a new Scout version. The same applies for performing any configuration and maintenance activities on the database.

The users must be mapped to the default schema dbo.

Note:

For environments with SQL Server clusters, additional permissions such as VIEW SERVER STATE and VIEW ANY DEFINITION are required.

SQL application role

In the case of Windows authentication for SQL Server, users are authorized to also log on to the SQL Server Management Studio. For scenarios like this we recommend using a dedicated user group with limited database rights for console-only users.

Permissions to access SQL Server tables can additionally be controlled and restricted for all databases via a system-wide SQL application role. The name and password of the application role must be defined in the relevant database.

A console user for SQL Server then only needs the database role **public** in order to execute the stored procedure for activating the application role. Once the application role is active, the connection to SQL Server loses the user permissions and assumes the permissions of the application role. Other databases in which **guest** has been disabled will be inaccessible to the application role. The application role permissions remain active for the duration of a session.

Configuring console users and an application role in SQL Server Management Studio

The following instructions refer to the Scout database.

1. In SQL Server Management Studio, under **Security > Logins**, add a new user group (Example: **Console users**) with the following options:

Server role	Public
User mapping > Database role membership	db_datareader
Securables	Set non-required permissions to Deny.Example on server level: **Alter any database** > Deny

2. Below the Scout database, under **Security > Schemas**, add a new schema (Example: **Console users schema**). Under **Schema owner**, select the **Console users**.
3. Below the Scout database, under **Security > Roles > Application roles**, add an application role (Example: **Console users role**) with the following options:

Default schema	Console users schema
Password	Choose freely
Securables	Select your Scout database and set all permissions except **Connect** to Grant

Defining the application role in the Scout database

1. To specify the application role data in encrypted mode, in a first step, encrypt the name and password of the role. For further information, see Encrypting values.
2. Back in SQL Server Management Studio, for the Scout database, edit the **System** table.
3. Add one row for the name and one row for the password of the application role. Paste the encrypted values into the table:

SYSTEMID	PARAMNAME	PARAMVAL
…		
< *n*>	`RNameEx`	*< encrypted name of application role>*
< *n*>	`RPassEx`	*< encrypted password of application role>*

When the Scout Console is started, the fields are read and the access rights of the application role are set.

Scout Server cluster

If you use an SQL database, several Scout Servers can connect to the same Scout database concurrently. Concurrent Scout Servers enable failure load balancing as well as the option to configure load balancing (ManagerLoadBalancing).

Client devices that connect to a Scout Server receive a list of all currently running servers that access the shared Scout database.

FailureLoadBalancing

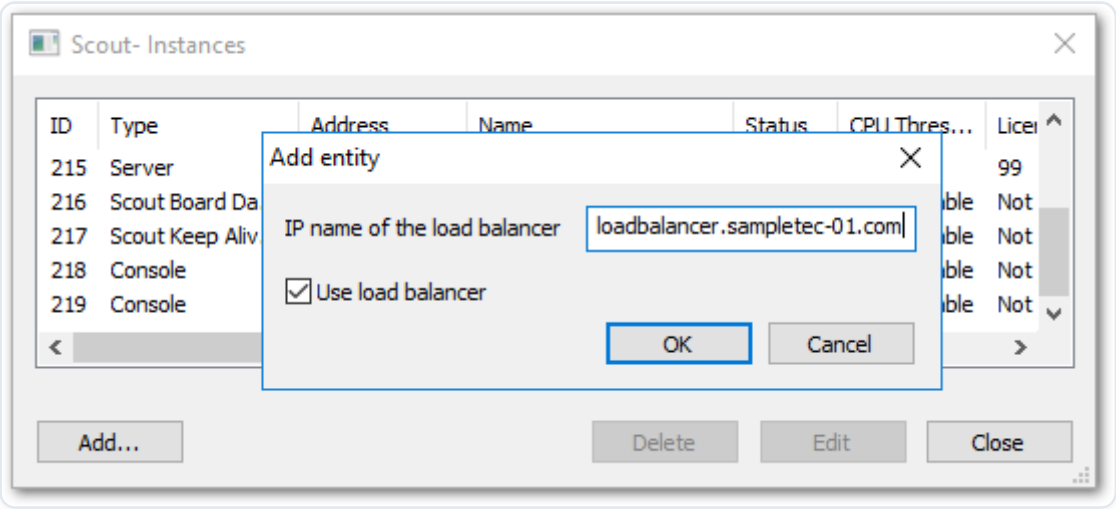
On start-up, a device tries to connect to the Scout Server it was connected to last time. If, however, that server is not available, it connects to the next server from the servers list which, subsequently, becomes the server the client tries to access by default from then on.

The FailureLoadBalancing mechanism restarts as soon as the client fails to connect to the same Scout Server.

ManagerLoadBalancing by dedicated load balancer

To define a dedicated load balancer, predefine the preferred manager address (IP address or name) for load balancing you want the devices to connect to:

- In the Scout Console, in **View > Scout entities**, add a new entity and select the option Use load balancer.



The load balancer entry refers to an existing load balancer pointing to the relevant Scout Server. The load balancer entry allows you to assign devices to a particular Scout Server without changing the Scout configuration.

The load balancer name is evaluated by the devices on each client restart.

Procedure:

- The device restarts
- The device connects to the load balancer and then is forwarded to the appropriate Scout Server

If, however, the Scout Server identified by the DNS entry `ManagerLoadBalancer` is not available, the FailureLoadBalancing mechanism described above is used and the client accesses the next server on the list.

ManagerLoadBalancing by DNS entry

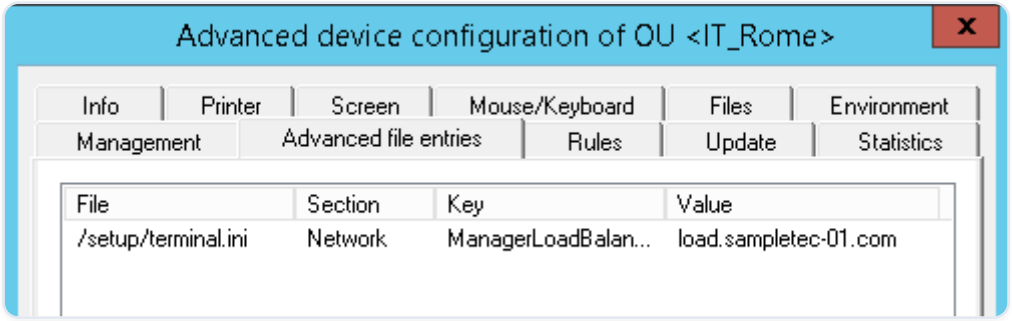
Alternatively, to predefine a preferred Scout Server you want the devices to connect to, you can use a DNS entry that you refer to in the advanced file entries.

- Use the Scout Console feature **Advanced file entries** for all devices, for an OU or an individual device:

FILE	`/SETUP/TERMINAL.INI`
Section	`Network`

FILE	`/SETUP/TERMINAL.INI`
Entry	`ManagerLoadBalancer`
Value	*< FQDN DNS entry>*

For further information, see Advanced file entries.



`ManagerLoadBalancer` refers to a DNS entry pointing to the relevant Scout Server. In a separate step, the DNS entry must be defined on the DNS server. The DNS entry allows you to assign devices to a particular Scout Server without changing the Scout configuration.

The `ManagerLoadBalancer` parameter is evaluated by the devices on each client restart.

Procedure:

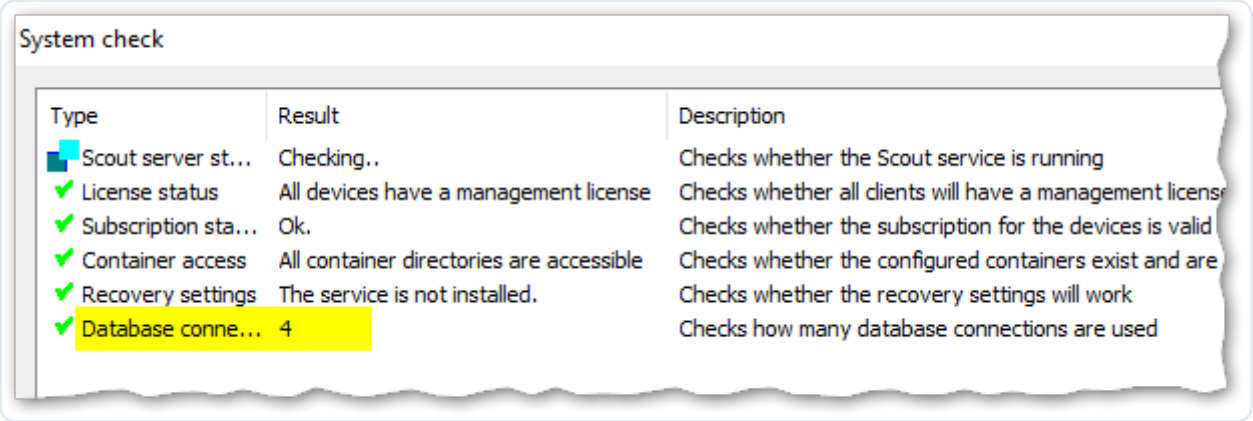
- The device restarts
- DNS entry `ManagerLoadBalancer` is resolved
- The device connects to the appropriate Scout Server

If, however, the Scout Server identified by the DNS entry `ManagerLoadBalancer` is not available, the FailureLoadBalancing mechanism described above is used and the client accesses the next server on the list.

Number of ODBC connections

The number of ODBC connections between the Scout Server and Scout SQL database is defined dynamically on start-up of the server service. Normally, for each CPU kernel, two ODBC connections are defined and used.

The number of database connections currently used can be viewed by using the system check feature (Scout Console **View > System diagnostics > System check**).



From experience, two ODBC connections for each CPU kernel lead to good results considering

- maximum communication performance between Scout Server and SQL database and
- optimum CPU utilization.

Static versus dynamic ODBC connections

You can specify a fixed number of ODBC connections to meet the particular system requirements of a Scout installation. For this, you must define the following parameter in the `eluxd.ini` configuration file of the Scout Server:

FILE	`%SYSTEMDRIVE%\USERS\PUBLIC\DOCUMENTS\UNICON\SCOUT\SERVER\ELUXD.INI`
Section	[ELUXD]
Parameter	`DatabaseConnections=`
Value	*n (n=1-128)*

> **Note:**
>
> Increasing the number of database connections manually, can lead to CPU overload.

> For further information on modifying `.ini` files, see Advanced file entries in the **Scout** guide.

Preparing for installation

Scout Server and the Scout Keep Alive service or Scout Statistics Service can be installed on the same machine or on different machines.

Make sure that the operating system is provided with up-to-date patches and the required software is installed. For further information, see System requirements.

Before you start the installation, read the following.

Registering Scout Server in the network

Opening ports

Preparing SQL Server databases

Permissions and certificates

Downloading software

Registering Scout Server in the network

To allow automatic registration of the devices, assign the IP address of the Scout Server via DNS or DHCP.

- DNS: Assign the host name ScoutSrv to the IP address. This is the easiest way.

Or:

- Configure one or more DHCP options. For further information, see DHCP configuration.

> **Notes:**
>
> - If you want to assign another Scout Server at a later time, use the Scout function **Client relocation**. Do not change the DHCP configuration while the devices are running.
> - If you do not use DHCP options for Scout, we recommend that you select the option **Ignore Scout Server DHCP options** in the **Device configuration > Network**.

DHCP configuration

-- optional --

> **Note:**

> DHCP options can only be applied to eLux® devices.

> A new device booting for the first time can retrieve the following information from a DHCP server:

- IP address or name of the Scout Server (option 222)
- List of Scout Servers (option 224)
- ID of the destination OU on the Scout Server (option 223)

This requires configuring the DHCP server via one of the two following methods.

In method 1 (recommended), you define a new vendor class, set the new options, and then apply the values. Method 2 uses the DHCP Standard Options 222, 223 and 224.

The following instructions are based on the DHCP manager of Windows Server 2012.

METHOD 1: DEFINING USER-DEFINED VENDOR CLASS

REQUIRES
DHCP server compliant with RFC 2132, supporting user-defined vendor classes. Otherwise use method 2.

1. Open the DHCP manager.
2. Select the relevant DHCP server, and then click **Action > Define**
3. Click **Add** to create a new class:

OPTION	VALUE
Display name	eLux NG
Description	eLux specific options
Code	ELUXNG
(in ASCII column)	The entry is automatically extended with the related hexadecimal number (45 4C 55 58 4E 47).

4. Click **Action > Set Predefined Options**. Then, in the **Option class** list field, select eLux NG.
5. To define one Scout Server, click **Add** and edit the new option as follows:

OPTION	VALUE
Name	Scout Server
Data type	String
Code	222
Description	Name or IP address of the Scout Server

6. To define more than one Scout Server, click **Add...** and edit the new option as follows:

OPTION	VALUE
Name	Scout Server list
Data type	String
Code	224
Description	Server names/IP addresses, comma-separated

7. To define a specific OU that you can assign new devices to, click **Add...** and edit the new option as follows:

OPTION	VALUE
Name	Scout OU ID
Data type	Long
Code	223
Description	ID of the destination OU

8. To assign the options, for the relevant DHCP server, select either the **Server Options**, the **Scope options** or the **Reservations**. Then click **Action > Configure Options... > Advanced**.

In the **Vendor class** list field, select . Select each option defined and enter its value into the **Data entry** field:

OPTION	VALUE
222 Scout Server	Name or IP address of the Scout Server
223 Scout OU ID	ID of the destination OU
224 Scout Server list	<Names or IP addresses of the Scout Servers, separated by comma>

METHOD 2: USING DHCP STANDARD OPTIONS

REQUIRES
The DHCP Standard Options [222], [223] and [224] must be available. Otherwise use Method 1.

1. Open the DHCP manager.
2. Select the relevant DHCP server, and then click **Action > Set Predefined Options...**. In the list field, select .
3. Click **Add...** to create the following Standard Options, as described for Method 1:
   - , String, 222
   - Scout Enterprise server list, String, 224
   - OU ID, Long, 223
4. To assign the options, for the relevant DHCP server, select either the **Server Options**, the **Scope options** or the **Reservations**. Then click **Action > Configure Options... > General**.

   Select each option defined and enter its value into the **Data entry** field:

OPTION	VALUE
222 Scout Server	
223 Scout OU ID	ID of the destination OU

OPTION	VALUE
224 Scout Server list	<Names or IP addresses of the Scout Servers, separated by comma>

DISABLING DHCP OPTION 12 AS SOURCE FOR HOST NAMES

If you have configured DHCP option 12 (host name), when connecting new devices, you can have the host names set via DHCP. To obtain the host name not via DHCP but from another source, such as the name template defined in the Scout Console, prevent the take-over from DHCP option 12. To do so, use a terminal.ini parameter:

- **File:** `/setup/terminal.ini`
- **Section:** `[Network]`
- **Entry:** `IgnoreDHCPHostname`
- **Value:** `true` (The default value is `false`.)

Assign new devices to specific Scout Servers

If you use more than one Scout Server, you can specify in advance to which of them a device will be assigned during the onboarding process.A filter (regular expression) on the MAC address is used as a criterion for the assignment.

The filter rules are defined in an `.ini` file, which is then transferred to the devices as part of a custom feature package in the image.This way, new devices receive the information to which Scout Server they are to connect, even before the first contact to Scout.

The `.ini` file, for example `scoutmapping.ini`, is a text file that is structured according to the pattern below:

```
[Mapping1]
```

```
 identifier=MAC   pattern=[AB][0-9A-F]
$   scoutsrv=scout1.sampletec-01.com   [Mapping2]   identifier=MAC   patte
rn=[CD][0-9A-F]
$   scoutsrv=scout2.sampletec-01.com   [Mapping3]   identifier=MAC   patte
rn=[EF][0-9A-F]$   scoutsrv=scout3.sampletec-01.com
```

Note the following:

- In Scout, the MAC address is displayed as a 12-digit number without separators (example: 901B0E01CE84)
- The filter must be a regular expression that filters on a substring of the MAC address.
- In a PostInstall script and PreUninstall script of the feature package, the `.ini` file must be referenced, example: `./setup/scoutmapping.ini`

For further details, please contact the Unicon™ support.

Opening ports

Open the following ports in the firewall: (provided the default ports are used)

PORT	TYPE	FROM	TO
1433	TCP	Scout Server	MS SQL Server
1434	UDP	Scout Server	MS SQL Server (Browser service)
22123	TCP	Scout Server (Scout Management / secure)	eLux endpoints
22125	TCP	Scout Server (Scout Management / TLS)	eLux endpoints
22124	TCP	Scout Server	Scout Keep Alive service
5900	TCP	Scout Console (Mirroring eLux® desktop)	eLux endpoints
5902	WebRTC (TCP, UDP)	Scout Board - web browser (Mirroring eLux desktop)	eLux endpoints
80/443	TCP	Clients (HTTP/ HTTPS)	Web server
80/443	TCP	Clients (Firmware-updates via HTTP/ HTTPS)	Web server

Note that, after the connection has been established, MS SQL Server assigns port numbers between 1024 and 5000 dynamically to its clients and that communication from 1433 to *ANY* must be allowed.

> **Note:**
>
> For detailed information on ports, protocols and network requirements, see Network requirements guide.
>
> The firewall service must be started.

Preparing SQL Server databases

Make sure that the following requirements are met:

- The Scout Server machines must be provided with suitable ODBC drivers to connect to the SQL database. For further information, see System requirements.
- We recommend that you create the required databases in Microsoft SQL Server before you install the Scout Enterprise Management Suite. Alternatively, new Scout databases can be created in Microsoft SQL Server during the installation, relevant rights provided.

  > **Note:**
  >
  > For all databases, the database tables are created automatically by the Scout installer.
  >
  > For further information, see Database support.

- SQL or AD users (**SQL serverInstance> / Security / Logins**) with relevant permissions for all databases

  For further information, see SQL Server users and application roles.

- The Browser service on the SQL server must be started.

Permissions and certificates

Permissions for the Scout Enterprise Management Suite® installation

- AD administrator account, member of the local administrators group on the target system

For information on how to authenticate to SQL Server, see Authentication to SQL Server.

> **Note:**
>
> The account of the installing administrator is the first active account of the Scout Console after the Administrator policies have been enabled.

Permissions for the web server

On the web server, one or more eLux® containers provide eLux software packages and images.

- Web Server (IIS) role or relevant permissions for the web server used
- Administrator rights on the root directory for the installing administrator
- Write access on the eLux container directory for all users that are allowed to create or modify images in ELIAS

Permissions for Scout Board

Logon via Active Directory is already supported in the Technical Preview. The administrator rights are applied as defined in the Scout Console under **Security > Activate administrator policies**.

SSL certificates for Scout Keep Alive service and Scout Board

The communication between eLux and the Scout Keep Alive service. For this, during installation, you are asked to specify a valid SSL certificate. Alternatively, you will be offered a self-signed certificate. For further information, see Certificate for Scout Keep Alive service.

For a secure connection to the Scout Board interface, also a valid SSL certificate is required. For further information, see Installing Scout Board.

eLux certificates for software packages

If you want to verify the signatures of the eLux software packages with ELIAS, you will need the relevant certificates:

- Download the package certificates from our myelux.com portal under **eLux Software Packages**.

Downloading software

Before you start the installation, download the `.zip` files required for the software you want to install:

- Scout Enterprise Management Suite®
- Scout Enterprise ELIAS 18 for creating individual firmware images
- eLux® software packages for the relevant operating system version
- USB stick image for the recovery installation of individual devices for the relevant operating system version
- …

1. Sign in to our myelux.com portal.
2. On the Download menu, choose the relevant software:

OPTION	DESCRIPTION / OPTION	DOWNLOAD
eLux	eLux Software Packages Latest operating system versions, LTSR (Long Term Service Release) or CR(Current Release)	Bundle: With `eLuxRP-x.x.x` AllPackages `-x`, you can install a container including all software packages in one step3 includes the installation file eLuxContainer.exe or import into ELIAS 18. Individual packages: Under **Release Packages**, all available software packages of an eLux version are provided for download.
eLux	eLux USB Stick Images Ready-to-use images for USB installation of the latest eLux version	< eLux CR version> Recovery Stick < eLux LTSR version> Recovery Stick includes the Citrix Workspace-App and the VMware Horizon client to connect against a backend
eLux	eLux Portable eLux on a USB stick, based on latest eLux RP version	eLux Portable (no installation needed)

OPTION	DESCRIPTION / OPTION	DOWNLOAD
Scout	Scout Enterprise Management Suite Latest LTSR and CR versions	< Scout CR version>* < Scout LTSR version> includes the installation file `ScoutInstaller.exe`
	ELIAS Latest ELIAS 18 version for creating and managing individual images The Scout Enterprise Management Suite alternatively includes the legacy ELIAS program.	ELIAS 18 < current version> for Windows ELIAS 18 < current version> for Linux
	Scout Agent for Windows for managing Windows devices	Latest version
Scout Cloud Gateway	Gateway to connect your eLux devices through the Internet (latest version)	Virtual Machine template (`.ova`) or Debian package
Tools	StickWizz	Latest version (StickWizz is also included in the eLux USB Stick images)
	eLux Builder Kit (eLux SDK VM) Development environment	Please contact sales(at)unicon.com
	Win2eLux Migrate from Windows to eLux	Win2eLux for 64-bit Windows Win2eLux for 32-bit Windows

3. To download the relevant file, click the file name or version number. *The software is downloaded in the form of* `.zip` *files.*
4. Unpack the `.zip` files.
5. Provide the installation files such as `ScoutInstaller.exe` on a local hard drive.
6. To create a recovery stick, connect an empty USB stick to a USB port. Start the `StickWizz.exe` application of the zip archive, and then write the image to the stick. For

further information, see Creating a USB recovery stick in the **eLux recovery procedures** guide.

Installation: Scout Enterprise Management Suite®

The Scout Enterprise Management Suite includes all features that are necessary or useful for managing a client infrastructure, in particular the Scout Server and the Scout Console.

To enable administrators to create individual firmware images, **one** of the following software components must be installed additionally:

- ELIAS 18

Here, you will import the eLux® software packages for creating client firmware images later on. For further information, see Installation: ELIAS 18.

or

- If you use the legacy ELIAS included in the Scout Enterprise Management Suite: (Choose user-defined installation and select it as a feature.)

eLux container with a compilation of software packages to be installed on a web server For further information, see Installation: eLux Container.

Features of the Scout Enterprise Management Suite

Installing Scout Enterprise Management Suite

After the initial installation

Unattended installation

Upgrading to newer versions

Modifying Scout Enterprise Management Suite

Uninstalling Scout Enterprise Management Suite

Features of the Scout Enterprise Management Suite®

Scout Enterprise Management Suite is the management solution for cloud devices, Hybrid Clients, mobile devices and PCs running the operating system eLux. In addition, Windows-based devices can be managed by using basic Scout management features.

Scout Enterprise Management Suite consists of several components. Most of the components listed below are included in the standard installation but can be optionally excluded when choosing custom installation.

COMPONENT	DESCRIPTION	INSTALLATION
Scout Server	The service controls and manages eLux devices as well as Windows devices on which Scout Agent for Windows has been installed.	`ScoutInstaller.exe`
Scout Console	User interface for the management of eLux devices and Windows-based devices on which Scout Agent for Windows has been installed Server communication only via database Multiple consoles can be managed with one Scout database.	`ScoutInstaller.exe`
Scout Board	Web user interface for the management of eLux devices and Windows-based devices on which Scout Agent for Windows has been installed	`ScoutInstaller.exe`
Recovery service	Customized TFTP service to realize a PXE recovery environment for eLux endpoints	`ScoutInstaller.exe`
ELIAS (The legacy ELIAS is no more included in the standard installation. To include it, choose User-defined.)	Legacy dialog program eLux Image Administration Service (ELIAS) for creating individual image definition files (`.idf`) for modular firmware updates of the eLux devices. The legacy ELIAS will be replaced by ELIAS 18.	`ScoutInstaller.exe`

COMPONENT	DESCRIPTION	INSTALLATION
ELIAS 18	New web-based platform-independent ELIAS application for creating individual image definition files (`.idf`)	separate(`EliasInstaller.exe`)
Scout Report Generator	Tool for creating freely definable reports across all currently existing devices, applications and OUs in the Scout ConsoleIs launched from the Scout Console	`ScoutInstaller.exe`
Scout Keep Alive service (Replaces the Scout Statistics Service from Scout 15 2209)	Windows service processing keep alive packets from eLux devices	`ScoutInstaller.exe`
Web API	Application programming interface for the management of eLux devices and for the management of Windows-based devices on which Scout Agent for Windows has been installed	`ScoutInstaller.exe`
Scout Command Interface	Command line interface for Scout commands	`ScoutInstaller.exe`
Scout Database Connection Editor	Tool for modifying database connection settings of the Scout Server and Scout Console	`ScoutInstaller.exe`

Further products

Scout Cloud Gateway	Cloud gateway with VPN backend for convenient connection of devices from	separate

	the Internet to a Scout infrastructure	
Scout Agent for Windows	Service providing an interface for Windows-based devices to be managed through Scout Enterprise Management Suite	separate

The features are described in the following guides:

- Scout Enterprise Management Suite: Configuration, control and management of the eLux devices using the Scout Console
- Scout Board
- Scout Cloud Gateway
- ELIAS
- ELIAS 18
- Scout Report Generator
- Scout Command Interface

Recovery procedures for eLux devices are described in the **eLux Recovery procedures** short guide.

> **Note:**
>
> To compose and use your own image files, in addition to the Scout Enterprise Management Suite installation, you need an ELIAS 18 installation. If you use the legacy ELIAS instead, you need an eLux container for the software packages, see Installing a container.

Installing Scout Enterprise Management Suite®

> **Note:**
>
> Before you start to install, make sure you have read the topics System requirements and Preparing for installation.

> When you run the Scout Enterprise Management Suite installer, we recommend to choose the standard installation scope which includes all features except for the legacy ELIAS. Then install the new web-based ELIAS 18 separately.

Alternatively, choose the `user-defined` option and select the features to be installed. Here you can still add the legacy ELIAS. For further information on the provided components, see Features of the Scout Enterprise Management Suite.

> **Notes:**

> - Run the installer from a local hard drive. Do not use a USB stick, CD-R drive or a network drive.
> - Anti-virus programs can have an impact on the installation. If required, disable the anti-virus program before you perform the installation.

1. Run the `ScoutInstaller.exe` file as administrator.
2. Select the installation language. Subsequently, read and accept the license terms.
3. Select the database type you want to use:
   - Microsoft SQL Server

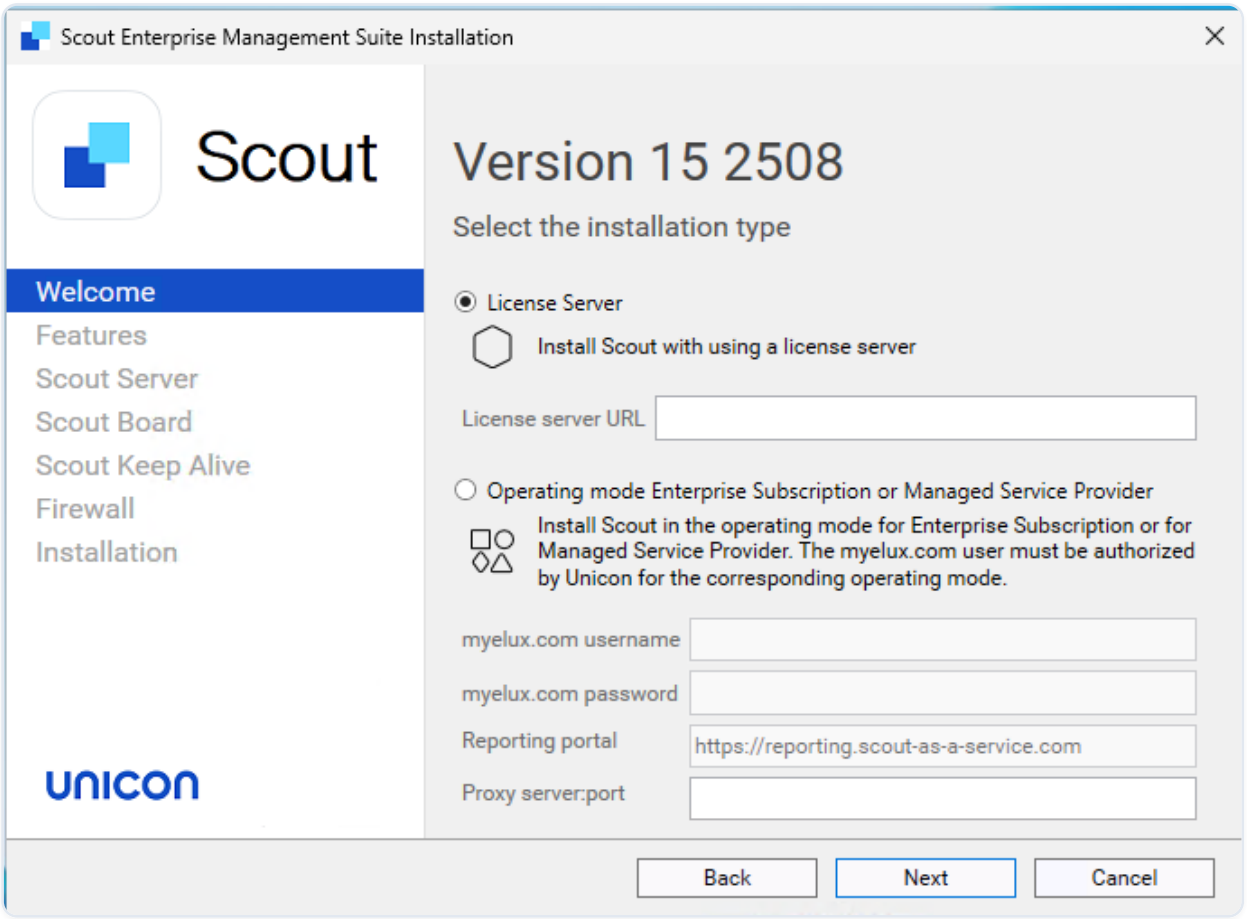
   For further information, see Database support.
4. Select the type of installation.
   - The operating mode Citrix License Server. This model is supported since the Scout version 15 2508 LTSR. After providing the hostname and the port (if needed), this license server will be used for receiving entitlements. You can continue without providing the Citrix License Server or by providing a not-yet reachable/configured one. In this case, Scout will go into a grace period, until a valid license server has been setup. The Citrix License Server can later be (re-) configured in the Scout Board UI at any time.

   > **Notes:** To see more information about the Citrix License Server and the licensing process, please follow the following links:

   > Citrix License Server

> Citrix License Manager

- The operating mode Managed service provider is only relevant for Managed Service Providers (MSPs) who want to offer Scout as a service and have an MSP account on our myelux.com portal.
- The operating mode Enterprise subscription is only relevant for customers that purchased an Enterprise Subscription and have an Enterprise Subscription account on our myelux.com portal.

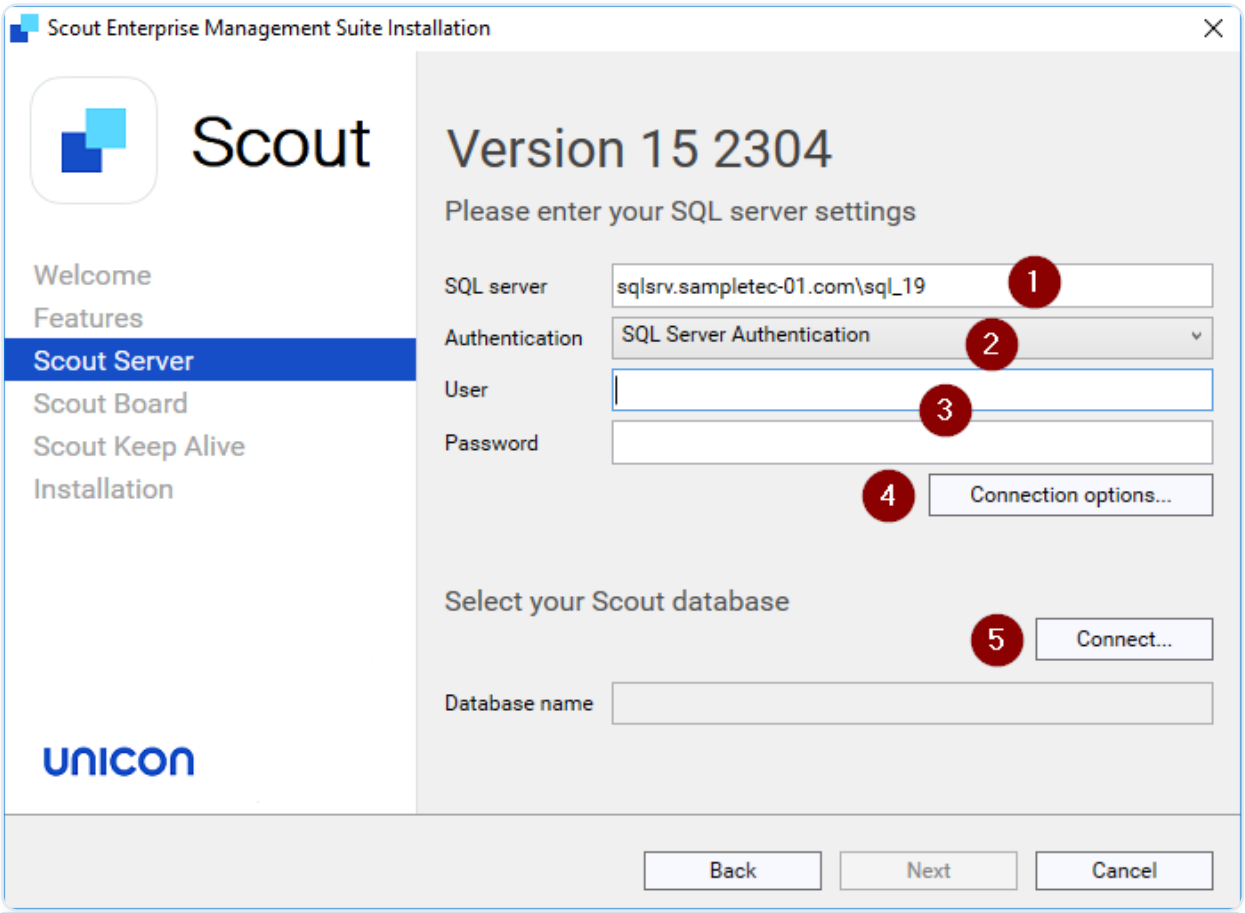


5. Choose the installation scope. To select individual features for installation or change the installation directory, select `User-defined`. For the standard installation, select `Standard`.
6. Specify the database connection data for your **Scout database**.

> **Notes:**

> Microsoft SQL LocalDB is not supported.

> Specify the required data to connect to your Microsoft SQL Server machine:



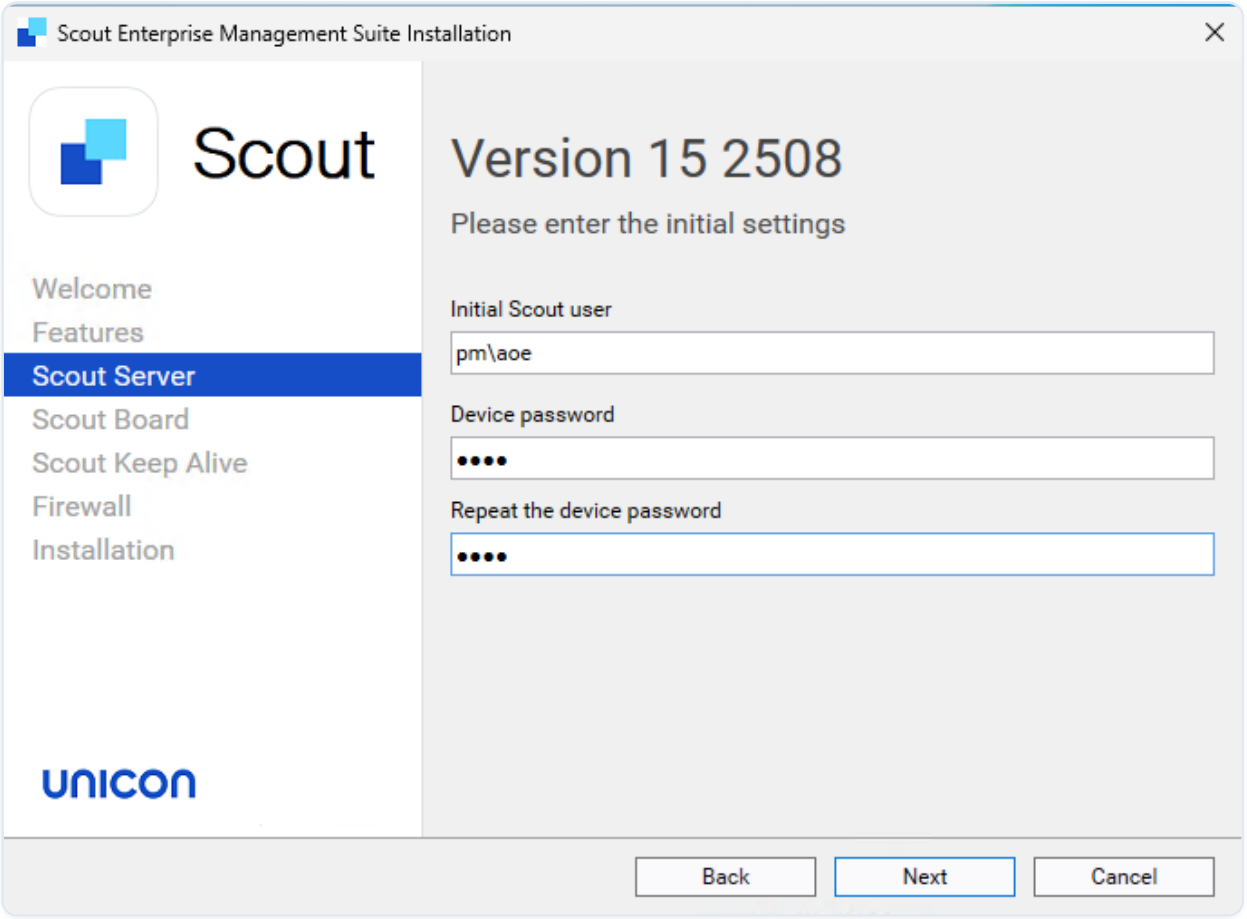
1	<*SQL Server machineinstance*> Example: `sqlsrv.sampletec-01.comsql_19`
2	`SQL Server authentication` or `Windows authentication` For further information, see Authentication to SQL Server.
3	SQL or Windows username and password for database access
4	SQL Server connection options: *For AlwaysOn Cluster: Faster reconnection after fail-over*, Use encrypted ODBC connection, and Trust server certificate (enabled by default)
5	Click to connect to the database server.

Click **Connect...**, and then, from the list-field, select your **Scout** database.

> **Note:**
>
> To show the databases on the specified SQL server, the SQL Server Browser service must be active.
>
> *Next to **Database name**, the selected database will be shown.*

> **Note:**
>
> If you choose "Windows Authentication" for the authentication method, all the services that need to connect to the database will be started by the given user and credentials. All other services will be started by the "Local System" user

7. In the dialog for initial settings, provide the initial admin user and the initial device password for eLux® endpoints. The initial admin user can be used to logon to the Scout Board UI.



8. In the next dialog, edit the **Scout Board** options. (optional component, from Scout 15 2209)

Enter the port number for the Scout Board service, and then the computer name Installing Scout Board in the **Scout Board** guide.

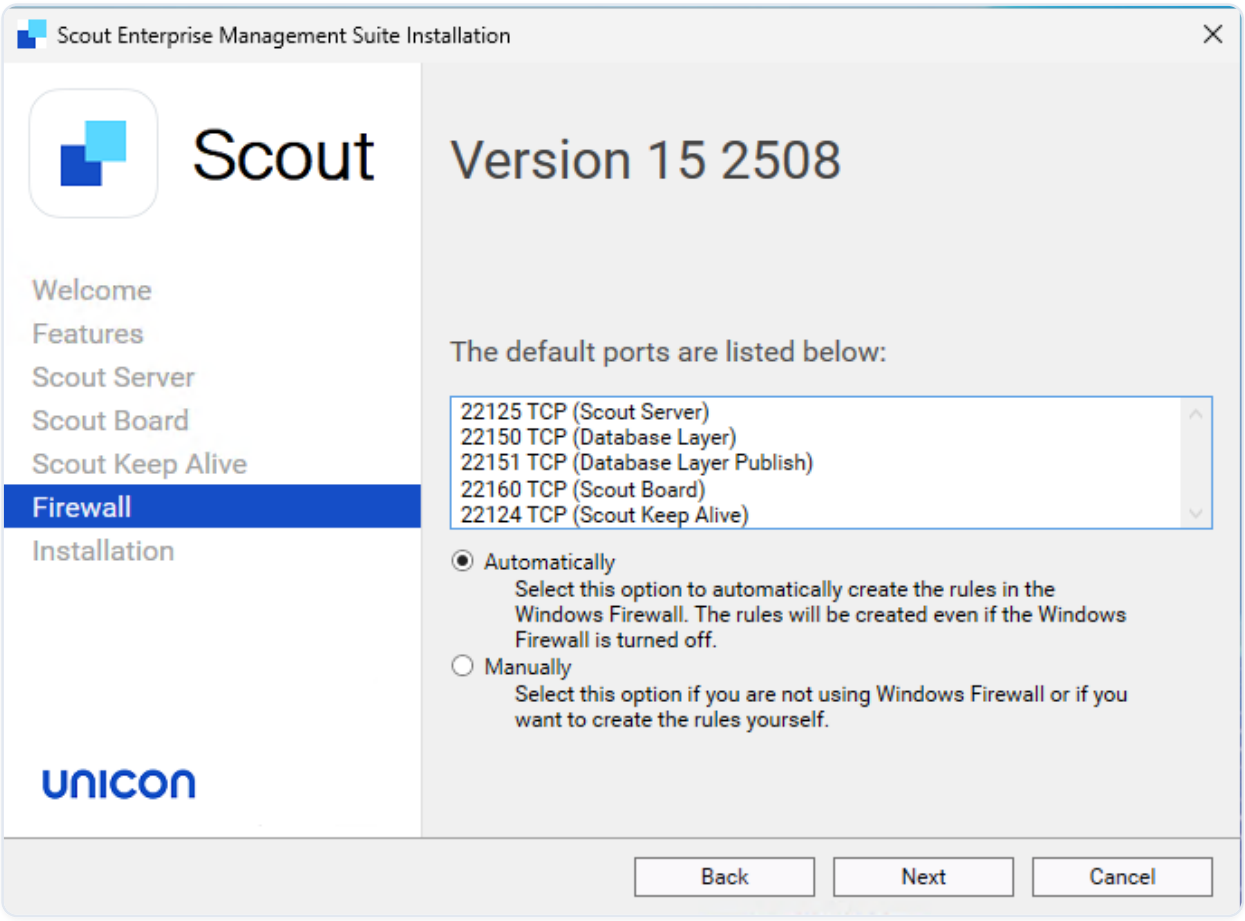
9. In the next dialog, edit the options for the **Scout Keep Alive service** database. (optional component, from Scout 15 2209)

   Specify the TCP port.

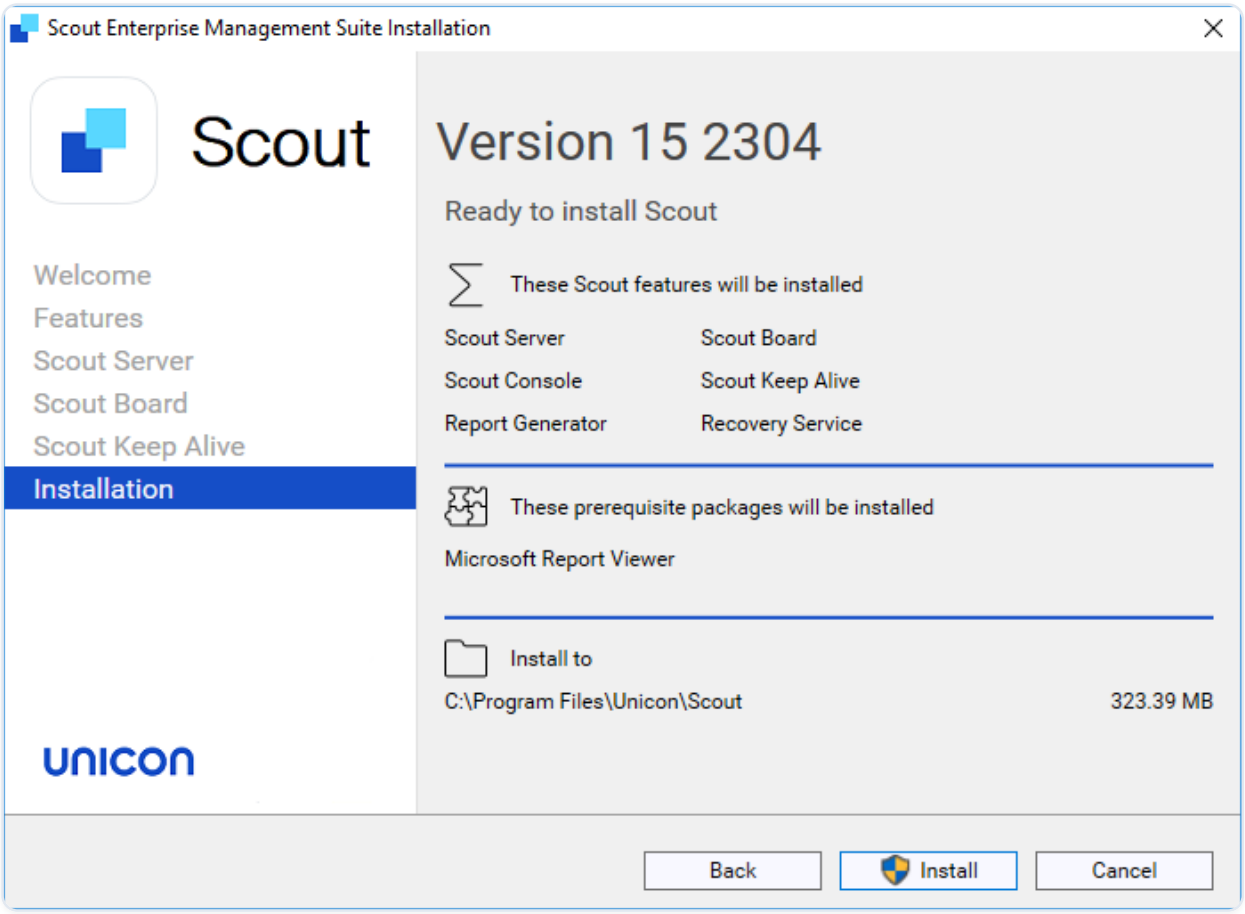
   With **HTTPS**, a secure connection to the interface is used. Specify a valid SSL certificate. Alternatively, let the system create a self-signed certificate, and then continue with the Scout installer.

   > **Note:**
   >
   > Click to update the list-field content and select the newly created certificate. Any certificates available are shown with their assigned friendly names. If there is no friendly name or it is assigned more than once, their serial number is shown.

10. In the firewall dialog, you can see all the ports that need to be activated, in order for Scout, Scout Board and Scout Services to work properly and for the communication with eLux endpoints. These firewall rules can be added automatically by the installer or you can activate them later manually.



11. In the last step, check the overview of all features to be installed.To start the installation process, click **Install**.



If required software components such as Visual C++ Redistributable or Microsoft Report Viewer are not yet available on the target system, they will be installed by the installer.

Once the installation is completed, you will find shortcuts for the Scout Console and Scout Board on the desktop. In the Scout group of the Windows Apps view, you will find all Scout applications including the Scout Database Connection Editor.

After the initial installation

The newly installed Scout Enterprise Management Suite® can be evaluated without licenses for up to 5 devices and up to 3 months without functional limitations. After 3 months or with more than 5 devices, licenses must be added.

After you have completed the installation with an empty Scout database, for your Scout Console the following is provided:

- A default account `administrator` with password `elux`
    - Change the password immediately to prevent unauthorized access:

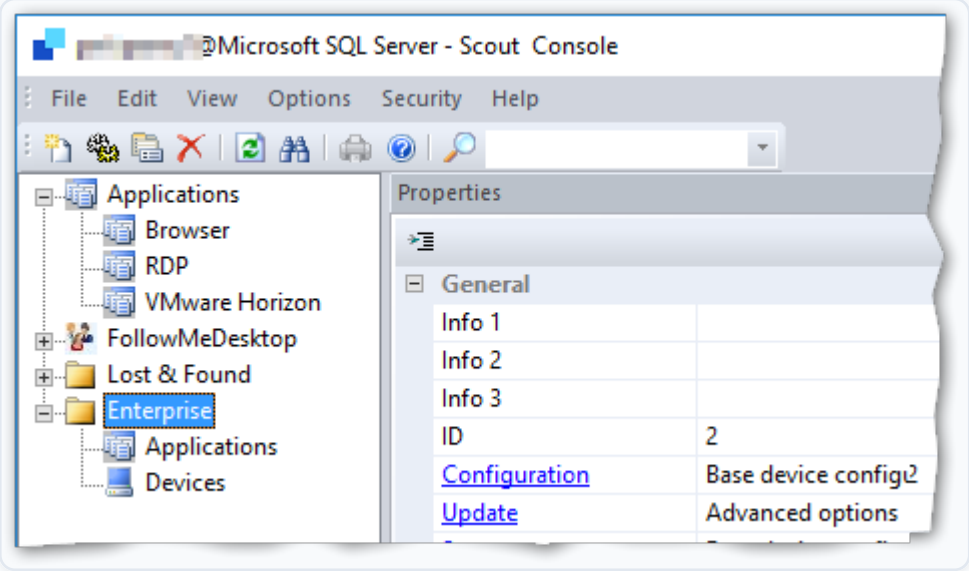
      -Change console password or

      -Activate administrator policies

- On the top level, three applications are predefined to connect to a back-end: **RDP**, **VMware Horizon** and the **Firefox** web browser.
  - To use one of the applications, modify the properties of an application definition and provide the relevant software by means of an IDF for the devices. For further information, see Defining applications.
- A top-level organizational unit (OU) named **Enterprise** is created.
  - Under the top-level OU, add further OUs corresponding to your corporate structure. For further information, see Organizational structure.

Note:

The hyperlinks above are related to the sections of the **Scout** guide.



Unattended installation

Note:

The following information refers to Scout Enterprise Management Suite 15.2302 and later. It includes the installation of Scout Board and Scout Keep Alive service.

Performing an unattended installation of the Scout Enterprise Management Suite®

- Run the `ScoutInstaller.exe` file along with the required command-line parameters: `"ScoutInstaller.exe" /s /v"<parameter>" /v"<parameter>"` You can add as many parameters as you like.

Creating a command line for unattended installation

Create a batch file for unattended installations by using a one-time manual installation according to the desired criteria.

1. Perform a manual installation of the Scout Enterprise Management Suite with the required components and options.
2. Open the log file created during the installation in a text editor: `%LOCALAPPDATA%\Temp\Scout_Enterprise_Management_Suite_<time stamp>.log`
3. Under `Silent install` command line, copy the command line created by the manual installation.
4. Create a batch file that contains the copied command line. *Passwords have been removed and must be entered manually.*
5. Replace the <`SET_PASSWORD`> strings for the database passwords by the relevant clear text passwords. If you want to use encrypted passwords, add_CRYPTED to the parameter names, see below.

List of parameters

The following tables summarize the available parameters and the values they can be given. On the left, **default values** are displayed in **bold**.

> **Note:**

> To encrypt passwords, you can use environment variables. For further information, see Encrypting values.

> **Parameters for /v**:

PARAMETER	DESCRIPTION
UCPROP_DBTYPE=`2`	`2` =Microsoft SQL Server

PARAMETER	DESCRIPTION
UCPROP_DBNAME=`Scout`	Name of the Scout database
UCPROP_DBSERVER=`sqlsrv.sampletec-01.comsql_12`	Database server (and instance) of the Scout database
DB_SCOUT_DB_AUTHENTICATION=`Windows Authentication`	Windows Authentication SQL Server Authentication
UCPROP_DBUSER=`Scout-Admin`	Only for SQL Server authentication:SQL username for the Scout database
UCPROP_DBPASSWORD_CRYPTED=`u[D Gqu[w_`	Only for SQL Server authentication:Crypted password for the Scout database, see eluxd.ini
UCPROP_DBPASSWORD=`My_Password`	Only for SQL Server authentication:Unencrypted password for the Scout database
UCPROP_SERVICEUSER	Only for Windows authentication:Windows username
UCPROP_SERVICEPASSWORD_CRYPTED	Only for Windows authentication: Encrypted Windows password
UCPROP_TRUSTSERVERCERTIFICATE=`1`	Trust the database server certificate
UCPROP_ENCRYPT=`0`	`0`: Use unencrypted ODBC connection, `1`: Use encrypted ODBC connection
UCPROP_MULTISUBNETFAILOVER=`0`	`0`: Default, `1`: Faster reconnection after fail-over (AlwaysOn Cluster)
UCPROP_DBCREATE=`0`	`0`: Scout database will not be recreated, `1`: Scout database will be recreated
UCPROP_LANGUAGE=`1`	Display language: `0`: German, `1`: English (If the parameter is not set, the language defined in the operating system is used.)

PARAMETER	DESCRIPTION
RUNSCOUTSERVICE=`true`	`true`: Scout services will be started during installation, `false`: Scout services will not be started

DEFAULT SETTINGS FOR THE DEVICES (BASE DEVICE CONFIGURATION)

UCPROP_DESKTOP_LANGUAGE=`en_US`	Display language for the desktop
UCPROP_KEYBOARD_LANGUAGE=`en`	Keyboard language
UCPROP_TIMEZONE=`US/Eastern`	Time zone

LICENSE SERVER

LSURL=`my.license.server`	Citrix License Server URL (the port can be added optionally `:port`)

> **Note:**
>
> If you want to install without the License server, use the parameter `/v"LS=0"`

FEATURES

ADDLOCAL=`Server,Console,Report`	Only the specified Scout features will be installed. `CommonFeature`, `Server`, `Console`, `Recovery`, `Elias`, and `Report`
INSTALL_SCOUT_FEATURE=`1`	`0`: Scout components will be skipped, `1`: Scout components will be installed (as defined in ADDLOCAL)
INSTALL_SCOUTBOARD_FEATURE=`1`	`0`: Scout Board will be skipped, `1`: Scout Board will be installed

INSTALL_SCOUTKEEPALIVE_FEATURE=`1`	`0`: Scout Keep Alive service will be skipped, `1`: Scout Keep Alive service will be installed

SCOUT BOARD

UCPROP_SCOUTBOARD_HOST=`scout.sampletec-01.com`	Computer name (FQDN) of the machine on which you want the database layer to run
UCPROP_SCOUTBOARD_PORT=`22160`	Port number of the Scout Board service
UCPROP_SCOUTBOARD_DBLAYER_ADDRESS=`tcp://scout.sampletec-01.com:22150`	Scout Board database layer address
UCPROP_SCOUTBOARD_DBLAYER_ADDRESS_PUBLISH=`scout.sampletec-01.com:22151`	Public address of the Scout Board database layer
UCPROP_SCOUTBOARD_AUTH_TYPE=`AD, OKTA_OIDC, OKTA_SAML, ENTRAID_OIDC, ENTRAID_SAML`	Preferred authentication type
UCPROP_SCOUTBOARD_IDP_ISSUER=`your-issuer-url`	OIDC issuer URL
UCPROP_SCOUTBOARD_IDP_CLIENT_ID=`your-client-id`	OIDC client id
UCPROP_SCOUTBOARD_IDP_CLIENT_SECRET=`your-client-secret`	OIDC client secret
UCPROP_SCOUTBOARD_IDP_INITIAL_USER=`your initial IdP user`	The initial IdP user
UCPROP_SCOUTBOARD_IDP_SSO_ENTRY=`your-sso-entry-url`	SAML SSO entry URL
UCPROP_SCOUTBOARD_IDP_CERTIFICATE_PATH=`your-certificate-path`	SAML certificate path

UCPROP_SCOUTBOARD_IDP_PRIVATE_KEY_PATH=`your-provate-key-path`	SAML key path
UCPROP_SCOUTBOARD_IDP_LOGOUT_URL=`your-logout-url`	SAML logout URL

FIREWALL SETTINGS

UCPROP_FIREWALL_ADD_RULES_SCOUTBOARD=`true, false`	Automatically add the firewall rules, or not
UCPROP_FIREWALL_ADD_RULES_SCOUTKEEPALIVE`true, false`	Automatically add the firewall rules, or not
UCPROP_FIREWALL_ADD_RULES_SCOUTSERVER`true, false`	Automatically add the firewall rules, or not
UCPROP_FIREWALL_SCOUTBOARD_PORT=`22160`	Firewall rule to add
UCPROP_FIREWALL_DBLAYER_PORT=`22150`	Firewall rule to add
UCPROP_FIREWALL_DBLAYER_PUBLISH_PORT=`22151`	Firewall rule to add
UCPROP_FIREWALL_SCOUTKEEPALIVE_PORT=`22124`	Firewall rule to add
UCPROP_FIREWALL_SCOUTSERVER_PORT=`22125`	Firewall rule to add

SCOUT KEEP ALIVE SERVICE

UCPROP_SCOUTKEEPALIVE_DBSERVER=`sqlsrv.sampletec-01.com`	Database server (and instance) for the keep alive data. These are normally stored in the Scout database, but can also be stored in a separate database.
UCPROP_SCOUTKEEPALIVE_DBUSER	

UCPROP_SCOUTKEEPALIVE_DBNAME=`Scout`	
UCPROP_SCOUTKEEPALIVE_DBPASSWORD	
UCPROP_SCOUTKEEPALIVE_AUTHENTICATION=`Windows`	`Windows`, `SQL Server`
UCPROP_SCOUTKEEPALIVE_SERVICEUSER	
UCPROP_SCOUTKEEPALIVE_SERVICEPASSWORD	
UCPROP_SCOUTKEEPALIVE_TRUSTSERVERCERTIFICATE=`1`	
UCPROP_SCOUTKEEPALIVE_ENCRYPT=`0`	
UCPROP_SCOUTKEEPALIVE_MULTISUBNETFAILOVER=`0`	
UCPROP_SCOUTKEEPALIVE_CERTIFICATES=`CREATESELFSIGN`	Certificate name or automatically created self-signed certificate
UCPROP_SCOUTKEEPALIVE_PORT=`22124`	Port number of the Scout Keep Alive service

FURTHER PARAMETERS

`/s`	The installation will be performed in unattended (silent) mode.
`/uninstall`	The Scout Enterprise Management Suite will be uninstalled.
`/l "%PUBLIC%DocumentsUniConscoutlog.txt`	The log file will be forwarded to the specified file.

Example for an unattended installation

```
ScoutInstaller.exe /s /v"UCPROP_DBTYPE=2" /v"UCPROP_DBNAME=Scout"

/v"UCPROP_DBSERVER=sqlsrv.sampletec-01.comsql_12"

/v"DB_SCOUT_DB_AUTHENTICATION=SQL Server Authentication"

/v"UCPROP_DBUSER=Scout-Admin"

/v"UCPROP_DBPASSWORD_CRYPTED=u[D``Gqu[w_" /v"UCPROP_DESKTOP_LANGUAGE=de"

/v"INSTALL_SCOUTBOARD_FEATURE=1"

/v"UCPROP_SCOUTBOARD_HOST=scout.sampletec-01.com"

/v"UCPROP_SCOUTBOARD_PORT=22160"

/v"UCPROP_SCOUTBOARD_DBLAYER_ADDRESS=tcp:/scout.sampletec-01.com:22150"

/v"UCPROP_SCOUTBOARD_DBLAYER_ADDRESS_PUBLISH=tcp://
scout.sampletec-01.com:22151"

/v"ADDLOCAL=CommonFeature,Server,Console,Recovery,Report"

/v"INSTALL_SCOUTKEEPALIVE_FEATURE=0"

/v"INSTALLDIR=C:/Program Files/Unicon/Scout"
```

Note:

When you perform an attended installation and select the required options, an eluxd.ini file will be created in the Scout Server directory. This file contains several Scout values that might be useful for unattended installation.

Performing unattended uninstall

- Run the following command: `ScoutInstaller.exe" /s /uninstall`

Upgrading to newer versions

An existing Scout Enterprise Management Suite® installation can be updated to the latest version in just a few steps.

1. Perform a full database backup of your Scout databases.
2. Download the latest version of the Scout Enterprise Management Suite as a `.zip` file from our technical myelux.com
3. Unpack the `.zip` file and provide the installation file on a local hard disk.
4. Run the `ScoutInstaller.exe` file as administrator.
5. Follow the instructions of the installation Wizard. Select your existing Scout databases.

Depending on the extent of new features, upgrading to a new version might cause longer runtimes when the Scout database is converted.

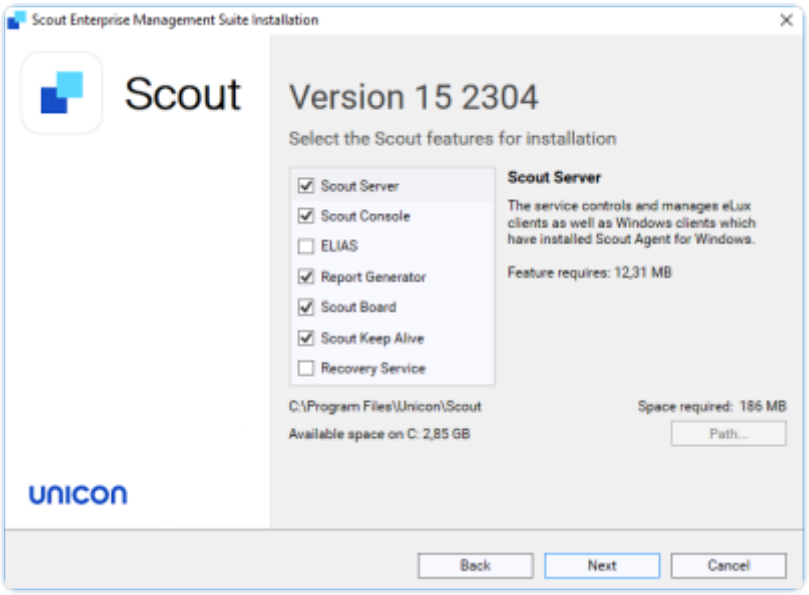
> **Important:**

> To upgrade to Scout 15 2204 or later versions, the upgrade base must be Scout 15.2.0 or later. Any older databases cannot be converted and will not be accepted by the installer.

Modifying Scout Enterprise Management Suite®

Installing additional features or uninstalling unneeded features

1. Use the control panel (Apps and Features) or run the `ScoutInstaller.exe` setup file as administrator.
2. In the Scout installation dialog, select the **Modify** option and click **Next**.



The installed features are shown with a checkmark.

3. Select the check boxes of the features you want to install. Clear the check boxes of the features you want to uninstall.

> **Note:**

> If you clear the selection of an installed feature, it will be uninstalled.

Repairing installation

1. Use the control panel (Apps and Features) or run the ScoutInstaller.exe setup file as administrator.
2. In the Scout installation dialog, select the Repair option and click Next.

The Scout Enterprise Management Suite is checked for missing files, shortcuts and registry settings and will be repaired if necessary.

Uninstalling Scout Enterprise Management Suite®

1. Use the control panel (Apps and Features) or run the `ScoutInstaller.exe` setup file as administrator.
2. In the Scout installation dialog, select the **Uninstall** option and click **Next**.

Installation: eLux® Container

> **Note:**

> If you use the web-based ELIAS 18, there is no need to install the eLux container. In ELIAS 18, you simply import the relevant software packages or the AllPackages bundle. For further information, see Importing software packages in the **ELIAS 18** guide.

> The eLux container is a compilation of software packages used to create the firmware (IDF) of a client. The administrator selects a subset of the software package pool to define an IDF (Image definition file) that can be installed on the devices.

The software packages can be provided via the container installation on a web or FTP server (legacy ELIAS) or via the ELIAS 18 web service in a MongoDB (locally or in combination with IIS).

For the main operating system versions , eLux RP 6 / 32-bit and eLux RP 5, dedicated containers are provided. Current eLux RP 6 / 64-bit versions are installed under `...eluxng\UC_RP6_X64`.

Installing a container

Exporting packages and containers

Importing packages into a container

Deleting packages from a container

Installing a container

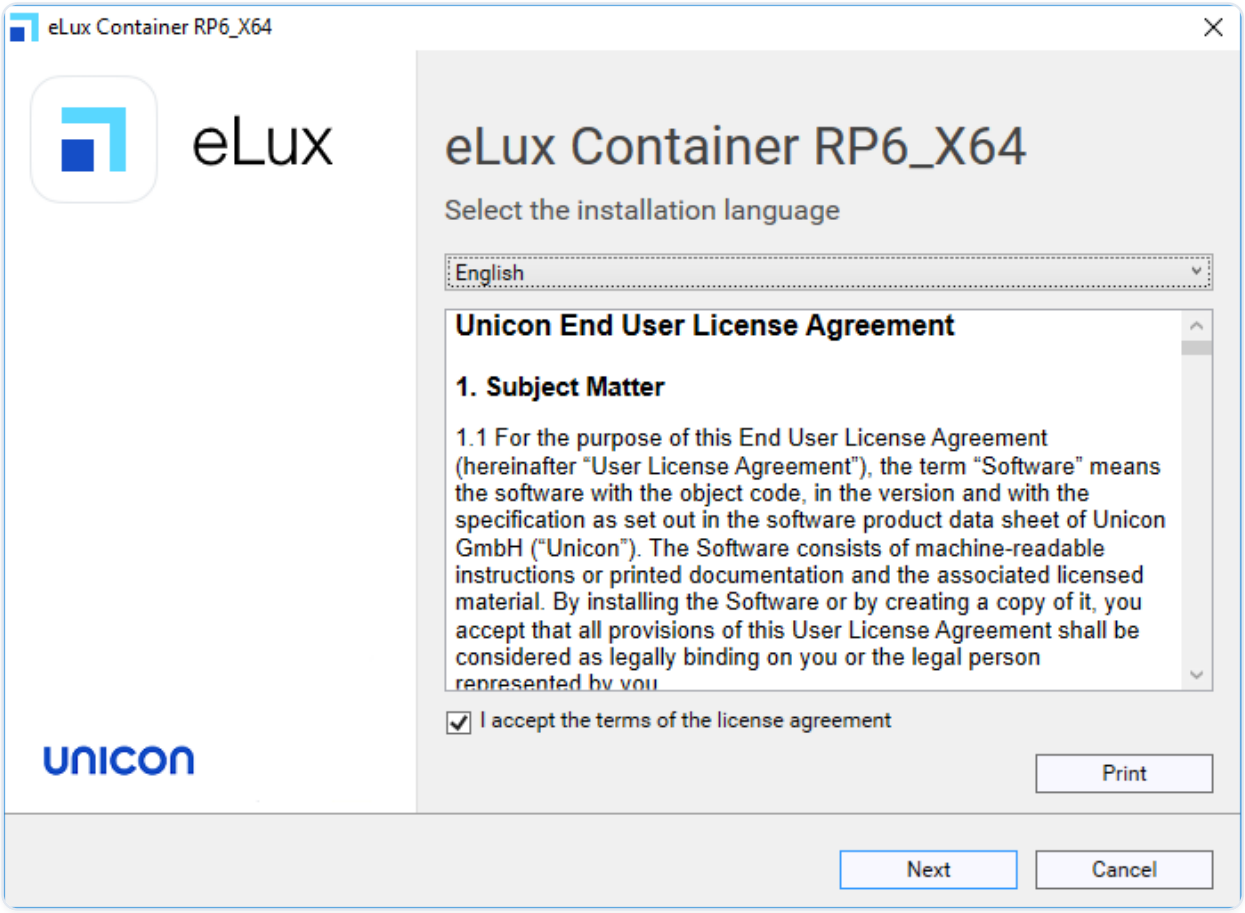
– applies only if you use the legacy ELIAS –

The following instructions show how to install an eLux® container including all software packages that are provided on our portal for the selected operating system version.

> **Note:**

> Before you start to install, make sure you have read the topics System requirements and Preparing for installation.

1. Run the `eLuxContainer.exe` (Extracted Allpackages archive file) as administrator from a local drive.



2. Select the installation language. Subsequently, read and accept the license terms.
3. Select the type of server that you use as the source server for firmware updates:
   - HTTP
   - FTP
4. Specify the access data for the FTP or HTTP server:

OPTION	DESCRIPTION	EXAMPLE
Root directory	Root directory of the server, local or network drive	`W:\inetpub\wwwroot` `C:\Program Files\inetpub\ftproot`
URL	complete URL to access the server	`http://update.sampletec-01.com` `ftp://update.sampletec-01.com`

5. If you use FTP, in addition, specify the logon data:

OPTION	EXAMPLE
Username	`anonymous`
Password	`elux@sampletec-01.com`

   The last dialog summarizes the required hard disk space.

6. To start the installation, click **Install**.

   On the web or FTP server, a `UC_RP6_X64` directory is created to hold the eLux RP 6 container. The `container.ini` file and the software packages (`.epm`, `.fpm` and `signature files`) are provided in the container. In ELIAS, you can create your own image now.

7. If the Scout Enterprise Management Suite is already installed, connect from the Scout Console to the eLux container in ELIAS: In the Scout Console, click **Options > ELIAS settings**... and specify the path pointing to the newly created container on the web/FTP server.

Updating to a later version

– applies only if you use the legacy ELIAS –

Updating the eLux® container can become necessary when the latest operating system version or fixes are provided, or when new versions of the client applications are available.

Updating to new major or minor version

New major or minor versions of eLux are provided as releases on our portal.

- Use the latest Allpackages bundle to install a container.

If you upgrade to a new major version, a new container (Example: `UC_RP6_X64`) is created into which the software packages of the new eLux version are installed.

If you update to a new minor version, the new software packages are added to the existing container. The existing software packages remain unchanged.

Updating individual software packages

1. On our portal, from the relevant container, download the required package as a `.zip` file.

   *The **Details** of each package in the container show the relevant enhancements and the history.*

2. Import the `.zip` file into your eLux container by using the ELIAS command **Container > Import Package**. For further information, see Importing packages into a container in the **ELIAS** guide.

Uninstalling an eLux® container

– applies only if you use the legacy ELIAS –

1. Use the control panel or run the `eLuxContainer.exe` setup file as administrator.
2. In the installation dialog, select the **Uninstall** option and click **Next**.

Installation: ELIAS 18

> **Note:**
>
> In ELIAS 18, you manage the eLux software packages for your firmware images. The ELIAS 18 installation replaces the container installation and the legacy ELIAS component of the Scout Enterprise Management Suite.
>
> ELIAS 18 is a web service that can be run stand-alone or in combination with Microsoft IIS. ELIAS 18 can be operated on Windows or Linux.

> **Note:**
>
> Other web servers can be used, but without configuration support
>
> ELIAS 18 is platform-independent and offers more functionality and convenience than the legacy ELIAS. For further information, see Overview in the **ELIAS 18** guide.

Installing ELIAS 18 / Windows

Installing ELIAS 18 / Linux

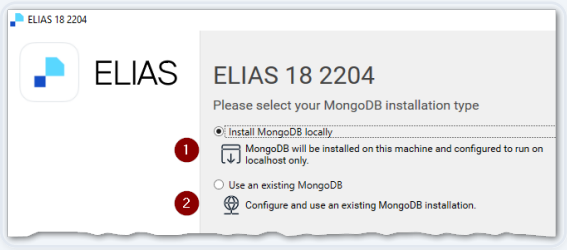
Starting ELIAS 18

Updating to newer ELIAS version

Installing ELIAS 18 / Windows

> **Note:**
>
> Before you start to install, make sure you have read the topics System requirements and Preparing for installation.

1. Run the `EliasInstaller.exe` file as administrator.
2. Select the installation language. Subsequently, read and accept the license terms.
3. Select whether to install MongoDB locally or to use an existing MongoDB installation.

	
1	The database will only be available on the local host.
2	To use an existing MongoDB installation, in the next step, specify the MongoDB connection data.

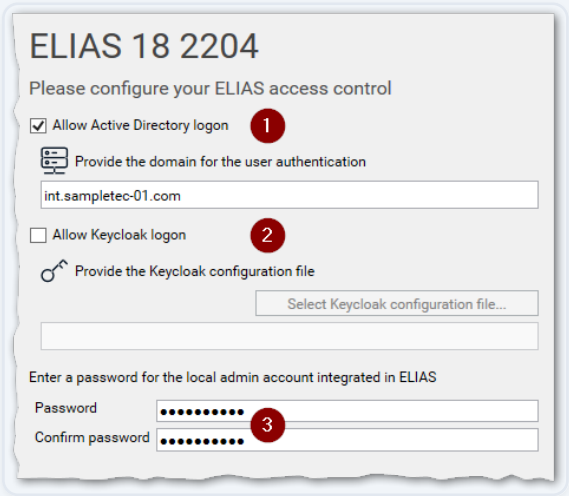
For an **existing** MongoDB installation, note the following:

- If you want your administrators to access the same MongoDB installation from multiple web servers, use different databases within the MongoDB installation.
- Specify the MongoDB server address with its port number, and the user credentials. Depending on the configuration of your MongoDB installation, specify additional options, for example, to connect to a specific replica set with timeout. MongoDB creates a URL connection string from the information provided. For further information, see `<https://docs.mongodb.com/manual/reference/connection-string/>`

> **Note:**

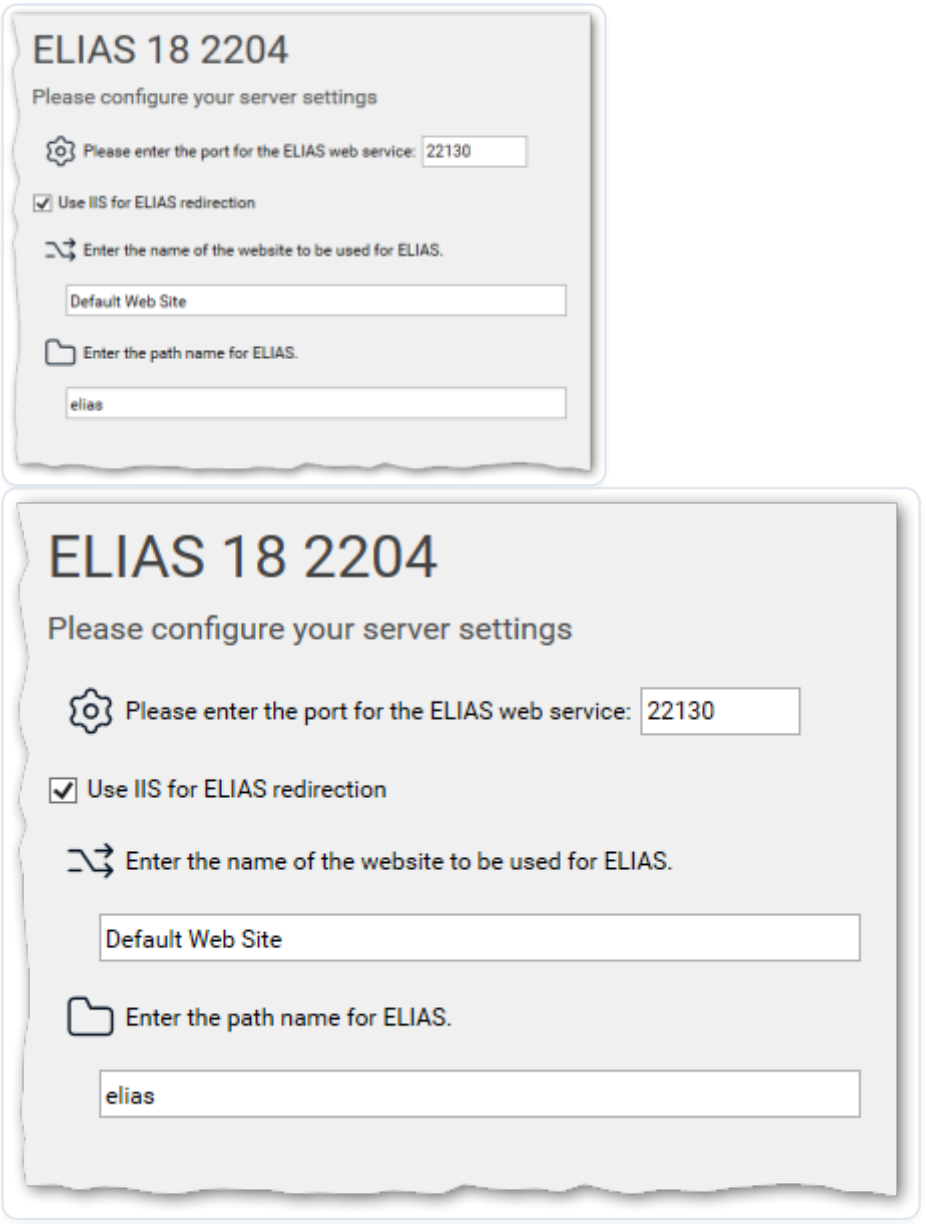
> From ELIAS 18 2104, all containers are stored in one database. This allows administrators to create additional containers later on, independent of the database user. The database prefix is no longer required.

4. In the next step, enter a name for your ELIAS database.
5. Then specify your ELIAS access control:

	
1	AD domain for Active Directory user authentication Example: `int.sampletec-01.com` Domain users need to be registered in a specific AD group. For further information, see Access management via AD in the **ELIAS 18** guide.
2	Configuration file of the Keycloak server for Keycloak authentication Example: `C:\install\ELIAS\keycloak.json` For further information, see Access management via Keycloak in the **ELIAS 18** guide.
3	Password for the local **admin** account

Note that the **Domain** and **Password** fields relate to two different logon types. For further information, see Access management and logon in the **ELIAS 18** guide.

6. Configure your web server settings:



If there is no IIS on your system, the ELIAS web service is installed on port 80 unless you specify another port number.

If an IIS already exists on the system, select the option Use **IIS for ELIAS redirection**. In this case, ELIAS is installed on port 22130 to avoid a conflict with web server port 80. To make ELIAS accessible via the default ports 80/443, it is registered using the reverse proxy procedure and the specified sub-directory. Specify the web site name and a path name for ELIAS.

To use HTTPS, an external web server such as IIS is required. The web site must be bound to port 443.

> **Note:**

> To configure your devices for updates, specify the path name defined here on the Firmware tab of the device configuration in the Scout Console

7. Confirm or change the installation path.
8. To start the installation, click **Install**.

*After the installation is completed, you will find an **ELIAS** desktop icon containing the URL of your ELIAS 18 installation. Double-click it to open ELIAS in the default browser.*

Cross-origin requests

After installation, cross-origin resource sharing (CORS) is allowed only from the installation server network (FQDN) for security reasons. This server is set by the installation routine under **allowedOrigins**. Administrators outside this network cannot access the installation.

- To allow access from other networks, add additional servers with their FQDN in the config.json file under allowedOrigins. Optionally add a port number after the server name with a colon. Separate multiple entries by comma.

Using the wildcard character, you can also configure global access. However, we do not recommend this for security reasons. Example: `"allowedOrigins": ["*"]`

Installing ELIAS 18 / Linux

– The following instructions refer to ELIAS 18 2209 or later versions –

ELIAS 18 can also be used in a Linux environment. A Debian package (.deb) tested with Ubuntu 20.04 is available for this purpose.

REQUIRES
A MongoDB database must be available either locally or remotely. The database requires sufficient hard disk space for container management, see also System requirements

1. From our myelux.com portal, under **Downloads > Scout> ELIAS**, download the provided Debian package of **ELIAS 18** < version> **for Linux**.
2. Install the Debian package by using appropriate package management tools (Debian/Ubuntu).

   Example: `sudo apt install ./elias-package.deb`

   The files are installed to /opt/unicon/elias.

3. To configure the connection to MongoDB and the web service (back-end configuration), edit the file `/opt/unicon/elias/server.json`: (This file remains local)

OPTION	DESCRIPTION	DEFAULT
"server"	MongoDB server name (as FQDN or IP address) For a local installation, use "localhost". Append the port number to the server name after a colon.	"localhost:27017"
"mongoUser"	MongoDB username If you do not use logon data, set an empty string.	""
"mongoPassword"	MongoDB password If you do not use logon data, set an empty string.	""
"mongoPasswordEncrypted"	Encrypted MongoDB password	false
"mongoOptions"	optional: Additional MongoDB options	
"adminPassword"	Encrypted password for the local admin account To encrypt passwords, use the **bcrypt** method with 13 passes.	"elias"
"adGroup"	AD group users need to belong to for logon For further information on available logon types, see Access management and logon in the **ELIAS 18** guide.	"ELIAS"
"logLevel"	Log level (debug, info, warn, and error)	"debug"

OPTION	DESCRIPTION	DEFAULT
"port"	Port used by the ELIAS API	"22130"
"iisWebsite"	IIS redirection website name, not used on Linux	—

Note:

To configure ELIAS for access via Keycloak, see Access management via Keycloak in the **ELIAS 18** guide.

4. For the front-end configuration, edit the file `/opt/unicon/elias/config.json`:

OPTION	DESCRIPTION	DEFAULT/EXAMPLE
"pollingInterval"	Interval in milliseconds for polling the API by the user interface	3000
"api"	Host name of the device ELIAS is running on (FQDN or IP address)	"`<Host name>`"
"domain"	AD domain for user authentication Domain users need to be registered in a specific AD group. To enable only the local admin account, set an empty string.	""
"redirectPath"	Any path that redirects to the API (for example, if Apache is used) Must end in api	"api"

This path must also be specified in the Scout Console, on the Firmware tab of the client device configuration.

"PROTOCOL"	EITHER "HTTP" OR "HTTPS"	"HTTP"
"`apiVersions`"	Must be set to	
"`base.database`"	Name of your ELIAS database If you create multiple containers, all of them will be stored in this database.2 from ELIAS 18 2104	"`ELIAS-Database`"
"port"	Port on which the ELIAS website is reachable externally	"8080"
"`allowedOrigins`"	List of URLs on which the ELIAS website is reachable externally (separated by comma) Example: "http://*<Hostname or IP address>*:8080"	

5. Restart the ELIAS service.

   Example: `sudo systemctl restart scout-enterprise-elias`

Reverse proxy for the web server

For performance or security reasons, you can configure a simple reverse proxy to handle SSL encryption and decryption. A configuration for **Apache** might look like the following:

Copy

```
<IfModule mod_ssl.c>

<VirtualHost *:443>

    ServerName elias.dev.sampletec-01.com

    ProxyPreserveHost Off

    ProxyRequests off
```

```
    SSLProxyEngine On

    <Proxy *>

        Require all granted

    </Proxy>

    ProxyPass / http://localhost:22130/

    ProxyPassReverse / http://localhost:22130/



# Insert your certificate file names:

SSLCertificateFile /etc/ssl/private/ca-certificate.crt.pem

SSLCertificateKeyFile /etc/ssl/private/ca-certificate.key.pem



SSLEngine on



# Intermediate configuration, tweak to your needs:

SSLProtocol             all -SSLv2 -SSLv3 -TLSv1 -TLSv1.1

SSLCipherSuite          ECDHE-ECDSA-AES128-GCM-SHA256:ECDHE-RSA-AES128-
GCM-SHA256:ECDHE-ECDSA-AES256-GCM-SHA384:ECDHE-RSA-AES256-GCM-
SHA384:ECDHE-ECDSA-CHACHA20-POLY1305:ECDHE-RSA-CHACHA20-POLY1305:DHE-RSA-
AES128-GCM-SHA256:DHE-RSA-AES256-GCM-SHA384

SSLHonorCipherOrder     off

SSLSessionTickets       off



SSLOptions +StrictRequire
```

```
# Add vhost name to log entries:

LogFormat "%h %l %u %t "%r" %>s %b "%{Referer}i" "%{User-agent}i""
vhost_combined

LogFormat "%v %h %l %u %t "%r" %>s %b" vhost_common



</VirtualHost>

</IfModule>
```

Customize the data such as server name and certificate files to your configuration.

We also recommend creating a second configuration file to redirect communication arriving unencrypted via port 80 to port 443. Example:

Copy

```
<VirtualHost *:80>

RewriteEngine on

RewriteRule ^ https://%{SERVER_NAME}%{REQUEST_URI} [END,NE,R=permanent]

</VirtualHost>
```

Again, customize the data to fit your configuration.

For the configuration files to be executed, proceed as follows:

1. Copy your **.conf** files to `/etc/apache2/sites-available`
2. Provide them to `/etc/apache2/sites-enabled` for example with `a2ensite <filename>.conf` or via symlink, for example ln `-s sites-available/<filename>.conf sites-enabled/<filename>.conf`
3. Restart the web server: `systemctl reload apache2`

Starting ELIAS 18

REQUIRES
ELIAS has been installed successfully. For further information, see Installation: ELIAS 18 in the **Installation** guide.

ELIAS is started with a URL pointing to your installed ELIAS web service.

Without IIS:

- In the web browser, type the following URL: `http://< host name>:< port number>` or `https://< host name>:< port number>`

  `< host name>` refers to the computer name or IP address of the computer ELIAS is installed on.

  `< port number>` is the port you have specified for the ELIAS web service

With IIS redirection:

- In the web browser, type the following URL: `http://< host name>/path` or `https://< host name>/path`

  `< host name>` refers to the computer name or IP address of the computer ELIAS is installed on / your web server.

  `< path>` is the specified ELIAS path name under your web site (elias in the example above)

> **Note:**

> The computer ELIAS is installed on provides an ELIAS desktop icon.

> For access from outside the network, see Installation: ELIAS 18 in the **Installation** guide.

Updating to newer ELIAS version

An existing ELIAS 18 installation can be updated to the latest version in just a few steps.

> **Note:**

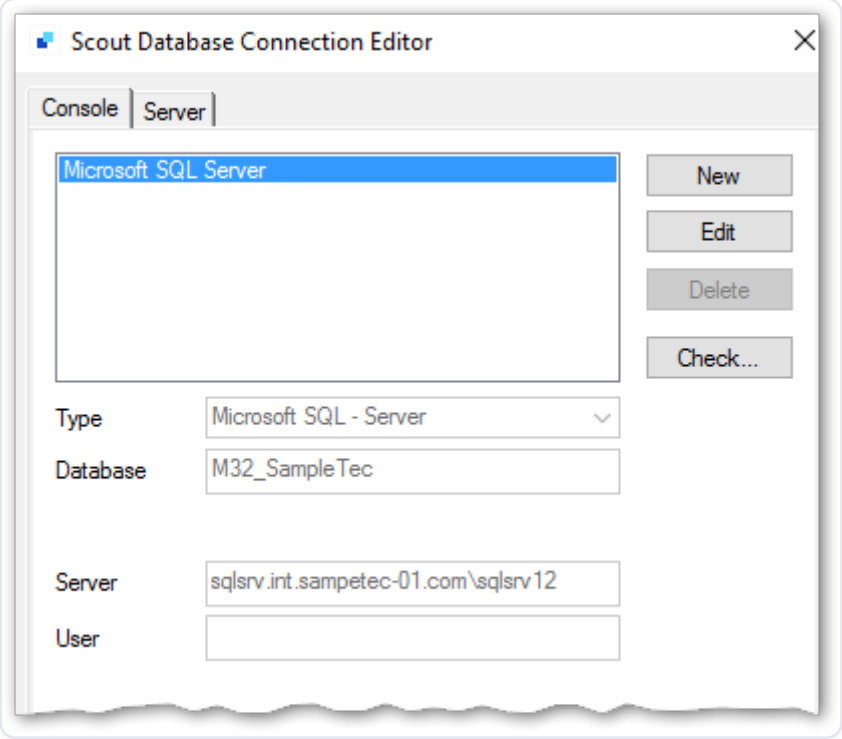
> From ELIAS 18 2104, all containers are stored in one database. If you manage multiple containers in an earlier ELIAS-18 installation, the update installation will merge the corresponding databases into one database.

1. Download the latest ELIAS 18 version as a `.zip` file from our technical myelux.com portal (**Download > Scout > ELIAS**).
2. Unpack the `.zip` file and provide the installation file on a local hard disk.
3. Run the `EliasInstaller.exe` file as administrator and follow the instructions of the installation Wizard.

Database connections

When you install the Scout Enterprise Management Suite®, you are asked to specify the Scout database. Check or modify the database connections at any point in time after the installation.

For the Scout Server and Scout Console, you can use the **Scout Database Connection Editor** to specify one or more connections to the Scout database. You can find the Scout Database Connection Editor as a stand-alone program within the Scout group of the Windows Apps view.



Certificates

Various features and applications require certificates to be provided. For (root) certificates on the client, note the following:

- Unless otherwise stated, the certificates must be Base64-encoded (ASCII) with file name extension `.crt`.
- To transfer certificates to the client, use the Scout feature Files configured for transfer. For further information, see Files configured for transfer in the **Scout** guide.
- On the client, the certificates are stored in the local certificate store `/setup/cacerts/` or in a sub-directory.

The following table provides an overview:

FEATURE	COMPONENT	DIRECTORY
Smart card user logon The certificates are specified in the Scout Console under **Security > User authentication > Certificates**	User authentication / AD+smart card	`/setup/cacerts/login`
Secure connection (TLS)	Firefox	`/setup/cacerts/browser` (Earlier eLux versions have also used `/setup/cacerts/firefox`)
Secure connection (TLS)	Chromium	`/setup/cacerts/browser`
Secure connection (TLS)	Builtin BrowserKiosk mode	`/setup/cacerts/browser`
Secure connection (TLS)	Citrix Workspace-App	`/setup/cacerts/` and `/setup/cacerts/intcerts`
Secure connection (TLS)	VMware Horizon client	`/setup/cacerts/`
Secure connection (TLS)	eLuxRDP	`/setup/cacerts`
Network logon	WLAN drivers / WPA-Supplicant (802.1X)X509/ RadiusNetwork Access Control / SCEP	`/setup/cacerts/setup/cacerts/scep`
VPN / OpenVPN	BaseOS	`/setup/openvpn`

FEATURE	COMPONENT	DIRECTORY
VPN / Cisco AnyConnect	Cisco AnyConnect	`/setup/cacerts` (from eLux RP 6 2302. Earlier versions require the certificate in `/setup/cacerts/ca` and `/setup/cacerts/client`)
Firmware update including certificate check	BaseOS	`/setup/cacerts`

> **Note:**
>
> StoreFront™ can be called using a Citrix® session or a browser.

Certificate for Scout Keep Alive service

As the eLux® devices and the Scout Keep Alive service communicate via HTTPS, for the installation of the Scout Keep Alive service, a valid SSL certificate for server authentication is required which is bound to port 22124 by default.

As soon as a certificate becomes invalid, you need to bind a new certificate to the port to keep the Scout Keep Alive service running. To do so, on the system the Statistics Service is running on, use the netsh.exe tool of the Windows command-line interface.

> **Note:**
>
> If the computer has more than one network adapter, the certificate must be bound to all IP addresses.

Viewing the current SSL certificate bindings

1. Launch the command-line interface.
2. Use the following command: `netsh.exe http show sslcert`

All ports with certificate bindings are shown including the relevant information.

Deleting an SSL certificate from a port

1. Launch the command-line interface.
2. Use the netsh.exe tool as shown in the following example: `netsh.exe http delete sslcert ipport=<IP address of host>:22124`

Viewing the thumbprints of certificates

> **Note:**
>
> Thumbprint corresponds to the certificate hash value.

1. Launch the Powershell. Note that the command is not supported by the normal command-line interface (cmd).
2. Use the following command depending on the certificate store: `dir cert:LocalMachineMy`

For all certificates available in the Microsoft Management Console, the thumbprints are shown under Local ComputerPersonal (with and without binding).

Binding a new SSL certificate to a port

1. Launch the command-line interface.
2. Use the netsh.exe tool with the following command: `netsh.exe http add sslcert ipport=0.0.0.0:22124 certhash=<*thumbprint of your certificate*> appid={957ba029-e2a1-4a13-b426-645a5e3802e2}`

The ipport parameter specifies the IP address and port.

The certhash parameter specifies the thumbprint of the certificate.

The appid parameter is the ID of the Scout Keep Alive service and must not be changed.

Management protocol

Communication between the Scout Server and the eLux® devices can be established via port `22123` or port `22125`.

If you are using a firewall, enable the relevant port.

Certificate-based management protocol

Configuring the trust level on the devices

Configuring Scout Server for communication via CA certificates

Certificate-based management protocol

The certificate-based management protocol provides secure communication between the Scout Server and devices via end-to-end encryption with TLS 1.2.

Starting with Scout 15 2107, only devices with **eLux RP 6.2 or later** are supported. For further information, see Compatibility client platform and Scout Enterprise Management Suite in the Whitepaper **Releases, Lifecycles and Compatibility**.

The certificate-based encryption of the management protocol is carried out via a self-signed certificate automatically generated by the Scout service. Alternatively, you can use a CA certificate that must be configured on the Scout Server.

For the encrypted communication with the Scout Server, port 22125 is used. (Up to version Scout 15 2107, devices with earlier eLux® versions could be used via port 22123 with AES-256 encryption.)

For TLS 1.2 communication, the following requirements must be met:

- On the devices, the trust level must be specified by using **TlsVerifyOption**. By default, the trust level is set to 0 and the certificate check is disabled.

  For further information, see Configuring the trust level on the devices.
- If you are using a certificate issued by a CA (instead of self-signed), the certificate must be provided in the form of a pfx or pem file on the Scout Server. Note that the certificate must not be password-protected. The devices must be equipped with the corresponding root certificates.

  For further information, see Configuring Scout Server for communication via CA certificates.

> **Note:**

> To check the communication via TLS, view the `eluxd.log` log file of the Scout Server service.

Configuring the trust level on the devices

The certificate-based encryption of the management protocol for communication between the Scout Server and the eLux® client requires the verification of the relevant certificates (Chain of

trust). By default, encryption is carried out via a self-signed certificate automatically generated by the Scout Server.

Important:

If you use a certificate issued by a CA (CA certificate), make sure you transfer the corresponding root certificates to the devices. If the root certificate does not exist on a device and certificate check is enabled, the device can no longer be reached by the Scout Server. To make it easier, you can perform both steps, enabling the certificate check and transferring certificates in one move. **Important:**

In addition, the Scout Server must be configured accordingly and provide the certificate locally.

1. To enable the certificate check, configure the trust level for the relevant devices with the option **TlsVerifyOption**.

   To do so, use the Advanced file entries feature of the Scout Console:

FILE	/SETUP/TERMINAL.INI
Section	Security
Entry	TlsVerifyOption
Value	
0	Certificate is not verified
1	Certificate is verified
3	Certificate is verified with additional verification that the Scout Server name matches the Subject Common Name (CN) or Subject Alternative Name (SAN) in the certificate.

   For further information, see Advanced file entriesin the **Scout** guide.

2. If you use a CA certificate, make sure you transfer all corresponding root/intermediate certificates of your CA to the devices to /setup/cacerts/scoutsrv. This is where the system searches for the required certificates once the certificate check is enabled (Chain of trust).

   For further information, see Files configured for transfer in the **Scout** guide.

3. If you use a CA certificate, in the next step, configure the Scout Server. For further information, see Configuring Scout Server for communication via CA certificates.
4. Restart the devices.

> **Note:**
>
> After the `terminal.ini` file has been updated on the device, one more device restart might be required to enable the new setting.

> *Once you have enabled trust level 1 or 3 for a device, it can only communicate with its Scout Server by using valid certificates. With trust level 3, the device name is verified in addition.*

Configuring Scout Server for communication via CA certificates

> **Note:**
>
> This configuration is only required if you are using a certificate that has been issued by a CA.

1. Save the certificate file locally on the Scout Server.
2. On the server machine, in the file system under `%PUBLIC%\Documents\Unicon\Scout\Server` open the `eluxd.ini` file for editing.

   Add the following entries:

SECTION	ENTRY	DESCRIPTION
ELUXD	`UseSelfsignedCertificate=0`	Requires a certificate issued by a CA Use a certificate that is not protected by an additional password. If you set this option (with value 0),

SECTION	ENTRY	DESCRIPTION
		you are required to define the next values. Default: 1
ELUXD	`CertificateFile` =*Path to certificate file*	Path to the location of the certificate file Example: `C:\Users\Public\Documents\Unicon\Scout\Server\sampletec-01.pfx`
ELUXD	`CertificateKeyFile` = *Path to private key file*	Only required, if the certificate file is not in pfx format

3. Restart the Scout service.
4. Ensure that the certificate check is enabled and the required root/intermediate certificates of your CA are available on the devices (Chain of trust). For further information, see Configuring the trust level on the devices.

From now on, the Scout Server will only communicate with devices that trust the CA certificate.

Troubleshooting

ERROR MESSAGE / PROBLEM	REASON	SOLUTION
Scout desktop icons are not displayed correctly.	Desktop icons are cached by Windows. This may lead to the fact that after you update Scout, new icons are not displayed correctly.	Delete the Windows icon cache. Please refer to the relevant documentation.
File access error while checking HTTP/FTP server (error number = 404)	Possibly caused by missing MIME type entries for the used file extensions on the web server	In the MIME type settings of the web server, add the file extensions used in eLux containers and assign them to the relevant MIME types. When you install ELIAS 18 or an eLux container on Microsoft Internet Information Server (IIS), the mappings

ERROR MESSAGE / PROBLEM	REASON	SOLUTION
		will be performed automatically.

The following MIME types are required and installed with the container:

EXTENSION	MIME TYPE
.dd	text/plain
.epm	text/plain
.fpm	text/plain
.gz	application/x-gzip
.idf	text/plain
.ini	text/plain
.rdf	text/plain
.sig	text/plain
.xz	application/x-xz

The following additional assignments might become necessary:

EXTENSION	MIME TYPE	DESCRIPTION
.bin	application/octet-stream	BIOS update via Scout
.bup	text/plain	BIOS update via Scout
.cab	application/vnd.ms-cab-compressed	UEFI update via Scout
.mee	text/plain	Migration of eLux RP 5 devices to eLux RP 6 with whitelist (for current eLux versions done by the container installer). Different

EXTENSION	MIME TYPE	DESCRIPTION
		images depending on the hardware model (/en-us/unicon-elux-scout/2603-ltsr-cu1/admin_guides/scout/setup/firmware/firmware_macro_hardware.html)
.udf	text/plain	UEFI updates in analogy to firmware updates (from Scout 15 2107 and eLux RP 6 2107)

Troubleshooting for an installation with Microsoft SQL Server

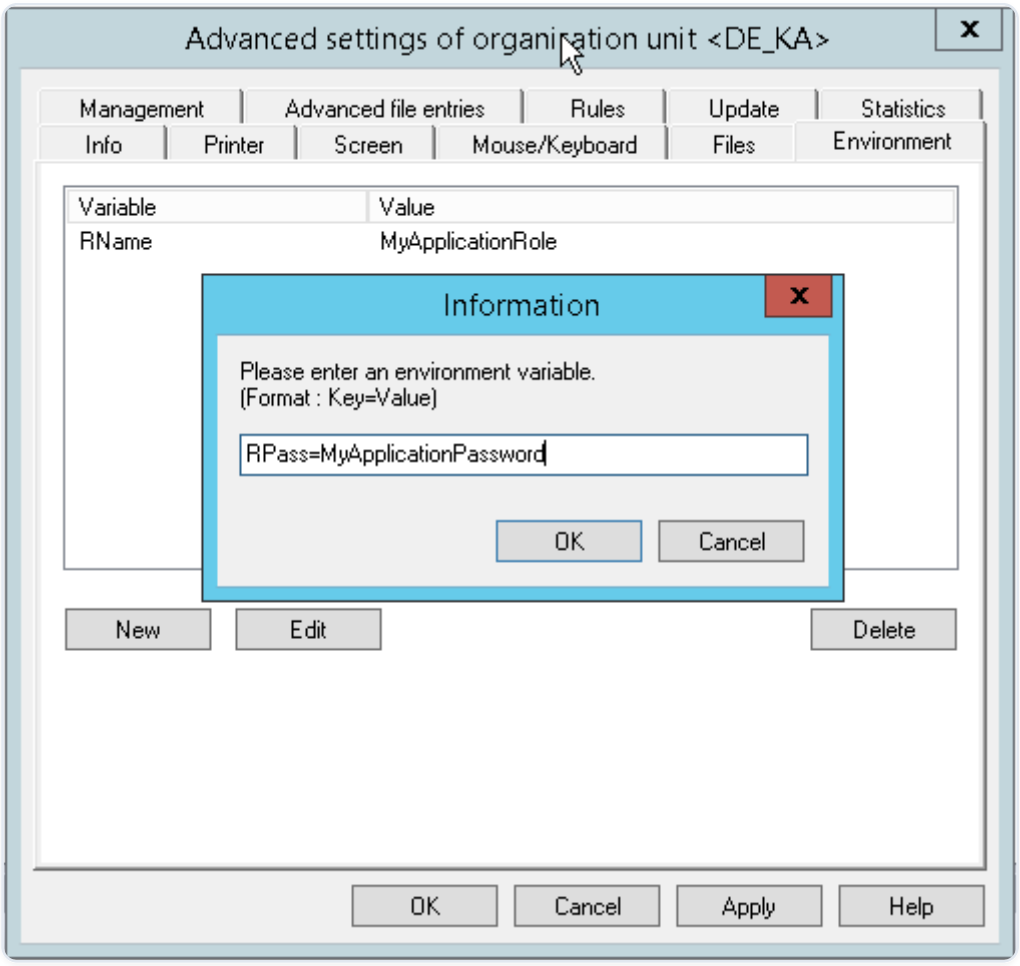
ERROR MESSAGE	REASON	SOLUTION
Cannot initialize the license database.	When checking the database ID, a problem was identified. A database backup was restored on an SQL Server on which the database does not exist. Note: A unique database ID ensures that a Scout license database cannot be used more than once.	Preventive: Restore a database backup only on the SQL Server on which the database backup was created and the database still exists. In case of error:Contact Unicon™ Support. The **Repair database** button displays a checksum that can be used by Unicon Support to generate a validation code for repairing the license database.

Encrypting values

Whenever you need to encrypt values, we recommend that you add and encrypt variables in a temporary OU, and then copy the encrypted values to their target position.

1. In the Scout Console, create a temporary OU such as `TEMP`.
2. For the TEMP OU, from the context menu, choose **Advanced device configuration > Environment**.

3. Add a new variable and its value. Confirm with OK.



*The new variable and its value are shown in the **Environment** tab.*

4. On the **Environment** tab, right-click the variable, and from the context menu, choose **Encrypt value**.

   The value of the variable is shown in encrypted mode.

5. Select the variable and click **Edit**. Then, copy the encrypted value to the Clipboard and paste it on the target position.
6. Delete the temporary OU.

Appendix

Program and file directories

eLux RP 6 partitions

IP ports

SNMP

Program and file directories

Program directory

The Scout Enterprise Management Suite® by default is installed to

`%PROGRAMFILES%\Unicon\Scout`

ELIAS 18 is installed according to your specifications during installation, for example on the web server IIS.

The eLux® container (only required for the legacy ELIAS) is installed on the web server to

`< root directory>\eluxng`

File path for Scout Server files

Scout log files, configuration files and more are saved to a subdirectory of

`%PUBLIC%\Documents\Unicon`

- To open the server files directory in the Windows Explorer, in the Scout Console, click **View > System diagnostics > Server files** (only if console and server are installed on the same machine).

File path for user files

User files are saved to a subdirectory of the local user directory in

`%USERPROFILE%\Documents\Unicon`

Diagnostic files that are requested via the console are saved to

`%USERPROFILE%\Documents\Unicon\Scout\Console\Diag`

Diagnostic files that are requested via Scout Board are downloaded by the browser used and saved in the download directory, if configured.

> **Note:**
>
> If you use anti-virus software on your Scout Server, we recommend that you exclude the specified directories from the virus scan to avoid side effects.

eLux® RP 6 Partitions

An eLux device's flash memory is generally divided into three or four partitions when eLux is installed. Each partition is reserved for a dedicated purpose and is only touched when you perform special tasks related to this partition.

All partitions are created during a recovery installation.

PARTITION	REQUIRES	PURPOSE	RECREATED WITH	OTHER
System		Reserved for the firmware (software packages)	Scout Update command with option Format system partition before update	Size for eLux RP 6 2104 LTSR and earlier versions: 1.77 GB / 1.84 GB with/without encryption, Size for eLux RP 6 2107 and later versions: 2.35 GB / 2.41 GB with/without encryption
Boot	only UEFI and USB	Boot section	–	
Setup		Device configuration Local application definitions	Factory reset command	Does not affect the system partition with installed firmware
Update	4 GB flash memory	Software delivery in advance (before firmware update) via Scout command or notification Signature	Scout Delivery command with option Format update partition before delivery	The size of the update partition complies with the storage space provided. The update partition is no larger than the storage space provided.

PARTITION	REQUIRES	PURPOSE	RECREATED WITH	OTHER
		check for eLux software packages		Devices with less than 4 GB flash memory are not provided with an Update partition.

Note:

In the Scout Console, in the Properties window of a device, the system, setup, and update partitions are listed including their sizes.

Extended System Partition Starting with eLux RP 6 2107

When you perform an update installation or a new installation (recovery) to eLux RP 6 2107 or later, the system partition is created with 2.35 GB / 2.41 GB (with/without encryption) instead of the previous almost 2.0 GB. This creates more space for the firmware and allows larger images to be used.

UPDATE INSTALLATION

An update installation (firmware update) is still based on the previous partition sizes. The image size is thus still limited to the earlier values. Afterwards, the extended system partition is available and you can install images that may be up to 2.35 GB / 2.41 GB in size. This means, to install larger images on the freshly resized partition of the devices, a second firmware update is required.

RECOVERY INSTALLATION

Provided an up-to-date recovery system is available, with a PXE or USB recovery installation the system partition can be partitioned to the new size directly during the installation process and a larger image with up to 2.35 GB / 2.41 GB can be written in the same process. A new installation or recovery installation thus allows the partition to be resized and used in one step.

DOWNGRADE

To downgrade devices with the extended system partition (eLux RP 6 2107 or later) to an earlier version that only supports the previous system partition with less than 2 GB, you will have to go back to eLux RP 6 2104 LTSR.

We therefore recommend that you update test devices to eLux RP 6 2107 or later as the first step to thoroughly test functionality.

For further information, see Update to new partition layout (1.77 / 1.84 GB to 2.35 / 2.41 GB) in the Scout guide.

Network requirements

All contents relating to ports, protocols and network requirements are covered by the **network requirements** guide. For further information, see Network requirements guide.

SNMP

SNMP (Simple Network Management Protocol) is a network protocol for monitoring and controlling network devices.

For eLux® RP 6 and newer, version SNMPv3 is used.

> **Note:**
>
> The command line program **snmpget** is not included in the software package. To query SNMP status information, please use third party software.

> Configuring SNMP

SNMPD and SNMP Configuration Directives

Configuring SNMP

1. From our myelux.com portal, under **eLux Software Packages**, for your eLux version, under **Add-On**, download the package SNMP Environment and deploy it to the devices.
2. If there is no /setup/snmp/snmpd.conf on the devices, transfer the configuration file snmpd.conf to the devices to `/setup/snmp/snmpd.conf` by using the Scout feature advanced_files Files configured for transfer

   Or:

   Modify the `terminal.ini` file by using the Adding individual file entries feature of Scout. Example:

File	`/setup/terminal.ini`
Section	SNMPD
Entry	rocommunity
Value	secret

3. Optionally, to define further SNMPD and SNMP Configuration Directives, use the Adding individual file entries feature and modify the `terminal.ini` file under SNMPD. Examples:

   `syscontact=contact@sampletec.com` `syslocation=testcenter` `doDebugging=1`

   For further information on SNMPD Configuration Directives, see http://www.net-snmp.org.

The section `SNMPD` of the `terminal.ini` file is evaluated by the client, and the file `/setup/snmp/snmpd.local.conf` is created, overwriting any existing `/setup/snmp/snmpd.conf`.

If the configuration file does not exist, the file `/setup/snmp/snmpd.local.conf` is created with default values.

NOTES ON CONFIGURING SNMP V3

- When you define users (**createUser**), set a password with at least 8 characters.
- For the authentication method, define `authPriv` or `authNoPriv`.

> **Note:**
>
> For SNMP v2, you can use `noAuthNoPriv` as the authentication method.

SNMPD and SNMP Configuration Directives

The following table refers to the eLux® software package snmp.

For further information on using SNMP with eLux, see SNMP

For further information on SNMP commands, see http://www.net-snmp.org.

APPLICATION	COMMAND
authtrapenable	1 | 2 (1 = enable, 2 = disable)

APPLICATION	COMMAND
trapsink	host [community] [port]
trap2sink	host [community] [port]
informsink	host [community] [port]
trapsess	[snmpcmdargs] host
trapcommunity	community-string
agentuser	agentuser
agentgroup	groupid
agentaddress	SNMP bind address
syslocation	location
syscontact	contact-name
sysservices	NUMBER
interface	name type speed
com2sec	name source community
group	name v1|v2c|usm security
access	name context model level prefix read write notify
view	name type subtree [mask]
rwcommunity	community [default|hostname|network/bits] [oid]
rocommunity	community [default|hostname|network/bits] [oid]
rwuser	user [noauth|auth|priv] [oid]
rouser	user [noauth|auth|priv] [oid]

APPLICATION	COMMAND
swap	min-avail
proc	process-name [max-num] [min-num]
procfix	process-name program [arguments...]
pass	miboid command
pass_persist	miboid program
disk	path [minspace | minpercent%]
load	max1 [max5] [max15]
exec	[miboid] name program arguments
sh	[miboid] name program-or-script arguments
execfix	exec-or-sh-name program [arguments...]
file	file [maxsize]
dlmod	module-name module-path
proxy	[snmpcmd args] host oid [remoteoid]
createUser	username (MD5|SHA) passphrase [DES] [passphrase]
master	specify 'agentx' for AgentX support
engineID	string
engineIDType	num
engineIDNic	string

SNMP CONFIGURATION DIRECTIVES

APPLICATION	COMMAND
doDebugging	(1|0)

APPLICATION	COMMAND
debugTokens	token[,token...]
logTimestamp	(1|yes|true|0|no|false)
mibdirs	[mib-dirs|+mib-dirs]
mibs	[mib-tokens|+mib-tokens]
mibfile	mibfile-to-read
showMibErrors	(1|yes|true|0|no|false)
strictCommentTerm	(1|yes|true|0|no|false)
mibAllowUnderline	(1|yes|true|0|no|false)
mibWarningLevel	integerValue
mibReplaceWithLatest	(1|yes|true|0|no|false)
printNumericEnums	(1|yes|true|0|no|false)
printNumericOids	(1|yes|true|0|no|false)
escapeQuotes	(1|yes|true|0|no|false)
dontBreakdownOids	(1|yes|true|0|no|false)
quickPrinting	(1|yes|true|0|no|false)
numericTimeticks	(1|yes|true|0|no|false)
suffixPrinting	integerValue
extendedIndex	(1|yes|true|0|no|false)
printHexText	(1|yes|true|0|no|false)
dumpPacket	(1|yes|true|0|no|false)
reverseEncodeBER	(1|yes|true|0|no|false)

APPLICATION	COMMAND
defaultPort	integerValue
defCommunity	string
noTokenWarnings	(1|yes|true|0|no|false)
noRangeCheck	(1|yes|true|0|no|false)
defSecurityName	string
defContext	string
defPassphrase	string
defAuthPassphrase	string
defPrivPassphrase	string
defVersion	(1|2c|3)
defAuthType	(MD5|SHA)
defPrivType	(DES)
defSecurityLevel	(noAuthNoPriv|authNoPriv|authPriv)

Network requirements

For deploying eLux® and management solutions, please make sure that the network requirements are implemented. Different products need different protocols and ports for the communication.

eLux

PORT	TYPE	DESCRIPTION	HOW TO DEACTIVATE	COMPONENTS	IN/OUT
	ICMP	ping must be supported to verify the		eLux --> Scout Server	Incoming/ Outgoing

PORT	TYPE	DESCRIPTION	HOW TO DEACTIVATE	COMPONENTS	IN/OUT
		status of the eLux devices			
80	TCP	Firmware update by using HTTP (and proxy port, if used)		eLux --> Web/File Server, serving the Images and packages.	Outgoing
				Best practice: Use HTTPS for a secure connection	
443	TCP	Firmware update via HTTPS/TLS		eLux --> Web/File Server, serving the Images and packages	Outgoing
443	TCP	Onboarding via uPilot		eLux --> `https://onboarding.unicon.com`	Outgoing
5900	TCP	Mirroring eLux desktop	In Device configuration > Security, disable mirroring or uninstall VNC server from the	Scout Console (VNC Client) --> eLux (VNC Server)	Incoming

PORT	TYPE	DESCRIPTION	HOW TO DEACTIVATE	COMPONENTS	IN/OUT
			X.Org package		
5902	WebRTC (TCP, UDP)	Mirroring eLux desktop	In Config > Device configuration > Security, disable mirroring or uninstall VNC server	Scout Board UI (Web Browser) --> eLux (webrtcify --> VNC Server).	Incoming
				End to end encrypted (WebRTC). High ports (UDP) are used by WebRTC for the data channels.	
				You can use the enterprise policy `WebRtcUdpPortRange` (Chrome v54+, Edge v77+) to restrict the local UDP port window (e.g. 10000-11999)	

PORT	TYPE	DESCRIPTION	HOW TO DEACTIVATE	COMPONENTS	IN/OUT
22124	TCP	Scout Keep Alive Service (optional)		eLux --> Scout Server	Outgoing
22125	TCP	Scout Server (Scout Manager / TLS)		eLux --> SCG, eLux --> Scout Server, Scout Server/SCG --> eLux.	Incoming/ Outgoing
				A certificate can be used with the TlsVerifyOption	
22129	TCP	VPN (SCG)	In SCG Admin Configuration > Gateways > Public port, you can change the port.	eLux --> SCG	Outgoing

Note:

Depending on the application and the use-cases, other optional ports might be needed; like DNS, DHCP, TFTP, AD/Kerberos, SNMP, Wake on LAN, etc.

Scout Server

PORT	TYPE	DESCRIPTION	COMPONENTS	IN/OUT
	ICMP	ping must be supported to verify the status of the eLux devices	Scout Server --> eLux	Incoming/ Outgoing
1433	TCP	MS SQL Server	Scout Server --> SQL Server	Outgoing
1434	UDP	MS SQL Server (Browser service)	Scout Server --> SQL Server, Scout Board DBLayer --> SQL Server	Incoming/ Outgoing
22124	TCP (HTTPS)	Scout Keep Alive Service offers an evaluation of configurable status messages of the managed devices.	eLux --> Scout Keep Alive service	Incoming
		These status messages are called keep alive messages.		
		This service runs on the Scout Server		
22125	TCP	eLux Endpoints (Scout Manager / TLS)	Scout Server --> eLux, eLux --> Scout Server	Incoming/ Outgoing

PORT	TYPE	DESCRIPTION	COMPONENTS	IN/OUT
8083 (WSLPORT)	HTTPS	The Citrix License Server port - can be defined/ changed during installation	Scout Server --> Citrix License Server	Outgoing

> **Note:** For details on the ports and firewall requirements, please see Citrix License Server

Scout Board

PORT	TYPE	DESCRIPTION	COMPONENTS	IN/OUT
1433	TCP	MS SQL Server	Scout Board Database Layer --> SQL Server	Outgoing
5902	WebRTC (TCP, UDP)	Mirroring eLux desktop	Scout Board --> eLux (webrtcify --> VNC Server)	Outgoing
22160	TCP (HTTPS)	Scout Board HTTPS web application. Administrator can define a custom port during installation	Administrator --> Scout Board web UI	Incoming

Scout Console (legacy)

PORT	TYPE	DESCRIPTION	HOW TO DEACTIVATE	COMPONENTS	IN/OUT
1433	TCP	MS SQL Server		Scout Console --> SQL Server	Outgoing
1434	UDP	MS SQL Server (Browser service)		Scout Console --> SQL Server	Outgoing
5900	TCP	Mirroring eLux desktop	In Device configuration > Security, disable mirroring or uninstall VNC server from the X.Org package	Scout Console (VNC Client) --> eLux (VNC Server)	Outgoing

Scout Cloud Gateway (SCG)

PORT	TYPE	DESCRIPTION	COMPONENTS	IN/OUT
22125	TCP (TLS)	Scout Server (Scout Manager / TLS) - during onboarding only	eLux --> SCG, SCG --> eLux	Incoming/ Outgoing (if 2 network interfaces exist: External Interface)
22129	TCP	VPN (SCG) - can be changed in the SCG admin UI	eLux --> SCG	Incoming (if 2 network interfaces exist: External Interface)

PORT	TYPE	DESCRIPTION	COMPONENTS	IN/OUT
636	TCP	AD Authentication (LDAP)	SCG --> LDAPS (AD)	Outgoing (if 2 network interfaces exist: Internal Interface)

Licensing Guide

The Licensing Guide explains the available licensing models for Citrix Unicon OS Management (Scout), their technical requirements and compatibility, and provides step-by-step instructions to help you choose and implement the model that best fits your needs.

See the licensing guide for detailed information.

Theming guide for eLux 7

The look and feel of eLux 7 can be modified to realize corporate identity requirements to some extent.

This guide shows,

- how to create themes for eLux 7,
- where to place the required files and configurations,
- how to activate a theme

Components of a theme

A theme consists of a couple of predefined components. The configuration and the naming of the files follow predefined conventions.

COMPONENT	REQUIREMENTS	INFORMATION
name	No spaces or special characters	The name of the theme
accentColor	Hex RGB color value (e.g. #RRGGBB)	The main color, which is used on different parts of the theme throughout eLux 7

COMPONENT	REQUIREMENTS	INFORMATION
desktopBackgroundImage	JPG image, i.e. Full-HD resolution image	The background image for the desktop
productImage	330x330 px	The company/CI logo (shown also during the boot process)
brandImage	Height: 32px, width: Can vary	The company/CI brand image

Creating a custom theme

To create a custom theme, follow the following steps:

Create the theme configuration file

Each theme requires a configuration file for the theme, called `config.json`.

```
{
  "name": "my-theme",
  "accentColor": "#CD5767",
  "desktopBackgroundImage": {"type": "jpg", "scaling": "crop"},
  "productImage": {"type": "png", "scaling": "contain"},
  "brandImage": {"type": "png", "scaling": "contain"}
}
```

Note:

The name of the theme and the accent color need to be defined by the theme requirements. The config file must be named `config.json` by convention.

Prepare the theme files

Prepare the image files for the desktop background, product image, and brand image, according to the requirements.

Note:

The file names for the images must be named `desktopBackgroundImage.jpg`, `productImage.png` and `brandImage.png` by convention.

Create the theme folder

Create the folder for your theme on eLux 7 using Scout Board and a custom command.

```
mkdir /setup/elux/themes/my-theme
```

The theme files will all go into this theme folder.

Note:

The default theme with all its files is located at `/setup/elux/themes/unicon_default` in case a template is needed.

Transfer the theme files to eLux 7

Transfer all the theme related files to eLux 7 in Scout Board, using the advanced file entries.

LOCAL THEME FILE	DESTINATION THEME FILE ON ELUX 7
`config.json`	`/setup/elux/themes/my-theme/config.json`
`desktopBackgroundImage.jpg`	`/setup/elux/themes/my-theme/desktopBackgroundImage.jpg`
`productImage.png`	`/setup/elux/themes/my-theme/productImage.png`
`brandImage.png`	`/setup/elux/themes/my-theme/brandImage.png`

Note

> After applying the configuration, reboot the endpoint, so that all the configuration and theme files are being transferred.

Set the theme

Once you have transferred all the required files to eLux 7, the theme needs to be activated.

For setting the active theme on eLux 7, the following custom command needs to be sent to eLux from Scout Board:

```
ucsettingsctl -o Desktop -v CurrentTheme=my-theme
```

After applying the theme, the corresponding values and files will be used by the system and shown accordingly.

Overview

eLux® Portable is an ideal solution for secure work-from-home and remote access environments. It is compatible with any x86 endpoint. The key benefits for the users are that it is immediately available without any installation efforts. So it is a real Plug & Work approach.

Features

- Any home PC equipment or other x86 device with a standard USB port can be used without modifying the underlying operating system.
- Any state-of-the-art USB drive with a capacity larger than 4 GB can be used.
- No company data are stored on the USB drive.
- No need for special security measures or dedicated hardware
- No installation
- The First Configuration Wizard guides you through the first steps, no additional configuration needed.
- The license always remains on the server, so a broken USB drive is easily replaced and remote work can continue seamlessly.

Requirements

- A stable internet connection
- On your device, change the boot order for boot from USB and disable secure boot.

Creating the eLux® Portable stick

REQUIRES
• PC running Windows 10 or later versions
• USB stick on local port with 4 GB memory space

1. From our myelux.com portal, under **Downloads > eLux > eLux Portable**, download the current **eLux Portable** archive.
2. Unpack the downloaded .zip file on your device.
    - The eLux USB Stick image is available as a .dd file in a directory on your PC, along with the StickWizz.exe program (and a describing text file with check sum).
3. Continue with Writing an image to USB.

Writing an image to USB (with StickWizz)

REQUIRES
• Image file as a .stw or .dd file on a PC running Windows 10 or later versions
• StickWizz application on your PC (/en-us/unicon-elux-scout/2603-ltsr-cu1/short_guides/recovery_general/recovery_usb/recovery_usb_prepare.html)
• USB stick on local port with 4 GB memory space, write access

1. Plug the USB stick into the PC. Note that the Windows Explorer must not be open on the USB drive.
2. Run the StickWizz.exe program.

- The language depends on the Windows display language selected.
- The checksum is calculated and compared to the one provided.

3. Check whether the appropriate source under Image file (1) and the appropriate destination under Device (3) are selected.
4. Click **Write** (7).

The image is written to the USB stick. A message confirms the successful completion.

1	After StickWizz is started, it shows a .stw or .dd file located in the same directory.
2	Browse to select your image file from another directory, if required.
3	Device /drive, on which a USB stick is connected. Choose a different drive if required.
4	Delete all data from the selected device (before writing the image).
5	During the write process, each written data packet will be compared to the source.
6	As soon as StickWizz opens with the specified file, the checksum is calculated and displayed.
7	Start writing.

Using eLux® Portable

REQUIRES
PC, notebook or other x86 device providing a boot option from **USB HDD**

Starting eLux from USB

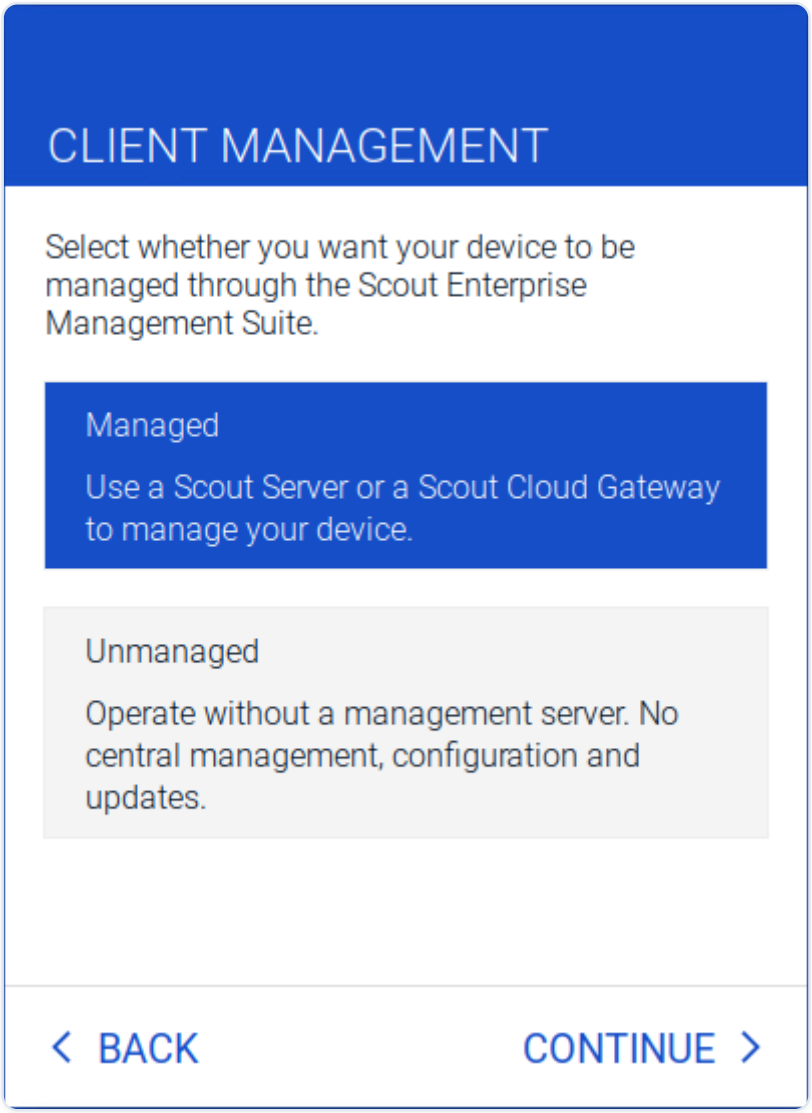
1. Plug the USB stick into a USB port and boot the device.
2. To enter the BIOS/UEFI menu, interrupt the boot process by pressing ESC, F2, F12, or the key specified by the manufacturer.

3. Configure the BIOS/UEFI of your device to boot from USB. Disable **Secure boot**.
4. Restart the device.

The device boots from the USB stick and eLux loads with the First Configuration Wizard.

Configure access to your infrastructure

1. In the First Configuration Wizard, select your preferred display language and keyboard language.
2. Agree to the license agreement.
3. Enter the data of your WLAN or your LAN connection..
4. To configure client management via Scout, select **Managed**.



5. Enter the address of your Scout Server (or Scout Cloud Gateway) as FQDN or IP address.
6. Enter your logon data.
7. Click **Finish**.

Your device connects to its Scout Server and your desktop is loaded. The connection data is retained, so you will access your eLux desktop without reconfiguration after the next device restart from USB.

> **Note:**

> The operating system and the data saved on your hard disk will remain completely unaffected.

Overview

A recovery installation allows you to install the eLux operating system on a device's flash memory and/or to reset the device to the initial state.

In the following cases a recovery installation can be suitable:

- The client does not boot anymore (=> restore)
- The flash memory of the client does not provide an image/is empty
- The local device password was lost
- The operating system on the flash memory shall be replaced by eLux®
- A factory reset of the image on the flash memory is required
- Migration to a later eLux version (alternatively see the **Migration** short guide)
- If critical feature-packages `(.fpm)` in the Base OS are updated, the client requires a recovery installation before an update can be performed.

> **Note:**

> A recovery installation will overwrite all data on the storage medium and install the eLux software. It cannot be undone.

> A recovery installation can be performed in two different ways:

- via USB stick if the client hardware supports booting from USB mass storage devices
- via network if the client's network adapter supports PXE

Recovery via USB

On our myelux.com portal, you can download **eLux USB Stick** images for all supported eLux versions. By using the tool included in the .zip file, transfer the image file to a USB stick and create an eLux recovery stick which you can use for installing eLux on your devices.

To recover your individual firmware image, export your image and the defined software packages in ELIAS via the Stickwizz option and create a USB stick with it. For further information, see Exporting an image in the **ELIAS 18** guide.

For further information on creating an individual USB recovery stick in the legacy ELIAS, see USB-recovery stick in the **ELIAS** guide.

- Preparing a recovery image for USB
- Writing an image to USB (with StickWizz)
- Performing a USB recovery

Preparing a recovery image for USB

REQUIRES
PC running Windows 10 or later versions

eLux® USB Stick Image

1. From our myelux.com portal, under **eLux USB Stick Images**, download the package for the required eLux version.
2. Unpack the downloaded `.zip` file.

The eLux USB Stick image is available as a .dd file in a directory on your PC, along with the StickWizz.exe program (and a describing text file with check sum).

Individual Image

1. Provide your individual image on your PC. The image can be one of the following:
    - `.stw` file that has been exported from ELIAS 18 via the **Stickwizz** option. For further information, see Exporting an image in the **ELIAS 18** guide
    - `.stw` file that has been created via **Create StickWizz image** in the legacy ELIAS. For further information, see Creating a StickWizz file as installation base in the **ELIAS** guide
    - `.dd` file that has been created as a disk image via **Create USB stick** in the legacy ELIAS. For further information, see Creating USB recovery stick in the **ELIAS** guide

2. From our myelux.com portal, under **Tools**, download the **StickWizz** program , and then unpack the `.zip` file on your PC.

Your individual image is available as a `.stw or .dd` file in a directory on your PC, along with the `StickWizz.exe` program. Continue with Writing an image to USB.

Writing an image to USB (with StickWizz)

REQUIRES
Image file as a .stw or .dd file on a PC running Windows 10 or later versions
StickWizz application on your PC (recovery_usb_prepare)
USB stick on local port with 4 GB memory space, write access

1. Plug the USB stick into the PC. Note that the Windows Explorer must not be open on the USB drive.
2. Run the `StickWizz.exe` program.
    - The language depends on the Windows display language selected.
    - The checksum is calculated and compared to the one provided.
3. Check whether the appropriate source under **Image file** (1) and the appropriate destination under **Device** (3) are selected.
4. Click **Write** (7).

The image is written to the USB stick. A message confirms the successful completion.

1	After StickWizz is started, it shows a .stw or .dd file located in the same directory.
2	Browse to select your image file from another directory, if required.
3	Device /drive, on which a USB stick is connected. Choose a different drive if required.
4	Delete all data from the selected device (before writing the image).

5	During the write process, each written data packet will be compared to the source.
6	As soon as StickWizz opens with the specified file, the checksum is calculated and displayed.
7	Start writing.

Performing a USB recovery

REQUIRES
• *PC, mobile device or other client device providing a boot option from **USB HDD**
• 4 GB (2 GB for limited functionality) memory space on the flash memory / hard disk

1. Configure the BIOS of your device for boot from USB.
2. Connect the USB stick to a USB port and boot the device.
3. To call the boot menu, during the boot process, press F12 or the relevant key specified by the supplier.
4. On the eLux® boot menu, select the Install eLux XX option.

Important:

During the installation process, all data and the entire operating system are deleted from the hard disk (system recovery). All eLux and Management license information is kept as is.

After the installation is completed, the client restarts. Depending on the network configuration, the device is automatically assigned to a Scout Server via DHCP or DNS, or the First Configuration Wizard starts.

Network recovery (PXE)

The PXE network protocol (Preboot Execution Environment) enables devices to boot via LAN. The devices do not need a bootable storage medium but their BIOS must support PXE.

The PXE code in the BIOS of the network adapter allows the devices to communicate with the DHCP and TFTP server. From the DHCP server, the client receives the IP configuration. The TFTP server provides the boot file. The boot file calls the core installer which installs a minimal system, formats the flash memory and prepares the installation of the eLux® operating system from the web server.

With the **Recovery service** of the Scout Enterprise Management Suite®, a TFTP server and the required functions are installed. The **Recovery service** is included in the standard installation.

- PXE recovery requirements
- Configuring PXE recovery in Scout Console
- DHCP configuration
- Performing a PXE recovery
- Troubleshooting

PXE recovery requirements

- The relevant devices are provided with a PXE-capable BIOS and a LAN connection. UEFI devices can be installed via a PXE recovery.
- Scout Enterprise Management Suite® including the **Recovery service** (included in standard installation)
- eLux® software container for the relevant eLux version, installed on an HTTP or FTP server
- DHCP server (included in Microsoft Windows Server operating system)

The TFTP server is installed and registered as a service by the Scout Enterprise Management Suite. Make sure that there is no other TFTP server running on the system.

Configuring PXE recovery in Scout Console

1. In Scout Board, go to **Preferences > Recovery settings**. (Or, in the Scout Console, click **Options > Recovery settings**).
2. Edit the following fields:

OPTION	DESCRIPTION
Protocol	Network protocol of the web server for the software package transfer to the devices (HTTP, HTTPS, FTP, FTPS)
Server	Name (FQDN) or IP address of the web server providing the eLux software packages and image file, optionally followed by a port number. Example: `192.168.10.1:80`
Proxy (optional)	IP address and port (3128) of the proxy server. Example: `192.168.10.2:3128`
User and Password (optional)	Username and password (if required) to access the eLux® software container of an FTP server
Path	Directory path of eLux software packages on the web server, Use slashes / to separate directories. For ELIAS 18, specify the path name defined during installation. Example: `elias/UC_ELUX`. For the legacy ELIAS, use `eluxng/<container name>` To account for different eLux major versions, use a specific parameter. For further information, see Different eLux versions in the **Scout** guide.
Image file	Name of the image file in the **Scout** guide. To update devices with an earlier partition layout to a current version with an extended system partition, see Update to new partition layout in the **Scout** guide.
User must confirm	On the devices, users are required to confirm before the recovery installation is started.
Operating system	Relevant eLux version

3. Confirm with **OK**.

DHCP configuration

Next to selecting DHCP options for the router, DNS server and domain name which are already required for eLux®, for a PXE recovery installation, you need to define the boot server and boot file as DHCP options.

1. Open the DHCP manager as administrator.
2. For the relevant DHCP server, select either the **Server options, the Scope Options or the Reservations** for individual devices, and then click **Action > Configure Options... > General**.
3. Edit the following DHCP options:

OPTION	VALUE
066 Boot Server Host Name	`<IP address of the TFTP server>` (normally same as Scout Server IP address)
067 Bootfile Name	grub/pxex64.efi for UEFI devices (x64) grub/pxe.i386 for non-UEFI devices

4. Confirm with **OK**.

Performing a PXE recovery

The recovery installation is initiated by the client. The client is required to use the boot medium **LAN** and, depending on the hardware, must be configured accordingly.

- Turn the device on and then, on the boot menu, select the boot medium option LAN.

or

- If the boot menu does not provide boot medium options, in the BIOS, configure the **First boot device** to LAN and then restart the device.

After the core installer has prepared the system, the software packages are transferred from the web server and are installed. After the installation is completed, the client restarts. Depending on the network configuration, the device is automatically assigned to a Scout Server via DHCP or DNS, or the First Configuration Wizard starts.

Troubleshooting

Consulting the server log files helps you analyze problems with the recovery installation:

- `tftpd.log` to be found in `Documents\UniCon\Scout\Recovery`

- DHCP server log file

While the software packages are installed, a progress bar is shown on the screen. To switch from graphic mode to text mode, press CTRL+ALT+F4 to read texts or any error messages.

For problems and error messages, see the following table.

PROBLEM	REASON	SOLUTION
PXE recovery cannot be used for UEFI devices.	The eLux version being used does not support PXE recovery for UEFI devices.	Use current eLux and Scout versions. For further information, see DHCP configuration.
When starting the PXE recovery, the device reports a DHCP timeout.	The DHCP server does not respond.	*Check the network connection. In the DHCP server log file, check whether the client has received an IP address. Modify your DHCP server if necessary.
The device starts the PXE recovery but then boots normally or reports a TFTP timeout: TFTP open timeout	The TFTP server does not respond.	Check if the TFTP service is available. Analyze the log file of the TFTPD daemon. Check the entries for the router/ gateway and boot server of your DHCP service.
After starting the PXE recovery, the device reports a TFTP error: TFTP Error – File not found	The TFTP server failed to send the boot file (pxelinux.0)	Check the boot file entry of your DHCP server. Analyze the TFTP log file. Check the access rights for the TFTP server root directory.
The recovery stops and a black screen is displayed with the following message: could not find image: eluxrp.krnboot	The TFTP server failed to provide the eluxrp.krn file.	Analyze the TFTP log file.* Check the access rights and whether the recovery files are available in the TFTP server root directory.
The recovery stops and a black screen is displayed with the following message: could	The TFTP server failed to provide the ramfs.rp file.	Analyze the TFTP log file.* Check the access rights and if the recovery files are

PROBLEM	REASON	SOLUTION
not find ramdisk image: ramfs.rp boot		available in the TFTP server root directory.
The recovery stops after the following messages are displayed: `ec = 406 ... elux-library.... failed` `http://user:password@webserverorfailed` `ftp://user:password@ftpserver`	The recovery image file cannot be transferred from the web or FTP server.	Wait for the FTP or HTTP timeout to occur and check whether the displayed address is valid: `http://user:passwort@webserver` or `ftp://user:passwort@ftpserver`. Modify the values in the Recovery settings of the Scout Console if necessary.

IEEE 802.1X authentication for eLux

Certificate-based logon with 802.1X

IEEE 802.1X is an IEEE standard for port-based Network Access Control in IEEE 802 networks. It provides an authentication mechanism for client devices (supplicants) wishing to attach to a LAN or WLAN. The supplicant provides credentials such as a digital certificate to an authenticator, and the authenticator forwards the credentials to an authentication server (RADIUS) for verification. The authenticator can be an IEEE 802.1X-capable Ethernet switch or wireless access point. If the authentication server determines the credentials are valid, the supplicant is allowed to access the protected side of the network.

As a RADIUS server, you can use the Microsoft Network Policy Server (NPS) or a freeware program such as freeRADIUS.

The supplicant is implemented as a software program. We support the free software wpa_supplicant.

The standard recommends the Extensible Authentication Protocol (EAP) or the PPP-EAP-TLS Authentication Protocol for authentication.

> **Note:**

> For certificate enrollment and management, you can use SCEP which is supported by eLux.

Certificate-based logon with 802.1X and TPM 2.0

In combination with SCEP, private keys on TPM 2.0 devices can be stored in the TPM 2.0 module. To further secure authentication, starting with eLux RP 6.2021, you have the option to have cryptographic keys generated inside the TPM 2.0 chip of a device. These keys can neither be displayed nor exported.

For further information, see Certificates for SCEP in the SCEP guide.

IEEE 802.1X authentication for eLux

Certificate-based logon with 802.1X

IEEE 802.1X is an IEEE standard for port-based Network Access Control in IEEE 802 networks. It provides an authentication mechanism for client devices (supplicants) wishing to attach to a LAN or WLAN. The supplicant provides credentials such as a digital certificate to an authenticator, and the authenticator forwards the credentials to an authentication server (RADIUS) for verification. The authenticator can be an IEEE 802.1X-capable Ethernet switch or wireless access point. If the authentication server determines the credentials are valid, the supplicant is allowed to access the protected side of the network.

As a RADIUS server, you can use the Microsoft Network Policy Server (NPS) or a freeware program such as freeRADIUS.

The supplicant is implemented as a software program. We support the free software **wpa_supplicant**.

The standard recommends the Extensible Authentication Protocol (EAP) or the PPP-EAP-TLS Authentication Protocol for authentication.

> **Note:**
>
> For certificate enrollment and management, you can use SCEP which is supported by eLux.

Certificate-based logon with 802.1X and TPM 2.0

In combination with SCEP, private keys on TPM 2.0 devices can be stored in the TPM 2.0 module. To further secure authentication, starting with eLux RP 6.2021, you have the option to have

cryptographic keys generated inside the TPM 2.0 chip of a device. These keys can neither be displayed nor exported.

For further information, see Certificates for SCEP in the SCEP guide.

Configuring 802.1X for eLux

> **Note:**

> 802.1X can be configured for LAN or WLAN.The procedure differs in the location of the configuration file and in some parameters.For further information, see Configuring WPA supplicant.

1. Make sure that the eLux package **WLAN drivers** and the included feature package **WPA supplicant** are installed on the devices. This may require modifications of the image definition file on the web server via ELIAS.
2. Transfer the required certificates to the devices to /setup/cacerts by using the Scout feature Files configured for transfer.

   > **Note:**

   > The certificates of your RADIUS environment require the FQDN for the Common Name (CN).

3. Configure the file wpa.conf, and then transfer the configuration file to the devices by using the Scout feature Files configured for transfer.For further information, see Configuring WPA supplicant.
4. If you use SCEP, alternatively generate the file wpa.conf from the template wpa.conf.scep.For further information, see WPA configuration via template.
5. Only LAN: In the Scout Console, for the relevant OU, in the device configuration under **Network > LAN > Advanced > IEEE 802.1X authentication**, select the **Activate** option.

If the configuration is correct and the required certificates are rolled out, you can use devices on the 802.1X port.

Configuring WPA supplicant

> **Note:**
>
> You can use the example files on the devices to configure the WPA supplicant: `/setup/scep/wpa.conf`.

1. Create an individual wpa.conf configuration file.

   By default, the file contains the following information:

```
ctrl_interface=/var/run/wpa_supplicant

ctrl_interface_group=0

ap_scan=0

network={

    key_mgmt=IEEE8021X

    eapol_flags=0

    eap=TLS

    identity=`"<Common Name as specified in certificate>"`

    priority=6

    ca_cert="/setup/cacerts/scep/serverca.pem"

    client_cert="/setup/cacerts/scep/client.pem"

    private_key="/setup/cacerts/scep/client.key"

        }
```

   Add further entries according to your CA implementation.

You can specify the certificate file and path in one of the following ways, for example if you access an external root certification authority. Note that only one file may be referenced via `ca_cert`. However, this file may contain several certificate entries.

```
ca_cert="/setup/cacerts/<root_extern>.pem"

ca_cert="/setup/cacerts/<subordinate_int>.pem"

ca_cert=`/setup/cacerts/<radius>`.ssl"
```

If the RADIUS certificate contains the NetBIOS name instead of the FQDN, you may use the following entry:

```
ca_cert=`/setup/cacerts/<root>`.pem
```

Important:

The spelling and case-sensitivity of the certificate file names must be identical to the names of the transferred certificate files.

2. If you want to use TPM 2.0 via WLAN, add the following engine parameters for the network:1 from eLux RP 6 2103

```
network={

ssid="WLAN-ABC"

scan_ssid=1

key_mgmt=WPA-EAP

proto=WPA2

eap=TLS

engine=1            # Value must always be 1

engine_id="tpm2tss"    # Private Key is taken from TPM 2.0 module
```

```
dentity="__IDENTITY__"

priority=6

ca_cert="/setup/cacerts/scep/serverca.pem"

client_cert="/setup/cacerts/scep/client.pem"

private_key="/setup/cacerts/scep/client.key"    # Public part

      }
```

3. To transfer the `wpa.conf` file to the devices, use the Scout feature **Files configured for transfer**. Use the following destination:

LAN	`setup/scep/`
WLAN	`setup/wlan/`

For further information, see Advanced device configuration > Files in the **Scout** guide.

Use of multiple WiFi networks

- To use multiple SSIDs, set the **network** entry multiple times.

Example:

```
```

ctrl_interface=/var/run/wpa_supplicant

ctrl_interface_group=0

ap_scan=1

network=

{ ssid="" scan_ssid=1 key_mgmt=WPA-EAP eap=TLS identity="" priority=5
ca_cert="" ca_cert="" client_cert="" private_key="" }

network=

```
{ ssid="" scan_ssid=1 key_mgmt=WPA-EAP eap=TLS identity="" priority=6
ca_cert="" ca_cert="" client_cert="" private_key="" }

```

Use of variables

For identity and host name, you can alternatively set variables:

SPELLING	DESCRIPTION	OTHER
%IDENTITY%	Common name from certificate	For compatibility reasons, the legacy spelling before eLux RP 6.9.100 (variable name in uppercase letters and 2 x 2 underscores is still supported). Example: **IDENTITY**
%HOSTNAME%	Hostname from `terminal.ini`	

Variables may also be used for a part of a value. Prefixes and suffixes of a variable are pure strings that are passed through. Example: `identity="host/%HOSTNAME%"`

For further information on configuring 802.1X for WLANs, see WPA support in the **Scout** guide.

WPA configuration via template (SCEP)

– only if SCEP is used –

> **Note:**

> The following information is related to 802.1X configuration of LANs.

> If you use SCEP, you can benefit from the template file /setup/scep/wpa.conf.scep provided on the devices:

```
ctrl_interface=/var/run/wpa_supplicant

ctrl_interface_group=0

ap_scan=0

network={

    key_mgmt=IEEE8021X

    eapol_flags=0

    eap=TLS

    identity="%IDENTITY%"

    ca_cert="/setup/cacerts/scep/serverca.pem"

    client_cert="/setup/cacerts/scep/client.pem"

    private_key="/setup/cacerts/scep/client.key"

        }
```

Note:

The %IDENTITY%1 Former spelling: `__IDENTITY__` variable is replaced by the CN of the client certificate.

The client evaluates this template and creates the wpa.conf file if the following requirements are met:

- SCEP is configured in the terminal.ini file of the client. For further information, see Configuring SCEP for eLux in the SCEP guide.
- There is no individual /setup/scep/wpa.conf file available.

If required, modify the template and transfer it to the devices to /setup/scep/wpa.conf.scep. To do so, use the Scout feature **Files configured for transfer**. Note that if you have an individual /setup/scep/wpa.conf file,2 see Configuring WPA supplicant it has precedence.

> **Note:**
>
> The `wpa.conf` file generated from the template is created in a temporary directory and can only be viewed via the diagnostic files.

Diagnosis for 802.1X

Show client certificate in a shell

- Use the following command: `openssl x509 -in /setup/cacerts/scep/client.pem -noout -text`

All information about the certificate is displayed.

View log files

1. For the relevant OU, in the device configuration, enable enhanced logging.
2. Request the diagnostic files. For further information, see Requesting diagnostic files in the **Scout** guide.

`/tmp/systemd-journal.log | Network log file`

Certificate handling with SCEP

Certificate handling with SCEP

SCEP is a protocol designed to simplify secure and scalable issuing of certificates to network devices in large-scale environments.

With SCEP, the devices take care of their certificates themselves. To do so, they require a one-time password created by an authorized person. With such a time-limited password (Windows: 60 minutes), a device can request a certificate from the SCEP service.

To prevent the certification authority (CA) from issuing arbitrary certificates with the one-time password, create a template in which you restrict the certificate classes.

By using SCEP you can simplify certificate issuing and management for uses such as

- 802.1X authentication for LAN and WLAN
- VPN connection via Cisco AnyConnect

- Connection via Citrix NetScaler® Gateway

The eLux® implementation is based on the **OpenSCEP** project. The following description requires the use of Windows NDES.

Windows as SCEP server

> **Note:**
>
> For Windows Server 2012 and later versions, NDES (Network Device Enrollment Service) is integrated in the Certification Authority (CA). The described procedure is to be seen as an example and can vary depending on the version and environment.

Installing NDES

1. Install the server role **AD CS** with the **Network Device Enrolment Service** feature .
2. Set up a service account for NDES which must be authorized on the CA later on.
3. Enter the information for the certificate from the Registration Authority (RA) to issue a signing certificate for this enrollment process.
4. Define the key length.

The new virtual directories are displayed in the IIS console below the default website.

Configuring templates

The SCEP server uses templates for submission to the CA.

1. Set the LDAP name of the template in the registry:

Default:	IPSecIntermediateOffline
EncryptionTemplate	`<templatename>`
GeneralPurposeTemplate	`<templatename>`
SignatureTemplate	`<templatename>`

2. Add the new template to the CA.
3. Configure the use of a one-time password in the registry (validity period, maximum number).

The one-time password allows a client that supports SCEP/NDES to contact the SCEP server directly and request a certificate. The client connects to the Windows SCEP server with the URL http://<CA_FQDN>/certsrv/mscep/mscep.dll

Configuring one-time password

Either

- disable passwords (option A): set to 0

or

- define fixed password (option B):

Permissions for NDES service account

- Set the permission to full control.

Modify SCEP application pool defaults

- Set **Load User Profile** option from false to true.

Requesting one-time passwords

1. Sign in to the SCEP Administrator website using your NDES service account: http://<CA_FQDN>/certsrv/mscep_admin/
2. Copy the challenge password to the configuration file scep.ini

Configuring SCEP for eLux®

How to enable eLux devices to use SCEP for certificate deployment:

> **Note:**

> The eLux package **Network Access Control** and the included feature package **SCEP** must be installed on the devices. This may require modifications of the image definition file on the web server via ELIAS.

1. To configure the SCEP agent, use the scep.ini file.Then select the Scout feature Files configured for transfer to transfer the configuration file to the devices.For further information, see Configuring SCEP ini file.

2. Ensure that the devices are running on the correct time.We recommend that you configure a time server.

   The local time must match the time of the certificate authority.

3. To activate the SCEP agent, add the following entry to the terminal.ini file on the devices.To do so, use the Scout feature **Advanced file entries**:

File	/setup/terminal.ini
Section	Network
Entry	SCEP
Value	true

   Please note case sensitivity. For further information, see Advanced file entriesin the **Scout** guide.

4. To use SCEP for 802.1X authentication for LAN, in the Scout Console, for the relevant OU, in the Network device configuration for the LAN profile, select **Advanced > IEEE 802.1X authentication > Activate**.For further information, see Configuring 802.1X for eLux in the **802.1X** guide.

   When using SCEP for other purposes, however, make sure that the above-mentioned option is not selected for the network profile **LAN**.

The client creates a WPA configuration file from the template `/setup/scep/wpa.conf.scep`*.However, if you have transferred an individual WPA configuration file to* `/setup/scep/wpa.conf`*, it will take precedence over the template.*

Certificates for SCEP

The SCEP agent stores the following certificates in /setup/cacerts/scep by default.

CLIENT.PEM	CLIENT CERTIFICATE	
client.key	Private key of client certificate	2048 or40961 from eLux RP 6 2204 RSA key may be stored TPM 2.0 chip
serverca.pem	Server CA certificate	

CLIENT.PEM	CLIENT CERTIFICATE	
serverca.key	Server CA key	required renewal of certificates
serverca.sig	Server CA signature	required renewal of certificates
serverra.pem	RADIUS server certificate	

Devices with TPM 2.0

The authentication procedures via WPA supplicant for wired Ethernet connections and via IEEE 802.1X for WLAN components consider both storage options for the SCEP private key – the file system and the TPM 2.0 chip.

Client certificates continue to be stored in the file system. The roll-out procedure, with which the devices initially receive the client certificates, is carried out exclusively via wired Ethernet connections. Existing certificates can be renewed both via wired Ethernet connections and via WLAN.

Also the private key file of the SCEP client certificate is stored in the file system, by default. It is located under /setup/cacerts/scep or in the directory you have defined in the scep.ini file. When you perform an eLux firmware update to a version with TPM 2.0 support, existing private key files of the SCEP client certificate remain stored in the file system.

Private key in TPM 2.0 chip

- To use the TPM 2.0 module for the private key, make the following Advanced file entry:

File	scep.ini	
Section	Certificate	
Entry	UseTPM2	
Value	true	Default is false.

This parameter ensures that the private key is generated for each TPM 2.0 device in its TPM module.2 from eLux RP 6 2101 It is stored there only and cannot be displayed or exported. The key never leaves the TPM chip of the device.

When the certificate is renewed, the private key is reused.

The SCEP agent creates a certificate signing request (CSR) via openssl with the public key of the TPM module. Each time a connection is set up to an 802.1X-secured port, the system checks for a valid certificate. The certificate must be signed with the private key of the TPM chip. If the certificate does not exist or is invalid, the connecting process is rejected.

The client.key file remains in the file system but only contains information on the public key. The retained file name allows devices without a TPM to use the same certificate.

To delete the key from the TPM 2.0 chip, delete the TPM in the BIOS. We recommend that you first revoke certificates that are still valid.

Configuring the SCEP ini file

> **Note:**
>
> You can use the example file on the devices to configure the SCEP agent: /setup/scep/scep.ini.sample

1. Create the text file scep.ini. Then insert the sections and make the entries described below.
2. To transfer the configuration file scep.ini to the devices under /setup/scep/, use the Scout feature Files configured for transfer. For further information, see Advanced device configuration > Files in the **Scout** guide.

The certificate request is created from the scep.ini data. You can optionally extend the certificate request with additional attributes. For further information, see Extended SCEP certificate request.

Sections and entries for scep.ini

SECTION	ENTRY	DESCRIPTION	EXAMPLE
Admin	URI	SCEP server address (URI)	`URI=http://ca.w2k12.samplete c-01.com/certsrv/mscep/`
	PROXY	Proxy server (optional)	PROXY=proxy.sampl etec-01.com:3800

SECTION	ENTRY	DESCRIPTION	EXAMPLE
	ReNew	Number of days before the certificate expires and the device requests a new certificate (time span for renewal) From this day on, the client tries to renew the certificate. The value must be less than the validity period of the certificate.	ReNew=30 (default)
	ReNewCheckOnlyClient1 from eLux RP 6 2302.1000	Alternative option to ReNew: Unlike ReNew, only the client certificate is checked, but not CA and RA certificates	ReNew=30 (Standard)
	ExpireCheck	Time interval in days, how often the ReNew date is checked	ExpireCheck=1 (default)
	challengePassword	One-time password for the request Valid for 60 minutes once only The password is deleted after the certificates have been successfully transferred.	(A): challengePassword=12345 (B): challengePassword=<password requested via http://CA_FQDN/certsrv/mscep_admin/>
Certificate	CNTYPE	Type (autoip,ip,dns,autodns,dnsfqdn,email)	CNTYPE=email

		autoip - The IP address of the device is used as **CN**.	
		ip - The IP address you specify as **CN** in the scep.ini (see next option) is used.	
		dns - The name you specify as **CN** in the scep.ini (see next option) is used.	
		autodns - The host name specified in the terminal.ini is used as **CN**.	
		dnsfqdn - The host name specified in the terminal.ini with the domain name appended is used as **CN**.	
		email - The email address you specify as **CN** in the scep.ini (see next option) is used.	
	CN	Certificate/Name Is filled by the system for CNTYPE=autoip,autodns,dnsfqdn	`CN=userxxx@sampletec-01.com`
	OU	Certificate/Attribute (optional)	OU=TestLab

	ORGANIZATION		ORGANIZATION=SampleTec
	LOCALITY		LOCALITY=Karlsruhe
	STATE		STATE=BW
	OU1	Further attributes for up to 6 OUs (optional).The OUs specified here can be used for a certificate request	OU1=Testlab
	OU2		OU2=KA_QA
	OU3		OU3=KA_DEV
	OU4		
	OU5		
	OU6		
	COUNTRY	Certificate/Country	COUNTRY=DE
	KEYLEN	Key length (from certificate template) Allowed values: 2048 or 4096 An incorrect key length leads to the event log entry **Incorrect Challenge Password** If a 4096 key cannot be generated in the TPM (for TPM 2.0 and UseTPM2), as a fallback a 4k RSA key is generated in the file system.	KEYLEN=2048
	CertStore	The certificate store can be freely selected. This parameter allows you to specify a new	CertStore=/setup/cacerts/scep

		directory that the SCEP agent creates on the device. If you specify the default path /setup/cacerts/scep, the private key for the SCEP client certificate for TPM 2.0 devices is stored in the TPM 2.0 module.	

Additional information

- The storage location can be defined individually for each certificate file.

The three file storage parameters have higher priority than the CertStore parameter. The specified directories must exist. Client certificate files can be merged.

- Access rights can be defined for the private key (if in file system).
- Certificates can be checked for validity according to the certificate revocation list (CRL)2 from eLux® RP 6 2101

Certificate	CA_Path	Path and file name of the server CA certificate	CA_Path=/setup/cacerts/scep/serverca.pem (default)
	Client_Path	Path and file name of the client certificate	Client_Path=/setup/cacerts/scep/client.pem (default)
	ClientKey_Path	Path and file name of the private key file of the client certificate	ClientKey_Path=/setup/cacerts/scep/client.key (default)

Note:

Specify the same value for **Client_Path** and **ClientKey_Path**, for example `/setup/cacerts/scep/client.pem`, if you want to merge the two files.

	RA_Path	Path and file name of the RADIUS server certificate	/setup/cacerts/scep/serverra.pem (default)
	MODE	For additional use of a RADIUS server certificate: RA Allowed values: CA (default), RA	CA
	ClientKey_Permission	Access rights for the private key file,0400 - Can only be read with root rights (default), 0444 - Can also be read by users	ClientKey_Permission=0400
	UseTPM2	For devices with TPM 2.0 chip From eLux RP 6 2101: The private key is generated in the TPM 2.0 module and remains there. eLux RP 6.10 and 11: The private key file is stored in the TPM 2.0 chip. Allowed values: false (/en-us/unicon-elux-scout/2603-ltsr-cu1/short_guides/scep/scep_config_elux/	UseTPM2=true

		scep_config_certificate.html).	
	TPM2Fallback3 from eLux RP 6 2204	If the private key cannot be stored in the TPM 2.0 module (for example, on devices with other TPM versions), it is stored in the file system. Allowed values: true (default), false false: There is no fallback. On devices without TPM 2.0, the key is not stored at all.	TPM2Fallback=true
	CrlCheck4 from eLux RP 6 2101	Optionally, specify up to five certificates (in addition to CA_Path, Client_Path and RA_Path) you want the SCEP agent to check for validity according to CRLs. The check is done each time a connection is set up. If the connection remains active for longer, the check is carried out after the time interval you set under ExpireCheck.	CrlCheck1=/setup/cacerts/myrootca.pem CrlCheck2=/setup/cacerts/myintermediate.pem

	CrlCheckEnabled5 from eLux RP 6 2110	true (default): All certificates are checked for validity according to CRLs false: No revocation check is done.	CrlCheckEnabled=true
	FirstCrlOnly6 from eLux RP 6 2204	true: If multiple CRLs are registered, only the first one is downloaded and checked. This reduces the load on the PKI infrastructure. false (default): All existing CRLs are downloaded and checked.	FirstCrlOnly=true

Extended SCEP certificate request

Note:

Use this function to configure the SCEP agent with additional attributes that are not included with the scep.ini file.

The certificate request is created from the values in the scep.ini file and is available on the device in the file `/setup/scep/clientreq.conf.in.` This file is installed with SCEP as a template and contains the following entries by default:

```
[req]

prompt=no
```

```
distinguished_name=req_distinguished_name

string_mask=nombstr

attributes=req_attributes



[req_attributes]

challengePassword=__CHALLENGEPASSWORD__



[req_distinguished_name]

C=__COUNTRY__

ST=__STATE__

L=__LOCALITY__

O=__ORGANIZATION__

OU=__OU__

1.OU=__OU1__

2.OU=__OU2__

3.OU=__OU3__

4.OU=__OU4__

5.OU=__OU5__

6.OU=__OU6__

CN=__CN__



[__X509V3__]
```

```
subjectAltName=critical,__CNTYPE__:__ALTNAME__
```

The fields marked with underscores are parameters and are replaced by the relevant values in scep.ini.

If the attributes provided in the scep.ini are sufficient for your purposes, there is no need to edit the clientreq.conf.in file.

Extending certificate request with additional attributes

1. Retrieve the /setup/scep/clientreq.conf.in file installed with SCEP from a device, for example by using the Diagnostics feature.
2. Edit the file. Add any other openSSL sections and attributes. For further information, see `<https://www.openssl.org/docs/man1.0.2/man1/openssl-req.html>`
3. Transfer the file to the relevant devices under /setup/scep/clientreq.conf.in by using the Scout feature Files configured for transfer.

This file – not the scep.ini – is the source for the certificate request. Only delete fields that you do not need!.

Diagnosis for SCEP

Show client certificate in a shell

Use the following command:

```
openssl x509 -in /setup/cacerts/scep/client.pem -noout -text
```

All information about the certificate is displayed.

View log files

1. For the relevant OU, in the device configuration, enable enhanced logging.
2. Request the diagnostic files. For further information, see Requesting diagnostic files in the **Scout** guide.

/tmp/systemd-journal.log	Network log file

/setup/logs/scepagent.log1 temporary version under /var/log/scepagent.log	Log file of the SCEP agent, contains last certificate transfer This file is not included in the template #System, but must be defined in an individual template. For further information, see Device diagnostics in the **Scout** guide.

Performing the migration

The migration from eLux® RP 6 to eLux 7 can be carried out easily via a firmware update. Below you will find details on the requirements and the procedure.

Prerequisites

- The migration enablement 1.6.3-1 package or any newer version is available in the UC_RP6_X64 container and is defined in the current eLux RP 6 image.
- The following migration files are available in the UC_ELUX7 container:

```
- `migrationfiles-<version>.tar.gz`

- `migration-system-<version>.idf`

- recovery_systemEFI.rdf
```

- The eLux 7 target image (.idf) is defined in the UC_ELUX7 container and has the same name as the current eLux RP 6 image (.idf).
- In the device configuration, the container variable `__CONTAINER__` is specified for the firmware path.

Preparing the migration

The migration enablement package for eLux RP 6 required for migration and the migration files for eLux 7 are provided via a separate download link.

Web-based ELIAS 18

1. Import the migration enablement package via ELIAS 18 into the eLux RP 6 Container UC_RP6_X64.
2. Define the migration enablement package with activated FPM for the One Step Migration to the current eLux RP 6 image (.idf) in the UC_RP6_X64 container.

3. Import the archive `migration-package-<version>.zip` into the container UC_ELUX7.
4. Define the eLux 7 target image (.idf) in the UC_ELUX7 container.
    - The image should have the same name as the current eLux RP 6 image (.idf).
    - If a name other than the current eLux RP 6 image (.idf) is to be used for the eLux 7 target image (.idf), an Advanced file entry must be set before the migration:

File	Section	Key	Value
/setup/terminal.ini	UpdateInfo	MigrationIdfUrl	`<URL>`

    The URL must not contain the container variable `__CONTAINER__` in the firmware path. Instead, the container name must be specified. *Example: `https://MyWebServ.domain.com/elias/container/UC_ELUX7/MyImage.idf`

Legacy ELIAS

1. Import the migration enablement package via ELIAS into the eLux RP 6 Container UC_RP6_X64.
2. Define the migration enablement package with activated FPM for the One Step Migration to the current eLux RP 6 image (.idf) in the UC_RP6_X64 container.
3. Unzip the archive migration-package-`<version>.zip` and copy the migration files (migrationfiles-`<version>.tar.gz`, migration-system-`<version>.idf`, recovery_systemEFI.rdf) into the UC_ELUX7 folder.
4. Define the eLux 7 target image (.idf) in the UC_ELUX7 container.
5. The image should have the same name as the current eLux RP 6 image (.idf).
6. If a name other than the current eLux RP 6 image (.idf) is to be used for the eLux 7 target image (.idf), an Advanced file entry must be set before the migration:

File	Section	Key	Value
/setup/terminal.ini	UpdateInfo	MigrationIdfUrl	`<URL>`

    The URL must not contain the container variable `__CONTAINER__` in the firmware path.Instead, the container name must be specified. *Example: `https://MyWebServ.domain.com/eluxng/UC_ELUX7/MyImage.idf`

Important:

The migration files must not be changed manually.

The devices must not be switched off during the migration.

Devices should be connected via LAN, WLAN, VPN and IEEE 802.1X during the migration. Using WWAN connections (i.e. LTE) is not supported. During the migration via VPN the VPN-connection needs to be reconnected after each reboot.

Please do not use the firmware settings “Check for new image version on start / on shutdown” during the migration process.

This can lead to an incorrect device configuration (e.g. Citrix StoreFront application definition), which requires a remote factory reset.

Migration process

The migration is triggered by the installation of the migration enablement package with activated One Step Migration FPM on eLux RP 6, if the device fulfils the migration requirements for eLux 7.

The migration requirements to be checked are:

- 8 GB HDD or more
- 4 GB RAM or more
- Boot Mode UEFI
- eLux RP 6 versions: 2302 LTSR, 2209 CR, 2104 LTSR CU3, 2104 LTSR CU4

If the device does not meet the migration requirements, a new check is carried out each time the device is restarted as long as the migration enablement package is installed.

A system message informs the user that the migration is due in a few minutes. During the migration, an eLux 7 base image is installed first and the eLux 7 target image is installed automatically in a further step.

Note:

> Once the migration is complete, it is no longer possible to downgrade to an eLux RP 6 version with the old partition layout using a firmware update. Please use PXE recovery or USB stick recovery if a downgrade is required.

Migration from Windows to eLux 7

Migration from Windows to eLux 7

Overview

The **win2elux.exe** migration tool enables automated migration of endpoints running **Windows** to **eLux 7**.

When you execute win2elux.exe, a new partition is created on your local hard drive. This partition serves as the starting point for the migration process, where the eLux 7 operating system, as defined by the image, is installed.

The migration can be executed:

- Locally or from a network location
- As part of automation tools like **Microsoft SCCM** to enable deployment of win2elux.exe and remote initialization

After migration, the device boots directly into eLux 7.

How the migration works

The migration script must be executed on the device that needs to be migrated, when the migration script is executed:

1. A temporary recovery partition is created on the local disk.
2. Migration files are copied to the partition.
3. The device restarts automatically.
4. eLux 7 is installed using the specified Image Definition File (IDF).
5. The device boots into eLux.
6. Optional onboarding to Scout can occur automatically.

Requirements

Device requirements

- Device must be running Windows 10 or below. Windows 11 is **NOT supported**.
- Device must boot using **UEFI**.
- Legacy BIOS mode is not supported.
- Network connectivity must be available during migration.
- Sufficient disk space must exist for the recovery partition.

> **Important:**
>
> - Devices not booted in UEFI mode cannot be migrated.
> - Migration is supported whether Secure Boot or BitLocker are enabled or disabled.

Software requirements

1. Sign in to myelux.com.
2. Navigate to:

   Products & Downloads > Tools > Migration tools
3. Download the latest **win2elux.zip** package.

Perform the migration

Extract the migration tool

From the device to be migrated, extract the provided migration tool package to a known location on the computer:

Run the migration script

Open the Command Prompt as **Administrator** and run the tool with the required parameters.

Example 1: Migrate the Windows endpoint to the given eLux image and reboot automatically and begins installation.

Execute the script:

```
win2elux.exe URL=<IDF_URL>
```

Example:

```
win2elux.exe URL=https://repository.example.com/elux/image.idf
```

Example 2: Migrate the Windows endpoint to the given eLux image, write a log file, use a proxy and don't reboot

Example:

```
win2elux.exe URL=https://repository.example.com/elux/image.idf Log=migration.log
Proxy=http://proxy.citrix.com Port=8888 Reboot=0
```

win2elux.exe command line reference

```
Usage: win2elux.exe [URL=<URL>] [Proxy=<proxy>] [Port=<proxyport>] [ARCHIVE=<URL|
localpath>] [OnlyCheck=<0|1>] [Reboot=<0|1>] [Debug=<0|1>] [Log=<logfile>]
```

PARAMETER	INFO	DESCRIPTION
URL	Required	Define to the relevant software container and the image definition file (IDF) which specifies the scope of the installation
Proxy		Defines a proxy server, e.g. `1.1.1.1`
Port		Defines the port of the proxy server, e.g. `8090`
ARCHIVE	URL or local path	Path to the local win2elux.zip ramdisk file or URL when hosting remote
Log		Defines a storage path for the log file, such as `C:`

PARAMETER	INFO	DESCRIPTION
		`\Temp\migration.log`. By default, the log file is saved in the same directory as the win2elux tool. Alternatively, you can use a network drive to ensure a persistent log file.
OnlyCheck	`0` or `1`	Perform a dry run without any modifications (default is 0). This parameter verifies if the requirements for UEFI, network, disk space are met and generates a log file.
Reboot	`0` or `1`	Define if the endpoint should reboot after the installation (default is `1`) If the reboot option is `0`, the migration will not start automatically, but on the next boot
Debug	`0` or `1`	Verbose installation messages are not displayed in the console output by default (set to `0`). The installation process stops in case of errors and provides error messages for different scenarios. If the parameter is not provided, the device will restart in case of an error without displaying an error message.

Advanced configuration using RAM disk

A RAM disk configuration can be used when additional configuration is required during migration, such as:

- IEEE 802.1X authentication to initiate the first connection to the network during the migration process.

- HTTPS repositories requiring certificate trust.
- Zero-touch onboarding to Scout Server and Scout Cloud Gateway.

The configuration is provided as a RAM disk archive that **must be named `8021conf.xz` and included inside the migration package**. This archive contains the required configuration for the migration process and is loaded into memory during execution for temporary use by the migration tool. The RAM disk is created as a cpio archive compressed with XZ and **must be generated on a Linux-compatible system**, Windows Subsystem for Linux (WSL) can be used to create the RAM disk.

> **Tip:** You can create the directory structure for each configuration, by extracting the example archive `8021conf.xz` from the working folder using the following commands.
>
> ```
> mkdir conf && cd conf && xz -cd ../8021conf.xz | cpio -iv
> ```

> After executing the command above, the directory structure will be created under the `conf` folder.

Configuring 802.1X authentication for migration

This configuration allows the administrator to initiate device migration when connecting to a network that enforces 802.1X authentication. During the migration process, the device uses a generic certificate, defined in the `wpa.conf` file, to establish the initial network connection.

After the migration is completed, the device receives a configuration from the Scout Server that defines how it should connect going forward. The device then reconnects to the network using a device-specific certificate and configuration defined in `wpa.conf.elux`, enabling access to the Scout Server and retrieval of the initial configuration.

> **Note SCEP use cases:** If SCEP certificate support is required during migration to replace the generic certificate used in the migration process, the custom_scep_renew-1.1.0-1.UC_ELUX7-1.0.zip package must be included to enable the SCEP request.

> **Note:** Only include 802.1X configuration when required. Including it on a migration that will connect to non-802.1X networks will cause migration failure.

> Create the following directory structure with the required files:

```
.
├── etc
│   └── wpa_supplicant
│       ├── client.key
│       ├── client.pem
│       ├── serverca.key
│       ├── serverca.pem
│       ├── serverca.sig
│       ├── wpa.conf
│       └── wpa.conf.elux
```

- `certificate` files used by the migration tool to connect to the 802.1x network. The certificates used during and after migration **must have the same filename**.
- `wpa.conf` is used during migration process.
- `wpa.conf.elux` is applied after the migration to enable eLux to reach the Scout server.

Example configuration for `wpa.conf`:

```
ctrl_interface=/var/run/wpa_supplicant

ctrl_interface_group=0

ap_scan=0

network={

    key_mgmt=IEEE8021X

    eapol_flags=0
```

```
    eap=TLS
    identity="eLuxMigration"
    ca_cert="/etc/wpa_supplicant/serverca.pem"
    client_cert="/etc/wpa_supplicant/client.pem"
    private_key="/etc/wpa_supplicant/client.key"
    private_key_passwd="PASSWORD_HERE"
}
```

Example configuration for `wpa.conf.elux`:

```
ctrl_interface=/var/run/wpa_supplicant
ctrl_interface_group=0
ap_scan=0
network={
    key_mgmt=IEEE8021X
    eapol_flags=0
    eap=TLS
    identity="eLuxMigration"
    ca_cert="/setup/cacerts/scep/serverca.pem"
    client_cert="/setup/cacerts/scep/client.pem"
    private_key="/setup/cacerts/scep/client.key"
    private_key_passwd="PASSWORD_HERE"
}
```

After successfully migrate the device, the client and server certificates defined above will be deleted from the device.

Adding SSL certificates for migration

This configuration allows the migration tool to trust the web server hosting the IDF image used to prepare the device during migration. If the web server is not trusted, the migration process will fail when attempting to access the website to download the required packages.

To establish HTTPS trust during the migration process, create the following directory structure and add the web server's root certificate.

```
.
├── etc
│   └── ssl
│       └── certs
│           ├── sslrootcertificate.pem
```

Enabling zero-touch onboarding

This configuration automatically onboards the device to the Scout Server or Scout Cloud Gateway, eliminating the need for users to manually enter the server URL and token during the onboarding process.

Create the json file and save as `zero_touch_onboarding.json` under the following structure:

```
.
├── user
│   └── setup
│       └── elux
│           ├── zero_touch_onboarding.json
```

Example of `zero_touch_onboarding.json` file for onboarding through SCG:

```
{

  "scgUrl": "scg.example.com",

  "scgToken": "onboarding-token"

}
```

Example of `zero_touch_onboarding.json` file for onboarding through Scout Server:

```
{

  "scgUrl": "scout-server.example.com",

  "scgToken": ""

}
```

Note: This configuration applies only to onboarding initiated through the migration tool. After a factory reset, the device will not automatically trigger onboarding again.

Create the RAM disk configuration archive

Generate the RAM disk archive:

```
./initrd-xz.sh <conf-folder> 8021conf.xz
```

Example:

```
./initrd-xz.sh /tmp/migration_tool/conf 8021conf.xz
```

Tip: (Optional) In case you can check the content of the `8021conf.xz` file, execute the following command:

```
xz -dc 8021conf.xz
```

Add it to the migration package:

```
zip win2elux.zip 8021conf.xz
```

Running the migration script with RAM disk package

From the Windows device to be migrated, execute the script:

```
win2elux.exe URL=<IDF_URL> ARCHIVE=<win2elux.zip path or URL>
```

Example:

```
./win2elux.exe URL=https://repository.example.com/elux/image.idf
ARCHIVE=win2elux.zip
```

The device restarts automatically and begins installation.

Debugging

When calling the win2elux.exe to perform the migration, you can enable the debugging by adding the `DEBUG=1` parameter. In case the migration fails due to 802.1X issues, you will be dropped into a debug shell and check for the log file `/tmp/wpasupplicant.log` for more details and any other log file under `/tmp/*`.

```
win2elux.zip URL=<IDF_URL> DEBUG=1
```

Logs are written to:

```
/tmp/
```

Example log:

```
/tmp/wpasupplicant.log
```

Use debug logs to diagnose:

- Network authentication failures
- Download issues

- Migration interruptions

Result of migration

After successful migration:

- eLux replaces the existing operating system.
- Temporary migration files are removed, including certificates used for 802.1x authentication when configured.
- The device boots directly into eLux.
- The device can automatically onboard to Scout if configured.

Migration from IGEL OS 11 or Stratodesk NoTouch OS to eLux 7

Migration from IGEL OS 11 or Stratodesk NoTouch OS to eLux® 7

Overview

The **igel2elux** migration tool enables automated migration of endpoints running **IGEL OS 11** or **Stratodesk NoTouch OS** to **eLux 7**.

The migration installs eLux directly on the device by creating a temporary recovery environment from which the operating system is deployed.

The migration can be executed:

- Centrally through **IGEL Universal Management Suite (UMS)**
- Through **Stratodesk NoTouch Center**
- Locally or from a network location

After migration, the device boots directly into eLux 7.

How the migration works

The migration script must be executed on the device that needs to be migrated, when the migration script is executed:

1. A temporary recovery partition is created on the local disk.
2. Migration files are copied to the partition.

3. The device restarts automatically.
4. eLux 7 is installed using the specified Image Definition File (IDF).
5. The device boots into eLux.
6. Optional onboarding to Scout can occur automatically.

Requirements

Device requirements

- Device must boot using **UEFI**
- Legacy BIOS mode is not supported
- Network connectivity must be available during migration
- Sufficient disk space must exist for the recovery partition

> **Important:**
>
> Devices not booted in UEFI mode cannot be migrated.

Software requirements

1. Sign in to myelux.com.
2. Navigate to:

   Products & Downloads > Tools > Migration tools
3. Download the latest **igel2elux.zip** package.

Perform the migration

Extract the migration tool

From the device to be migrated, unzip the provided migration tool package to a known location on the computer:

```
unzip igel2elux-<version>.zip -d /tmp/migration_tool
```

In the above example, /tmp/migration_tool is used as the working directory. The following steps in this article assume all commands are executed from this directory.

Set execute permissions for igel2elux.sh

Simply having the migration script on the device is not sufficient. The script must have executable permissions before it can be run.

After uploading the `igel2elux.sh` script to the device, set the correct permissions using the following command:

```
chmod +x igel2elux.sh
```

Run the migration script

Execute the script:

```
./igel2elux.sh --url <IDF_URL>
```

Example:

```
./igel2elux.sh --url https://repository.example.com/elux/image.idf
```

The device restarts automatically and begins installation.

> **Tip:**
>
> When executing the migration from **IGEL Universal Management Suite (UMS)**, ensure that **read, write, and execute permissions** are assigned to both the owner and others for the `igel2elux.sh` file.

igel2elux.sh command line reference

Usage: `igel2elux.sh [--url <URL>] [--proxy <proxy>] [--port <proxyport>] [--archive <URL | local path>] [--noreboot] [--log <logfile>] [–-debug]`

PARAMETER	DESCRIPTION
`--url <URL>`	Define to the relevant software container and the image definition file (IDF) which specifies the scope of the installation

PARAMETER	DESCRIPTION
`--proxy <proxy>`	Defines a proxy server
`--port <port>`	Defines the port of the proxy server
`--log <logfile>`	Defines a storage path for the log file. By default, the log file is saved in the same directory as the win2elux tool.
`--noreboot`	Does not restart the system after installation of the recovery image and the boot loader.
`--debug`	writes debug information in `/tmp/`.
`--archive`	Path to igel2elux.zip (URL or local path)

Advanced configuration using RAM disk

A RAM disk configuration can be used when additional configuration is required during migration, such as:

- IEEE 802.1X authentication to initiate the first connection to the network during the migration process.
- HTTPS repositories requiring certificate trust.
- Zero-touch onboarding to Scout Server and Scout Cloud Gateway.

The configuration is provided as a RAM disk archive that **must be named `8021conf.xz` and included inside the migration package**. This archive contains the required configuration for the migration process and is loaded into memory during execution for temporary use by the migration tool. The RAM disk is created as a cpio archive compressed with XZ and **must be generated on a Linux-compatible system**.

> **Tip:**

> You can create the directory structure for each configuration, by extracting the example archive `8021conf.xz` from the working folder using the following commands.

```
# mkdir conf && cd conf && xz -cd ../8021conf.xz | cpio -iv
```

After executing the command above, the directory structure will be created under the `conf` folder.

Configuring 802.1X authentication for migration

This configuration allows the administrator to initiate device migration when connecting to a network that enforces 802.1X authentication. During the migration process, the device uses a generic certificate, defined in the `wpa.conf` file, to establish the initial network connection.

After the migration is completed, the device receives a configuration from the Scout Server that defines how it should connect going forward. The device then reconnects to the network using a device-specific certificate and configuration defined in `wpa.conf.elux`, enabling access to the Scout Server and retrieval of the initial configuration.

Note SCEP use cases:

If SCEP certificate support is required during migration to replace the generic certificate used in the migration process, the `custom_scep_renew-1.1.0-1.UC_ELUX7-1.0.zip` package must be included to enable the SCEP request. **Note:**

Only include 802.1X configuration when required. Including it on a migration that will connect to non-802.1X networks will cause migration failure.

Create the following directory structure with the required files:

```
.

├── etc

│   └── wpa_supplicant

│       ├── client.key

│       ├── client.pem

│       ├── serverca.key
```

```
|       ├── serverca.pem
|       ├── serverca.sig
|       ├── wpa.conf
|       └── wpa.conf.elux
```

- `certificate` files used by the migration tool to connect to the 802.1x network. The certificates used during and after migration **must have the same filename**.
- `wpa.conf` is used during migration process.
- `wpa.conf.elux` is applied after the migration to enable eLux to reach the Scout server.

Example configuration for `wpa.conf`:

```
ctrl_interface=/var/run/wpa_supplicant

ctrl_interface_group=0

ap_scan=0

network={

    key_mgmt=IEEE8021X

    eapol_flags=0

    eap=TLS

    identity="eLuxMigration"

    ca_cert="/etc/wpa_supplicant/serverca.pem"

    client_cert="/etc/wpa_supplicant/client.pem"

    private_key="/etc/wpa_supplicant/client.key"

    private_key_passwd="PASSWORD_HERE"

}
```

Example configuration for `wpa.conf.elux`:

```
ctrl_interface=/var/run/wpa_supplicant

ctrl_interface_group=0

ap_scan=0

network={

    key_mgmt=IEEE8021X

    eapol_flags=0

    eap=TLS

    identity="eLuxMigration"

    ca_cert="/setup/cacerts/scep/serverca.pem"

    client_cert="/setup/cacerts/scep/client.pem"

    private_key="/setup/cacerts/scep/client.key"

    private_key_passwd="PASSWORD_HERE"

}
```

After successfully migrate the device, the client and server certificates defined above will be deleted from the device.

Adding SSL certificates for migration

This configuration allows the migration tool to trust the web server hosting the IDF image used to prepare the device during migration. If the web server is not trusted, the migration process will fail when attempting to access the website to download the required packages.

To establish HTTPS trust during the migration process, create the following directory structure and add the web server’s root certificate.

```
.
├── etc
│   └── ssl
│       └── certs
│           ├── sslrootcertificate.pem
```

Enabling zero-touch onboarding

This configuration automatically onboards the device to the Scout Server or Scout Cloud Gateway, eliminating the need for users to manually enter the server URL and token during the onboarding process.

Create the json file and save as `zero_touch_onboarding.json` under the following structure:

```
.
├── usr
│   └── setup
│       └── elux
│           ├── zero_touch_onboarding.json
```

Example of `zero_touch_onboarding.json` file for onboarding through SCG:

```
{
  "scgUrl": "scg.example.com",
  "scgToken": "onboarding-token"
}
```

Example of `zero_touch_onboarding.json` file for onboarding through Scout Server:

```
{

  "scgUrl": "scout-server.example.com",

  "scgToken": ""

}
```

Note:

This configuration applies only to onboarding initiated through the migration tool. After a factory reset, the device will not automatically trigger onboarding again.

Create the RAM disk configuration archive

Generate the RAM disk archive:

```
./initrd-xz.sh <conf-folder> 8021conf.xz
```

Example:

```
./initrd-xz.sh /tmp/migration_tool/conf 8021conf.xz
```

Tip: (Optional)

In case you can check the content of the `8021conf.xz` file, execute the following command: `bash` `xz -dc 8021conf.xz`

Add it to the migration package:

```
zip igel2elux.zip 8021conf.xz
```

Running the migration script with RAM disk package

Execute the script:

```
./igel2elux.sh --url <IDF_URL> --archive <igel2elux.zip path or URL>
```

Example:

```
./igel2elux.sh --url https://repository.example.com/elux/image.idf --
archive igel2elux.zip
```

The device restarts automatically and begins installation.

Debugging

When calling the igel2elux script to perform the migration, you can enable the debugging by adding the `--debug` parameter. In case the migration fails due to 802.1X issues, you will be dropped into a debug shell and check for the log file `/tmp/wpasupplicant.log` for more details and any other log file under `/tmp/*`.

```
./igel2elux.sh --debug
```

Logs are written to:

```
/tmp/
```

Example log:

```
/tmp/wpasupplicant.log
```

Use debug logs to diagnose:

- Network authentication failures
- Download issues
- Migration interruptions

Result of migration

After successful migration:

- eLux replaces the existing operating system.

- Temporary migration files are removed, including certificates used for 802.1x authentication when configured.
- The device boots directly into eLux.
- The device can automatically onboard to Scout if configured.

Overview

UEFI/BIOS update

By using the Scout Console, you can perform an update of the UEFI or legacy BIOS of your devices remotely. For UEFI devices, the UEFI Capsule Update procedure is supported.

Download the latest manufacturer-specific UEFI or legacy BIOS firmware data from the relevant platform (such as the LVFS portal) or the manufacturer's homepage and provide them on your web server.

Updating UEFI devices to a newer UEFI firmware works in analogy to a software update and also supports configurable control by end users (defer and cancel). Administrators configure their devices in the Scout Console accordingly and start the UEFI update with the **UEFI update** Scout command. The UEFI update on the end points is performed via the standardized Capsule Firmware Update procedure.

Alternatively, for legacy BIOS devices, you have the option of performing a BIOS update via the respective tool of the corresponding manufacturer. As an administrator, you start the legacy BIOS update via a user-defined Scout command with the parameters relevant for the respective manufacturer tool. The legacy BIOS update on the end device is performed via the respective manufacturer tool. The required manufacturer tools are provided through the companies Fujitsu, HP and Dell. These tools are integrated by Unicon into an eLux software package and made available on our myelux.com portal.

The devices can be equipped with different hardware.

UEFI/BIOS update for devices with encrypted system partition

Updating the UEFI or BIOS of devices with an encrypted system partition can be done via the described procedures without any problems. The only exception is Fujitsu FUTRO devices, regardless of whether they are UEFI devices or legacy BIOS devices. Any attempt to update the UEFI or BIOS of these devices, if encrypted, is therefore prevented by eLux® and documented in the log file.

UEFI/BIOS configuration

In addition to a UEFI/BIOS update, some devices allow for configuring their BIOS remotely and, for example, setting a new BIOS password. This option depends on the hardware model.

Reference BIOS

Some devices (Fujitsu, Dell/Wyse, HP) allow for copying and transfer of a reference BIOS to other devices of the same type. Via a reference BIOS, new manufacturer firmware and/or a reference configuration can be transferred to other devices.

Disclaimer

Please note that Unicon supplies manufacturer-specific BIOS tools, updates and data exclusively for the purpose of software packaging and simplified logistics of provision. These manufacturer-specific BIOS tools, updates and data are taken over by Unicon from the respective manufacturer "as is", are not changed and passed on to you free of charge. This means that Unicon neither has influence on the creation nor knowledge of the concrete contents and functionalities of such manufacturer-specific BIOS tools, updates and data and therefore neither assumes responsibility for their individual suitability and/or usability nor responsibility for the consequences of implementing such manufacturer-specific BIOS tools, updates and data on your systems. Any liability and/or warranty of Unicon for such manufacturer-specific BIOS tools, updates and data shall therefore be expressly excluded, unless warranty or liability claims result from Unicon's own willful or fraudulent conduct in the context of passing on the BIOS tools, updates and data or from any other mandatory, non-negotiable, statutory reason for liability.

Requirements

Supported devices / UEFI

Dell	Wyse 3040, 5070, 5470 Latitude 3410, 3330, 3420, 3440, 5420, 5430, 5520, 5530, 5540 Optiplex 3000, 3090 Ultra, 5060 Precision 3260, WD19
Fujitsu	S 540, S 5010, S 5011, S 740, S 7010, S 7011, S 940, S 9010, S 9011 Q 940, LIFEBOOK E 5512, E 5410, E 5412, E 5511
Hewlett Packard	USB-C Dock G5 Device Update, Thunderbolt G4

Lenovo	ThinkCentre M75n, M625q ThinkPad X260, L14 Gen 1, L14 AMD, L14 Gen4

For a list of the devices that support the UEFI Capsule Update standard, refer to the **Linux Vendor Firmware Service** (LVFS): `<https://fwupd.org/lvfs/devicelist>`

This platform also serves as an online repository to which hardware manufacturers upload new firmware packed in .cab archives.

If UEFI Capsule Update is not yet supported by the vendor and the UEFI device is not listed via LVFS, use the procedure for legacy BIOS devices, provided a corresponding manufacturer tool is available.

Supported devices / legacy BIOS

Dell	Wyse 5020
Fujitsu	S 540, S 740, S 940, S 5011, S 7011, S 9010
Hewlett Packard	mt21, mt45, mt645, t430, t530, t610, t640, t740

Requirements for performing a UEFI/BIOS update via Scout

- eLux® RP 6
- Scout Enterprise Management Suite® including ELIAS
- Web server (IIS) for UEFI/BIOS firmware of the hardware vendors

> **Important:**

> Do not perform the updates described, unless you have extensive experience with our software solutions Scout and eLux.

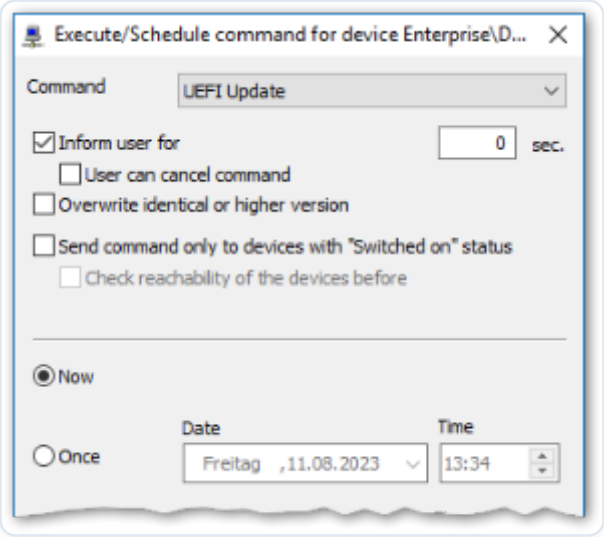
UEFI update in analogy to firmware update

– from Scout 15 2107 and eLux® RP 6 2107 –

For UEFI devices, you can conveniently update the UEFI firmware via a dedicated Scout command. The procedure is the same as for firmware updates (software): In a *.udf file you define the assignment of vendor-specific UEFI firmware to the device model types used. This UEFI file and the vendor-specific UEFI firmware archives are located in the eLux container. Finally, the UEFI file is referenced in the devices' device configuration. This allows UEFI devices that receive a UEFI update command to update to the appropriate UEFI firmware.

UEFI updates can be applied to individual devices, OUs, Dynamic Device Groups or devices selected in the **All devices** window. Administrators may schedule a UEFI update using the command dialog or create a UEFI update notification that will be evaluated on the next connection to the devices.

The options defined in the device configuration under **Firmware > Reminder settings** for users to defer an update are applied to both, firmware updates (software) and UEFI updates. Note that as with firmware updates, the administrator needs to additionally enable the **Inform user** option for each update command.



Since the UEFI update procedure and firmware update procedure (software) are structured similarly, the same features apply for both procedures:

- Initialization via Scout command or notification
- Update start can be controlled by users with appropriate configuration (**Firmware > Reminder settings** and user options in command dialog / notification).
- In the Scout Console, the update status is displayed for each device in its **Properties** window. Click to define which fields you want to show.
- Each update process is logged. For further information, see Command results and update information in the Scout guide.

- If an update fails, no system-side efforts will be made to retry.

> **Note:**
>
> The UEFI of Fujitsu devices with an encrypted system partition cannot be updated. Any attempt to do so will be stopped by eLux and documented in the log file. Whether a downgrade can be performed and to which version is specified by the manufacturer in the .cab file.

- Preparing UEFI devices
- Providing new UEFI firmware
- Configuring UEFI updates
- Performing UEFI updates via Scout command
- Performing UEFI updates via notification
- Logging of UEFI updates

Preparing UEFI devices

For updating UEFI systems, a special update daemon is required, which is provided by Unicon™ in an eLux software package. This package needs to be integrated into the image of your devices.

1. From our myelux.com portal, under **Downloads > eLux Software Packages**, download the following software package:
   - **UEFI BIOS update daemon**
2. Import the packages into ELIAS. For further information, see Importing software packages in the **ELIAS 18** guide or Importing packages into a container in the **ELIAS** guide.
3. In ELIAS, integrate the software package into the relevant image.

   For further information, see Defining and pinning packages in the **ELIAS 18** guide or Creating an IDF in the **ELIAS** guide.
4. For the relevant devices, perform an eLux® firmware update to the updated image.

The eLux software package **UEFI BIOS update daemon** is installed on the devices.

Providing new UEFI firmware

The vendor-specific UEFI firmware data must be made available on your web server. The manufacturers provide the UEFI firmware of their devices on the LVFS platform in the form of .cab archives. The way the device types are listed is up to the individual vendors and is unfortunately not consistent.

For models that come with different chipsets, make sure to use the appropriate .cab archive for the UEFI update. Example: Dell Latitude 5530

The chipset installed in your model is displayed in the Scout Console in the device properties.

> **Note:**

> If you cannot find a specific UEFI device via LVFS, please contact the vendor. It is also possible that **UEFI capsule update** is not yet supported for this type of device.

1. For all relevant device types, download the latest UEFI firmware archives from the LVFS platform.

   To shorten the names of the .cab archives or make them more meaningful, you may modify the file names. Leave the file extension .cab.

   Example original name: `860fceb3052faa6955ff4f3dfe337ca1736e08515f5ac402a59c2af5b9bbf8d1-firmware.cab` Example short name: `Dell_Latitude_3330_1130.cab`
2. Store the archives on your web server inside the eLux® container.

   Alternatively, you may create a dedicated container for UEFI firmware. The corresponding path must then be specified in the UEFI file (*.udf).
3. Check the MIME type settings of your web server and add any missing entries:

.cab	application/vnd.ms-cab-compressed	default setting of IIS
.udf	text/plain	for UEFI updates via Scout **UEFI Update** command

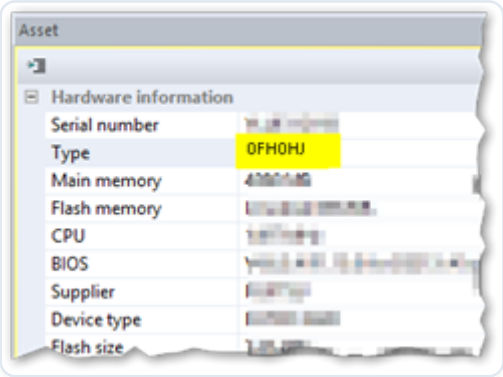
4. Follow the further steps in Configuring UEFI updates.

The new UEFI firmware is provided on the web server.

Configuring UEFI updates

The UEFI firmware archives need to be mapped to the relevant device types. This is done in a dedicated UEFI file which then is specified in the device configuration of all devices that are supposed to receive UEFI updates.

1. For all relevant device types, identify the type name. For a selected device, look up the Type name in the Scout Console, in the Asset window under **Type**.

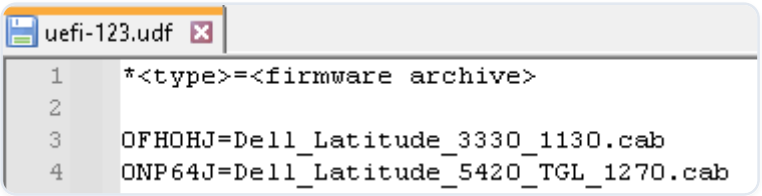


2. As a **UEFI file**, create a text file with the following:

   - File name extension *.udf
   - Any file name

   For each model type, insert a line with `<type>=<firmware archive>`

   Example: 0FH0HJ=Dell_Latitude_3330_1130.cab1 original name of the firmware archive changed



   If you have the UEFI firmware in a dedicated container, specify each model type with `<type>=<path/firmware archive>`

   Example: 0FH0HJ=../UC_UEFI/dell_latitude_3330/Dell_Latitude_3330_1130.cab2 dedicated UC_UEFI container with different directories

3. Import your UEFI file (*.udf) into your eLux® container on the web server.
4. To modify the device configuration, for the relevant devices, open **Device configuration > Firmware**. Under **UEFI file**, enter the name of your *.udf file with the file extension.

1	UEFI file in the eLux container
2	From the specified data, the system generates a URL that is used by the devices to update their UEFI.
3	Users may postpone a UEFI update just like a firmware update if configured under Reminders. For further information, see Update deferment by users in the **Scout** guide.

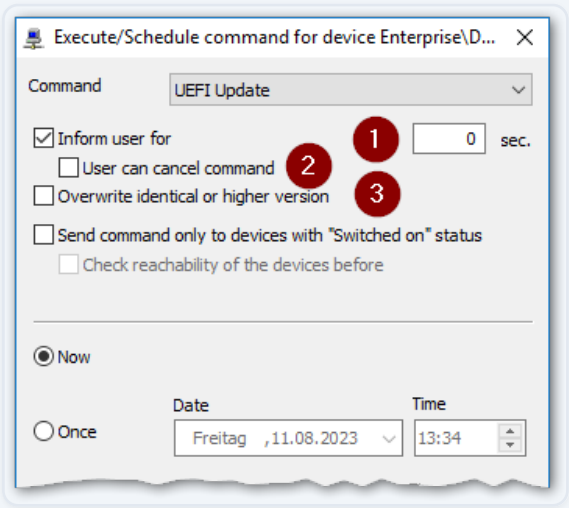
The UEFI file is located on your web server in the eLux container. In the device configuration of the relevant devices, the UEFI file is referenced.

> **Note:**
>
> UEFI files can be globally predefined by the administrator just like image files. For further information, see Predefined images and containers in the Scout guide.

Performing UEFI updates via Scout command

REQUIRES
• The relevant devices have the **UEFI BIOS update daemon** installed (/en-us/unicon-elux-scout/2603-ltsr-cu1/short_guides/elux_bios_update/uefi_update_gui/uefi_epm.html).
• The UEFI firmware archives are available on the web server (/en-us/unicon-elux-scout/2603-ltsr-cu1/short_guides/elux_bios_update/uefi_update_gui/uefi_binaries.html).
• The*.udf file is located in the eLux container and is referenced in the device configuration (/en-us/unicon-elux-scout/2603-ltsr-cu1/short_guides/elux_bios_update/uefi_update_gui/uefi_update_configure.html).

1. Select a device, an OU, a Dynamic Device Group or devices in the **All devices** window. .
2. On the context menu, click **Commandos > UEFI update**.
3. Edit the following options:

	
1	Users will be informed before a UEFI update is performed. The system message displayed provides users with rescheduling options depending on the configuration (**Firmware > Reminder settings**). See also (User information before update) in the Scout guide. Optionally, specify a display duration of the system message in seconds. With 0 seconds, the message will be shown until the user clicks one of the buttons.
2	Users are additionally allowed to cancel the command.
3	The UEFI system will be overwritten in any case. This option is mandatory for a downgrade.

4. Specify a time for the update process. For further information, see Executing commands in the Scout guide.
5. Click **Execute**.
6. Enter your device password.

The update process is triggered at the specified time. While the UEFI system is updated, for each device, the UEFI update in progress status is shown. In addition, detailed information about the current action is shown with a time stamp.

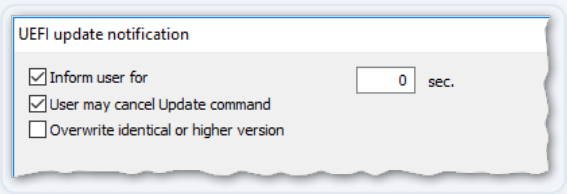
Performing UEFI updates via notification

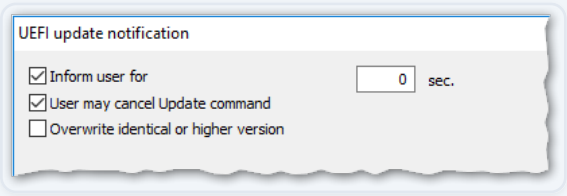
– from Scout 15 2107 and eLux® RP 6 2107 –

REQUIRES
• The relevant devices have the **UEFI BIOS update daemon** installed (/en-us/unicon-elux-scout/2603-ltsr-cu1/short_guides/elux_bios_update/uefi_update_gui/uefi_epm.html).
• The UEFI firmware archives are available on the web server (/en-us/unicon-elux-scout/2603-ltsr-cu1/short_guides/elux_bios_update/uefi_update_gui/uefi_binaries.html).
• The*.udf file is located in the eLux container and is referenced in the device configuration (/en-us/unicon-elux-scout/2603-ltsr-cu1/short_guides/elux_bios_update/uefi_update_gui/uefi_update_configure.html).

By using UEFI update notifications, you send an explicit one-time update request to selected devices that are evaluated with the next connection.

1. Select a device, an OU, a Dynamic Device Group or devices in the **All devices** window. .
2. On the context menu, click **Notifications > Initiate UEFI update**.
3. Edit the following options:

	
1	Users will be informed before a UEFI update is performed and are allowed to reschedule depending on the configuration.

	
2	Users are additionally allowed to cancel the command.
3	The UEFI system will be overwritten in any case. This option is mandatory for a downgrade.

The options correspond to the dialog options for execution by command, see Performing UEFI updates via Scout command.

4. Confirm the notification and confirmation.

The notifications for UEFI updates are activated for the relevant devices.

For each device, in the **Properties** window, the UEFI update notification field shows the status Activated.

Deleting a UEFI update notification for one or more devices

UEFI update notifications can be deleted before the firmware has been updated:

- On the context menu, click **Notifications > Delete UEFI update notification**.

> **Note:**
>
> Notifications are always set or deleted for all selected devices regardless of whether they are only available for individual devices.

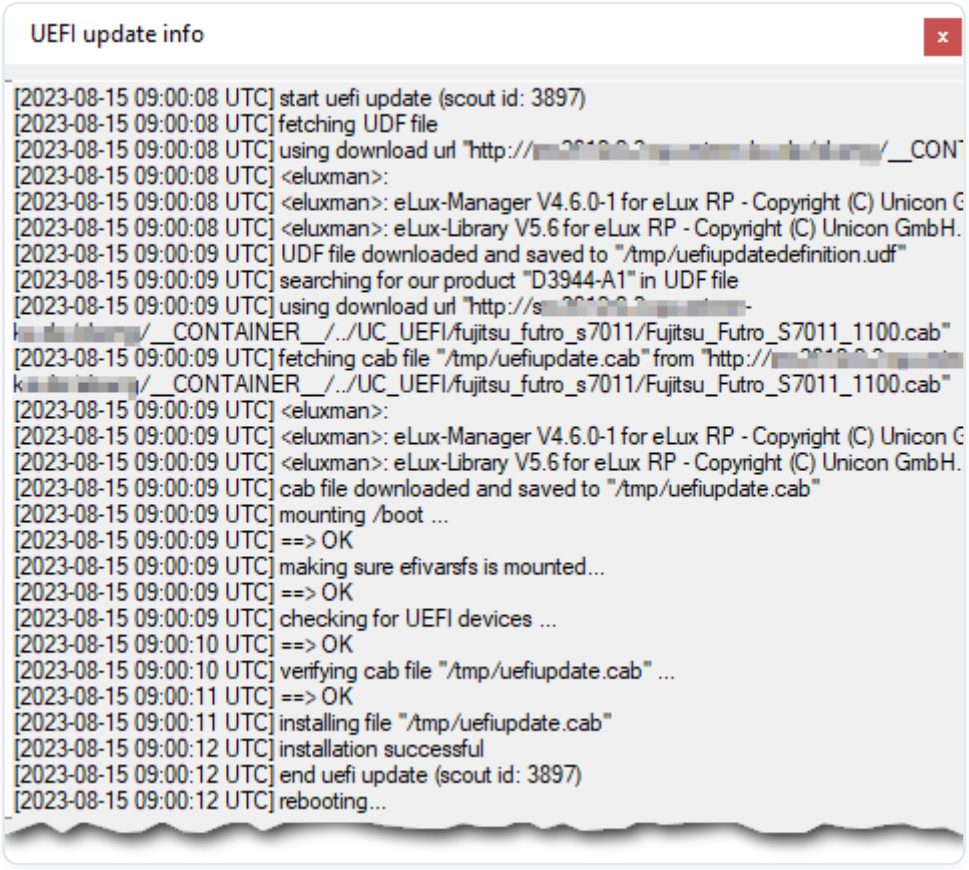
Logging of UEFI updates

UEFI updates can be triggered by a Scout command or by a Scout notification. For each device, in the Scout Console in the **Proerties** window, you may check the command status and other information:

1	Make sure that all relevant UEFI update fields are displayed. To show or hide fields, click the toolbar button.
2	Under **UEFI update**, you get relevant information about the update result and the time, size, provider, and image.
3	To view details, double-click **UEFI update state**.

Via the configured UEFI file (.udf), for each device type , a download-URL with the assigned UEFI firmware is used to transfer the appropriate .cab file. After a device restart, the installation of the UEFI firmware is started.

After the installation, the devices will restart again, this time using the new UEFI.



If it is a Fujitsu FUTRO device type and the system partition is encrypted, it will be found on the blacklist and the command will be canceled.

BIOS update for devices without UEFI

A BIOS update for legacy devices that are not yet equipped with UEFI is still possible. The update to newer BIOS firmware is initiated with the help of a user-defined command from the Scout Console and executed on the relevant devices.

We recommend to avoid downgrading to earlier firmware versions.

> **Note:**
>
> The BIOS of Fujitsu Futro devices with an encrypted system partition cannot be updated. Any attempt will be blocked by eLux® and documented in the log file.

- Preparing BIOS devices
- Providing new BIOS firmware
- Performing a BIOS update
- Different hardware models

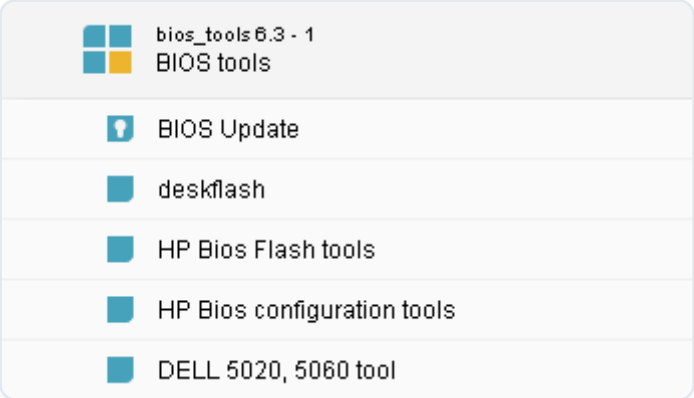
Preparing BIOS devices

The manufacturer tools required for UEFI/BIOS updates are provided by Unicon in a special eLux software package. This package needs to be integrated into the image of your devices.

1. From our myelux.com portal, under **Downloads > eLux Software Packages**, download the following software package:
   - BIOS tools
2. Import the package into ELIAS. For further information, see Importing software packages in the **ELIAS 18** guide or Importing packages into a container in the **ELIAS** guide.
3. In ELIAS, integrate the software package into the relevant image.

   Inside the BIOS tools package, select the subordinate feature according to your hardware.

   deskflash for Fujitsu HP BIOS Flash for HP (BIOS update) HP BIOS configuration tools for HP (BIOS configuration) Dell tools for Dell



For further information, see Defining and pinning packages in the **ELIAS 18** guide or Creating an IDF in the **ELIAS** guide.

4. For the relevant devices, perform an eLux® firmware update to the updated image.

The eLux software package BIOS tools is installed on the devices.

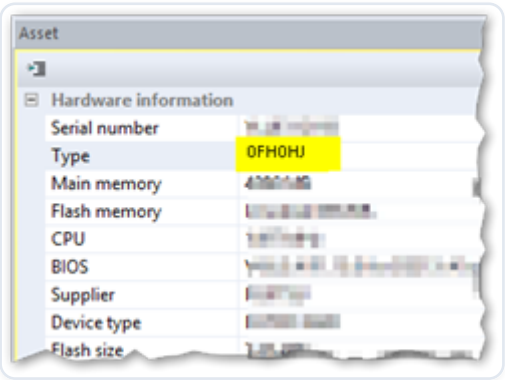
Providing new BIOS firmware

The vendor-specific BIOS firmware data must be made available on your web server.

1. For "legacy" BIOS devices, download up-to-date BIOS firmware from the vendor's homepage.

   The data are normally available as .bin or .bup archives.

2. On the web server, create a container named UC_BIOS
3. For each device type, under UC_BIOS, create a sub-directory. The name of the sub-directory must correspond exactly to the Type name of the devices. For a selected device, look up the **Type** name in the Scout Console, in the **Asset** window under **Type**.



> **Note:**

> It is important that the sub-directory names of the UC_BIOS container exactly match the type names in the Scout Console, in the Asset window (Hardware information). In the example, the name D3313-A1 represents the Fujitsu S920 device.

4. Copy the vendor-specific BIOS firmware archives into the sub-directories of the UC_BIOS container named according to the device types.

   Example: `eliasUC_BIOSD3313-A1`

5. Check the MIME type settings of your web server and add any missing entries:

.bin	application/octet-stream	default setting of IIS
.bup	text/plain	

Note:

> If you want to use individual names for the UEFI/BIOS files, we recommend following the 8.3 convention (especially for HP devices), which means a maximum of 8 characters for the file name and 3 characters for the file extension. For individual file extensions, add an appropriate MIME type.

> The new BIOS firmware is provided on the web server.

Performing a BIOS update

REQUIRES
• On the devices, the eLux® software package **BIOS Tools** is installed
• On your web server, the UC_BIOS container with subdirectories and latest BIOS firmware archives is available Providing new BIOS firmware.

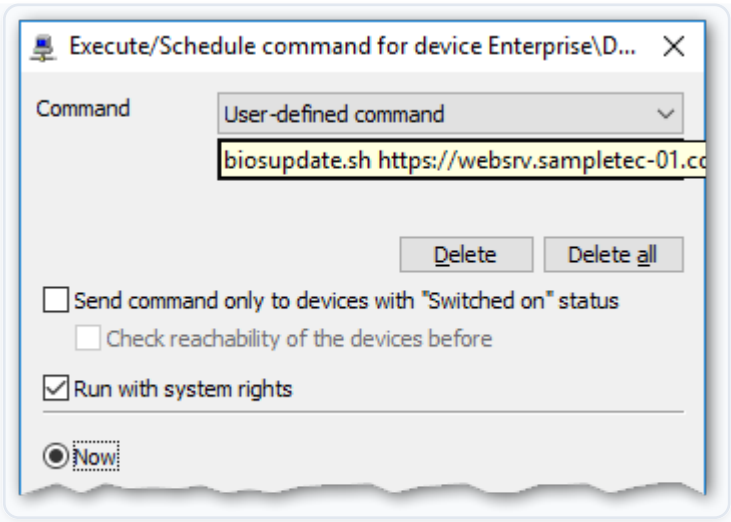
Important:

> Avoid downgrading to earlier BIOS firmware versions.

1. For the relevant device, OU or Dynamic Client Group, open the context menu and click **Commands > User-defined command**.
2. In the command field, enter the script name biosupdate.sh and the URL to the manufacturer-specific container on the web server.

   If the BIOS is protected by a password, add the BIOS password. Use the following syntax: biosupdate.sh `<URL> <BIOS password>` Example: `biosupdate.sh https://websrv.sampletec-01.com/elias/UC_BIOS/__TYPE__/D3313-B1x.R1.15.0.bup <password>.`

   > **Note:**
   >
   > To update the BIOS of different device types via a common command, use the type parameter `__TYPE__`. For further information, see Different hardware models in the `__Scout__` guide.

   The biosupdate.sh script starts the relevant manufacturer BIOS update tool.
3. Select **Run with system rights**.
4. Click `__Execute__`.
5. Enter your device password.

On the target devices, the BIOS update to the firmware data provided on the web server will be initiated.

> **Note:**

> You can predefine a user-defined command as a template: Create a predefined command and specify the password as a variable. The password is requested when the administrator executes the command. For further information, see Creating predefined commands in the **Scout** guide.

Result

The Scout Console triggers the transfer of the provided BIOS firmware data to the devices. The manufacturer tools driven by the script then install the new BIOS firmware on the devices. At this point, the devices already report the status successful: The initialization of the BIOS update was completed successfully. After the installation, the devices will be rebooted and then start with the new BIOS.

In the Scout Console, you may check the command status for each individual device in its **Properties** window

To view details, double-click **Command state**:

- The **Command info** dialog shows the individual steps performed for each command sent.
- The figure shows on the right side under **Command** the relevant user-defined command with exact syntax.
- Under **Result** it can be seen that this Fujitsu device type was found on the blacklist: For these devices the command will be aborted if the system partition is encrypted Otherwise the specified firmware archive is transferred.

Different hardware models

To have the UEFI/BIOS of different device types updated via a common command, use the type parameter `__TYPE__`. This is very helpful when you are dealing with a mixed hardware infrastructure.

The parameter is automatically resolved to the correct folder name of the UEFI/BIOS file.

> **Note:**

> The UC_BIOS container with correctly named subdirectories must be made available on the web server.

- In the user-defined command, following UC_BIOS, replace the directory name by the string `__TYPE__`. Example: `biosupdate.sh http://mywebserver/eluxng/UC_BIOS/__TYPE__/xxx.bin <password>`

The container parameter is resolved by the devices according to their hardware model, for example to D3313-A1.

Spelling of the container parameter

Make sure to use the correct spelling:

Two underscores followed by the string TYPE (all uppercase) followed by two more underscores: `__TYPE__`

Updating UEFI/BIOS configuration

For many devices, BIOS or UEFI configuration changes can also be made remotely from the Scout Console.

- Changing individual UEFI/BIOS settings
- Exporting and changing a UEFI configuration /Fujitsu
- Using a BIOS as a reference /Fujitsu
- Updating a UEFI/BIOS configuration /HP

Changing individual UEFI/BIOS settings

For some devices, UEFI/BIOS configuration changes can also be performed remotely from the Scout Console. This has been successfully tested for the following devices:

Fujitsu Futro S740, S940, S7010, S7011, S9010, S9011

REQUIRES
The eLux® software package BIOS Tools is installed on the devices.

> **Note:**

> To reference boot entries, check the manufacturer documentation for your devices to see which entries are available.

Setting/deleting the UEFI/BIOS password

1. For the relevant device, OU or Dynamic Client Group, open the context menu and click **Commands > User-defined command**.
2. Use the following commands: Replace current password with new password: `/opt/deskview/bin/biosset -PWD=<Old password> -NEWPWD=<New password>`

   Delete password: `/opt/deskview/bin/biosset -PWD=<Old password> -NEWPWD=`

   Set password, if not set: `/opt/deskview/bin/biosset -PWD= -NEWPWD=<New password>`
3. Select Run with system rights.
4. Click **Execute**.

Retrieving current boot order

- Use the following command:

  `/opt/deskview/bin/biosset -BOU –PWD=<password>`

For non-UEFI devices, the result is shown in the command info.

Changing boot order

REQUIRES
• The UEFI/BIOS CPU option VT-d must be disabled.

Use the following commands for **UEFI** devices:

`/opt/deskview/bin/biosset -BOU=<boot order> -PWD=<password>`

Example:

`/opt/deskview/bin/biosset -BOU=0x01-UEFI:FilePath0,0x0a-LAN0 -PWD=<password>` `(first entry for HDD)`

- Use the following commands for **non-UEFI** devices:

  Changing boot order:

`/opt/deskview/bin/biosset -BOOTORDER=1HDD,2LAN -PWD=<password>`

Disabling or enabling boot entry

REQUIRES
The UEFI/BIOS CPU option VT-d must be disabled.

- Use the following command to disable a boot entry:

  `/opt/deskview/bin/biosset -BOUD=<boot option> -PWD=<password>`

  Example for **UEFI** devices:

  `/opt/deskview/bin/biosset -BOUD=UEFI:FilePath0 -PWD=<password>`

  Example for non-UEFI devices:

  `/opt/deskview/bin/biosset -BOUD=LAN -PWD=<password>`

- Use the following command to enable a boot entry:

  `/opt/deskview/bin/biosset -BOUE=<boot option> -PWD=<password>`

  Example for **UEFI** devices: `/opt/deskview/bin/biosset -BOUE=UEFI:FilePath0 -PWD=<password>`

  Example for non-UEFI devices: `/opt/deskview/bin/biosset -BOUE=LAN -PWD=<password>`

Configuring UEFI/BIOS with manufacturer tool

UEFI/BIOS configuration changes can be made for DELL devices, for example, via the relevant manufacturer tool. To do so, include the call of the tool in the user-defined command. Note that the entire parameter for the call must be set in quotation marks.

- Use a command according to the following example:

  `/usr/sbin/biosconfig.sh --tool "/opt/dell/dcc/cctk --ValSetupPwd=<password> -I" <URL>`

For the required parameters, refer to the manufacturer documentation, for example, the documentation for the DELL Client Command Suite. Note that the command is executed on the device without any further checks. The responsibility lies with the executing administrator.

> **Note:**

> You can predefine a user-defined command as a template: Create a predefined command and specify the password as a variable. The password is requested when the administrator executes the command. For further information, see Creating predefined commands in the **Scout** guide.

Exporting and changing a UEFI configuration /Fujitsu

– from eLux® RP 6 2204 –

The UEFI configuration can be exported as an .xml file for some devices. In the .xml file, you may change as many settings as you like before importing the file to the target devices. This has been successfully tested for the following devices:

Fujitsu Futro S540, S740, S940, S7010, S9010

REQUIRES
The eLux software package **BIOS Tools** and included feature packages **BIOS Update** and **Deskflash** must be installed on the devices.

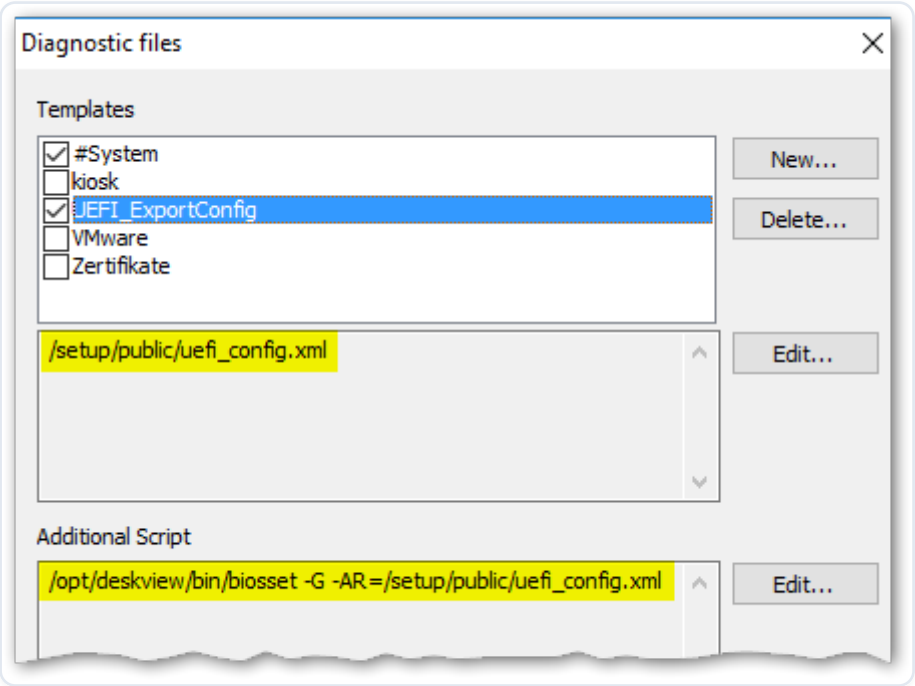
To use the **Diagnostics** feature, the object right **Edit diagnostic templates** is required (disabled by default)

1. To export the UEFI configuration of a device, use a user-defined command. We recommend that you execute the command using the **Diagnostics** feature. With it, you will transfer the .xml file in the same step.
2. Create a diagnostic template for which you specify the target file in the file list section.
3. In the **Additional script** section, create the actual command:

   ```
   /opt/deskview/bin/biosset -G -AR=/<path>/<targetfile.xml>
   ```

   Make sure to specify the .xml file extension for the target file. A password is not required.

   Example:



Note:

As an alternative to the Diagnostics feature, you can execute the user-defined command directly on a device and get the target file by other means.

4. Change the desired settings in the .xml file using a text editor.

   For each entry, both the value **(SettingValue)** and descriptive text **(SettingValueText)** must be changed. For information on the settings and their possible values, see the Fujitsu User Manual of the **DeskView Client**, under **Expert mode**.

   Save the .xml file.

5. Place the .xml file on your web server below UC_BIOS in the relevant device type specific directory. For further information, see Providing new BIOS firmware.

6. To import the updated UEFI configuration, open the context menu of the target devices, for example, of a Dynamic Client Group, and choose **Commands >User-defined command**. Enter the following command:

   ```
   `biosconfig.sh https://<server>/<path>/<sourcefile.xml> [UEFI password]`
   ```

   The password is required if the UEFI is password-protected.

   Example without password: `biosconfig.sh https://https://websrv.sampletec-01.com/eluxng/UC_BIOS/D3544-B1/BIOS_settings/uefi_config.xml`

   > **Important:**
   >
   > The .xml file must have the filename.xml format.

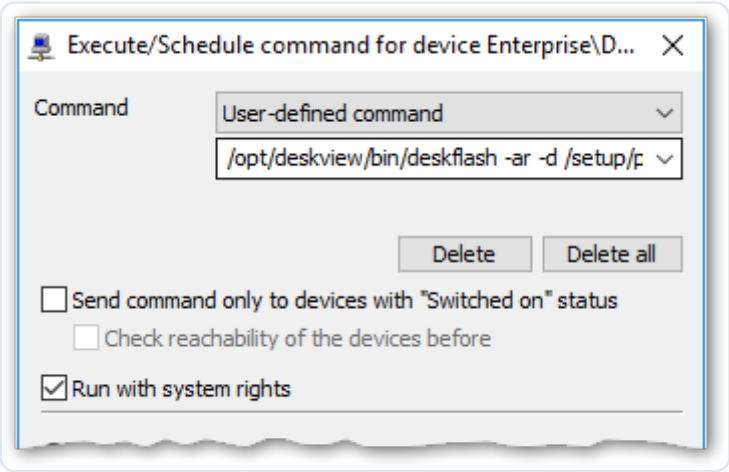
7. Check in the **Command info** or **Command history** whether the command has been executed successfully.

Using a BIOS as a reference /Fujitsu

As an alternative to updating the BIOS configuration via user-defined commands in multiple steps, you can copy the BIOS from a fully configured client and transfer it to other devices within the same model series.

> **Note:**
>
> The following procedure applies to Fujitsu devices within the same Futro model series.

1. For the configured device (reference client), open the context menu and click Commands > User-defined command.
2. Enter the following command: /opt/deskview/bin/deskflash -ar -d /setup/public
3. Select **Run with system rights**.



4. Click **Execute**.

   The BIOS is saved to a .bup file in the local directory /setup/public of your Futro client.

5. Copy the .bup file to a USB stick or use the **Diagnostics** feature to save it on the stick.
6. Make sure that the relevant vendor-specific directory exists within the UC_BIOS container on the web server. For further information, see Providing new BIOS firmware.
7. Copy the .bup file from the USB stick to the relevant subdirectory on the web server. Example: `UC_BIOS/D3313-A1 for device type D3313-A1`.
8. For your target devices, perform a BIOS update.For further information, see BIOS update for devices without UEFI

Updating a UEFI/BIOS configuration /HP

– currently available for the LTSR version only –

To update any UEFI/BIOS settings, configure the UEFI/BIOS of a reference system and then transfer it to the target devices of the same type.To only change the UEFI/BIOS password, you do not need a reference system and can start with step 6.

> **Note:**
>
> The following procedure applies to the HP devices t730 and t630.

1. For the configured device (reference client), open the context menu and click **Commands > User-defined command**.
2. Enter the following command: `/usr/AMI-HP630/LnxBCU.sh /Get:/setup/public/biossettings.txt`

3. Select **Run with system rights**.
4. Click **Execute**.
    - The UEFI/BIOS settings are saved to the text file biossettings.txt in the local directory / setup/public of your HP device.
5. Copy the `biossettings.txt` file to a USB stick or use the **Diagnostics** feature to save it on the stick.
6. Create the file `biossettings.zip.` If the UEFI/BIOS of your devices is secured by password or if you want to change the UEFI/BIOS password, you will need encrypted password binaries, see below.

    The `biossettings.zip` must contain the following files:

Update UEFI/BIOS settings on devices without UEFI/BIOS password	`biossettings.txt`
Update UEFI/BIOS settings on devices with UEFI/BIOS password	`biossettings.txt and curpassword.bin`
Update UEFI/BIOS settings and UEFI/BIOS password	`biossettings.txt, curpassword.bin and newpassword.bin`
Update UEFI/BIOS password only (You can skip the previous steps for this.)	`curpassword.bin and newpassword.bin`

7. Make sure that the relevant manufacturer-specific directory exists within the UC_BIOS container on the web server.For further information, see Providing new BIOS firmware.
8. Copy the biossettings.zip file from the USB stick to the relevant sub-directory on the web server.
9. For your target devices, perform a user-defined command (with system rights) to update the UEFI/BIOS configuration.To do so, use the following command: `biosconfig.sh https://<server>/<path>/biossettings.zip`

For further information, see Providing new BIOS firmware.

The UEFI/BIOS of your devices is updated with the UEFI/BIOS settings of the reference device.To check whether the command has run through without errors on all devices, use the Scout Console option **View > Command history**.

Creating password binaries

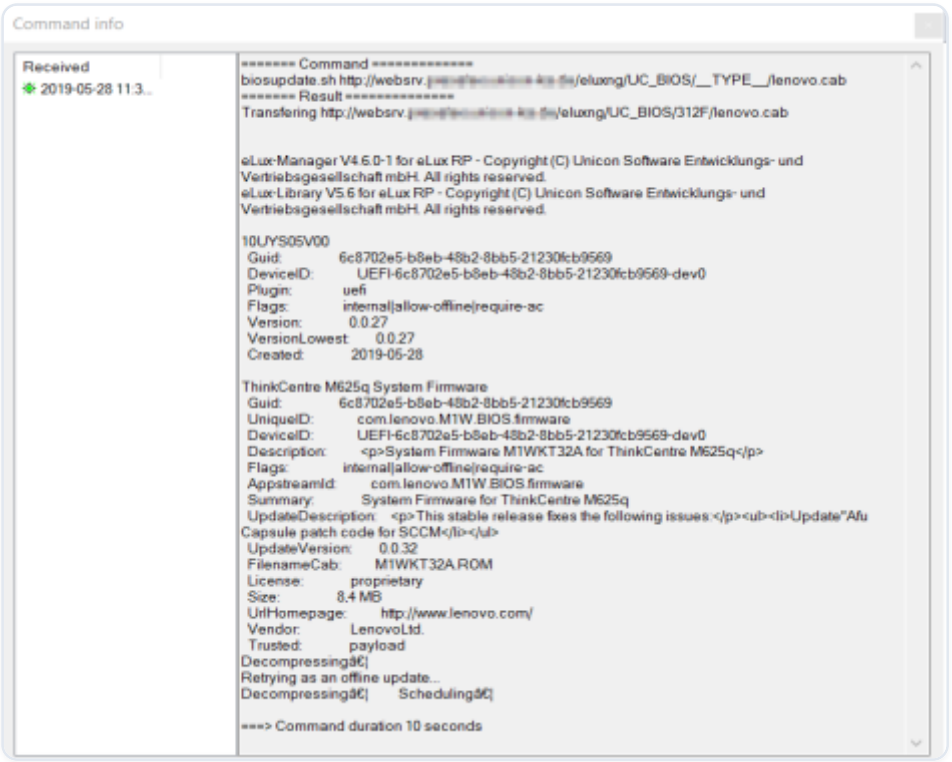
The passwords must be encrypted with the **HPQPswd.exe** tool supplied by HP for their UEFI/BIOS firmware.

1. Run **HPQPswd.exe** and then, in the password fields, enter the current password.
2. Save the encrypted binary file under the name curpassword.bin
3. Run **HPQPswd.exe** a second time and then, in the password fields, enter the new password.
4. Save the encrypted binary file under the name newpassword.bin

Analyzing UEFI/BIOS updates and configuration changes

We recommend that you check the UEFI/BIOS update and UEFI/BIOS configuration history immediately after performing an update or making changes to the UEFI/BIOS.

1. To open the command history, in the Scout Console, right-click the relevant device and click **Commands > Command info**.



   The **Command info** dialog lists all commands executed on an individual device with their single steps.

2. After a UEFI/BIOS update or UEFI/BIOS configuration changes have been performed, restart the relevant devices with the new UEFI/BIOS.

OEM Guides

Guides and integration documents with OEMs

Seamless Printing with ThinPrint

Printing is a common pain point in VDI and DaaS. End users expect it to "just work", while IT teams often face driver, network, and mapping issues. With Citrix Unicon OS (eLux) and ThinPrint Client, local and network printers are seamlessly available in Citrix sessions - without extra print servers.

The installation and configuration guide provides detailed information on ThinPrint.

Whitepapers

A whitepaper is a document that gives an overview on benefits, standards and technical topics, for example our licensing white paper.

Long Term Service Releases and Current Releases

Since January 2017, Unicon has been offering a Long Term Service Release (LTSR) of eLux RP and the management software Scout Enterprise Management Suite for download on our myelux.com portal.

By extending the life and support cycle of our software, Long Term Service Releases enable enterprises to retain a particular release for an extended period of time during which Unicon™ periodically publishes Cumulative Updates (CU). Cumulative Updates only include fixes for the LTSR components and features that help simplify the on-going maintenance. Opting for LTSR, customers obtain high-quality versions and can reduce update management efforts while they are sure to be provided with security fixes if necessary.

In parallel, Unicon carries on providing so called Current Releases (CR) for eLux RP and the Scout Enterprise Management Suite® on a quarterly basis. They are ideal for dynamic environments where the latest features and the most innovative technology are required.

For LTSR and CR product versions, consecutive version numbers are used. This means that a particular product version is either related to LTSR or to CR. Product versions will be labeled clearly to help customers distinguish LTSR from CR.

LTSR is included in the Scout Management license which must only be acquired once. Additional cost or new licensing is not required. Customers can choose between LTSR or Current Releases but they are also free to use both of them in a mixed environment with LTSR and CR versions for Scout Management Suite and/or eLux RP. Changing from CR to LTSR and vice versa is possible at any time. LTSR and CR versions are generally compatible unless explicitly stated otherwise in the release notes of the relevant product version.

Benefits of Long Term Service Releases

- Extended life and support cycle

- Simplified on-going maintenance and greater predictability
- Optimized management efforts through extended periods of time between product upgrades result in lower Total Cost of Ownership
- Additional fixes and security patches in Cumulative Updates

Differences between Current Releases and Long Term Service Releases

- Latest functionality and support of latest hardware platforms in CR
- Release cycle on a quarterly basis for CR
- New technology and fixes are combined in CR – Fixes are available with next CR

LTSR and CR are subject to different cycles. For further information, see Release cycles.

Long Term Service Releases and Current Releases

Since January 2017, Unicon has been offering a Long Term Service Release (LTSR) of eLux RP and the management software Scout Enterprise Management Suite for download on our myelux.com portal.

By extending the life and support cycle of our software, Long Term Service Releases enable enterprises to retain a particular release for an extended period of time during which Unicon™ periodically publishes Cumulative Updates (CU). Cumulative Updates only include fixes for the LTSR components and features that help simplify the on-going maintenance. Opting for LTSR, customers obtain high-quality versions and can reduce update management efforts while they are sure to be provided with security fixes if necessary.

In parallel, Unicon carries on providing so called Current Releases (CR) for eLux RP and the Scout Enterprise Management Suite® on a quarterly basis. They are ideal for dynamic environments where the latest features and the most innovative technology are required.

For LTSR and CR product versions, consecutive version numbers are used. This means that a particular product version is either related to LTSR or to CR. Product versions will be labeled clearly to help customers distinguish LTSR from CR.

LTSR is included in the Scout Management license which must only be acquired once. Additional cost or new licensing is not required. Customers can choose between LTSR or Current Releases but they are also free to use both of them in a mixed environment with LTSR and CR versions for Scout Management Suite and/or eLux RP. Changing from CR to LTSR and vice versa is possible at any time. LTSR and CR versions are generally compatible unless explicitly stated otherwise in the release notes of the relevant product version.

Benefits of Long Term Service Releases

- Extended life and support cycle
- Simplified on-going maintenance and greater predictability
- Optimized management efforts through extended periods of time between product upgrades result in lower Total Cost of Ownership
- Additional fixes and security patches in Cumulative Updates

Differences between Current Releases and Long Term Service Releases

- Latest functionality and support of latest hardware platforms in CR
- Release cycle on a quarterly basis for CR
- New technology and fixes are combined in CR – Fixes are available with next CR

LTSR and CR are subject to different cycles. For further information, see Release cycles.

Release cycles

The table below compares the release schedules and related options. Note that the specified time periods can only provide general guidance. Unicon™ does not make product roadmaps public.

	LONG TERM SERVICE RELEASE (LTSR)	CURRENT RELEASE (CR)
Software Subscription	required (standard terms)	required (standard terms)
Access to Releases	myelux.com portal	myelux.com portal
Cumulative Updates (CU)	yes (CU typically only contain fixes for LTSR).	no (Rapid release cycles allow customers to add new features and all public fixes by upgrading to the next CR.)
Includes Fixes	yes (LTSR customers are required to update to next CU or next LTSR that contains the fix.)	yes (CR customers are required to upgrade to next CR that contains the fix.)
CU Schedule	4-9 months	n/a
Release Schedule	12-18 months	3-6 months

	LONG TERM SERVICE RELEASE (LTSR)	CURRENT RELEASE (CR)
Enhancements	No enhancements until next LTSR	Rapid release cycles, enhancements included

Unicon always supports the most recently published Long Term Service Release with the latest Cumulative Update, and the most recently published Current Release.

For further information, see Product and support lifecycle.

Product and support lifecycle

> **Note:**

> Unicon™ reserves the right to make changes to the time frames based on business needs or technical requirements.

> The Unicon products eLux and Scout Enterprise Management Suite® pass through three life-cycle phases. Each phase is induced by a milestone.

The life-cycle of Unicon products is driven by the customer requirements for new features and constantly evolving technologies including enhancements to existing products.

Milestones in the product lifecycle

General Availability (GA)	Date on which a major product release is made available for download on our technical portal. With GA, the maintenance period is started. Unicon customers receive continued support exclusively for the latest version of the Long Term Service Release (LTSR) provided on our myelux.com portal with the latest Cumulative Update (CU), and for the latest version of the Current Release (CR) provided.
End of Maintenance (EOM)	After EOM, Unicon is no longer obliged to give software support. Unicon offers Extended Support on request (chargeable).

End of Life (EOL)	With EOL, no further security fixes are published. Support and downloads are no longer available for this product. EOL is a minimum of six months from the EOM date. After EOL, the customer may continue to use the product version within the product license terms. As the maintenance period has expired, the product can only be used on an as-is basis. The customer might be asked to update or migrate to the latest version.

Life-cycle milestones for individual products

ELUX® BASE OPERATING SYSTEM (X86)	GA	EOM	EOL
eLux 7	2024-05-29	2029-06-30	2029-12-31
eLux RP 6	2017-07-14	2025-03-31	2025-09-30

SCOUT ENTERPRISE MANAGEMENT SUITE			
Scout 15	2017-07-14	2029-06-30 (or upon GA of next major Scout version)	n/a

SCOUT CLOUD GATEWAY			
Scout Cloud Gateway 1	2019-07-10	2027-06-30	n/a

SCOUT AGENT FOR WINDOWS PLATFORMS			
Scout Agent for Windows 1	2015-07-14	2029-06-30 (EOM ends depending on Scout 15)	n/a

The Unicon product life-cycle and the related services apply exclusively to already purchased products of current Unicon customers. They do not apply to other products or to any other third parties. Unicon reserves the right to make changes to life-cycles and support terms if required.

Support terms

During the product life-cycle GA to EOM, services are provided as listed above and as stated in the EULA terms.

For EOM to EOL, Unicon offers additional support (Extended Support) on request. Extended Support is subject to a charge.

Compatibility client platform and Scout Enterprise Management Suite®

The following table shows the product versions required to manage devices through the Scout Enterprise Management Suite and to connect them via the Scout Cloud Gateway.

CLIENT-PLATFORM	SCOUT ENTERPRISE MANAGEMENT SUITE	SCOUT CLOUD GATEWAY
eLux 7 2404 and later	15 2309 / 15 2302 LTSR CU1 and later	
eLux RP 6.2 and later	15.2 and later	
Scout Agent for Windows platforms	15.2 and later	
eLux® 7 2404 and later		1 2209 and later
eLux RP 6.8 and later		1.1 and later

Legal notes

- Terms and Conditions of Use
- Privacy policy
- Cookie policy

Terms and Conditions of Use

Terms and Conditions of Use for

- `www.unicon.com`
- `uDocs.unicon.com`
- `myelux.unicon.com`

We are delighted that you have found our Unicon™ Internet portal (hereafter referred to as the “Unicon website”). We hope that you will find a product to suit you and take away plenty of ideas to help you design your commercial IT environment. Please take some time to read through these terms and conditions; if required, you can print them out by clicking on the relevant icon or save them to your own computer.

1. General

   Use of the Unicon website is permitted exclusively under the terms and conditions for use set out below. Unicon reserves the right to amend, supplement or replace these terms of use at any time. On logging in or, in the event that a separate login is not required, on start of use of the site, the user accepts the validity of these terms and conditions of use in their current version. Users use the Unicon website at their own risk.

2. Access to the Unicon Web Site

   Due to the nature of the Internet and computer systems, Unicon does not accept any liability for the uninterrupted availability of the Unicon website. Moreover, Unicon reserves the right to cease or limit the operation and/or provision of the website either in whole or in part at any time.

3. Use of the Unicon Website and the Content and Information Provided

All information published by Unicon is made available to users free of charge, unless otherwise specified. Each user may use and store (download) the information provided for his own personal needs. Any other use of the information, especially replication, modification or integration in publications or websites of any kind, is only permitted with prior consent from Unicon or the particular owner of the information. Unicon would like to point out that content and information may be subject to property rights; such property rights (in particular rights to the use of names and trademark rights) remain exclusively with Unicon or the relevant third-party owners.

All information provided by Unicon has been researched and compiled with the greatest care and to the best of its knowledge and belief, and, if originating from a third party, has been adopted without any alteration. However, it is the responsibility of every user to check that the information provided is correct, complete and up-to-date. Any Unicon liability due to incompleteness, incorrectness or outdatedness of the information concerned as well as any liability for damages resulting from any misinterpretation of the information provided by the user shall be limited to gross negligence or intent on the part of Unicon.

When using the Unicon website, the user shall also ensure that such use or any associated activity does not, under any circumstances:

- cause damage to persons, in particular minors, or infringe their personal rights;
- transgress standards of public decency;
- violate industrial property rights and copyright or other property rights;
- convey contents infected by viruses, so-called Trojan horses or other programs that can damage the software;
- enter, save or send hyperlinks or contents, which the user is not entitled to enter, save or send, particularly if these hyperlinks or contents violate obligations to maintain confidentiality, are unlawful or in breach of contract and are simply deliberately incorrect; and/or
- distribute advertising or unsolicited e-mails (so-called "spam") or inaccurate warnings of viruses, malfunctions or the like or incite persons to take part in competitions, snow-ball systems, chain letters, pyramid games and comparable activities.

Unicon shall not be liable for damages resulting from any misuse by the user of the information offered.

4. Links/Content Provided by Third Parties

The Unicon website may contain links and/or other references to websites of third parties. Unicon accepts neither any liability for the content of these third-party websites nor any ownership of these web-sites and/or their content, as Unicon does not have any influence over the linked content and information and does not monitor this content and information on a regular basis. Similarly, Unicon shall also not be liable for the quality, correctness and/or completeness of information from third parties where such information is explicitly indicated as such on the Unicon website. This also applies expressly to all information and content provided by Unicon sales partners on the Unicon website.

5. Software Downloads

   The use of any software offered by Unicon for download is governed by the applicable license conditions of the authorized supplier or the manufacturer concerned (license agreement). These terms and conditions are transferred with the appropriate software as a file or can be obtained from the manufacturer/authorized supplier of the software on request. As a rule, the software cannot be installed without the user's prior acceptance of the terms and conditions of the license agreement.

   The software offered is – unless otherwise specified – intended only for downloading and usage by end users and may only be used by end users. Any other use will be pursued according to criminal and civil law. Mandatory user rights shall remain unaffected within the particular scope of the law as applicable for the respective area of validity.

   Unicon accepts no liability for loss occurring directly or indirectly through the use of downloaded files.

6. Important Information About Computer Viruses

   Although Unicon makes every endeavor to keep the Unicon website virus-free, Unicon cannot provide any warranty or accept any liability for the Unicon website being virus-free. Before downloading information, software and documents, the user shall ensure that appropriate security precautions are taken and appropriate virus scanners used, both for his own protection and to prevent viruses infecting the Unicon website.

7. Rules for Use and Responsibility for Passwords

   The user is solely responsible for safeguarding any password issued for use of certain areas of the Unicon website against misuse. Should a user discover that his password has unintentionally been made accessible to any unauthorized persons or third parties or that there is a risk of misuse, the user is requested to inform Unicon of this immediately. Unicon will then take appropriate action to block the affected password. Notwithstanding this, however, Unicon shall not be liable for any damages resulting from any misuse of passwords, unless such

misuse was caused by gross negligence or willful intent on the part of Unicon or a vicarious agent of Unicon. Unicon shall also be entitled to revoke or restrict access authorization at any time by blocking the access data, without being required to cite reasons.

8. Limitation of Liability

   Unicon shall not be liable for any consequential damages, especially lost profits, production losses, interruption of operations or loss of data unless such liability is mandatory by law. As far as the liability can be excluded or limited according to law, but has not already been excluded or limited elsewhere within these terms and conditions of use, Unicon's liability to the user shall be limited to an aggregate maximum sum of €12,500. Mandatory legal liability exists in particular in the event of a grossly negligent violation of important obligations or willful conduct on the part of Unicon.

9. Court of Jurisdiction/Applicable Law

   Unicon does not accept any responsibility for the fact that information, software and/or documents from the Unicon website may also be accessed or downloaded at locations outside the country concerned, or that the downloading of such information, software and/or documents is permitted in these locations. Users who access the Unicon website from locations outside the country concerned are solely responsible for compliance with regulations that apply under the laws of that particular country. Accessing or downloading information, software and/or documents from the Unicon website from countries where such access or download is illegal is explicitly prohibited by Unicon.

   The laws of the Federal Republic of Germany are applicable. The court of jurisdiction is Munich as long as such place of jurisdiction can rightfully be agreed with the user.

10. Legal information

    © 2024 Unicon GmbH

    This document is copyrighted. All rights reserved.

    eLux® and Scout Enterprise Management Suite® are registered trademarks of Unicon GmbH in the European Union, GB and the United States.

    ScoutaaS® is a registered trademark of Unicon GmbH in the European Union, GB, the USA and Japan.

    All other product names are registered trademarks of their relevant owners.

Written Offer for eLux® and Scout products:

For software in binary form which is integrated in the product and which is licensed under an open source license that gives you the right to receive the source code for that binary, you can obtain a copy of the applicable source code from Unicon Entwicklungs- und Vertriebsgesellschaft mbH on physical media by sending a request by email to the following email address: `support@unicon.com`.

Your request by email must be sent within three (3) years of the date you received the product including the binary that is the subject of your request.

Your request should include:

- The name of the binary for which you are requesting the source code
- The name and version number of the product containing the binary
- The date you received the product
- Your name/company name and
- Your return mailing address and email.

Please note, that

Unicon GmbH may charge you a fee to cover the cost of physical media and processing.

Adobe, Acrobat Reader and PostScript are registered trademarks of Adobe Systems Incorporated in the United States and/or in other countries.

Broadcom is a registered trademark of Broadcom Corporation in the U.S. and/or in other countries.

CardOS is a registered trademark and CONNECT2AIR is a trademark of Siemens AG in Germany and/or in other countries.

CUPS and the Common UNIX Printing System are the trademark property of Easy Software Products.

DivX is a trademark of Project Mayo.

Citrix, Independent Computing Architecture (ICA), Program Neighborhood, XenApp, XenApp are registered trademarks or trademarks of Citrix Systems, Inc. in the U.S.A. and other countries.

Gemplus is a registered trademark and GemSAFE a trademark of Gemplus.

Linux is a registered trademark of Linus Torvalds in the United States and/or other countries.

Macromedia and Shockwave are registered trademarks of Macromedia, Inc. in the United States and/or in other countries.

Microsoft, MS, Windows, Windows NT, Windows CE, Excel, PowerPoint and Active Directory are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.

palmOne and Tungsten are among the trademarks or registered trademarks owned by or exclusively licensed to palmOne, Inc.

PCL is a registered trademark of Hewlett Packard Company in the United States and/or in other countries.

PS/2 and Token Ring are registered trademarks of International Business Machines Corporation.

RealPlayer is a registered trademark of RealNetworks, Inc.

Safesign is a registered trademark of Thales e-Security Ltd.

SAP and SAP product and service names are the trademarks or registered trademarks of SAP AG in Germany and in several other countries.

SICRYPT is a registered trademark of Infineon Technologies Security Solutions GmbH in Germany and/or in other countries.

SSH and SSH product and service names are the trademarks or registered trademarks of SSH Communications Security Ltd.

Sun, Sun Microsystems, Java and Solaris are trademarks or registered trademarks of Sun Microsystems, Inc. in the United States and in other countries.

UNIX is a registered trademark of The Open Group in the U.S.A. and in other countries.

All other trade names we refer to are service marks or trade marks of the respective manufacturers.

This product includes software developed by The X.org Foundation (www.x.org) and its contributors.

This product includes software developed by The XFree86Project, Inc. (www.xfree86.org/) and its contributors.

Privacy policy

Unicon™ and data privacy

Unicon GmbH, Germany ("Unicon") is committed to protecting and respecting your privacy.

This privacy policy sets out the basis on which we will process any personal information that we may collect about you as a visitor to our website or premises or a contact at one of our customers or potential customers, suppliers or potential suppliers or other business partners or in any other cases where we specifically state that this policy will apply.

This privacy policy also sets out how we protect your privacy and your rights in respect of our use of your personal information.

Unicon may have a privacy policy or statement specific to particular local laws, products or services in which case such policy or statement shall supersede or supplement this privacy policy.

What personal information do we collect from you?

We may collect and process the following personal information about you:

- **Personal information that you give us**: This is information about you that you give to us by filling in forms on our website (or other forms that we ask you to complete), giving us a business card (or similar) or corresponding with us by telephone, post, email or otherwise. It may include, for example, your name, address, email address and telephone number; information about your business relationship with Unicon ; and information about your professional role, background and interests.
- **Personal information that our website and other systems collect about you**:
    - If you visit our website it will automatically collect some information about you and your visit, including the Internet protocol (IP) address used to connect your device to the Internet and some other information such as the pages on our site that you visit. This is used to monitor the performance of the website and improve the experience of visitors to the website. Our website may also download "cookies" to your device – this is described in our separate cookie policy. As described further in the cookie policy, you can change the way in which we use cookies by changing your cookie preferences.
    - If you interact with us through social media services, we may be able to access certain information about such interaction, such as whether you have interacted with us through multiple social networks. We make reasonable efforts to ensure that the social media providers have permission from you to allow us to access certain information about you.

Please note that we are not responsible for the manner in which social media service providers handle your personal information that they may collect from you, as this is their responsibility. For instance, our website may use plugins of social media networks such as Twitter, Facebook, Linkedin, and Google+. If you use one of these plugins, these plugins are able to establish a direct connection between your browser and the sites of the respective social media networks. As this transfer takes place directly between your browser and the respective network, Unicon does not have any access, knowledge or control over any data sent or the use of this data within the respective social media networks. If you make use of a social media network and make content available to such networks, this is not covered by this privacy policy. Instead, the terms of use and privacy policies of the respective operators of those social media networks shall apply to any such content. We encourage you to read the privacy policies and terms and conditions of your social media service providers to understand how they handle your personal information.

- Our website may contain links to websites of third parties who are not affiliated to Unicon . If you access such third party websites by clicking on such links, we are not responsible for the way in which such third parties process your personal information that they collect.
- We may make web-based platforms available to our partners and suppliers and such platforms may collect your personal information if you use such platforms. This privacy policy applies to such personal information collected during the registration process and/or the use of such platforms.
- If you exchange emails, telephone conversations or other electronic communications with our employees and other staff members, our information technology systems may record details of those conversations, sometimes including their content.
- Some of our premises have closed circuit TV systems and other security and access management systems which may record you and certain information about your visit if you visit our premises, for security and safety purposes.

- **Other information**: We may also collect some information from other sources. For example:
  - If we have a business relationship with the organisation that you represent, your colleagues or other business contacts may give us information about you such as your contact details or details of your role in the relationship.
  - We sometimes collect information from third party data providers or publicly available sources for anti-money-laundering, export control, credit rating background checking and similar purposes, and to protect our business and comply with our legal and regulatory obligations.

How will we use your personal information?

We may collect, store and use your personal information for the following purposes:

- to operate, manage, develop and promote our business (including our products and services) and, in particular, our relationship with the organisation you represent (if any) and related transactions including, for example:
    - marketing purposes (when we have either gathered prior opt-in consent and/or have a legitimate interest to send you communications which we believe to be relevant and of use to you);
    - warranty processes; product compliance processes;
    - accounting and billing / payment purposes (including to offer financing solutions to customers, together with our finance partners);
    - to operate, administer and improve our website and premises and other aspects of the way in which we conduct our operations;
    - to offer you the services of our online shops based on the terms and conditions of the dedicated web-shop;
    - to provide you with services or information that you may have requested;
    - keep you informed and updated on relevant products or services you may be interested in;
    - to enable you to take part in our online assessments and surveys;
    - to manage and maintain the relationship with investors; and
    - to operate recruiting activities;
- to protect our business from fraud, money-laundering, breach of confidence, theft of proprietary materials and other financial or business crimes;
- to comply with our legal and regulatory obligations and bring and defend legal claims and assert legal rights; and
- if the purpose is directly connected with an assigned purpose previously made known to you.

We may from time to time review your information held in our systems – including the contents of and other information related to your email and other communications with us – for compliance and business-protection purposes as described above. This may include reviews for the purposes of disclosure of information relevant to litigation and/or reviews of records relevant to internal or external regulatory or criminal investigations. To the extent permitted by applicable law these reviews will be conducted in a reasonable and proportionate way and approved at an appropriate level of management. They may ultimately involve disclosure of your information to governmental

agencies and litigation counterparties as described below. Your emails and other communications may also occasionally be accessed by persons other than the member of staff with whom they are exchanged for ordinary business management purposes (for example, where necessary when a staff member is out of the office or has left Unicon).

We will only process your personal information as necessary so that we can pursue the purposes described above and where we have a legal basis for such processing. Where our lawful basis for processing is that such processing is necessary to pursue our legitimate interests, we will only process your personal information where we have concluded that our processing does not prejudice you or your privacy in a way that would override our legitimate interest. In exceptional circumstances we may also be required by law to disclose or otherwise process your personal information.

Disclosure and international transfer of your personal information

We may disclose your personal information, where reasonably necessary for the various purposes set out above:

- to the other members of the Fujitsu group of companies;
- to your colleagues within the organisation that you represent;
- to service providers who host our web services or other information technology systems or otherwise hold or process your information on our behalf, under conditions of confidentiality and security required by law;
- business partners, channel partners, service partner, agents, suppliers and sub-contractors for the performance of any contract we enter into with them or you;
- to a person who takes over our business and assets, or relevant parts of them; or
- in exceptional circumstances:
    - to competent regulatory, prosecuting and other governmental agencies, or litigation counterparties, in any country or territory; or
    - where we are required by law to disclose.

These disclosures may involve transferring your personal information overseas. If you are dealing with us within the European Economic Area, you should be aware that this may include transfers to countries outside the European Economic Area, which have not been determined by the European Commission to have an adequate level of data protection. In those cases, where we transfer your personal information to other members of the Fujitsu group or our service providers, we will ensure that our arrangements with them are governed by relevant legal mechanisms and safeguards

including data transfer agreements, designed to ensure that your personal information is protected, on terms approved for this purpose by the European Commission.

How long do we keep your personal information?

We will delete your personal information when we no longer need such personal information, for instance where:

1. it is no longer necessary for us to retain your personal information to fulfil the purposes for which we had collected it;
2. we believe that your personal information that we hold is inaccurate; or
3. in certain cases where you have informed us that you no longer consent to our processing of your personal information.

Sometimes, however:

1. there are legal or regulatory requirements which may require us to retain your personal information for a specified period, and in such cases we will retain your personal information for such specified period; and;
2. we may need to retain your personal information for certain longer periods for product liability purposes or in relation to legal disputes, and in such cases we will retain it for such longer periods to the extent required.

Note that we may retain some limited information about you even when we know that you have left the organisation that you represent, so that we can maintain a continuous relationship with you if and when we are in contact with you again, representing a different organisation.

What are your rights?

You have the following rights (subject to applicable local laws) in relation to the personal information that we hold about you:

- to access your personal information, and some related information, under relevant data protection law;
- to require any inaccurate personal information to be corrected or deleted;
- to object to our use of your personal information for direct marketing purposes at any time and you may have the right to object to our processing of some or all of your personal information (and require them to be deleted) in some other circumstances;
- to require us to delete your personal information in certain circumstances;

- to require us to restrict or block the processing of your personal information in certain circumstances (when processing is restricted, we can still store your personal information, but may not use it further);
- to obtain from us your personal information, in a structured, commonly used and machine-readable format in certain circumstances. Further, you may have the right to require us to transmit your personal information directly to another person (for instance a new provider) where it is technically feasible to do so; and
- where we are processing your personal information based on your consent to such processing, to withdraw your consent at any time.

Further information and advice about your rights can be obtained from the data protection regulator in your country.

If you wish to exercise any of these rights (subject to applicable local laws), or have complaints about our processing of your personal information, please contact us as set out below. You can also lodge a complaint about our processing of your personal information with the body regulating data protection in your country

Sensitive information

We request you not send to or share with us any sensitive personal information (e.g. information related to racial or ethnic origin, political opinions, religion or other beliefs, health, genetic, or biometric data, criminal background or trade union membership).

Personal information of children

Unicon respects the privacy of children and does not knowingly collect, use or disclose personal information of children under 16 years of age without verifiable consent from a parent or guardian.

If Unicon becomes aware that the information submitted to or collected by Unicon is personal information of children without first receiving verifiable consent from a parent or guardian, Unicon will take prompt steps to delete such personal information.

If a parent or guardian wishes to raise a concern regarding personal information pertaining to a child, please contact us using the details set out in the section titled “Contact Us” below.

Automated decision making

We may use technology which tracks your use of our website and your interactions with us and which helps us to build a profile of your preferred products and information requests. If we do this,

what this means for you is that you are more likely to receive offers and information that are tailored to your specific preferences, based on your previous profile activity.

If we use automated decision making in respect of any specific service or product and such automated decision making produces legal effects concerning data subjects or similarly significantly affects them, we will provide information required by applicable laws in respect of such automated decision making.

We may use automated decision making in the context of comparing delivery and personal data with export control sanction lists of the European Union, the United States of America, Japan and any other country from which an export takes place. Depending on the result of these controls Unicon is obliged to take appropriate measures within the context of export control regulations, which are stipulated by the European Union, the United States of America, Japan and any other country from which the export takes place.

Contact us

If you have any questions, comments and requests regarding this privacy statement or our processing of personal information, please contact us under `<http://www.unicon.com/en/contact/>`.

If you would like to contact our Data Protection Office, please contact us.

Changes to this policy

This privacy policy is valid from the time of its publication on the Unicon websites, and all previous versions of this privacy policy become invalid as of such time. Any changes we make to this privacy statement in the future will be posted to our website and also available if you contact us. Please check back frequently to see any changes.

We will also keep available previous versions of this privacy policy on our website indicating their date and version number.

Last updated: 28 May 2018

Cookie policy

Our website uses “cookies” to distinguish you from other users of our website and to understand how you use our website. This helps us to improve your browsing experience and also to improve our website.

A cookie is a small file of letters and numbers that we store on your browser or the hard drive of your computer if you agree.

You can learn more about our use of cookies, including the types and purpose of the cookies used by our website, by clicking the link for further information in the “cookie notice” which appears as a banner on our website when you first visit our website. Through the cookie notice, you may accept (for the purposes listed), or decline, the use of some or all cookies used by our website. The cookie notice will set out a list of any “essential” cookies which you may not be able to turn off without losing access to some or all parts of our website.

After you have selected your cookie settings through the cookie notice, you will be able to access a “cookie consent” tool on each web page of our website, on each subsequent visit to our website. Clicking on the cookie consent tool will allow you to gain up-to-date information about the cookies being used by our website and manage your consent to some or all of the cookies at any time.

Please consult our Privacy policy to know more about the basis on which we will process any personal information that we may collect about you as a visitor to our website and how we protect your privacy and your rights in respect of our use of your personal information.

If you have any questions, comments and requests regarding this cookie policy, please contact us.

This cookie policy is valid from the time of its publication on the Unicon™ websites, and all previous versions of this privacy policy become invalid as of such time.

Last updated: 28 May 2018

Third party software

Below you will find a collection of links to third-party software and open source projects that are integrated into our software solutions or can be optionally integrated.

For our technology partners, see our home page under unicon.com/partners.

COMPONENT	DESCRIPTION
Avahi	Networking of devices in local networks
CUPS	Print interface with central configuration
FreeRDP	eLuxRDP is based on FreeRDP
Fluendo	Audio/Video codec packages (subject to license)
Gnome	Free Unix/Linux desktop environment
Linux Kernel	Linux kernel
Maliit	Virtual keyboard
MATE-Desktop	Free Unix/Linux desktop environment
Mozilla	Firefox web browser
NoMachine	Remote desktop software
OpenVPN	VPN client
PCSC-Lite	Smart card reader drivers
PulseAudio	Free sound middle-ware
Qt Project	Cross-platform C++ framework
Squid	Free proxy server

COMPONENT	DESCRIPTION
SSH	Secure remote access
VNC	Mirroring/Remote maintenance
WPA Supplicant	Encryption for wireless networks, 802.1X network security
xfree86	X-Windows graphic interface (X11)