HD VISION
high dimensional vision systems

LumiScan Cam Driver

User Guide

HD VISION
high dimensional vision systems

Imprint

HD Vision Systems GmbH
Carl-Friedrich-Gauß-Ring 5
69124 Heidelberg

Tel.: +49 (6221) 67219-00
Fax: +49 (6221) 67219-01
Email: info@hdvisionsystems.com
WWW: https://hdvisionsystems.com

CEO: PD Dr. Christoph Garbe
Headquarters: Heidelberg
Register court: Amtsgericht Mannheim
Register number: HRB 726917

Version_2024_2_19

Table of Contents

1 Introduction 1

1.1 About This Manual 1

1.2 Formatting Convention 1

1.3 Disclaimer 2

2 LumiScan Cam Driver 3

2.1 Description 3

2.2 Requirements 3

2.3 General Concept of Handshake 3

2.3.1 Functionality Execution 4

2.4 Initializing the Camera 5

2.4.1 Addresses 5

2.4.2 Connection Steps 6

2.4.3 Status Code Values 8

2.4.4 MTU Size Change 9

2.5 Setup Camera Configuration 12

2.5.1 Edit with Built-In Website 12

2.6 Application Settings Content Example 14

2.7 Camera Example File Content Example 16

2.8 Creating New Camera Settings Files 18

2.9 Uploading Data with WebDAV 20

3 General Disclaimer 22

1 Introduction

1.1 About This Manual

This document contains important information about the installation and operation of the LumiScan Cam Driver application. Please read the manual carefully before using this software.

1.2 Formatting Convention

This guide uses special formatting to highlight certain words and phrases:

- Keywords/important information and buttons are highlighted in bold (e.g. XXX/**capture/ready**).
- Links and references are highlighted in green (e.g. info@hdvisionsystems.com).
- File and path names are highlighted in a special font (e.g. `hdvisionsystems/lumiscan-cam-driver/<camera name>/control/capture/ready`).

1.3 Disclaimer

NOTE: By accessing or using these commercial software products, you expressly agree to the following terms and conditions.

Any attempt to use a debugger to examine, analyze, or tamper with the software provided by HD Vision Systems is strictly prohibited and may have immediate and irreversible consequences.

If the software detects the presence of a debugger, security protocols will be activated to protect the intellectual property, functionality and stability of the software. This may result in, among other things, immediate suspension of the associated Software license, loss of data, and, in extreme cases, forced termination of all instances of the Software operating under the same license, as well as legal consequences.

By using the Software, you acknowledge that you have read, understood and accepted the terms of this Disclaimer (see also General Disclaimer).

2 LumiScan Cam Driver

2.1 Description

The LumiScan Camera Driver is a comprehensive image capture application. It supports all Ethernet-enabled, GenICam standard compliant cameras, runs on the ctrlX CORE and is configured and controlled by the ctrlX Data Layer. Accessible through any webbrowser, LumiScan Cam Driver provides easy access to raw data and allows you to quickly adjust basic settings such as exposure time or pixel format.

2.2 Requirements

- ctrlX CORE
- Ethernet-enabled, GenICam standard compliant camera

2.3 General Concept of Handshake

The communication process in the LumiScan Cam Driver application is based on the general concept of handshake. The naming convention for handshake parameters usually consists of the **functionality** and the names **-ready** or **-request**.

Table 1: General Concept of Handshake: Address

Address	Description
<parent address>/<functionality name>/ready	Read-Only, controlled by our app to indicate whether the relevant functionality is allowed to be executed.
<parent address>/<functionality name>/request	Writable. It is used by user to trigger our application to execute relevant functionality.

2.3.1 Functionality Execution

NOTE: To avoid repeating the entire address, in this section the **invariant part** containing the parent address and the **variable functionality name** (capture, connect-camera, disconnect-camera) have been replaced by **XXX**.

Thus, the address: **hdvisionsystems/lumiscan-cam-driver/<camera name>/control/connect-camera/ready** will be written as **XXX/ready**.

Accordingly, the address: **hdvisionsystems/lumiscan-cam-driver/<camera name>/control/connect-camera/request** will be written as **XXX/request**.

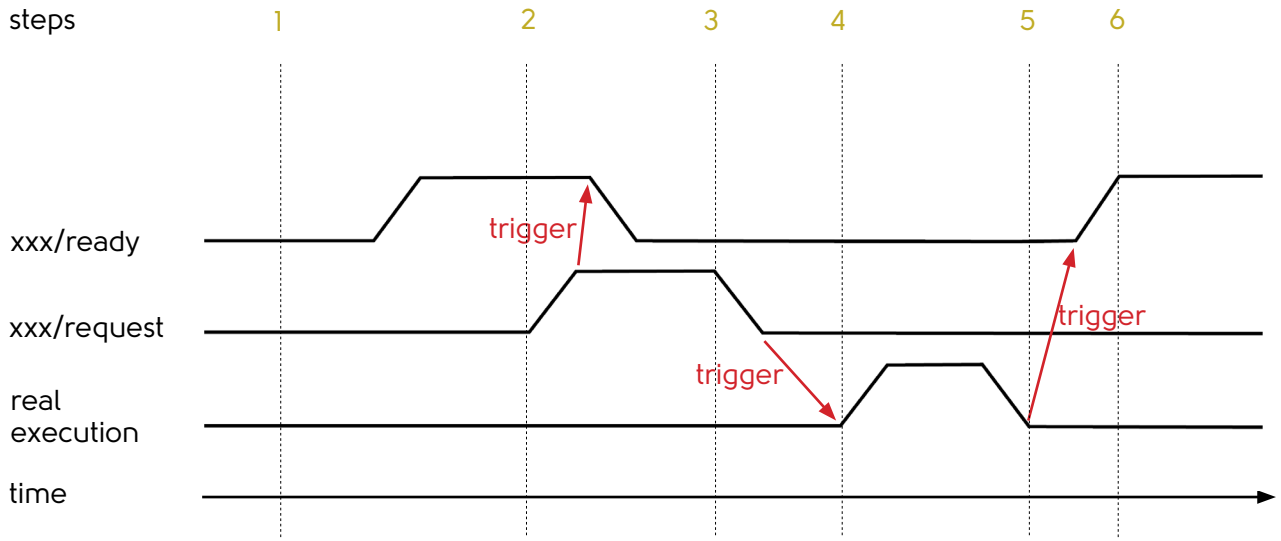
In order to execute a functionality, you need to run the following steps (See also image *Functionality Execution Steps*):

1. Wait until XXX/ready is set to **True**.
2. Set XXX/request to **True**.

NOTE: Make sure there is only one process is accessing XXX/request at a time. The result of execution is **undefined** when:

a. ProcessA set XXX/request to True.
b. ProcessB set XXX/request to False without considering ProcessA.
c. Our application might therefore miss XXX/request from ProcessA.

3. XXX/ready changes to **False**.
4. Set XXX/request to **False**.
5. Start executing relevant functionality.
6. Execution finished.
7. XXX/ready is automatically set to **True** again.



Functionality Execution Steps

2.4 Initializing the Camera

2.4.1 Addresses

Table 2: Functionality Addresses

Functionality	Permission	Address
Capture	Read-Only	hdvisionsystems/lumiscan-cam-driver/<camera name>/control/capture/ready
	Writable	hdvisionsystems/lumiscan-cam-driver/<camera name>/control/capture/request
Connect Camera	Read-Only	hdvisionsystems/lumiscan-cam-driver/<camera name>/control/connect-camera/ready
	Writable	hdvisionsystems/lumiscan-cam-driver/<camera name>/control/connect-camera/request
Disconnect Camera	Read-Only	hdvisionsystems/lumiscan-cam-driver/<camera name>/control/disconnect-camera/ready
	Writable	hdvisionsystems/lumiscan-cam-driver/<camera name>/control/disconnect-camera/request
Status	Read-Only	hdvisionsystems/lumiscan-cam-driver/<camera name>/control/status/code

2.4.2 Connection Steps

NOTE: To avoid repeating the entire address, in this section the **invariant part** containing the parent address has been replaced by **XXX**.

Thus, the address: **hdvisionsystems/lumiscan-cam-driver/<camera name>/control/connect-camera/ready** will be written as **XXX/connect-camera/ready**.

Accordingly, the address: **hdvisionsystems/lumiscan-cam-driver/<camera name>/control/connect-camera/request** will be written as **XXX/connect-camera/request**.

1. Start, connected to camera:
   a. Prerequest: camera is physically connected but is not yet connected in software (XXX/connect-camera/ready is False)
   b. Check XXX/connect-camera/ready = True

NOTE: If is false, prerequest is not met or no camera is found.
Check whether camera ip is in the same local network as ctrlX.

   c. Set XXX/connect-camera/request := True
   d. Check XXX/connect-camera/ready = False

NOTE: If the Cam Driver application hasn't seen that the variable has changed:

i. Make sure that only one process is accessing XXX/connect-camera/request at a time.
ii. Check that you are using the same data layer version as the camera application.
iii. Contact hdvisionsystems.

   e. Set XXX/connect-camera/request := False
   f. Check XXX/status/code (Camera) = 0 (ready)* -----> Camera is initialized, and ready to capture image

* The meaning of all status code values is explained in the Status Code Values table at the end of this chapter.

2. Capture image
    a. Prerequest: Camera is started (XXX/connect-camera/ready is False)
    b. Check XXX/capture/ready = True

NOTE: If XXX/capture/ready is False:

- Camera could be in error state, not started or still capturing image.
- Check status code for more information.

    c. Set XXX/capture/request := True
    d. Check XXX/capture/ready = False

NOTE: If XXX/capture/ready is True:

i. Make sure that only one process is accessing XXX/connect-camera/request at a time.
ii. Check that you are using the same datalayer version as the camera application.
iii. Contact HD Vision Systems.

    e. Set XXX/capture/request :=False
    f. Check XXX/status/code (Camera) = 0 ----> Capture image is successful
    g. Image captured, you can access it at address `hdvisionsystems/lumiscan-cam-driver/<camera name>/output/image`

3. Disconnect Camera
    a. Prerequest: Camera is started and is not capturing (XXX/disconnect-camera/ready is True (internally it checks XXX/connect-camera/ready is False, XXX/capture/ready is True))
    b. Check XXX/disconnect-camera/ready = True
    c. Set XXX/disconnect-camera/request := True
    d. Check XXX/disconnect-camera/ready = False
    e. Set XXX/disconnect-camera/request := False
    f. Check XXX/status/code (Camera) = 0 ----> Capture image is successful
    g. Image captured, you can access it at address `hdvisionsystems/lumiscan-cam-driver/<camera name>/output/image`

2.4.3 Status Code Values

Table 3: Status Code Values Description

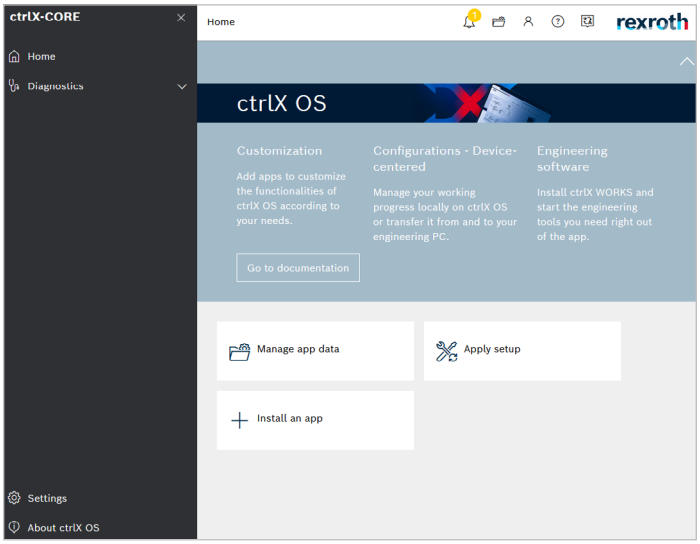
Status Code Number	String	Meaning
0	"ready"	The last functionality is successfully executed, the application is ready to execute the next functionality.
1	"settingUp"	The module or function is still setting up.
2	"running"	The function is currently running.
3	"waitingForClient"	Waiting for the client (usually to acknowledge the start of the function).
-1	"undefinedError"	Something went wrong and we did not create a special error code for it.
-2	"invalidSetting"	The supplied settings file could not be loaded.
-3	"invalidInput"	One of the input fields was not valid.
-4	"missinglicense"	Make sure you have the correct license installed.
-10000	"failedToConnectToCamera"	There was a problem connecting to the camera, please make sure the camera is connected.
-10001	"pixelFormatNotSupported"	Invalid pixel format: Pixel type typo, or camera doens't support this type of pixel format.
-10002	"couldNotSetPixelType"	The pixel format couldn't be set. Camera is still running, please stop and set again.
-10003	"couldNotSetExposureTime"	Exposure time is out of range. Please check with the camera supplier for the permissible range of values.
-10004	"couldNotCaptureImage"	There may be several causes of this problem: • The camera may be disconnected, please reconnect. If it happens often, it may be a communication failure due to poor connection quality. • Please make sure to use „Foil Shielded Twisted Pair" cable. • Check the MTU size setup in the ctrlX web page settings. It should be set to 8000 or 9000.* If you cannot change MTU size, switch to Ethernet port that supports large MTU size. (p.s. since 1.16 or 1.18 ctrlX supports changing MTU).

* For information on how to change the MTU size, see next section MTU Size Change.

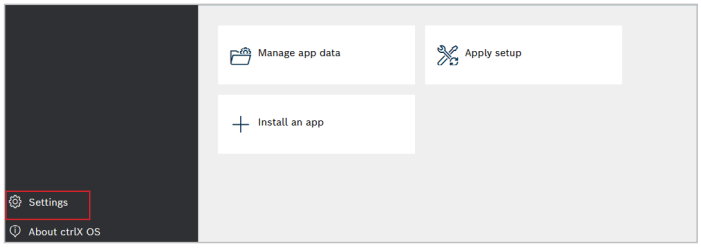
2.4.4 MTU Size Change

To change the camera's MTU size:

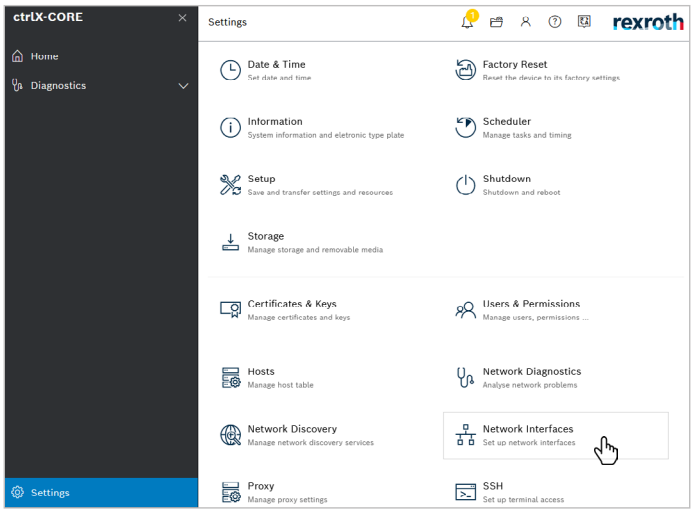
1. Log in to the ctrlX OS website.
2. Go to **Home**.



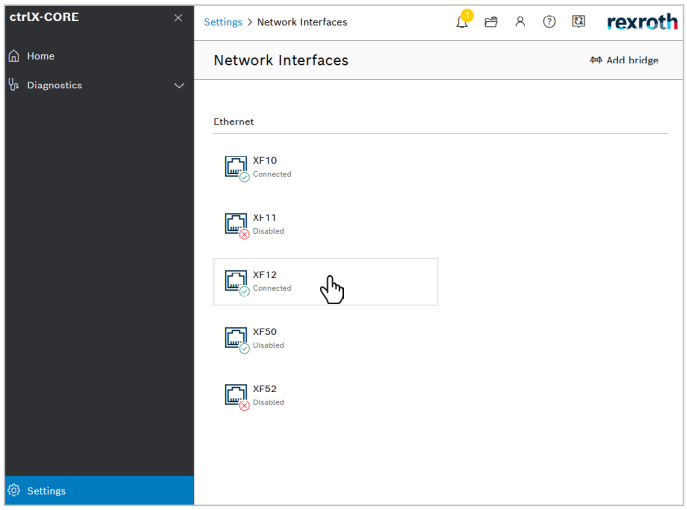
3. Select **Settings**.



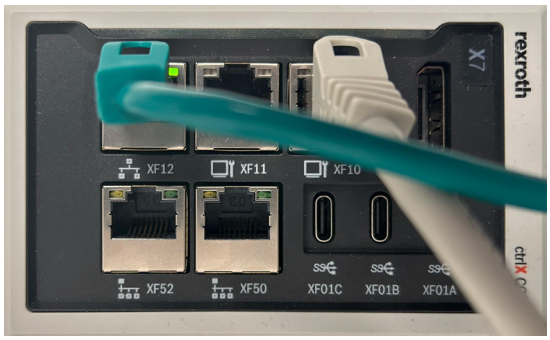
4. Select **Network Interfaces**.



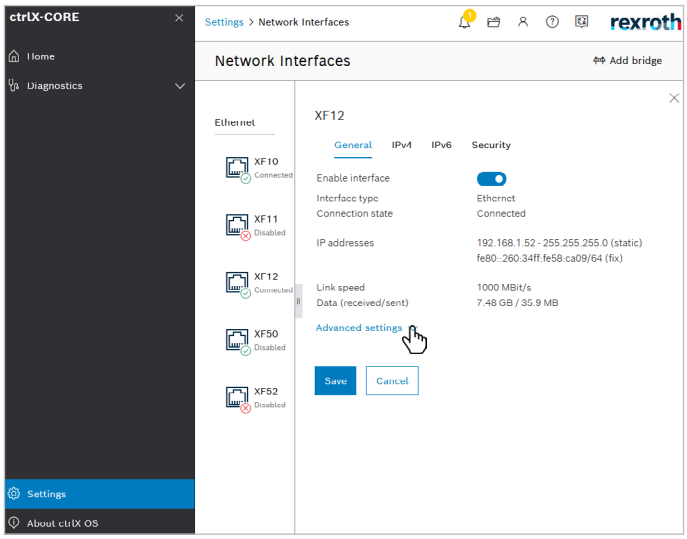
5. Select the currently used Ethernet port of your ctrlX device.



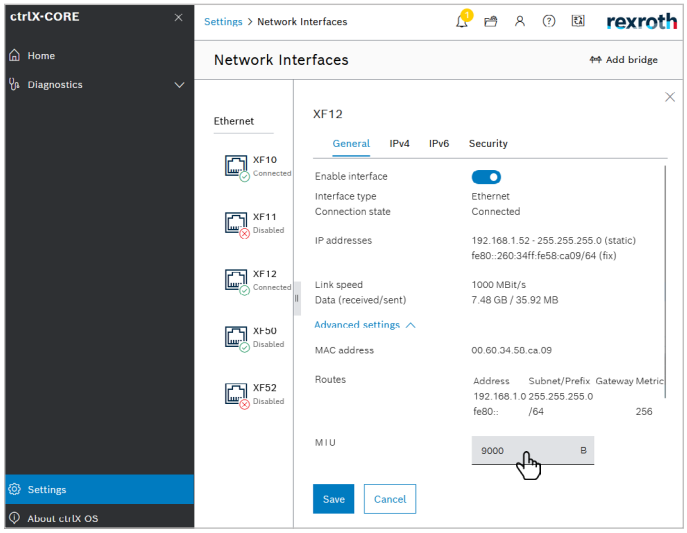
NOTE: The number of the currently used Ethernet port can be found next to the port entry on the ctrlX device. In this example, it is going to be XF12.



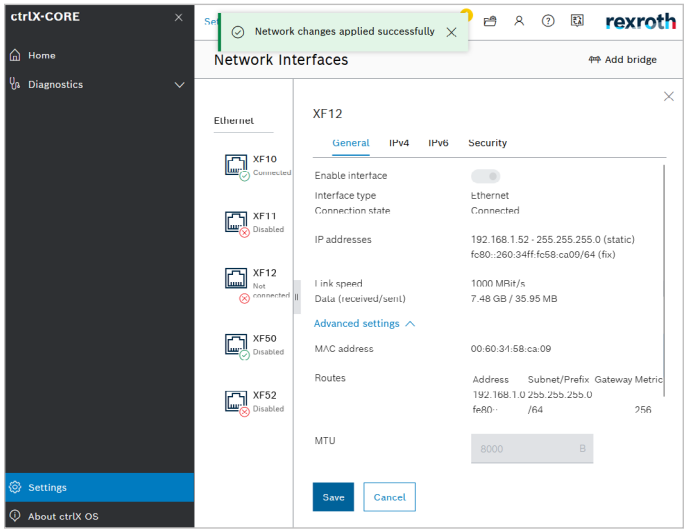
7. Expand **Advanced Settings.**



8. Enter the value you want to use in the MTU window (check status code **-10004** in Status Code Values).



9. Click **Save** to confirm.
10. A report confirming the change appears on the screen.

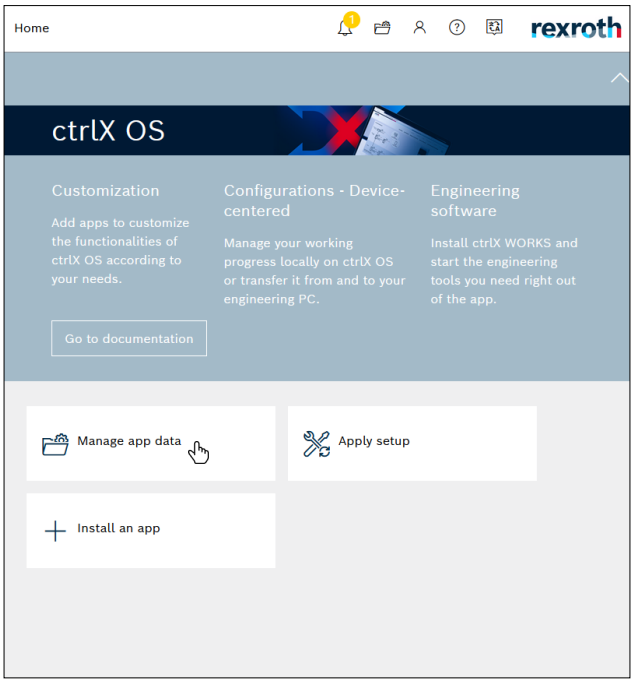


11. Once the changes have been saved, you can close the program.

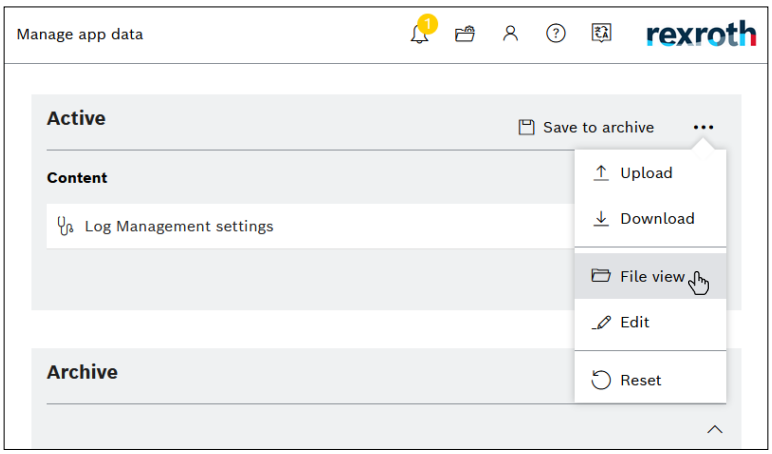
2.5 Setup Camera Configuration

2.5.1 Edit with Built-In Website

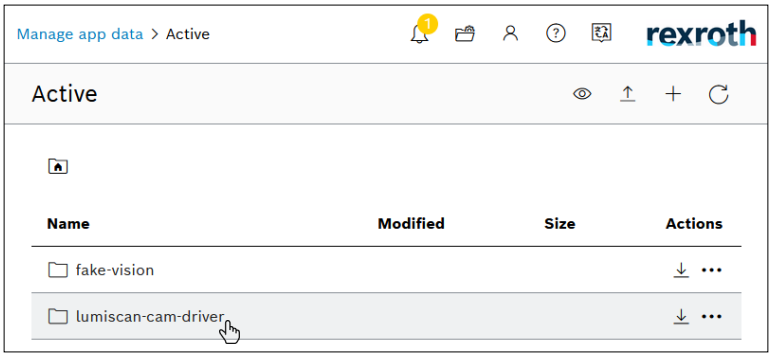
1. Log in to the ctrlX OS website.
2. Go to **Home.**
3. Select **Manage app data.**



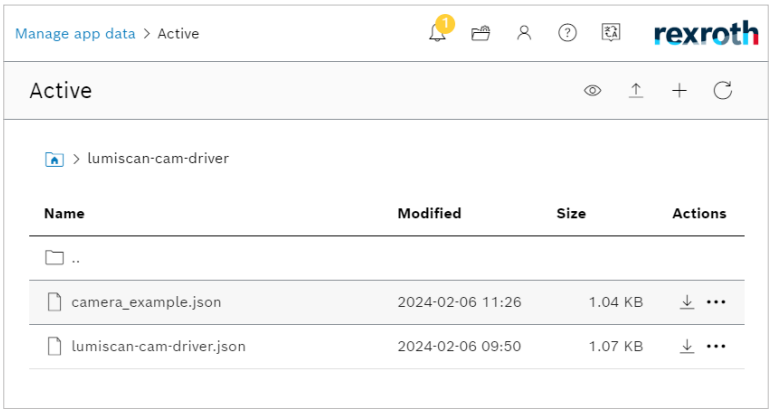
3. Click the three dots and select **File View.**



4. Go to the relevant application folder.

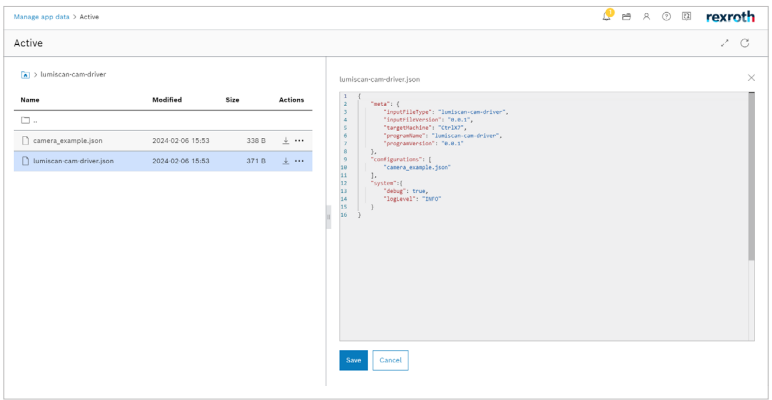


5. There are two files by default (You can add new files later).



- **lumiscan-cam-driver.json** contains application settings. Here you can specify configuration files for the camera you want to use.
- **<extra file name>** (here as example: camera_example.json) contains configuration for one camera. You can configure camera relevant parameter here.

6. Left-clicking on the desired filename will automatically open the JSON Settings window.



NOTE: See also Application Settings Content Example

2.6 Application Settings Content Example

NOTE: If you don't want to set the parameters each time you connect to the camera, you can set the camera to a specific configuration by specifying its name.

The settings of the Lumiscan Cam Driver application are located in the **lumiscan-cam-driver.json** file, and it is here that you should configure the general settings of the camera.
An example of such settings is shown below.

```
{
    "meta": {
        "inputFileType": "LumiScan-CtrlX-Cam-Driver",
        "inputFileVersion": "0.0.1",
        "targetMachine": "CtrlX7",
        "programName": "lumiscan-cam-driver",
        "programVersion": "0.0.1"
    },
    "configurations": [
        "camera_example.json"
    ],
    "system":{
        "debug": true,
        "logLevel": "INFO"
    }
}
```

The description of the parameters mentioned above can be found in the following Table 4.

Table 4: Application Settings File: JSON Settings Description

Parameter	Description
meta	Place to set meta information: inputFileType and inputFileVersion are required.
inputFileType	IIn this case it must be: `LumiScan-CtrlX-Cam-Driver`
inputFileVersion	Currently, only version 0.0.1 is supported.
targetMachine	Optional. Enter the name of the machine in use here to indicate that the current application configuration is being used for this particular machine.
programName	Optional. This parameter indicates by which application the current file is used. In this case: `lumiscan-cam-driver`
programVersion	Optional. Currently used program version.
configurations	This is a list of camera configuration files. Camera serials should be duplicated.
system	Setup ctrlX releated configuration.
debug	Shows detailed message of i/o of datalayer. Possible values: `true` or `false`
logLevel	Change log level, possible values are: INFO, DEBUG, WARN, FATAL, OFF.

NOTE: After changing the parameters, the LumiScan Cam Driver application should be restarted.

2.7 Camera Example File Content Example

NOTE:

- The camera parameters can be overridden by setting the appropriate parameters in the path `.../input/` in the ctrlX Data Layer.

The settings for a specific camera are located in the **<extra file name>.json** file, here as an example named: **camera_example.json**.
An example of such settings is shown below.

```
{
    "meta": {
        "inputFileType": "camera-configuration",
        "inputFileVersion": "0.0.1",
        "targetMachine": "CtrlX7",
        "programName": "lumiscan-cam-driver",
        "programVersion": "0.0.1"
    },
    "id": "camera1",
    "serial": "LX123456",
    "exposureTimeUS": 4321,
    "pixelFormat": "RGB8"
}
```

The description of all the parameters mentioned above can be found in the following Table 5.

Table 5: Camera Example JSON Settings Description

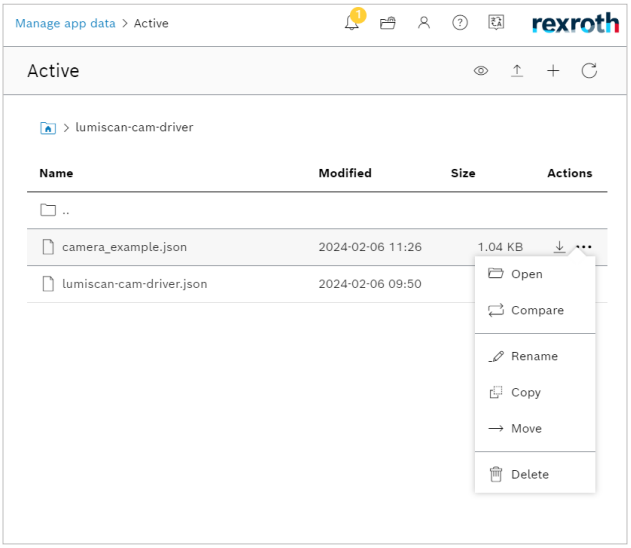
Parameter	Description
meta	Place to set meta information: inputFileType and inputFileVersion are required.
inputFileType	In this case it must be: `camera-configuration`
inputFileVersion	Currently, only version 0.0.1 is supported.
targetMachine	Optional. Enter the name of the machine in use here to indicate that the current application configuration is being used for this particular machine.
programName	Optional. This parameter indicates by which application the current file is used. In this case: `lumiscan-cam-driver`
programVersion	Optional. Currently used program version.
id	This is an id of current camera setting. It must be **unique** across all camera settings.
serial	Serial number of camera.
exposureTimeUS	Exposure time in macro seconds.
pixelFormat	Pixel format of the image, possible values are: RGB8, MON8.

NOTE: After changing the parameters, the LumiScan Cam Driver application should be restarted.

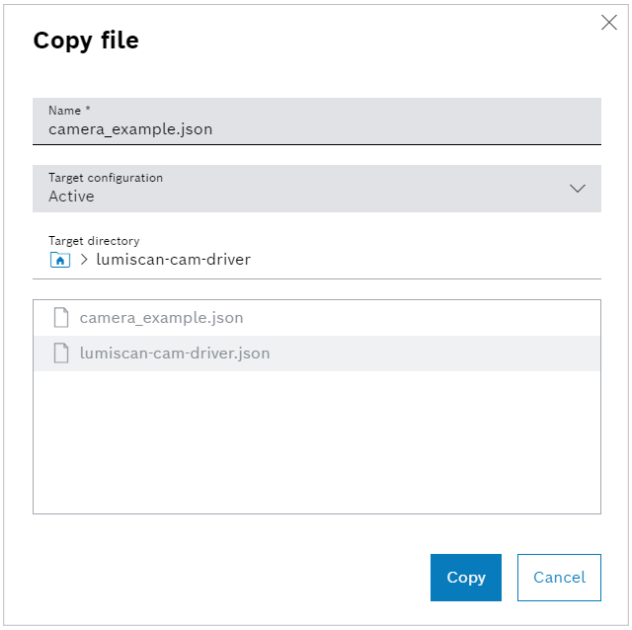
2.8 Creating New Camera Settings Files

You can create a new file with camera settings simply by duplicating an existing file. To do this:

1. Click the triple dots you see to the right of the file name.



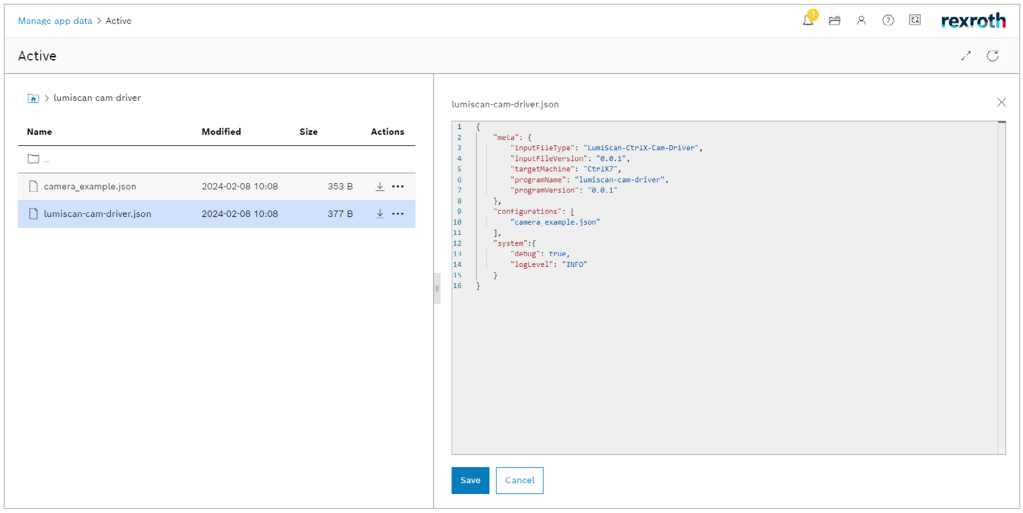
2. Select **COPY**
3. The copy file window appears.



4. Set new file name.

NOTE: You can also rename the new file later by clicking the triple dots to the right of the name and selecting **Rename** from the expanded list.

5. Confirm with clicking on **Copy.**
6. New camera settings file is created.
7. Now you can specify the newly created file name in the application settings (lumiscan-cam-driver.json):
   a. Left-click on the lumiscan-cam-driver.json file name.
   b. The JSON settings window will open on the right.



   c. Enter the name in the `configuration` field.

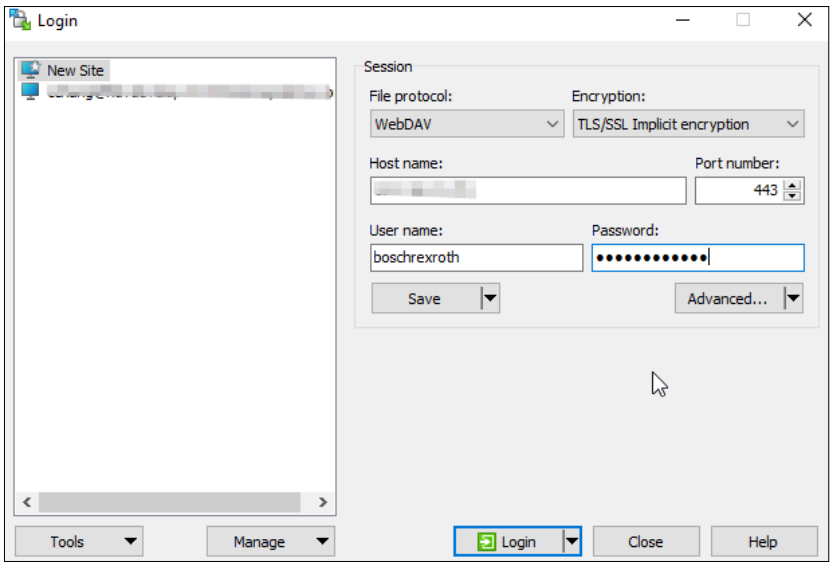
8. Save the changes.

2.9 Uploading Data with WebDAV

Configuration upload is an optional feature, in case the user wants to upload his own settings.

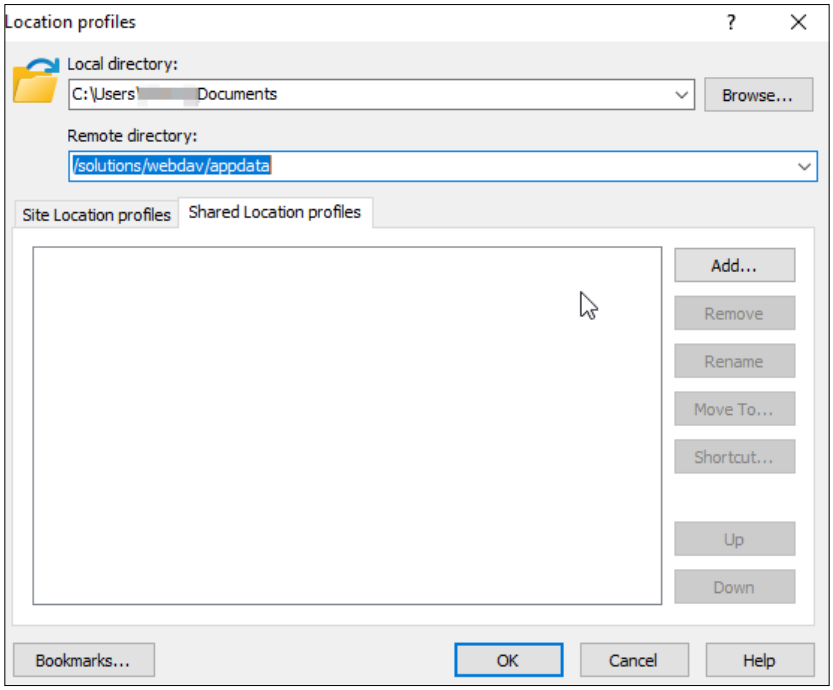
1. Go to the WinSCP website.

2. Download the latest software version.
3. Install it on your device.
4. Start WinSCP.
5. Login page will open.

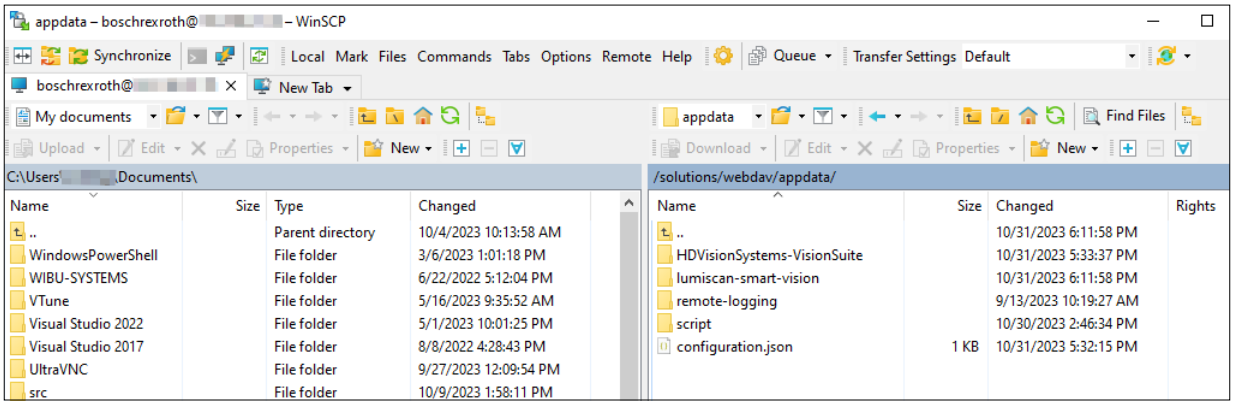


6. Select WebDAV as the **File protocol**.
7. Select TLS/SSL for **Encryption**.
8. Enter **user name** and **password**.

9. Navigate to **/solutions/webdav/appdata.**



10. You can see folder structure as below on the right panel.



You can drag and drop data, settings into the corresponding folder with the application name.

For an example configuration, see Application Settings Content Example.

3 General Disclaimer

1. The manufacturer is not liable for damage to life, body or health or damage to property resulting from improper use. Please note that operating and/or connection errors are beyond our control. We cannot accept any liability for damage resulting from this.
2. Any damage caused by unauthorized debugging activities to inspect, analyze or manipulate the software provided by HD Vision Systems is the sole responsibility of the user. Any attempt to use a debugger is strictly prohibited and may result in irreversible consequences such as loss of data, interruption of service, and even legal action. The manufacturers, developers and distributors of the software are not liable for any damage or loss resulting from the user's failure to comply with this warning.
3. Claims for damages on the part of the purchaser arising from culpa in contrahendo, breach of secondary contractual obligations and tort are excluded, unless they are based on intent or gross negligence on the part of HD Vision Systems GmbH itself or one of our vicarious agents.
4. Recourse claims in the sense of §12 of the Product Liability Act (Produkthaftungsgesetz) are excluded, unless the party entitled to recourse proves that the defect was caused by HD Vision Systems GmbH and was due to at least gross negligence.
5. If any provision of this Agreement is or becomes invalid or unenforceable, the validity of the remaining provisions shall not be affected. The invalid or unenforceable provision shall be replaced by a valid and enforceable provision whose effects come as close as possible to the economic purpose pursued by the parties with the invalid or unenforceable provision. The foregoing provisions shall apply mutatis mutandis in the event that the contract proves to be incomplete. German courts shall have exclusive jurisdiction over all disputes arising in connection with this disclaimer.
6. For more information about the warranty, please contact the manufacturer of the product.

HD Vision Systems GmbH
Carl-Friedrich-Gauß-Ring 5
69124 Heidelberg

T +49 6221 6721900
F +49 6221 6721901

info@hdvisionsystems.com
www.hdvisionsystems.com