Application Note

FESTO

Designer Studio 21 CFR Part 11 Compliance

CDPX

Designer Studio includes a set of functions for responding to the requirements specified in FDA 21 CFR Part 11. The standard is in-tended to provide a solution for securely handling electronic records and electronic signatures in industrial applications.

Title ..... Designer Studio 21 CFR Part 11 Compliance
Version 1.10
Document no. 100805
Original ..... en
Author ..... Festo

Last saved 10.02.2026

General information:

This document is intended for qualified, trained and instructed professionals. The data provided in this document are no guaranteed specifications, in particular with regard to functionality, condition or quality in the legal sense. The information in this document is intended only as simple indications for the implementation of a specific, hypothetical application, and in no way as a substitute for the operating instructions of the respective manufacturers or the design and testing of the respective application by the user. The respective operating instructions for Festo products can be found under www.festo.com. The user of this document must ensure that each function described herein works properly in his application. The user remains solely responsible for his or her own use of this document, even by studying this document and using the information mentioned therein. This also applies to any software (in source code and/or object code) that is made available to the user as an appendix to this document.

Rights of use of the software:

If software is made available to the user as an appendix to this document, the user is granted a simple and unlimited right of use hereto. This right also includes the processing and distribution of the software to third parties in edited or unedited form. The User is not permitted to use the name "Festo" to endorse or promote products derived from the Software without express prior written permission.

Software is provided to the user "as is". Festo does not assume any warranty or guarantee with regard to software. Festo's liability for damages of any kind arising from the use of the software is limited to intent and gross negligence.

Legal Notices:

Table of contents

1 Components/Software used

Type/Name	Version Software/Firmware	Date of manufacture
Designer Studio	4.5.2	

Table 1.1: Components/Software used

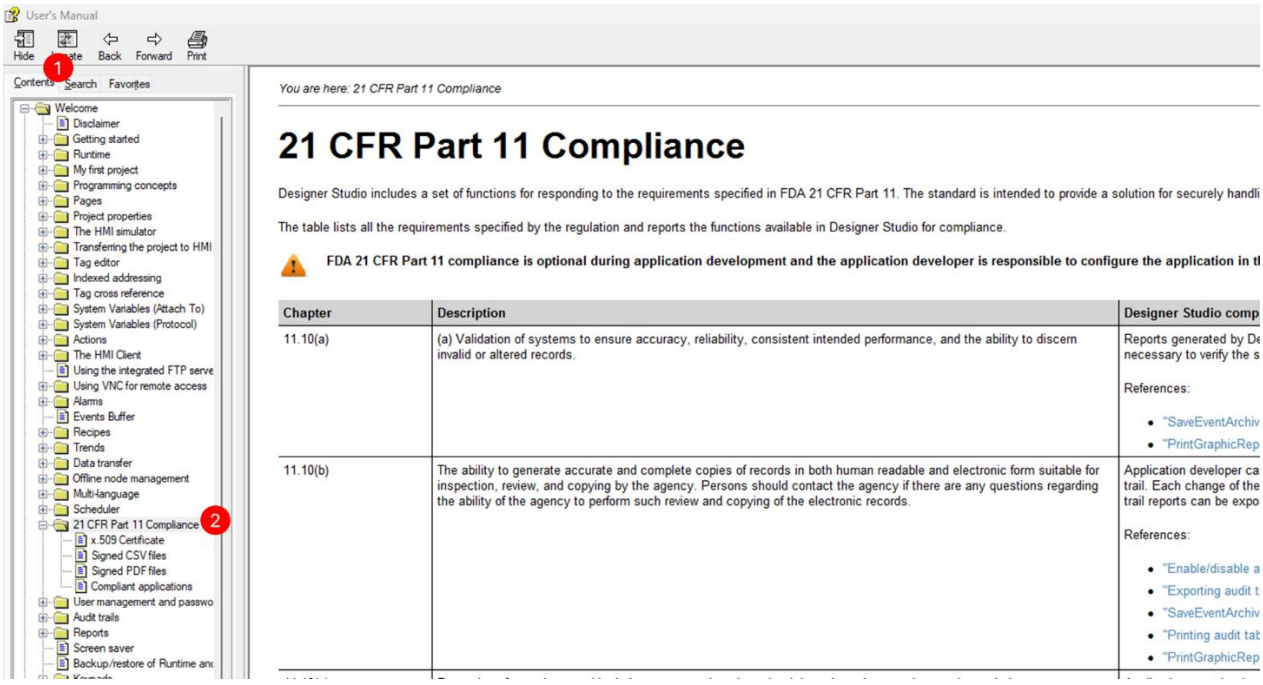
1.1 Designer Studio conformity with FDA 21 CFR Part 11

The table lists all the requirements specified by the regulation and reports the functions available in Designer Studio for compliance.

FDA 21 CFR Part 11 compliance is optional during application development and the application developer is responsible to configure the application in the proper way.

All this information can be accessed in the Designer Studio help, specifically in:

Path: Help > Help Contents > 21 CFR Part 11 Compliance



Chapter	Description	Studio compliance level 11.10(a)
11.10(a)	(a) Validation of systems to ensure accuracy, reliability, consistent intended performance, and the ability to discern invalid or altered records.	Reports generated by Designer Studio can be signed using x.509 Certificates. A certificate that includes the public key, necessary to verify the signature of reports, will be exported with the report. References: • "SaveEventArchive" • " • PrintGraphicReport"
11.10(b)	The ability to generate accurate and complete copies of records in both human readable and electronic form suitable for inspection, review, and copying by the agency. Persons should contact the agency if there are any questions regarding the ability of the agency to perform such review and copying of the electronic records.	Application developer can select the resources (process values, alarms, etc.) whose changes will be tracked to the audit trail. Each change of the selected resources will be recorded with the name of the operator doing the change. The audit trail reports can be exported to .csv or .pdf files. References: • " • Enable/disable audit trail" • "Exporting audit trail as .csv files" • "SaveEventArchive" • "Printing audit table" • " • PrintGraphicReport"
11.10(c)	Protection of records to enable their accurate and ready retrieval throughout the records retention period.	Applications can be developed to self-generate signed reports to external memory or network folders at predefined interval (e.g. at the end of the day) or when circular buffer is full. User is responsible to keep these reports saved for the retention period. References: • "SaveEventArchive" • " • PrintGraphicReport" • "Scheduler"

11.10(d)	Limiting system access to authorized individuals.	Application developer is responsible for the appropriate security configuration of the application. References: • "User management and passwords"
11.10(e)	Use of secure, computer-generated, time-stamped audit trails to independently record the date and time of operator entries and actions that create, modify, or delete electronic records. Record changes shall not obscure previously recorded information. Such audit trail documentation shall be retained for a period at least as long as that required for the subject electronic records and shall be available for agency review and copying.	Audit trail records are stored using a circular buffer (this is to ensure that the device will not run out of memory). Audit trails cannot be modified by the operator. Each record contains a sequential number to easily check the presence of all records. The application can be developed to save/export a copy of the data at regular intervals (e.g. at the end of each day); operator is responsible for storing copy of reports in a safe place. References: • "Exporting audit trail as .csv files" • "SaveEventArchive" • "Printing audit table" • " • PrintGraphicReport" • "Scheduler"
11.10(f)	Use of operational system checks to enforce permitted sequencing of steps and events, as appropriate.	Macros or JavaScript can be used to configure command sequences in the application.
11.10(g)	Use of authority checks to ensure that only authorized individuals can use the system, electronically sign a record, access the operation or computer system input or output device, alter a record, or perform the operation at hand.	The HMI application can be configured • to be accessible only after user sign in with its own password • objects can be configured to be available or not available depending on the user who logged in to the system • resources can be configured to require a password confirmation before be modified References: • "User management and passwords" • "Electronic Signature"

11.10(h)	Use of device (e.g., terminal) checks to determine, as appropriate, the validity of the source of data input or operational instruction.	Resources can be configured to be accessible only from selected user groups. List of allowed IP address can be configured from the User Management settings. References: • "Modifying access permissions"
11.10(i)	Determination that persons who develop, maintain, or use electronic record/electronic signature systems have the education, training, and experience to perform their assigned tasks.	Application developer is responsible to define and assign the appropriate user rights to each user that have access at the HMI device
11.10(j)	The establishment of, and adherence to, written policies that hold individuals accountable and responsible for actions initiated under their electronic signatures, in order to deter record and signature falsification.	Application developer is responsible for establishing appropriate procedures.
11.10(k)'	Use of appropriate controls over systems documentation including: (1) Adequate controls over the distribution of, access to, and use of documentation for system operation and maintenance. (2)Revision and change control procedures to maintain an audit trail that documents time-sequenced development and modification of systems documentation.	Application developer is responsible for establishing appropriate procedures.

11.30	Persons who use open systems to create, modify, maintain, or transmit electronic records shall employ procedures and controls designed to ensure the authenticity, integrity, and, as appropriate, the confidentiality of electronic records from the point of their creation to the point of their receipt. Such procedures and controls shall include those identified in 11.10, as appropriate, and additional measures such as document encryption and use of appropriate digital signature standards to ensure, as necessary under the circumstances, record authenticity, integrity, and confidentiality.	Designer Studio has been designed for operation in closed systems.
11.50(a)	Signed electronic records shall contain information associated with the signing that clearly indicates all of the following: (1) The printed name of the signer; (2) The date and time when the signature was executed; and (3) The meaning (such as review, approval, responsibility, or authorship) associated with the signature.	All records will be added to the audit trail with time stamp and user id of logged user. References: • "Exporting audit trail as .csv files" • "Table audit widget"
11.50(b)	The items identified in paragraphs (a)(1), (a)(2), and (a)(3) of this section shall be subject to the same controls as for electronic records and shall be included as part of any human readable form of the electronic record (such as electronic display or printout).	
11.70	Electronic signatures and handwritten signatures executed to electronic records shall be linked to their respective electronic records to ensure that the signatures cannot be excised, copied, or otherwise transferred to falsify an electronic record by ordinary means.	Application developer is responsible for avoiding using the macros that permit the import/export of user passwords.

11.100(a)	Each electronic signature shall be unique to one individual and shall not be reused by, or reassigned to, anyone else.	System will ensure that two users with the same id cannot be defined. It is user responsibility to avoid removal and reassignment of the same user id to a different user.
11.100(b)	Before an organization establishes, assigns, certifies, or otherwise sanctions an individual's electronic signature, or any element of such electronic signature, the organization shall verify the identity of the individual.	User responsibility.
11.100(c)	Persons using electronic signatures shall, prior to or at the time of such use, certify to the agency that the electronic signatures in their system, used on or after August 20, 1997, are intended to be the legally binding equivalent of traditional handwritten signatures. (1) The certification shall be submitted in paper form and signed with a traditional handwritten signature, to the Office of Regional Operations (HFC-100), 5600 Fishers Lane, Rockville, MD 20857. (2) Persons using electronic signatures shall, upon agency request, provide additional certification or testimony that a specific electronic signature is the legally binding equivalent of the signer's handwritten signature.	User responsibility.

11.200(a)	(a)Electronic signatures that are not based upon biometrics shall: (1) Employ at least two distinct identification components such as an identification code and password.	Designer Studio Security functions are based on the combination Username/ Password.
	(i) When an individual executes a series of signings during a single, continuous period of controlled system access, the first signing shall be executed using all electronic signature components; subsequent signings shall be executed using at least one electronic signature component that is only executable by, and designed to be used only by, the individual. (ii) When an individual executes one or more signings not performed during a single, continuous period of controlled system access, each signing shall be executed using all of the electronic signature components.	Users must enter name and password to access the system. Critical actions can be configured to require entering again the password before execution is started. References: • "User management and passwords" • "Electronic Signature"
	(2) Be used only by their genuine owners; and (3) Be administered and executed to ensure that attempted use of an individual's electronic signature by anyone other than its genuine owner requires collaboration of two or more individuals.	Each user is responsible to not divulge own password. Passwords defined by administrator for first access can be forced to be redefined at first use. References: • "Configuring users"

11.200(b)	Electronic signatures based upon biometrics shall be designed to ensure that they cannot be used by anyone other than their genuine owners.	Designer Studio does not support biometrics.
11.300(a)	Maintaining the uniqueness of each combined identification code and password, such that no two individuals have the same combination of identification code and password.	It is not possible to define to define two users with the same User ID
11.300(b)	Ensuring that identification code and password issuances are periodically checked, recalled, or revised (e.g., to cover such events as password aging).	System can be configured to force each users to define a new and different password after a configurable number of days References: • "Configuring users"
11.300(c)	Following loss management procedures to electronically deauthorize lost, stolen, missing, or otherwise potentially compromised tokens, cards, and other devices that bear or generate identification code or password information, and to issue temporary or permanent replacements using suitable, rigorous controls.	Users can change their password at any time. Administration can redefine each user's password and force them to redefine at the first login. References: • "User management actions" • "Configuring users"
11.300(d)	Use of transaction safeguards to prevent unauthorized use of passwords and/or identification codes, and to detect and report in an immediate and urgent manner any attempts at their unauthorized use to the system security unit, and, as appropriate, to organizational management.	Failed logging attempts are logged to audit trail.
11.300(e)	Initial and periodic testing of devices, such as tokens or cards, that bear or generate identification code or password information to ensure that they function properly and have not been altered in an unauthorized manner.	User is responsible for ensuring appropriate measures.

Table 1.2: 21 CFR Part 11 Compliance descriptions

A Technical appendix

A.1 Compliant applications

Suggestions to development a CFR11 compliant applications

User management macros

User management macros that could be use from any user

- Login
- Logout
- SwitchUser
- ChangePassword

User management macros that could be used from administrator only

- ResetPassword
- AddUser
- EditUsers
- ExportUsers,

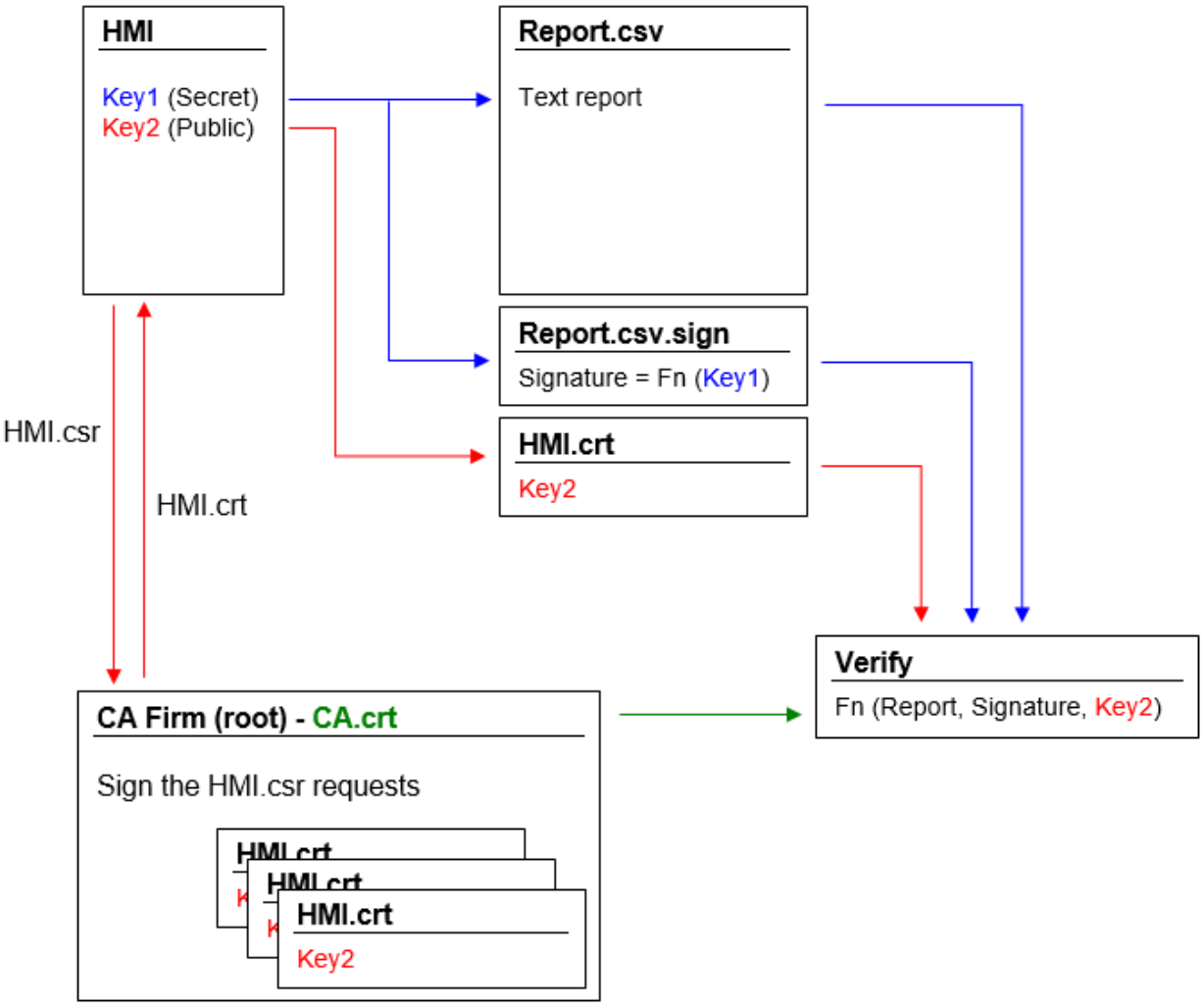
Deprecated macros that must not be used inside CFR 21 part 11 compliance applications

- ImportUsers
- DeleteUser
- DeleteUMDynamicFiles,

A.2 x.509 Certificate

To ensure authenticity of reports generated by HMI devices, HMI Runtime can generate reports with signed files to verify the authenticity and the integrity of the generated reports.

HMI Runtime uses asymmetric cryptography keys to sign files and x.509 standard to manage public key certificates. The picture shows the architecture.



The public key can be signed by a Certificate Authority (CA) that guarantees its authenticity.

Workflow

1. Each HMI device contains two keys:
    - Key1 is the secret key, that is used to sign the reports generated by the HMI device. This key is securely stored inside the HMI device.
    - Key2 is the public key that anyone can use to verify the authenticity of the reports signed by the HMI device.
2. The macros "SaveEventArchive" or "

3. PrintGraphicReport" can be used to generate signed reports (see "SaveEventArchive" or "

4. PrintGraphicReport" for additional details)
3. For the .csv file, you can use the public key and the signed file to verify the report is authentic and not tampered. (See "Signed CSV files")
4. For the .pdf file, you can use a PDF reader to verify the report is authentic and not tampered. (See "Signed PDF files")

The internal x.509 certificate files

Each HMI devices already have a self-signed certificate. You are free to use it, ask a Certificate Authority to sign it, create a new one using the information that you prefer or to upload and use your own certificate. All operations are available from the device "*System Settings*" in the Designer Studio Help.

Note that you can never retrieve the private key from the HMI device. You can instead provide a certificate with both private and public keys.

Use the self-signed certificate

To use the self-signed certificate you have to do nothing. Simply, use the macros that generate signed reports. Even if the certificate will be provided from the macros, you can use the "*System settings*" to retrieve your copy of the certificate (just to be sure of the originality of the certificate).

Use a certificate signed from a Certificate Authority

To use your signed HMI certificate from a certificate authority you must download the certificate signing request file from the "*System settings*" panel. Sending and asking a certificate authority to sign the certificate (generally this is a pay operation) and then upload the signed certificate to the HMI device.

After retrieving the "certificate signed request" file to send to the certificate authority, be sure to never regenerate a new certificate otherwise the internal private key associated with the certificate send to the authority will be lost.

Use your own certificate

If you have your own Certificate and you like to use it, you can upload it inside the HMI device from the "*System Settings*" panel. Note that you must provide both private and public keys.

When the certificate contains a private key, the current private key will be substituted with the key found in the certificate and it will not be possible to recover it.

Example of a certificate with both public and private keys (certificates are encoded base64).

ssl-certificate.crt

```
-----BEGIN CERTIFICATE-----
MIIDBDCCAewCCQDcBYW7PYwJsDANBgkqhkiG9w0BAQsFADBEMQswCQYDVQQGEwJJ
VDEPMA0GA1UEBwwGVmVyb25hMRMwEQYDVQQKDApUZXN0T2ZmaWNlMQ8wDQYDVQQD
DAZITUktMDQwHhcNMTcwNjI2MDgwOTQ1WhcNMTgwNjI2MDgwOTQ1WjBEMQswCQYD
VQQGEwJJVDEPMA0GA1UEBwwGVmVyb25hMRMwEQYDVQQKDApUZXN0T2ZmaWNlMQ8w
DQYDVQQDDAZITUktMDQwggEiMA0GCSqGSIb3DQEBAQUAA4IBDwAwggEKAoIBAQDc
Nlp2kswcbLh4IxS6eeCgQ4EAUHCRpaZ5YPfQ/un9/s0tejaa3Si3Pcqv3JqddJM8
mJEZaPF/+HhAEhtC+rv57TbgullUQJdoQpfoGChofpULforXZt2BfdWNx67plNoa
YM3ElaNtAKIW2o6S9HGEvlkf09XFLGkFgeMgC59+SejgguCNT0m99m6fNa5910I7
UDJFINkC3bxtONj+WiL/iEZYkHXacaN9q06fx+2NfmiSsXGPnmSys5mocqo89tMa
TjyeF7jYpDccCpJ9pY4xRjRpcIkDCM7PabVoG/ascSMUU6XPE2R0W4UJ6bPAygD6
QLKCCq0BUi6/eUj0pnanAgMBAAEwDQYJKoZIhvcNAQELBQADggEBAMLfIEXQOEjS
OpwVkzNxXmL/A6PLU5BKlhVYHb7ofb2Z37zN69vCn8ESglAFYK7QhkhJu3zAD+jH
fYBVKVdxfd3HS8EmcDWxpC6F2lfgqsSqepMRTbKbsSaO53a7JsXtwnHVNfG6EBZV
8tqSlGc4RwtJeVZJelUdmWSBD4Fc7asFeBCKqLrHJinz7buj3I4fLcyscTaMTBI9
fsE7poEpWvKc7NWtKYZglGG3AG6xONu3sEahcJ5k+UVdh/QQdAiCt3vG+JJ/owYU
sd30WIZ4pNzG/GUH9MbJyvI4ftA8IvEhGxHvi3xt7slJnvYQDaghOEDhdtGvilOr
nJZ2FZOBCEI=
-----END CERTIFICATE-----
-----BEGIN RSA PRIVATE KEY-----
MIIEpAIBAAKCAQEA3DZadpLMHGy4eCMUunngoEOBAFBwkaWmeWD30P7p/f7NLXo2
mt0otz3Kr9yanXSTPJiRGWjxf/h4QBIbQvq7+e024LtZVECXaEKX6BgoaH6VC36K
l2bdgX3Vjceu6dTaGmDNxJWjbQCiFtqOkvRxhL5ZH9PVxSxpBYHjIAuffkno4ILg
jU9JvfZunzWufZdCOlAyRSDZAt28bTjY/loi/4hGWJBl2nGjfatOn8ftjX5okrFx
j55ksrOZqHKqPPbTGk48nhe42KQ3HAqSfaWOMUY0aXCJAwjOz2mlaBv2rHEjFFOl
zxNkdFuFCemzwMoA+kCyggqtAVIuv3lI9KZ2pwIDAQABAoIBAGnamsuqrwDu5hGh
02H8GhUPvd/3ytTISujHyvgkwTf+FoTI3Zy9uMe0pUy5/3y2v9v9/qm3P3djafJq
gb5Fprxx4dJPXJZaYi2U7U585lesmVqoHneCk/GeGlyH4zWlwo2xgNgBkkhgaIoR
zz0m0bachVz+SCD6wxUJpbMOw0FBw54oPL0XS/gD+76S9ET7xmqZAS5xV/w8Khht
PtjPfT58GKhqVIC9cMrrBrkuGQPrNrDaJMPsQDxrFp7POQm4+GivrUJ0FA9Vtx46
C5QhXqVps/BODo3mjeOcj2b/FqsvG7WCc5PWOAcCqStmDxl+DQZOIVFSTrE4kdpg
mNn/80kCgYEA88Xfmqg0ta83lpe9b6U0BaLvvslgxgXmCmkyvK7Ru+iKyPUMzxB+
BjGWeeiZuigmIhXfFu3eBs5xOgDrUxf9j55sJAFamljG4LTyun378RnOdA87ff1q
rpF4oPKVfTrfXXzZkeIg0eX2tD6Lsn3+MJwYqpefovxmyJA3kPgcGv0CgYEA50H0
HQififZZ2nApgPf/jJpU7hBLC45cSXvE2MX2I3rd3ptGwzKRo/lZkslbvQutqRln
slyEF+c9LCz6g7FYhJoewChLqCVfeZ9GxBzHeJloxZwmxDXi8L4vmEDphwlcV8b3
ExHqUlMGuINHGelPIRlLKeEsbTQU+OVHuNv443MCgYEA7rMKYhllC6bYCsjowSMG
TqKembX84cqyl+zstp+EVbi99Usm0Lc4f/4cd6EQrplTwbqi6YPgDdAmRQLTalkp
e3FIOPVub4aQr0XgDEcC5bI8W57yxUrZJLjjYs5HHQoB4Dw5m0TOmFnS+enoxs3i
kly3Nowjz+fRCYFWN8kLVE0CgYEA43CLLK7ZcW9XKa2cNBo0PE1g8A4YMJJfk2nl
zKjNj1F9ujyO2NV4RYOsI+RSsFe3ARdJcS6xP20OTc8ixrh57VhCnAxFdGblQpFy
oNgJGkf9zjPoMJsqykjSOHTG+CctqaqmPxwkkLScbIW4PPSn/U6KDPNHpVNOuQeO
hXHak58CgYBLW17l9vgYhUiSWc9Gd3mCSxpAb6y8RcyTgqF76K8v4MalLPqFkEtD
0BaFtlA+PtMLk2ODTRH4XU18oc9eV+7VDFkPJ8T0A2VwjzjMgNAd+vKlm4nOEBTt
UhegY0k8yLxSlZvuYiVnHvKBIoF/G2ckwrxjO9KVE+SA45Ex0Px5qQ==
-----END RSA PRIVATE KEY-----
```

You can import inside each HMI device the same certificate file to have a unique public certificate file for all your HMI devices.

Personalize x.509 Certificate

HMI Device use a self-certificate to encrypt the Internet communication trough the HTTPS protocol. You can personalize the certificate with the data of your Company and ask to a Certificate Authority to firm it.

The procedure to personalize and firm your certificate is:

1. Enter in edit mode and fill the necessary parameters, then push GENERATE button to generate a self-signed certificate with your data.
2. Export the “Certificate Signed Request”
3. Sent the “Certificate Signed Request” to a Certificate Authority to firm it (general this is a paid service)
4. Import the signed certificate into the HMI device

Certificate's parameters

Parameter	Description
Device Name	The name of your device
Organization	The legal name of your organization
Unit	The division of your organization handling the certificate
State	The state/region where your organization is located
Location	The city where your organization is located
Country	The two-letter ISO code for the country where your organization is location
Valid (days)	Validity of the certificate
Key Length	Number of bits of the key used from the cryptographic algorithm

Table 1.3: Certificate's parameters descriptions

Managed certificates are base64 encoding

Required BSP v1.0.239 or greater

A.3 Signed CSV files

Reports generated in CSV format using the **SaveEventArchive** macro can be signed using the x.509 certificate included inside the HMI device. The signature makes sure that nobody tampered with the content of the document since it was signed.

See also:

- The SaveEventArchive parameters ("SaveEventArchive")
- How to provide an x.509 Certificate to Linux devices ("x.509 Certificate")
- How to provide an x.509 Certificate to WinCE devices ("x.509 Certificate")

When required, using Signed=True, the SaveEventArchive macro in addition of the [ReportName].csv generate other two files:

- [ReportName].csv.sign
- ssl-[CertificateName].crt

Where the [ReportName].csv.sign is the signature of the report and the ssl-[CertificateName].crt is a copy of the x.509 certificate of the HMI device. Note that you can retrieve the certificate of the HMI device even from the System Setting of the HMI device.

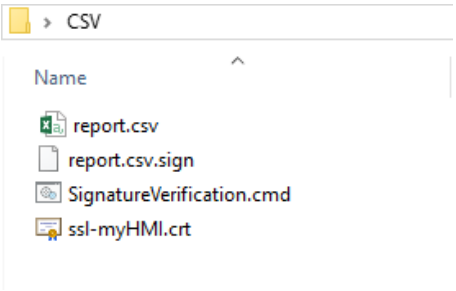
How to verify the report's signature using the public OpenSSL library

To verify that nobody has tampered the content of the report you need

- be sure the ssl-[CertificateName].crt is coming from the HMI device
- use a tool to verify the signature (e.g. OpenSSL-Win32)

Reference.: https://www.openssl.org/

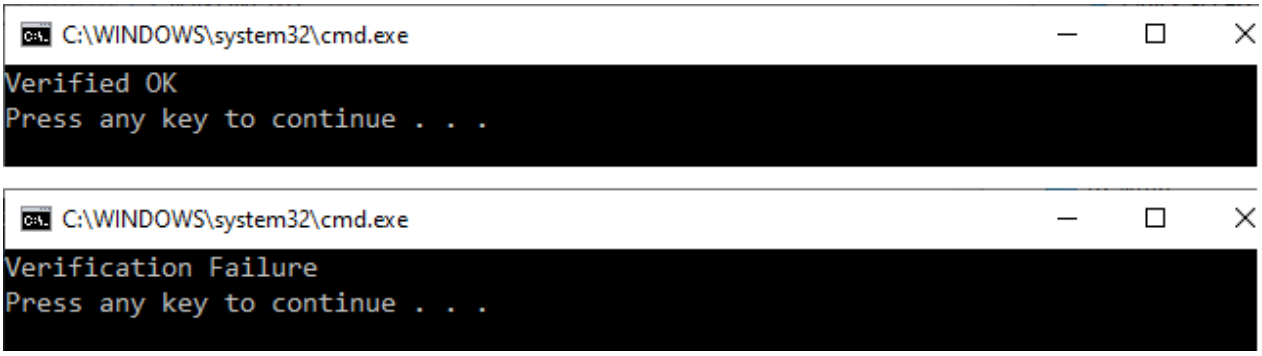
To verify that the .csv report generate from HMI device has not tampered you can install a public OpenSSL library, copy all files generated from the macro inside the same folder and use the below batch file



File: *SignatureVerification.cmd*

```
@echo off set OpenSSL="C:\Program Files (x86)\OpenSSL-Win32\bin\openssl.exe"
set FileToCheck=Report.csv
set hmiCertificate=ssl-myHMI.crt

rem Extract public key from the certificate
%OpenSSL% x509 -in %hmiCertificate% -pubkey -noout > publicKey.pem

rem Verify Signature
%OpenSSL% dgst -sha256 -verify publicKey.pem -signature %FileToCheck%.sign %FileToCheck%

rem Remove public key
del publicKey.pem

pause
```

The below pictures are showing the possible outputs of the batch file



On Linux devices, the BSP v1.0.239 or greater is required

On WinCE devices, the BSP v2.29 or greater is required

A.4 Signed PDF files

Reports generated in PDF format using the **PrintGraphicReport** macro can be signed using the x.509 certificate included inside the HMI device. The signature makes sure that nobody tampered with the content of the document since it was signed.

See also:

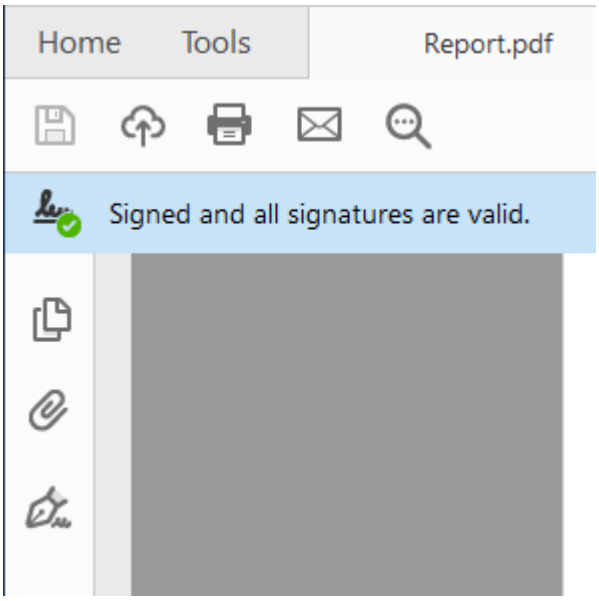
- The PrintGraficReport parameters ("

- PrintGraphicReport")
- How to provide an x.509 Certificate to Linux devices ("x.509 Certificate")
- How to provide an x.509 Certificate to WinCE devices ("x.509 Certificate")

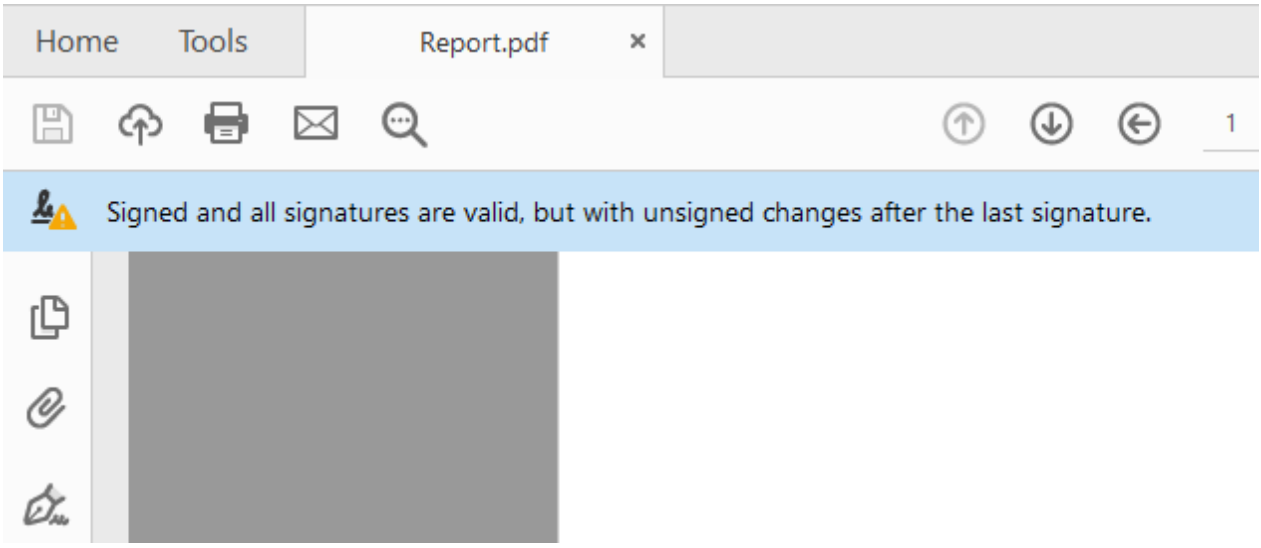
When you open the file, the PDF reader tries to decide if the signature is valid then it looks at the certificate used to sign the document.

x.509 certificate signed from a Certificate Authority

If you have uploaded to the operator panel a valid x.509 certificate, signed by a Certification Authority, when you open the generated PDF file you will get a message that highlights the document is valid.



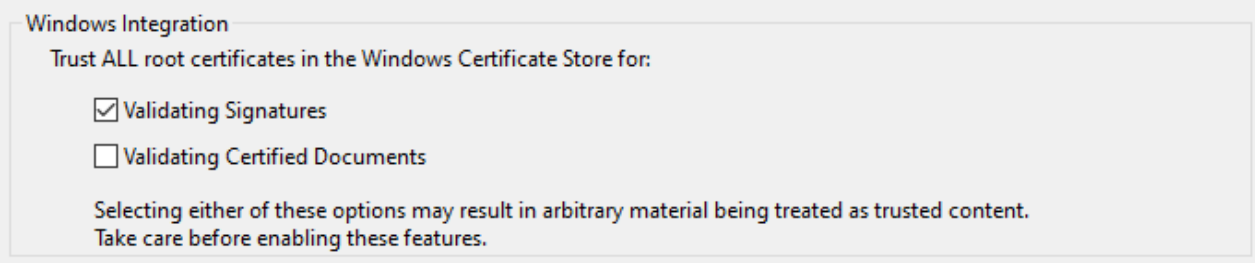
If the document has been modified, it will be highlighted with a different message.



Certificate Trust and Authenticity

Trust of signed certificates depends on the issuer of the certificate. The PDF reader will trust a certificate if you have told it to trust the issuer of that particular certificate. By default the Adobe Reader only trust certificates issued by Adobe or one of their partners. This means that it will show a warning if the certificate wasn't issued by one of these authorities. Microsoft Windows also uses certificates for validating software vendors and content providers. You can configure your Adobe Reader to trust these issuers in addition to the Adobe partners.

Check inside the preferences of the PDF reader if you want to enable the PDF reader to use even the Microsoft Windows certificates

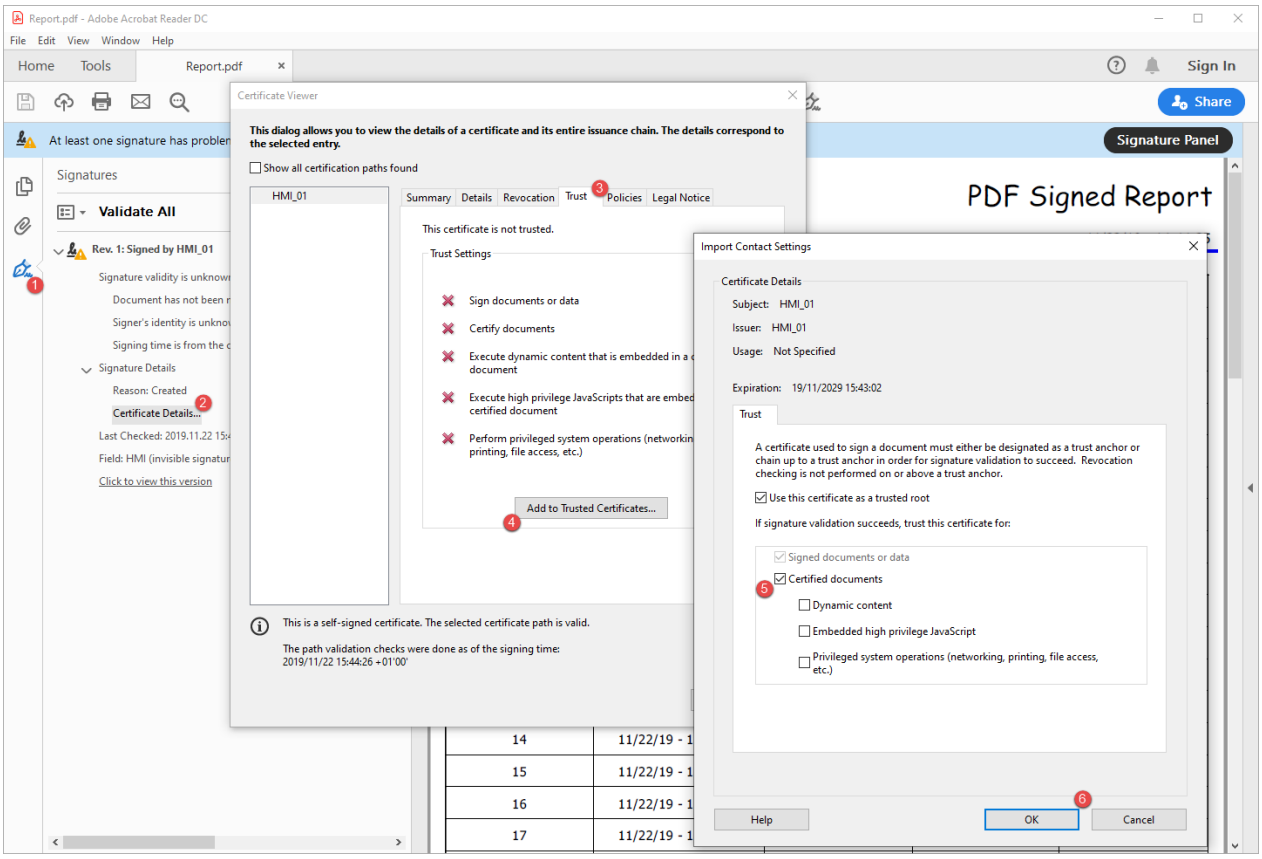


x.509 self signed certificate

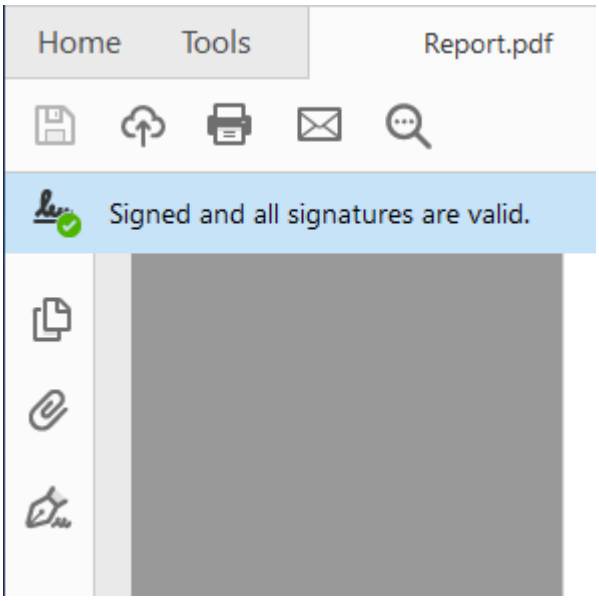
A self-signed certificate is a certificate that is not signed by a certificate authority (CA).

This means that PDF Reader can confirm the file is signed and not tampered, but cannot confirm the signature (alias the certificate) is authentic. Is the user have to take care to verify the certificate is authentic (for example, making sure that the document was actually produced by the panel) and confirm to the PDF reader that the certificate included in the document is valid and that can be considerate valid even for the next reports.

Steps to manual confirm that the certificate is authentic:



Now, if you close and reopen the PDF document you will get the valid signature. Moreover, even all other documents produced from the same HMI device will be shown with the correct signature because the information that the certificate is authentic has been stored inside settings of the PDF Reader.



On Linux devices, the BSP v1.0.507 or greater is required

On WinCE devices, the BSP v2.31 or greater is required

A.5 SaveEventArchive

Save the records located within the audit trail to a signed file. The file signature will ensure that the records within the report are not altered.

Parameter	Description
EventArchive	Name of buffer to dump data
FolderPath	Destination folder • Internal = \Flash\QTHMI\workspace\Dump • USB drive = \USBMemory • SD Card = \Storage Card • Public Network = \\<hostname or IP>\sharePath • Private Network = \\<username>:<password>@<hostname or IP>\sharePath ➔ Note: supported formats for external memory are FAT or FAT32 (NTFS format is not sup-ported). ➔ Note: Private networks are supported only from Linux devices with BSP 1.0.25 and above.

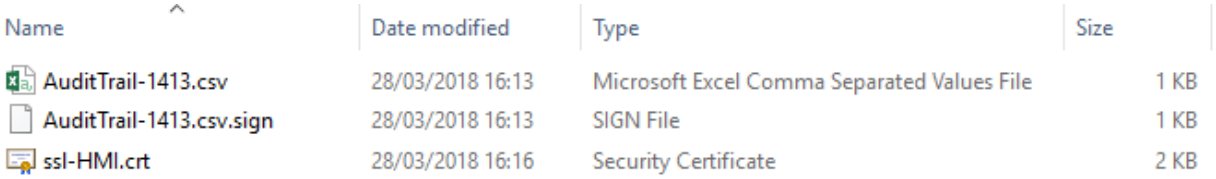
Parameter	Description
FileName	The below wildcards are supported • %n = Event archive name • %y = Year • %M = Month • %d = Day • %h = Hour • %m = Minutes • %s = Seconds Example: \%n\%y%M%d\%h%m%s
Format	Format of the output file • CSV
Signed	Generate the file signature.  On Linux devices, the BSP v1.0.239 or greater is required On WinCE devices, the BSP v2.29 or greater is required  The algorithm to use to signing is defined inside the project properties parameters See "Project" for the available algorithms See also: • "Signed CSV files"
TimeSpec	Time format: • Local = the time values exported are the time of the HMI device. • Global = the time values exported are in UTC format.
PeriodMode	Defines the time window to export • All events • Today • Yesterday • Last week • Last month • Current week • Current month • Custom The additional parameters "periodFrom" and "periodTo" will be shown
Separate Date and Time	Uses two separate columns for Date and Time
Date Format	Select the Date and Time format

Table 1.4: SaveEventArchive descriptions

Signed file

When the "Signed file" parameter is true, two files will be added in addition to fileame.csv:

- filename.csv.sign
  The file signature will ensure that the records within the file filename.csv file have not been altered
- ssl-HMI.crt
  A copy of the certificate of the HMI device required to verify the authenticity of the report.



Name	Date modified	Type	Size
AuditTrail-1413.csv	28/03/2018 16:13	Microsoft Excel Comma Separated Values File	1 KB
AuditTrail-1413.csv.sign	28/03/2018 16:13	SIGN File	1 KB
ssl-HMI.crt	28/03/2018 16:16	Security Certificate	2 KB

For more information about the certificate and how to verify signed files, see . "x.509 Certificate"

For more information about the exported information see "Exporting audit trail as .csv files".

A.6 PrintGraphicReport

Prints a graphic report.

Parameter	Description
reportName	Assigns a name to the report
silent	**false** = allows to set printer properties at runtime
FileName	File name (available only for PDF reports) Supported placeholders: • %n = Report name • %p = Project name • %y = Year, %M = Month, %d = Day • %h = Hour, %m = Minutes, %s = Seconds.
folderPath	Folder Path (available only for PDF reports) Note that the pathname of the arguments field uses native OS format • On WinCE devices Path for USB Device is "\USBMemory" • On Linux devices Path for USB Device is "/mnt/usbmemory" "testFolder" will be inside "/mnt/data/hmi/qthmi/deploy/testFolder"
Signed	Generate the file signature.  On Linux devices, the BSP v1.0.507 or greater is required On WinCE devices, the BSP v2.31 or greater is required  The algorithm to use to signing is defined inside the project properties parameters See also: • "Signed CSV files"

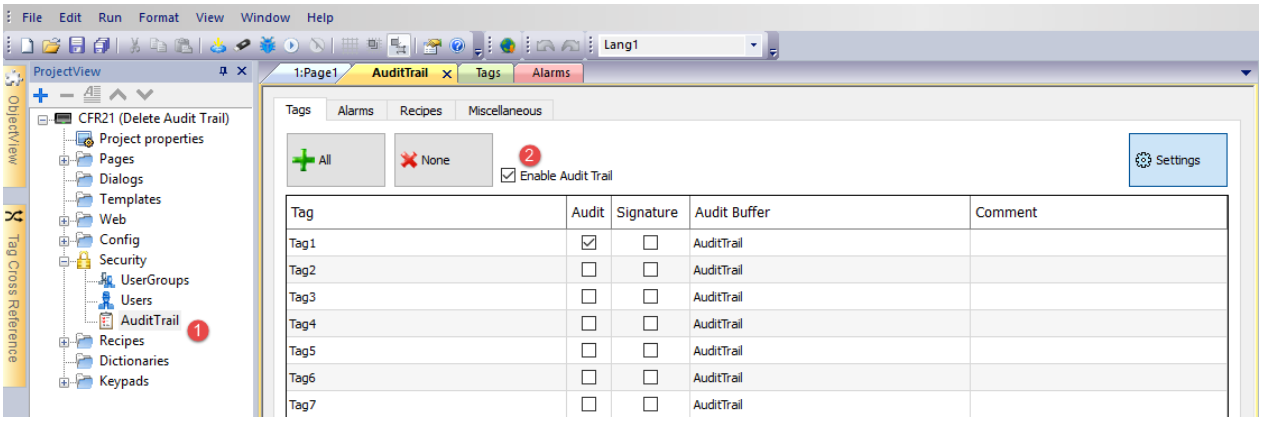
Table 1.5: PrintGraphicReport descriptions

A.7 Enable/disable audit trail

Path: **ProjectView**> **Security** > double-click **AuditTrail**

Audit trail logging can be enabled from the “Enable Audit Trail” check box

When enabled, all changes to the selected resources will be logged to the audit buffer with the time stamp, user name that performed the operation and some additional information concerning the modified resource (e.g. new value and previous value for tags)



From the main tabs (Tags, Alarms, Recipes and Miscellaneous) of the Audit trail Editor you can switch between the list views of the available resources.

Parameter	Description
Audit	Enable tracking of the selected resource
Signature	The user password is required before allowing the resource to be modified from the user (see "Electronic Signature" to additional information)
Audit Buffer	• Internal buffer where store the related audit events (see "Events Buffer" to additional information)
Comment	• Comment space available for the developers

Table 1.6: Audit descriptions

Tags

- Keep track of when tag value changes.

Alarms

- Keep track of when user acknowledges or resets an alarm event

Recipes

- Keep track of when user downloads or uploads recipes

Miscellaneous Resources

- User login details
  Keep track of when user login, logout or change password
- User management actions
  Keep track of when a user is added, removed or when the user properties are modified
- System actions
  Keep track of system actions (HMI Device Restart, Power On, Backup, Update, Download, enter in System Setting, open Project Manager)
- FTP actions
  Keep track of ftpGET, ftpPUT, OpenTextEditor, SaveTextEditor
- Buffer actions
  Keep track of dump and delete actions on alarms, audit or trends buffers

LogMessage Macro

In addition of that, the LogMessage macro gives the possibility to define additional events to log to the audit trail buffer.

This macro give the possibility to developer to decide to keep track of some events (e.g. when a button is pressed, when a page is activate, etc.) into the audit trail. The attach to tag to have the possibility to define the message to log at runtime is supported.

Parameter	Description
EventArchive	Name of the audit buffer where add the message
Message	Message to add inside the audit buffer

Table 1.7: LogMessage descriptions

Cache Memory

Data is temporarily saved in cache memory and flushed to file system when at least one of the following conditions is true:

- temporary cache buffer is full
- an explicit dump procedure has been called
- 5 minutes cycle time has expired

Warning: data in cache memory will be lost if there is a power failure before data has been flushed to the file system.

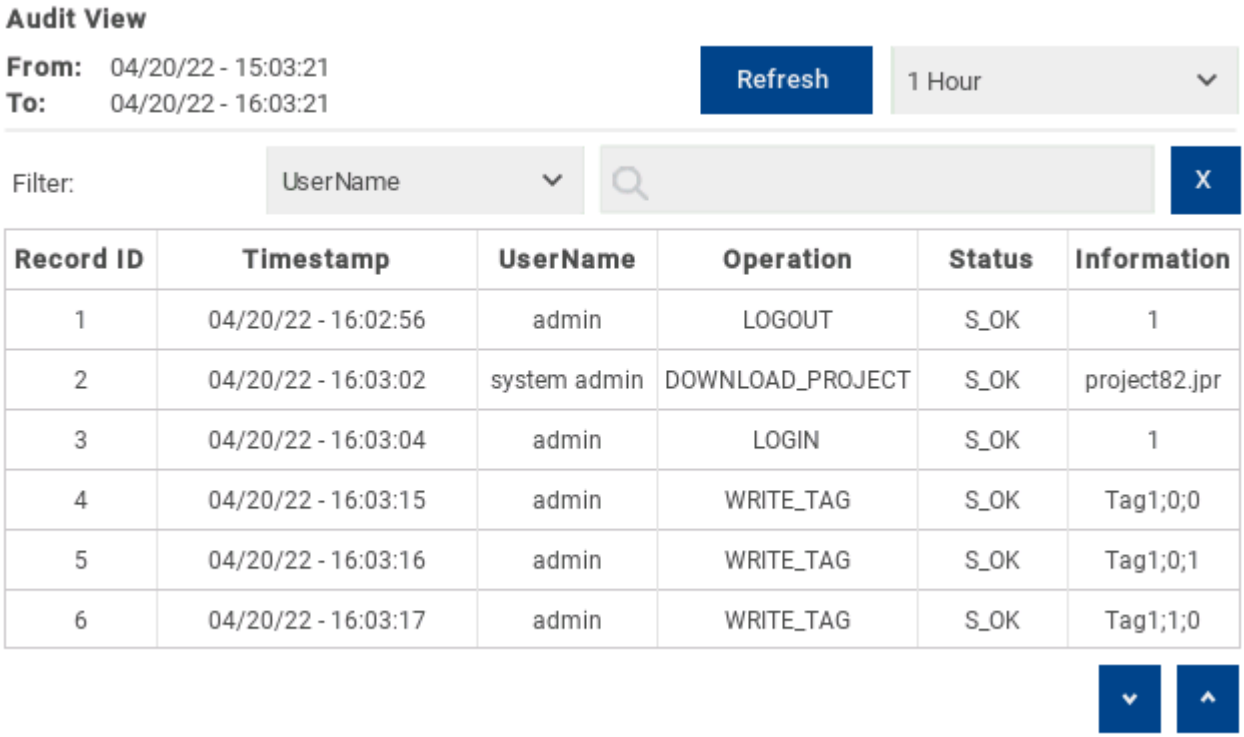
Backup audit events

From the "Events Buffer" you can configure the size of the audit buffer and activate the backup of the audit events when the buffer is full.

A.8 Table audit widget

Path: **Widget Gallery**> **Basic**> **Audit Tables**

Display contents of the audit trail inside a widget



Record ID	Timestamp	UserName	Operation	Status	Information
1	04/20/22 - 16:02:56	admin	LOGOUT	S_OK	1
2	04/20/22 - 16:03:02	system admin	DOWNLOAD_PROJECT	S_OK	project82.jpr
3	04/20/22 - 16:03:04	admin	LOGIN	S_OK	1
4	04/20/22 - 16:03:15	admin	WRITE_TAG	S_OK	Tag1;0;0
5	04/20/22 - 16:03:16	admin	WRITE_TAG	S_OK	Tag1;0;1
6	04/20/22 - 16:03:17	admin	WRITE_TAG	S_OK	Tag1;1;0

Buttons:

- REFRESH
  Retrieve trend data from internal buffer and refresh table view
- BACKWARD/FORWARD
  Move the display window forward or backward as specified in the duration parameter

Filter:

Use the combo box to select the column where search for and the text filed on the right to enter the string to search to.

Parameter	Description
AuditBuffer	Event Buffer from which the event list is retrieved (see "Events Buffer")
Heading	Heading label
Default Duration	Initial value of time window to show
End Time	Upper limit of the time displayed in the table in units of 1 second
Time Spec	Time format: • Local = show the time values of the HMI device. • Global = show the time values using UTC format.
Date Format	Select the Date and Time format
Filter List	Labels to show in filter column selection
Timestamp Sorting	Set how to sort the time stamp data • Ascending • Descending
Table Layout	Defines the characteristics of the scroll bar and allows to remove the header of the table

Table 1.8: Filter descriptions

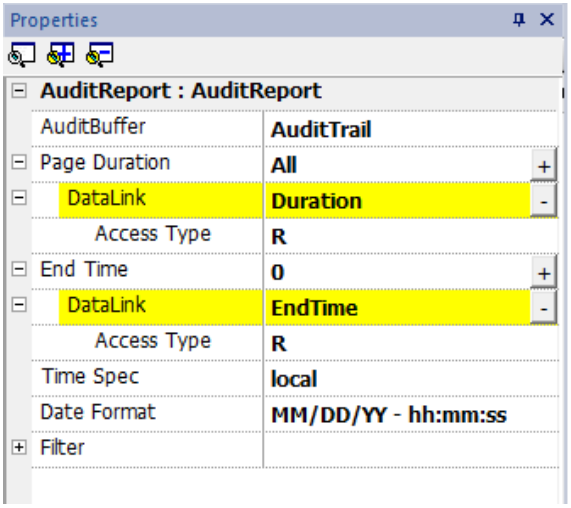
Printing audit table

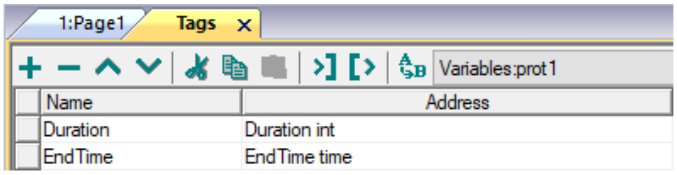
An audit table widget without buttons can be found and used from the print report gallery. The table can be drawn and enlarged to fill the entire page. If the number of lines to printed is greater of one page, the audit table will be printed using additional pages.

Using the “attach to tag” feature is possible to use tags to define some properties of the historical trend to print at runtime:

- Page Duration
- End Time

"Page Duration" with "End Time" define the piece of the audit buffer to print.





A.9 Exporting audit trail as .csv files

Data recorded inside the audit trail can be exported inside a csv file using the **SaveEventArchive** action. See "SaveEventArchive" for details.

File structure

	A	B	C	D	E	F	G	H	I	J	K	L
1												
2	**Record ID**	**Date**	**Time**	**User ID**	**Interface**	**Action**	**Status**	**Data**				
3	1	27/03/2018	14:22:06	SYSTEM_IDAL	SYSTEM_IDAL	SYSTEM_POWERON	S_OK					
4	2	27/03/2018	14:22:06	admin	LOCAL	LOGIN	S_OK	1				
5	3	27/03/2018	14:22:08	admin	LOCAL	WRITE_TAG	S_OK	Tag1	0	1		
6	4	27/03/2018	14:22:09	admin	LOCAL	WRITE_TAG	S_OK	Tag2	0	1		
7	5	27/03/2018	14:22:26	admin	LOCAL	WRITE_TAG	S_OK	Tag2	1	5	This is a test	
8	6	27/03/2018	14:22:50	SYSTEM_IDAL	SYSTEM_IDAL	RECIPE_WRITE_TAG	S_OK	Tag1	1	1		
9	7	27/03/2018	14:22:50	SYSTEM_IDAL	SYSTEM_IDAL	RECIPE_WRITE_TAG	S_OK	Tag2	5	3		
10	8	27/03/2018	14:22:50	SYSTEM_IDAL	SYSTEM_IDAL	RECIPE_WRITE_TAG	S_OK	Tag3	0	5		
11	9	27/03/2018	14:22:50	admin	LOCAL	DOWNLOAD_RECIPE	S_OK	Recipe0	set-00			
12	10	27/03/2018	14:22:54	admin	LOCAL	ACK_ALARM	S_OK	Alarm2				
13	11	27/03/2018	14:22:58	admin	LOCAL	RESET_ALARM	E_FAIL	Alarm2				
14	12	27/03/2018	14:23:02	admin	LOCAL	DUMP_AUDIT_BUFFER	S_NEEDNOT_NOTIFY	AuditTrail				
15												
16												
17	**Record ID**	**Date**	**Time**	**User ID**	**Interface**	**Action**	**Status**	**Data**				
18	13	27/03/2018	14:23:24	admin	LOCAL	DELETE_AUDIT_BUFFER	S_OK	AuditTrail				
19	14	27/03/2018	14:23:26	SYSTEM_IDAL	SYSTEM_IDAL	RECIPE_WRITE_TAG	S_OK	Tag1	1	2		
20	15	27/03/2018	14:23:26	SYSTEM_IDAL	SYSTEM_IDAL	RECIPE_WRITE_TAG	S_OK	Tag2	3	4		
21	16	27/03/2018	14:23:26	SYSTEM_IDAL	SYSTEM_IDAL	RECIPE_WRITE_TAG	S_OK	Tag3	5	6		
22	17	27/03/2018	14:23:26	admin	LOCAL	DOWNLOAD_RECIPE	S_OK	Recipe0	set-01			
23	18	27/03/2018	14:23:27	user1	CGI	LOGIN	S_OK	192.168.49.242				
24	19	27/03/2018	14:23:37	user1	CGI	WRITE_TAG	S_OK	Tag1	6	55		
25	20	27/03/2018	14:24:28	admin	LOCAL	DUMP_AUDIT_BUFFER	S_NEEDNOT_NOTIFY	AuditTrail				
26												

Parameter	Description
RecordID	Each record is stored with a progressive number which will give the possibility to easily identify missing records or confirm that they are not lost. Note that the progressive number is not reset to zero when the buffer is deleted.
Date, Time	Event time stamp. Time can be configured as local or global from the dump action.
User ID	User that perform the operation
Interface	LOCAL: when the action is performed in the HMI device CGI: when the action is performed by a remote client. SYSTEM_IDAL: when the action is performed from the HMI Runtime application
Action	Action executed.
Status	Result of the executed action • S_OK Action executed correctly • E_FAIL Action non executed • S_NEEDNOT_NOTIFY Action triggered (will be executed asynchronously)
Information	Additional info related with the executed action.

Table 1.9: SaveEventArchive descriptions

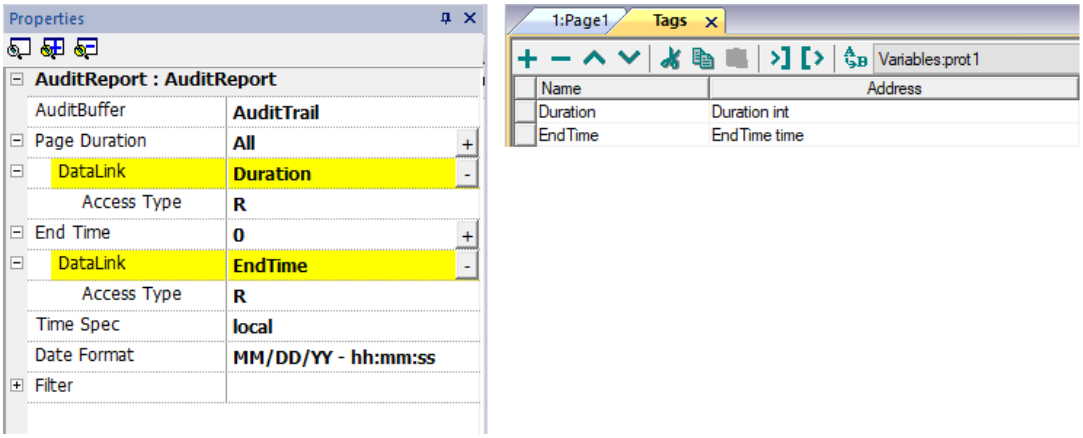
A.10 Printing audit table

An audit table widget without buttons can be found and used from the print report gallery. The table can be drawn and enlarged to fill the entire page. If the number of lines to printed is greater of one page, the audit table will be printed using additional pages.

Using the “attach to tag” feature is possible to use tags to define some properties of the historical trend to print at runtime:

- Page Duration
- End Time

"Page Duration" with "End Time" define the piece of the audit buffer to print.



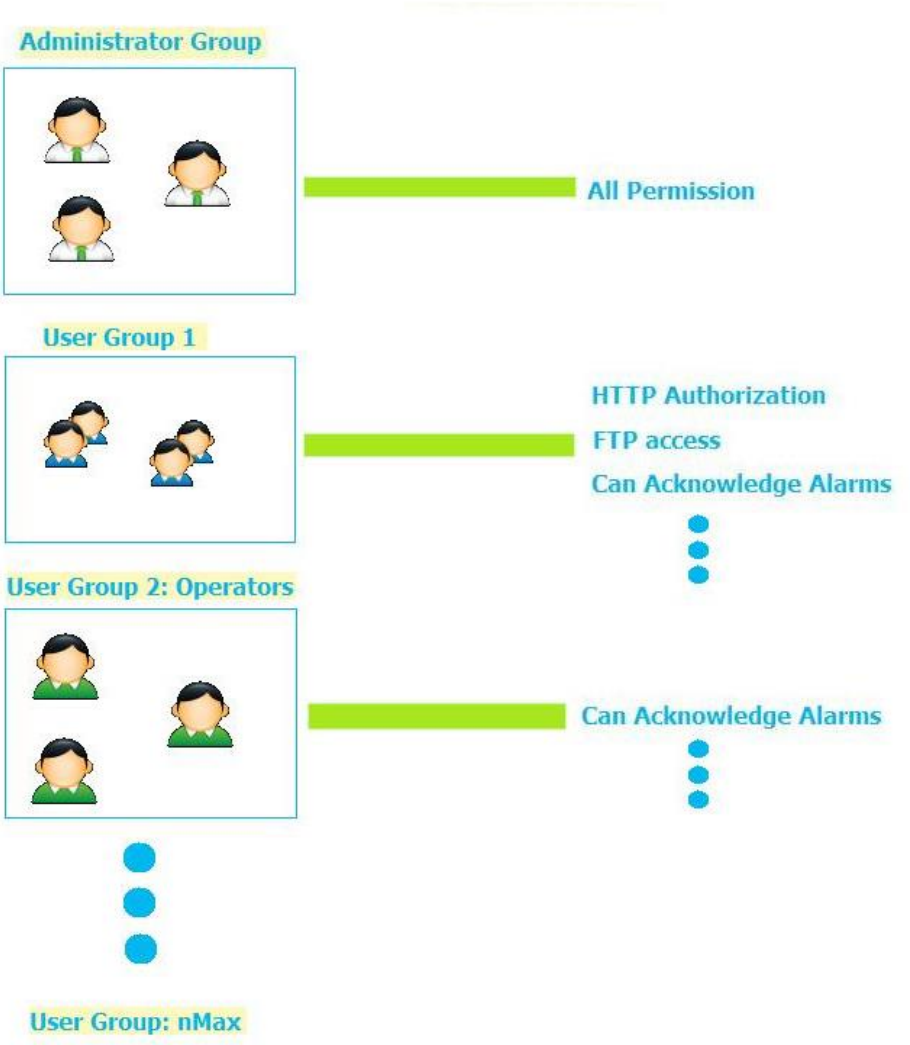
A.11 User management and passwords

You can restrict access to various widgets and operations by configuring users, users groups and assigning specific authorizations to each group.

Each user must be member of one and only one group. Each group has specific authorizations and permissions. Authorizations and permissions are divided in two categories:

- Widget permissions: hide, read only, full access
- Action permissions: allowed or not allowed.

By organizing permissions and groups you can define the security options of a project.

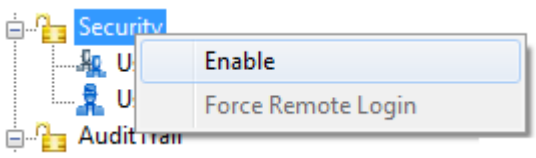


A.12 Configuring users

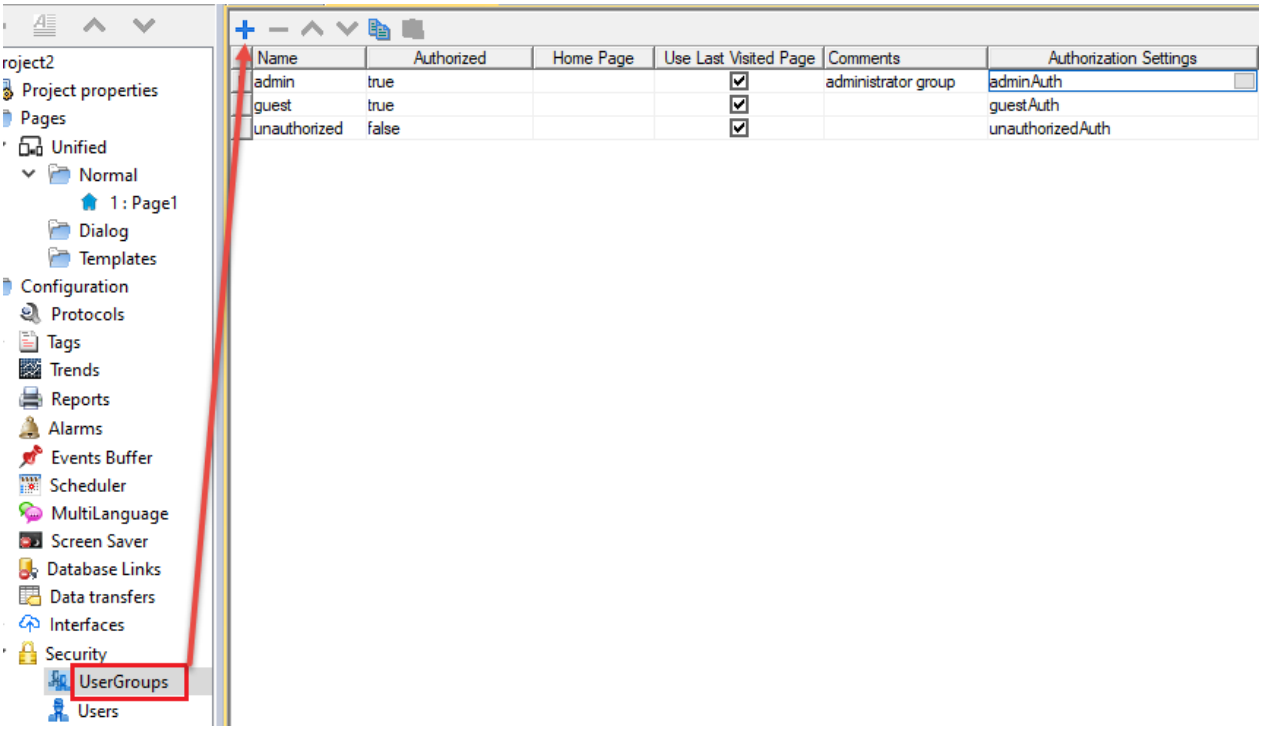
Enable/disable security management

Path: **ProjectView**> right-click **Security**> **Enable**

The padlock symbol indicates whether the function is enabled or disabled.



Path: **ProjectView**> **Security**> double-click **UserGroups**



Three predefined groups are available by default (**admin**, **guest** and **unauthorized**): they cannot be deleted nor renamed. You can, however, modify authorizations and other settings.

Adding a user group

Click **+** to add user group.

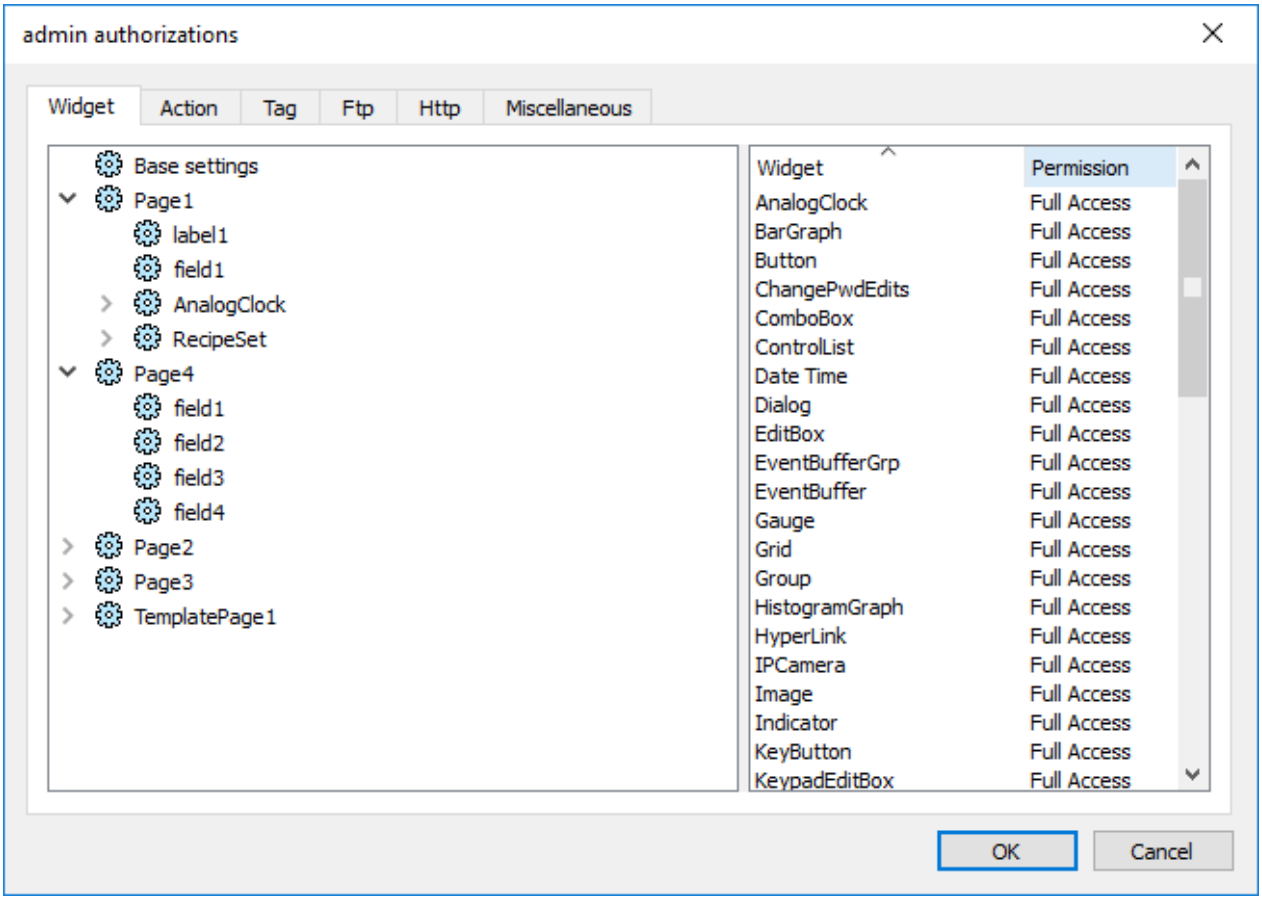
Parameter	Description
Name	Name of users group
Authorized	Authorization granted
Home Page	Page displayed when users belonging to this group log in
Use Last Visited Page	When selected, the last page displayed by the previous user will be displayed when users belonging to this group log in
Comments	Any comment or description for the group
Authorization Settings	Opens the Admin Authorization dialog to set access permissions. • See "Modifying access permissions" for details.

Table 1.10: UserGroups descriptions

A.13 Modifying access permissions

Path: ProjectView> Security> double-click UserGroups > Authorization Settings column

Click the button: a dialog appears with a list of widgets and actions. You can modify access permissions for each one in the list.



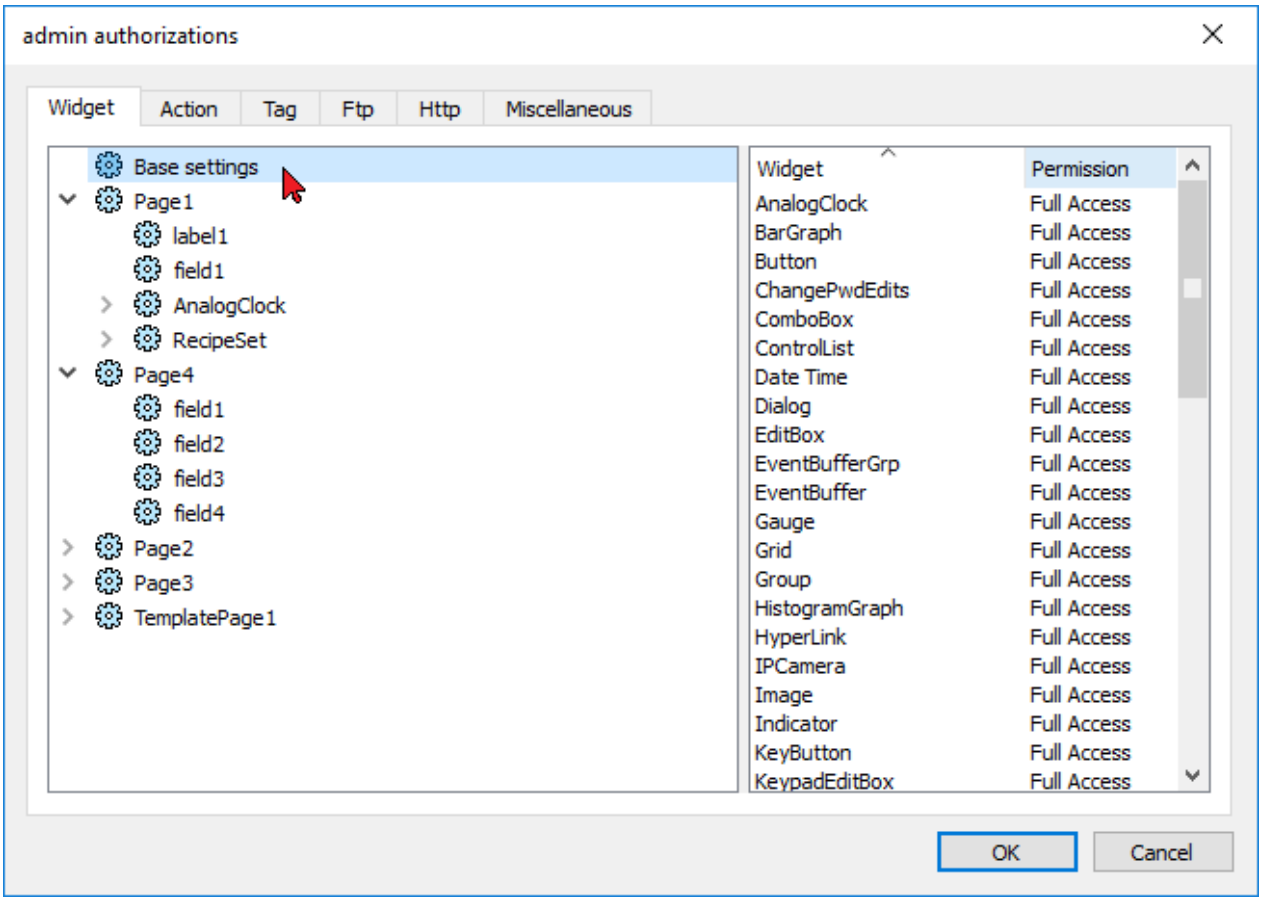
Widget permissions

In the **Widget** tab you can define widget access options at project level, at page level or at widget level for all the widgets used in the project. Lower levels permission (for example, widget level) overrides higher levels (that is, page and project levels).

Use **Base settings** to set default permissions at project level.

Possible settings are:

- **Full Access** to enable read/write access to the widget
- **Read Only** to enable readonly access to the widget
- **Hide** to hide widget for selected group



Changing a widget permission

To change access permission for an individual widget in a page of the project, navigate to that widget within its page on the right pane and customize its access options. Otherwise, all widgets take the permissions set at project or page level.

For example, if page permission for a widget is set at project level to **Read Only**, then all the same widgets will have permission **Read Only**. When you select a widget inside a page from the tree structure, permission is actually set to **Use Base Settings**. You can change this setting and modify access permissions only for this widget in this page.

Access priority

Widget permissions are considered with the following priority:

Permission level	Priority
Project level - Basic settings	Low
Page level	Medium
Widget level	High

Table 1.11: Permissions

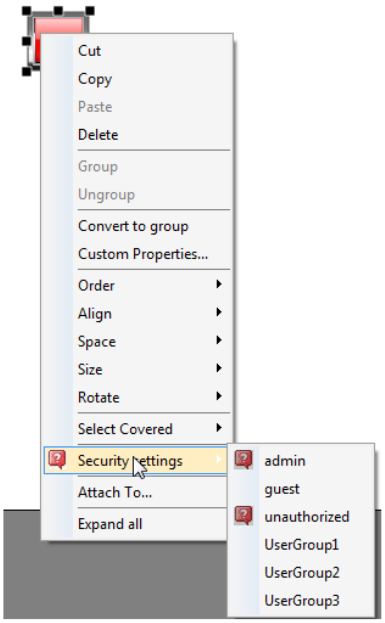
This allows you to specify exceptions for an action or a widget directly from the page view.

For example, if you set permissions for a widget at project level to Read Only and to Full Access at page level then the page level settings will prevail.

Access permissions can be modified directly from the project page.

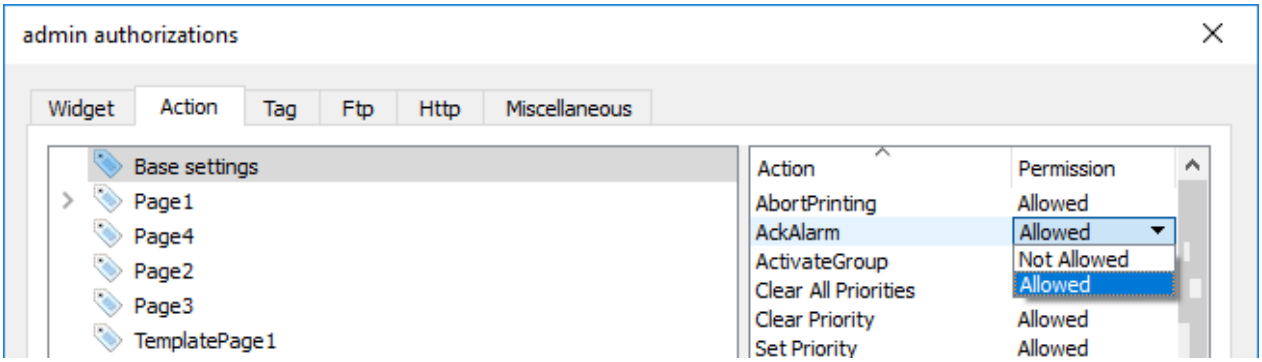
You can assign different levels of security, to different user groups, on a single widget, directly from the project pages.

1. Right-click on the widget and select **Security settings**.
2. Choose the group: the authorization dialog for the group is displayed.
3. Set the security properties to access the widget.



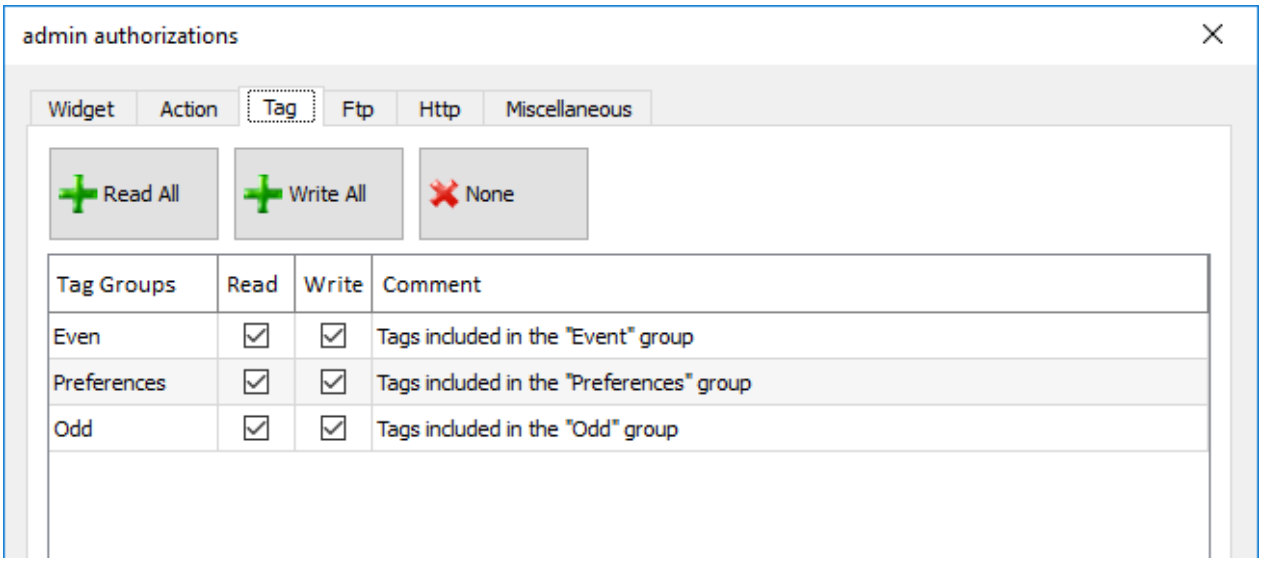
Action permissions

In the **Action** tab you can define action authorizations at project level, at page level or at widget level. Actions can be either **Allowed** or **Not Allowed**.



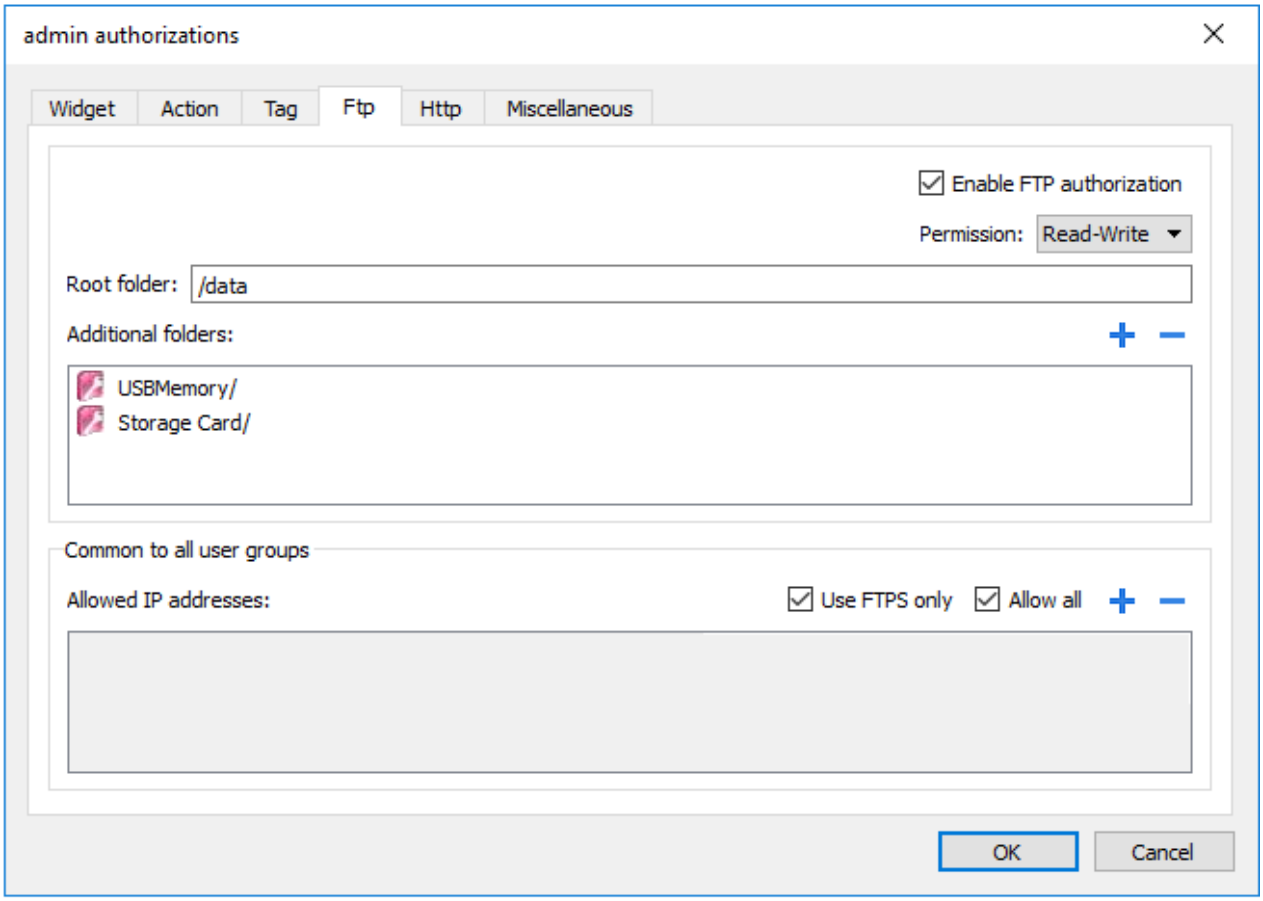
Tag permissions

For each group of tags, you can define the Read/Write access rights



FTP authorizations

In the **Ftp** tab you can set specific authorizations for the FTP server.



Element	Description
Enable FTP authorization	Enables the FTP function for the specific group
Permission	Type of permission: • Read-Only Read-Write
Root Folder	Folder to be used as root for FTP access. This is a relative path.
Additional folder	Extra folders to be used as root for FTP access (for example, on USB drive or SD card)
Allowed IP Addresses	List of IP addresses from which FTP connection can be accepted. **This setting is common to all users groups.**
Use FTPS only	• You can disable this flag if you need to use an old FTP client that does not support encrypted FTP mode, but please note that this is not a secure connection and all your data (even your password) are sent in the clear over the Internet.

Table 1.12: FTP authorizations descriptions

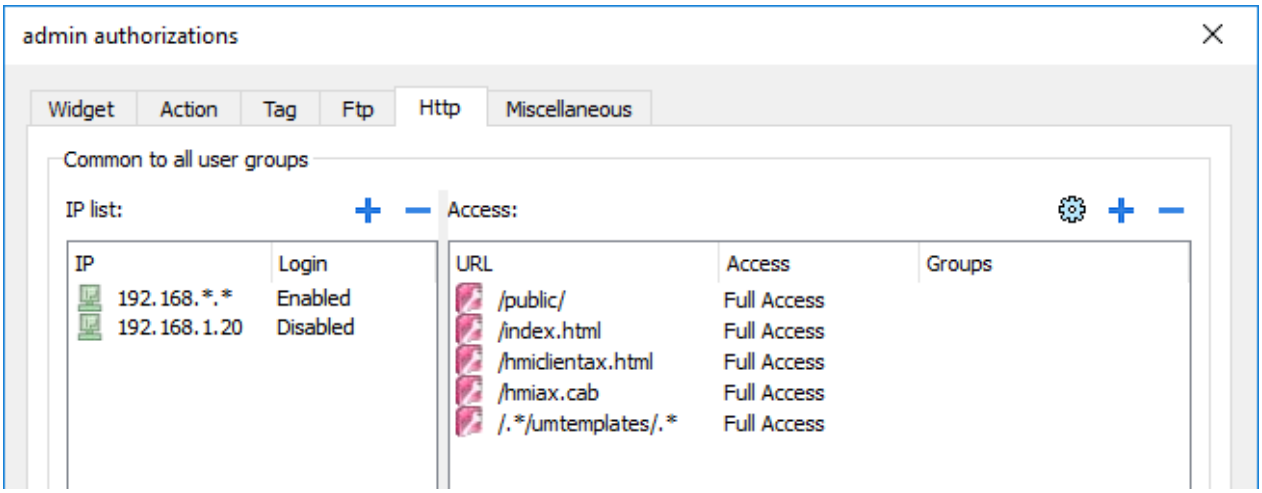
- **Security management must be enabled to use the FTP server**

HTTP authorizations

In the **HTTP** tab you set restrictions to HTTP access to the web server integrated in HMI Runtime.

Wildcards can be used to identify a range of IP addresses.

For example, the two following rules set the HMI device unit can only be accessed by all the IP addresses 192.168.*.* on your local network in which only the IP address of 192.168.1.20 can access the device without entering a login name.



Element	Description
IP list	IP addresses authorized to access the HTTP server. **By default the login is required from any IP address (IP=.*, Login=Enabled).**
Login	When disabled, the username and password are not required.
Access limits	List of resources for which access is limited

Table 1.13: HTTP authorizations descriptions

Effect of these settings depends on whether the option **Force Remote Login** has been selected.

Force Remote Login	Default Access to workspace	Access limits
-	**Full**	-
Disable	**Full**	Can be used to block access to some files/folders or to require authorization
Enable	**No Access**	Can be used to open access to files/folders

Table 1.14: Force Remote Login options

Important: This setting is common to all users groups.

Adding an HTTP configuration

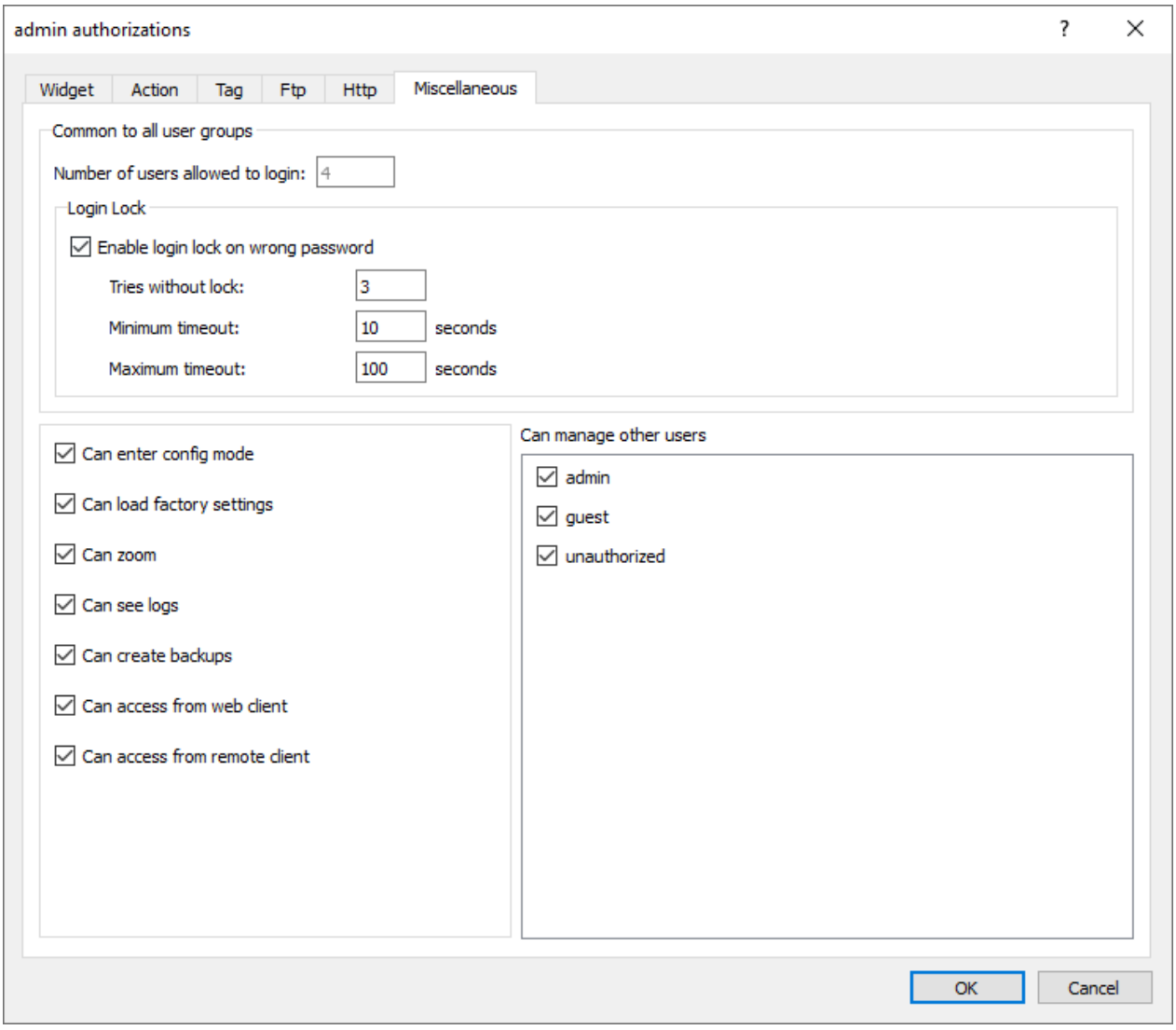
To add and configure a new access click **+**: the **Access limits** dialog is displayed.

To restore the default configuration click the **Set default access limits** icon. Default configuration allows access to the following:

- PUBLIC folder and Index.html

Miscellaneous settings

In the **Miscellaneous** tab you can define various authorization settings.



Option	Description
Number of users allowed to login	Maximum number of users that can be connected to the HMI Runtime at the same time. **This setting is common to all users groups.**
Login Lock	Option to prevent “brute force” attacks.When "Enable access lock on bad password" is selected after the number of bad passwords allowed has been exceeded, the system will introduce a delay between one password and another in order to prevent a possible brute force attack. It is possible to define: • Tries without lock Number of incorrect passwords accepted before inserting a delay between passwords • Minimum/Maximum timeout The initial delay and the maximum delay will not be further increased. Timeout uses an exponential growth. Example of usage: • Tries without lock: 3 • Minimum timeout: 2 • Maximum timeout: 10 First 3 attempts — no timeout Attempt 4 — 2 seconds of timeout Attempt 5 — 4 seconds of timeout Attempt 6 — 8 seconds of timeout Attempt 7 — 10 seconds of timeout Attempt 100 — 10 seconds of timeout Attempt 200 — 10 seconds of timeout
Can enter config mode	Enables switching from runtime to configuration mode. Normally used for maintenance.
Can load factory settings	Restores factory settings.
Can zoom	Enables zoom in/out in context menu at runtime
Can see log	Allows user to see logs at runtime
Can create backup	Allows user to backup project.
Can access from web client	Enables connecting from a web client
Can access from remote client	Enables connecting from HMI Client
Can manage other users	Gives super user privileges at runtime to manage the select groups. Allows adding, deleting and modifying users' permissions.

Table 1.15: Miscellaneous settings descriptions

A.14 User management actions

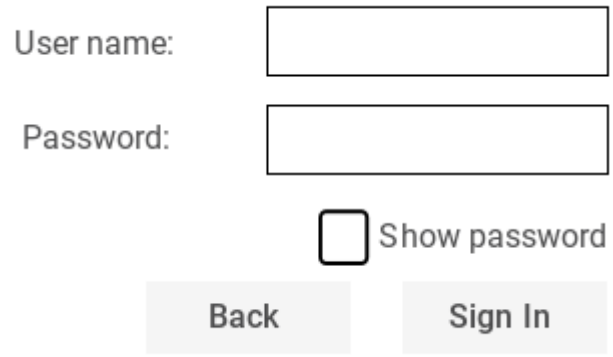
User management and security settings.

LogOut

Logs off the current user. The default user is then automatically logged in. If no default user has been configured, the logon window is displayed.

SwitchUser

Switches between two users without logging off the logged user: the user login dialog appears. User can click **Back** to go back to the previously logged user.



The server continues running with the previously logged user, until the next user logs on. One user is always logged onto the system.

ChangePassword

Change current user password: a dialog appears

No parameter is required.

ResetPassword

Restores the original password together with the settings specified in the project for the current user.

No parameter is required.

AddUser

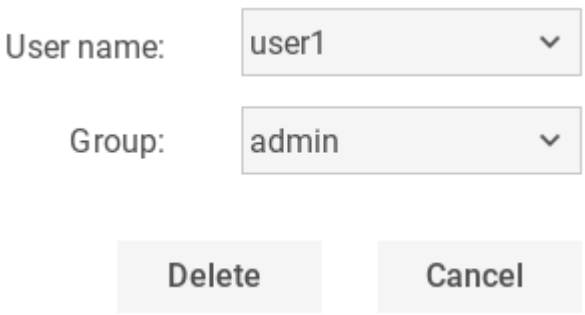
Reserved to users with **Can manage other users** property set.

DeleteUser

Reserved to users with **Can manage other users** property set.

Deletes a user at runtime: a dialog appears.

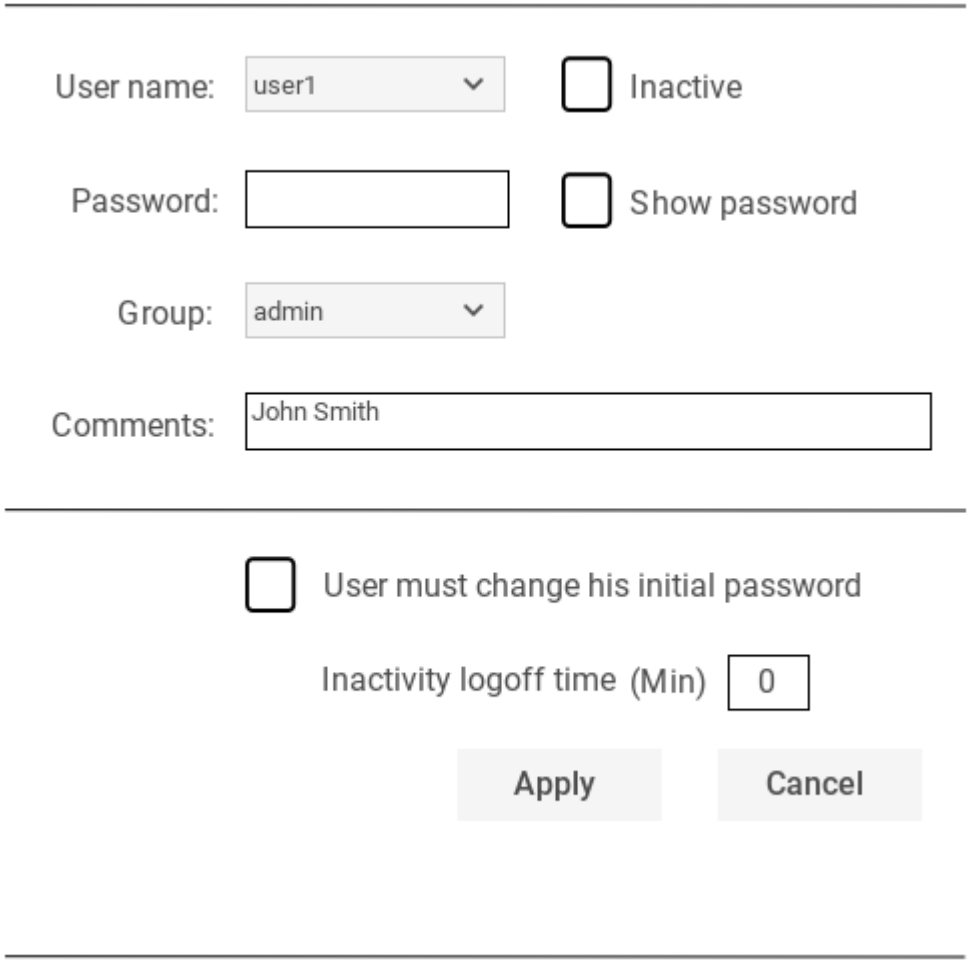
No parameter is required.



EditUsers

Reserved to users with **Can manage other users** property set.

Edits user settings.



Inactive

If you set the *Inactive* flag, the user will no longer be able to log in

DeleteUMDynamicFile

Deletes the dynamic user management file. Changes made to users settings at runtime are erased. The original settings are restored from the project information.

No parameter is required.

ExportUsers

Exports user settings to an .xml file (usermgnt_user.xml) in encrypted format to be restored when needed.

Set destination folder for the export file.

Important: The user file is encrypted and cannot be edited.

Note: supported formats are FAT or FAT32. NTFS format is not supported.

ImportUsers

Imports user settings from a previously saved export .xml file (usermgnt_user.xml).

Set source folder for the import file.

Note: supported formats are FAT or FAT32. NTFS format is not supported.

A.15 Scheduler

Designer Studio provides a scheduler engine that can execute specific actions at set intervals, or on a time basis.

Creating a schedule is typically a two-step process:

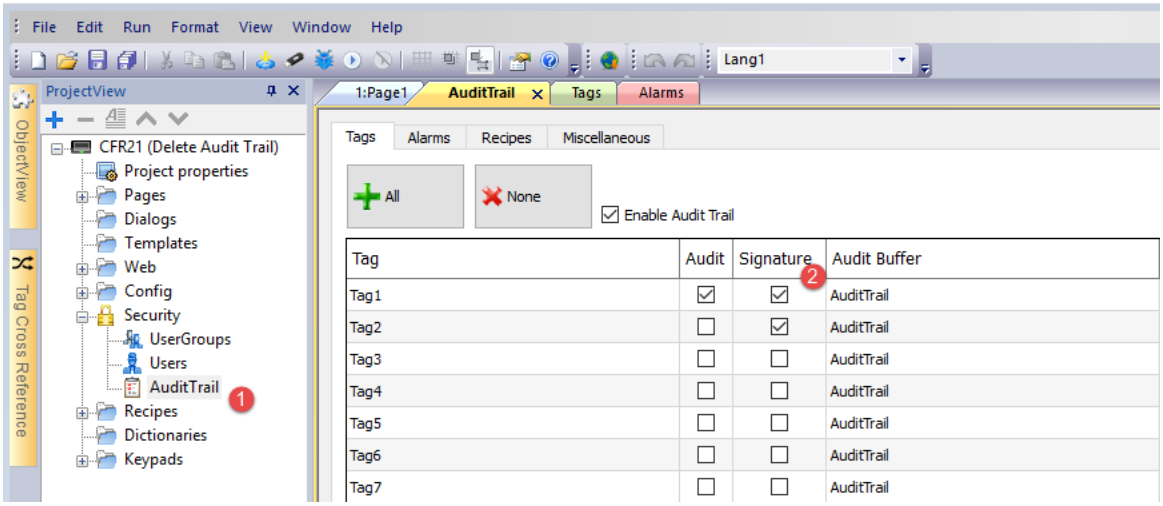
1. You create a schedule with a list of actions to be executed when the scheduled event occurs. You do this in the Scheduler editor
2. You create a runtime user interface that allows the end-user to change settings for each schedule. You do this adding a **Scheduler** widget to a page of your project and configuring it to fit user scheduling needs.

All operations are available from the scheduler engine in the Designer Studio Help.

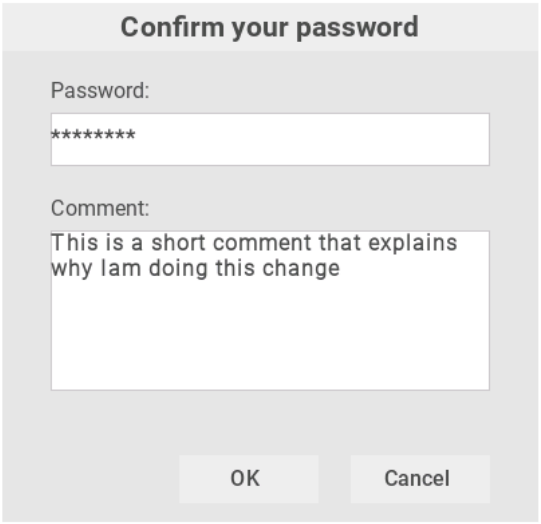
A.16 Electronic Signature

For each resources listed within the Audit Trail editor, it is possible configure the HMI Runtime to require the password confirmation before changing it. If the audit trail log is enabled, the user has the option of adding a comment that will be recorded within the Track Log.

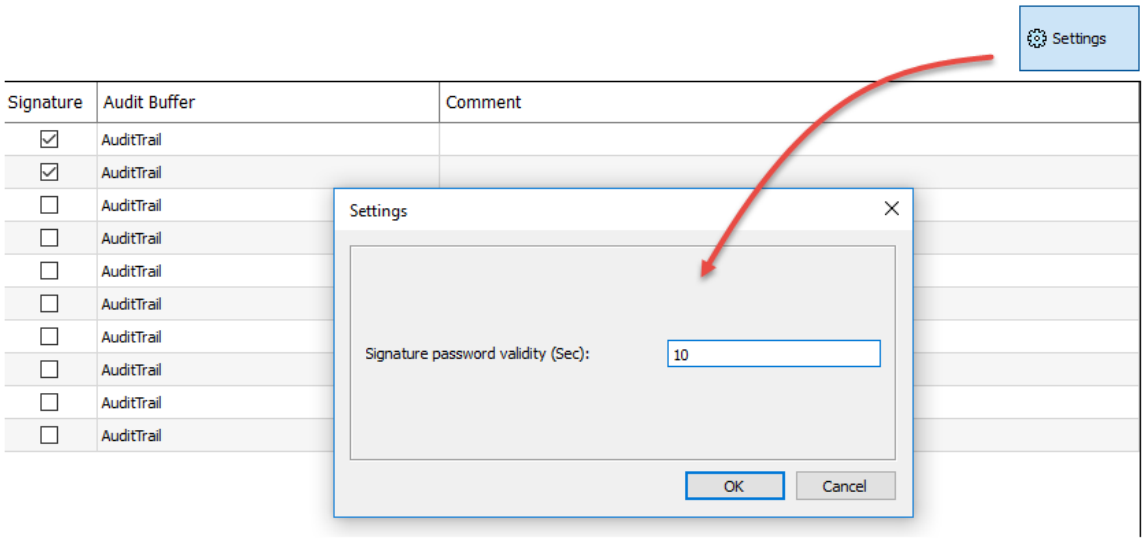
Path: **ProjectView**> **Security** > double-click **AuditTrail**



The user password is required before allowing the resource to be modified by the user

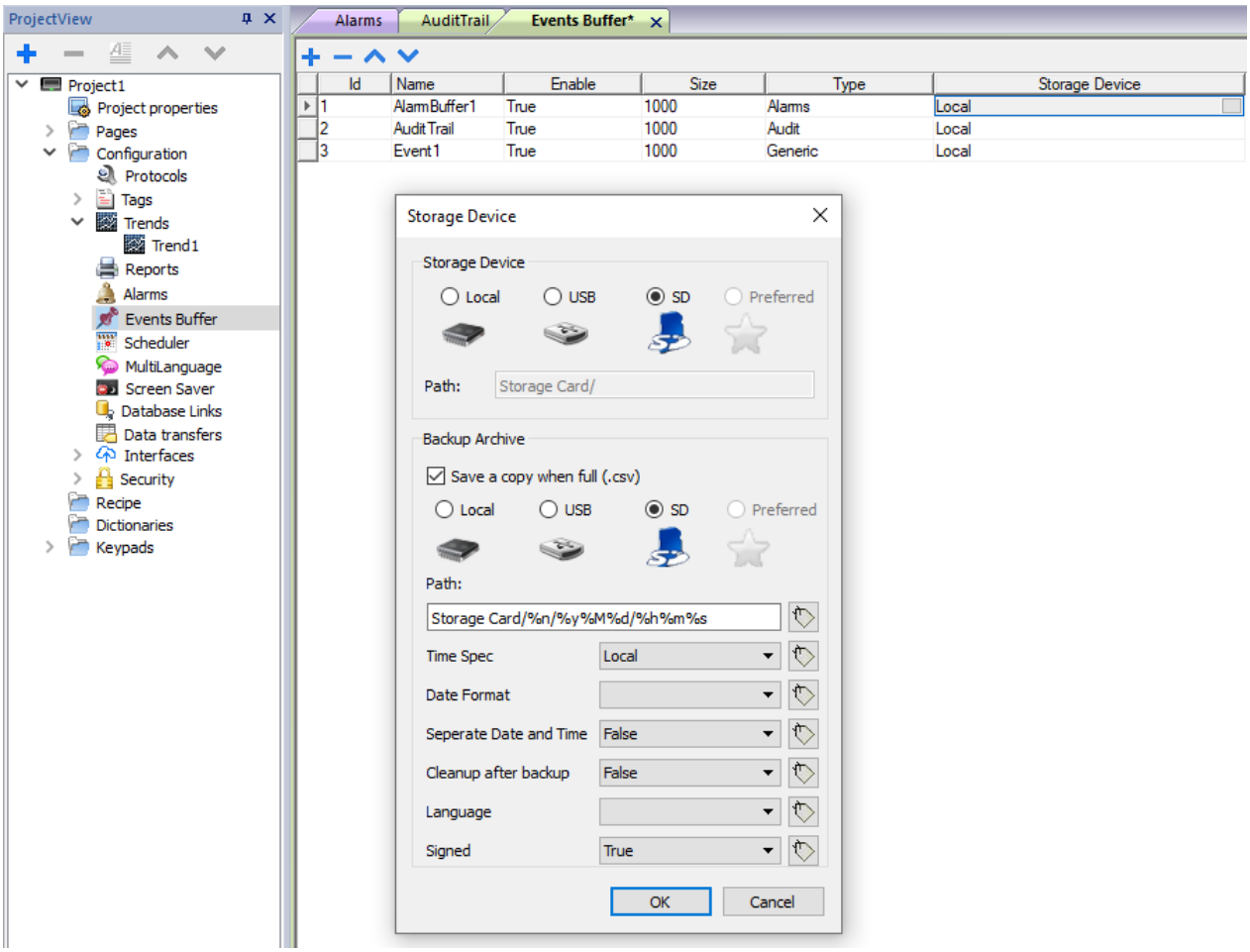


The introduced password will be not required again for the commands released in the next 10 Sec. The validity time can be modified from the Settings dialog.



A.17 Events Buffer

The "Events Buffer" page gives you the possibility to configure the current events buffers (used for store alarms or audit trail information) or add additional events buffers.



Parameter	Description
Id	Buffer identification number
Name	Buffer name
Enable	Enable/disable logging
Size	Number of events stored in the buffer (FIFO). Data is automatically saved to disk every 5 minutes.
Type	Type of events logged: • Alarms • Audit Generic
Storage Device	Device where the data will be stored

Table 1.16: Events Buffer descriptions

Backup Archive

If **Save a copy when full** option is enabled, the HMI device will save a copy when the events buffer is full before it is overwritten by newer data.

Parameter	Description
Path	Where events buffer data will be copied. The below wild cards are supported • %n = Events buffer name • %y = Year • %M = Month • %d = Day • %h = Hour • %m = Minutes %s = Seconds
Time Spec	Timestamp of events • Local Use the time of the HMI device where the project is running • Global Use global time (GMT)
Date Format	Time and Date format. Placeholders can be used
Separate Date and Time	When "true", the date and the time are placed into two different fields
Cleanup after backup	When "true", the event buffer is clean up after completing the backup. When "false", the older events are removed when new events are incoming (circular buffer)
Language	Language to use
Signed	When "true", the additional file with the signature is added (see"Signed CSV files")

Table 1.17: Backup Archive descriptions