IIoT Services manual

IIoT Services manual

v.12

Table of contents

1 Welcome to COPA-DATA help

GENERAL HELP

If you cannot find any information you require in this help chapter or can think of anything that you would like added, please send an email to documentation@copadata.com.

LICENSES AND SERVICES

If you find that you need other zenon services or licenses, our staff will be happy to help you. Email sales@copadata.com.

PROJECT SUPPORT

You can receive support for any real project you may have from our customer service team, which you can contact via email at support@copadata.com.

SUPPORT & SERVICES

Hungry for more information? Want to get your zenon questions answered fast and easy or trouble shoot yourself?

Visit our new Self Service Portal (https://selfservice.copadata.com/) to access hundreds of checklists and FAQs created by zenon experts. Access thousands of technical posts and get involved in our zenon community forum. Search all bugfixes and product changes to keep track of what has changed.

ZENON ACADEMY

If you want to learn about zenon, this is the right place. Easily increase your zenon knowledge, book online or face-to-face courses (https://www.zenon-academy.com/) and receive your zenon Certificates.

ZENON VIDEO TUTORIALS

You can find practical examples for project configuration with zenon in our YouTube channel (https://go.copadata.com/tutorials). The tutorials are grouped according to topics and give an initial insight into working with different zenon services. All tutorials are available in English.

2 IIoT Services

The IIoT Services link local production sites to a global production network.

The IIoT Services are an extension of the zenon software platform.

FUNCTIONS

The IIoT Services supplement the zenon software platform with the following functionality:

- Cross-site networking:
  Secure connection of selected zenon applications to remote sites using standard Internet connections.
- Integration of third-party applications:
  You can easily integrate third-party applications into the zenon software platform with the IIoT API.
- Central authentication service:
  Unified user authentication via the **Identity Service** for selected zenon applications.
- Distribution of software packages to devices:
  The distribution of zenon projects for Service Engine on devices is administered using the

Device Management service. You can create software packages from existing zenon projects. These packages can be deployed to devices instantly or scheduled.

The modular structure of the IIoT Services allows for a wide variety of application scenarios within the zenon software platform. Possible usage scenarios are described in the respective services.

COMPATIBLE APPLICATIONS

Selected zenon applications can be connected to IIoT Services via appropriate connectors.

Compatible zenon applications:

- Service Engine
- Report Engine
- Web Engine

You can also integrate a wide variety of third-party applications in the zenon software platform with **IIoT API**.

Compatible third-party applications include, for example:

- Applications developed by your company
- Third-party manufacturer applications (e.g. ERP systems or MES)
- Dashboard applications (e.g. Grafana)
- Mobile applications

In principle, every third-party application that can exchange data via a REST interface is compatible with the IIoT Services.

EXTENDED AREA OF APPLICATION

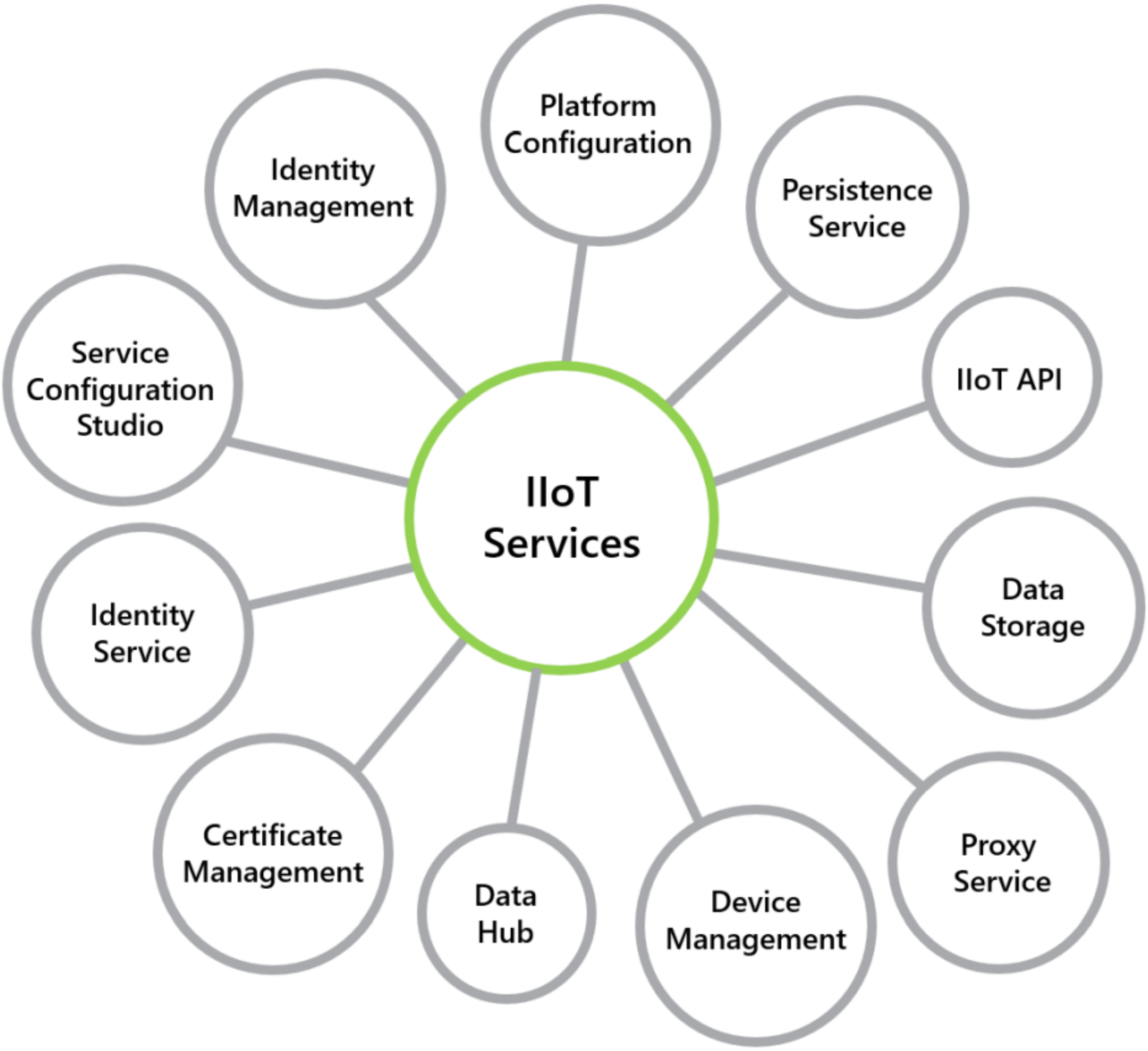
The table shows the benefits of using the IIoT Services:

Description	zenon without IIoT Services	zenon with IIoT Services
Network infrastructure	**Requirements:** - Stable local network connections - High bandwidth required - Protected network infrastructure	**Requirements:** - Connections may be interrupted temporarily - Connections with low bandwidth are generally sufficient - Public network infrastructure (unprotected) is sufficient
Network connections	No compensation of temporary	Compensation of temporary

Description	zenon without IIoT Services	zenon with IIoT Services
	interruptions of the connection.	interruptions of the connection.
Connection to third-party applications	**Either via:** - zenon API - Add-In Framework - Self-developed drivers Connection usually within the production network.	**Uniformly via:** - IIoT API Connection either within the production network or via the Internet.
Primary area of application	- secure local networks (LAN)	- secure local networks (LAN) - insecure wide area networks (WAN)
Cross-site networking	Not supported by default. Only possible by encryption on the network level (VPN).	Supported by default. IIoT Services autonomously encrypt all required connections at application level. Standard Internet connections are sufficient to securely connect sites with each other. A VPN is not required.

3 Basic knowledge

INTERNAL SERVICES OF THE IIOT SERVICES



The graphic shows some logical units and internal services of the IIoT Services.
Logical units are:
1. IIoT Services – umbrella term for the sum of all internal services.
All other elements of this illustration are internal services.

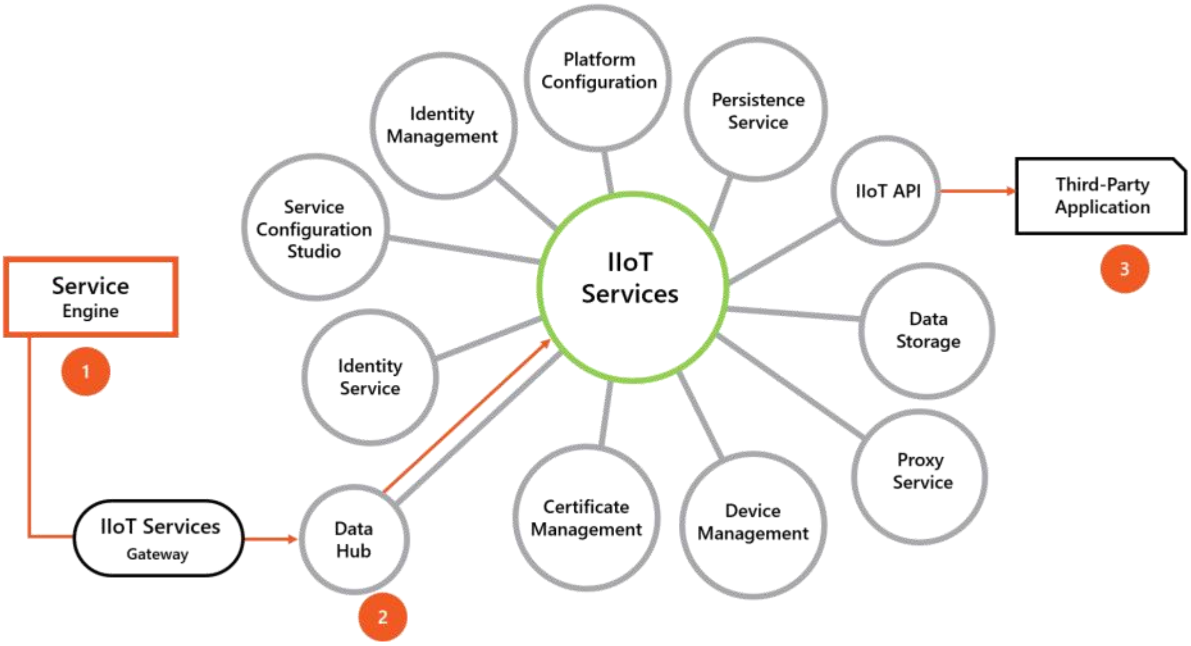
A typical IIoT Services installation consists of a multitude of internal services.

Internal services are for example:

- **Identity Service** – authorizes access of users and clients.
- **IIoT API** – allows third-party applications to access the IIoT Services.
- **Data Storage** – Central storage of archive data (historical variable values)

Internal services are automatically connected to each other and fundamentally set up during the installation of the IIoT Services. How the internal services are installed depends on the installation option selected.

EXTERNAL SERVICES FOR THE IIOT SERVICES



A use case for the IIoT Services:
1. A Service Engine instance generates process data as an external service.
2. The process data is fed into the IIoT Services using the Data Hub.
3. Third-party applications can access the process data as external services by means of the IIoT API.
The amount of internal services of the IIoT Services that are involved in a data transaction always depends on the specific use case.

You can connect different external services to the IIoT Services. External services are all applications that you can connect to IIoT Services but are not part of a IIoT Services installation themselves.

Examples of external services are:

- **Service Engine**
- **Report Engine**
- Third-party applications

You must connect external services to the IIoT Services yourself and configure them accordingly. In a typical use case (on page 46), you must also install and configure a number of other applications in addition to the IIoT Services.

⚠ Attention

zenon applications:

- zenon applications that are not part of the internal services of the IIoT Services are always treated as external services by the IIoT Services.
- This also applies where both zenon applications and the IIoT Services have been installed on the same computer through the same platform setup.

For this, see the installation recommendations.

3.1 Architecture

The data flows in the IIoT Services are protected by a multi-stage security concept. This begins in the transport layer of the network traffic.

Communication is secured by:

- Trusted **Certificate Bundles** for internal services and zenon applications.
- Automatic verification of the send and receive permissions of messages on the protocol level.
- Central user authentication with **Identity Service** for users and client applications.

The security concept of the IIoT Services thus also protects communication via public non-secure networks such as the internet. You do not require a VPN for cross-site connections. A standard internet connection is enough.

3.2 Internet connection

The IIoT Services have only low requirements for network connections. Basically, even narrowband internet connections are sufficient to connect applications. In addition, temporary connection interruptions are compensated for by IIoT Services.

However, the connection quality needed for the operation of a specific application always depends on the particular use case.

Important parameters for the quality of internet connections are:

- Latency
- Bandwidth
- Connection stability

When planning a use case, you must determine the quantities of data to be sent via the internet connection and in which intervals. This determines the required connection quality.

3.3 Configuration files (Docker)

You can download the configuration files for IIoT Services from the zenon website.

<u>**The download contains:**</u>

- IIoT Services configuration files: *.env* and *docker-compose.yml*
- PDF file: **IIoT Services Help** (including **Getting Started Guide**)

<u>**To download the configuration files:**</u>

- Go to the **copadata.com** website
- Go to the following subpage:
  Downloads -> Product-Downloads -> Software -> Current versions -> IIoT Services
- Download the *.zip* file with the latest version of **IIoT Services** (Docker).

Note: You must log in to the COPA-DATA website with your user account for this download. Registration is free.

.ENV FILE

The **.env** file contains configuration data for user names, logins and the database connection. The file is provided with empty variable fields. The configuration files are read when the IIoT Services are initialized.

<u>**To do this, you must:**</u>

- configure the **.env** file with the values for your system.

The file will be loaded again every time Docker is started. But not all values will be reset when an initialized system is restarted!

The values that have to be set once during the initial configuration can be found in the **.env** file.

> **Tip**
>
> **Subsequent editing of the .env file**
>
> Once the IIoT Services have been initialized for the first time, the **.env** file should not be edited again. This helps you avoid having to make extensive manual configurations of an existing system.

DOCKER-COMPOSE.YML

The **docker-compose.yml** file contains basic settings for the configuration of the IIoT Services.

<u>**You need this file to:**</u>

- Initialize the IIoT Services via the command line.

- Start the already-initialized IIoT Services via the command line.

The file is fully configured and should not be edited.

DOCKER-COMPOSE.OVERRIDE.YML

The **docker-compose.override.yml** file gives you the option to enable the **Persistence Service** port on the host system. You then have access to the **Persistence Service** through tools such as **Mongodump** and **Mongorestore**.

To do this, you must:

- Start **docker-compose.override.yml** together with **docker-compose.yml**.

The file is fully configured and should not be edited.

DOCKER-COMPOSE.WEB-ENGINE.YML

You need the optional **docker-compose.web-engine.yml** file if you want to use **Web Engine** together with the IIoT Services.

To do this, you must:

- Configure the relevant section for "**HTML Web Engine**" in the **.env** file.
- Start **docker-compose.web-engine.yml** together with **docker-compose.yml**.

Further configurations are not necessary in this file.

3.4 Minimum password requirements

The minimum password requirements must be met for every password assigned for IIoT Services. This is due to technical reasons and thus also applies for protected test environments.

Unsuitable passwords can cause malfunctions in IIoT Services.

The minimum password requirements in IIoT Services are:

- Password length: at least 8 characters
- One uppercase letter (*A - Z*)
- One lowercase letter (*a-z*)
- One numeric character (*0 - 9*)
- One special character (!#$%&'()*+,-./\:;<=>?[]@^_'`{}|~)

Information

The password complexity can be parameterized in the Service Configuration Studio in **Identity Management** in the settings node.

Only when all the requirements have been met is a password regarded as being suitable. Unsuitable passwords fail to meet at least one of the above-mentioned requirements.

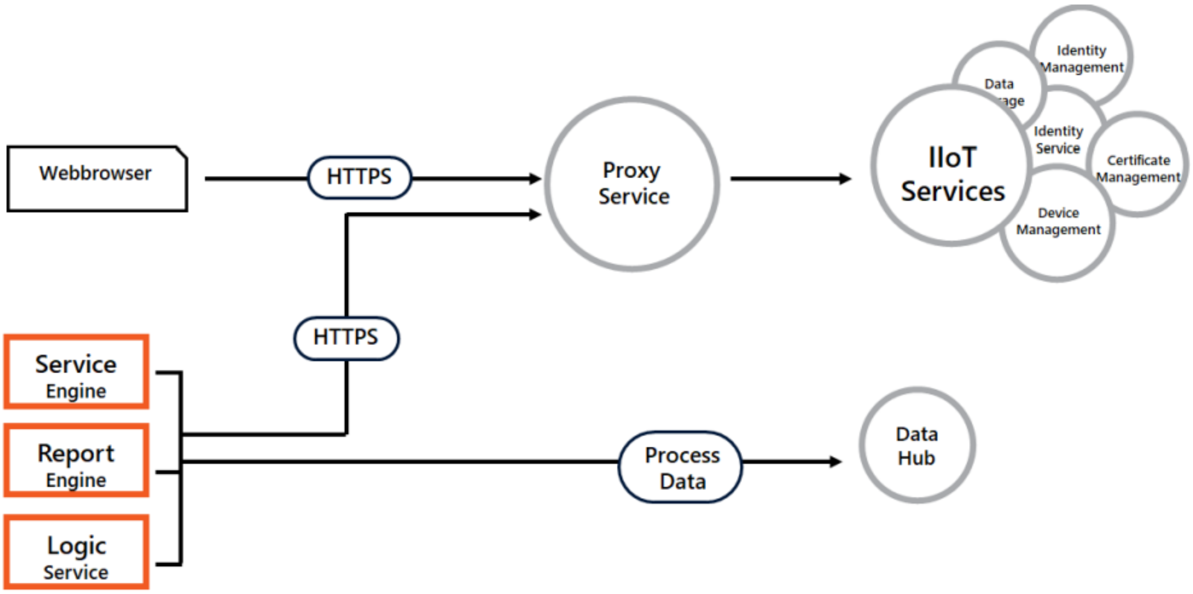
When configuring passwords, a difference must be made between two cases:

- Configuration using GUI:
  Parameterization of a password in Service Configuration Studio is validated during configuration. You are only able to configure suitable passwords here.
- Configuration using the configuration file:
  It is technically possible to configure unsuitable passwords in the *.env* file. This leads to authentication problems between services.

Important: Only use suitable passwords that fully meet the minimum password requirements.

3.5 Communication - Proxy Service

From zenon version 11.2, communication with the IIoT Services is carried out via a central Proxy Service.



The following is applicable for communication via Proxy Service:

- The Proxy Service is the central starting point for overall HTTP communication with the IIoT Services.

- As a result, only a central URL for the connection to the IIoT Services still needs to be configured.
  Note: Ensure that the port for the Proxy Service is not blocked by firewall rules. The forwarding to the individual services of the IIoT Services is carried out via the Proxy Service.
- If the central IIoT Services URL is entered in a web browser, Service Configuration Studio is opened.
- The Data Hub communicates process data. The protocol used is therefore not HTTP traffic and therefore does not use the Proxy Service.
  Note: Ensure that the port for the Data Hub is not blocked by firewall rules.

3.5.1 IIoT Services communication ports

IIoT Services service	Description	Default port
Proxy Service	The central entry point to the IIoT Services for HTTP traffic. The following IIoT Services are supported by the Proxy Service: - Service Configuration Studio - **Identity Service** - **Identity Management** - **Certificate Management** - **Data Storage** - **Platform Configuration** - **IIoT API** - **Device Management** HTTP-based communication is forwarded to the individual services internally. You can find a detailed list of the individual services and their internal ports in the **Services, Ports and URLs** (on page 20) chapter.	*9443*
Data Hub	Central data transfer hub that receives information from services and distributes it to other services. The	*9411*

IIoT Services service	Description	Default port
	communication is event-based.	

3.5.2 Services, ports and URLs

This table offers you an overview of all services and ports in the IIoT Services. This information applies both to when the IIoT Services runs in a Docker environment and to the IIoT Services on Windows.

Docker container or Windows Service	URL* (default port)	Description	Default status
proxy-service (Proxy Service)	*https: //[mycomputer.mydomain.com]:9443* Port *9443*.	The central starting point in IIoT Services services for HTTP traffic.	*running*
data-hub (Data Hub)	No configuration in Service Configuration Studio. Port *9411*.	Central data transfer hub that receives information from services and distributes it to other services. The communication is event-based.	*running*
data-storage (Data Storage)	No configuration in Service Configuration Studio. Port *9460*.	Service for saving data.	*running*
certificate-management (Certtificate Management)	*https: //[mycomputer.mydomain.com]:9410*	Controls access of services to the **Data Hub**.	*running*
identity-service (Identity Service)	*https: //[mycomputer.mydomain.com]:9430*	Central authentication. Verifies the login to the IIoT Services.	*running*
identity-management (Identity Management)	*https: //[mycomputer.mydomain.com]:9431*	Provides the configuration interface for the **Identity Service**.	*running*
data-modelling (Data Modelling)	*https: //[mycomputer.mydomain.com]:9440*	Administration and storage of data in the zenon software platform. Your own data models can also be created and administered.	*running*

Docker container or Windows Service	URL* (default port)	Description	Default status
device-management (Device Management)	*https: //[mycomputer.mydomain.com]:9415*	Deploys Service Engine files for configured devices.	*running*
platform-configuration (Platform Configuration)	*https: //[mycomputer.mydomain.com]:9470*	Provides support for configurations in the IIoT Services	*running*
iiot-api (IIoT API)	*https: //[mycomputer.mydomain.com]:9400*	Offers an API for data access of external systems and external applications to the zenon software platform.	*running*
persistence-service (Persistence Service)	No configuration in Service Configuration Studio. No port.	Database that is used by the various services for storing data. Runs in the background.	*running*
iiot-services-redis (---)	No configuration in Service Configuration Studio. Port *6379*.	Internal service for communication.	*running*
service-configuration-studio (Service Configuration Studio)	*https: //[mycomputer.mydomain.com]:9450* **Note:** From version 11, Service Configuration Studio can also be called up with *localhost:9450* if the IIoT Services are running in a Docker environment.	Central administration interface of IIoT Services.	*running*

*Replace *[mycomputer.mydomain.com]* with the computer host name.

Tip

You can also define ports for services yourself using advanced configurations (on page 219).

3.5.3 Addresses and URLS in Service Configuration Studio

The administration of IIoT Services is set up and configured in Service Configuration Studio. The following services are important for the use of the IIoT Services:

Web interface	Description	Login
Service Configuration Studio	Central administration interface for all user accounts and further services of the IIoT Services. Contains all web interfaces except **Identity Service**.	Only users with the *Administrator* user role can log in.
Identity Service	**Identity Service** is the central login service for users and clients. Every user can configure basic settings for their own user account in Service Configuration Studio.	All users with a user account in **Identity Service** can log in.

3.5.4 Configurable services in the Service Configuration Studio

A number of IIoT services have been integrated into Service Configuration Studio. The configuration interfaces of services can also be called up individually.

The following services are integrated in Service Configuration Studio:

Service	Description	Login	Service Configuration Studio URLs
Service Configuration Studio	Central administration interface for all user accounts and further services of the IIoT Services.	Only administrators of the respective services can work actively in Service Configuration Studio. In principle, the following applies: All services are displayed in Service Configuration Studio. However, only the nodes for which administrator rights are present due to login can be configured.	*https://[IIoT Services URL]:9443/* **Example:** *https://mydomain.local:9443/*

Service	Description	Login	Service Configuration Studio URLs
Identity Service	Central login service for users and clients. You can find further information in relation to this in the **Identity Service** (on page 102) section.	Every user can make basic settings for their own user account in the web interface.	*https://[IIoT Services URL]:9443/identity* **Example:** *https://mydomain.local:9443/identity*
Certificate Management	Administers Certificate Bundles that secure communication for the IIoT Services. You can find further information in relation to this in the **Certificate Managment** (on page 63) section.	Only users with the *Certificate Administrator* user role can use this interface.	*https://[IIoT Services URL]:9443/certificate-management* **Example:** *https://mydomain.local:9443/certificate-management*

Service	Description	Login	Service Configuration Studio URLs
Identity Management	Comprehensive administration of users, clients and connected applications. You can find further information in relation to this in the **Identity Management** (on page 126) section.	Only users with the *Identity Administrator* user role can use this interface.	*https://[IIoT Services URL]:9443/identity-management* **Example:** *https://mydomain.local:9443/identity-management*

Service	Description	Login	Service Configuration Studio URLs
IIoT API	You can connect any number of third-party applications to IIoT Services via the REST API. Service Configuration Studio offers a test environment and the documentation of the API. You can find further information in relation to this in the **IIoT API** (on page 211) section.	Configuration of this service in Service Configuration Studio is possible without login. - If a user wants to retrieve the data manually via the API, the user must authorize themselves in Service Configuration Studio by clicking a button. - The user requires access permissions for the zenon project of the data the user is retrieving.	*https://[IIoT Services URL]:9443/iiot-api* **Example:** *https://mydomain.local:9443/iiot-api*

Service	Description	Login	Service Configuration Studio URLs
Platform Configuration	Supports, for initial configurations of the IIoT Services: Creation of initial user account for **Identity Service** Configuration of the HTTPS certificates to be used. You can find further information in relation to this in the **Plattform Configuration** (on page 180) section.	In the course of setup, a user will be created for login.	*https://[IIoT Services URL]:9443/platform-configuration* **Example:** *https://mydomain.local:9443/platform-configuration*
Device Management	Deployment of zenon projects on devices. Provides project data as software packages for Service Engine on the devices. You can find further information in relation to this in the **Device Management** (on page 188) section.	Only users with the *Device Management Administrator* user role can use this interface.	*https://[IIoT Services URL]:9443/device-management* **Example:** *https://mydomain.local:9443/device-management*

Service	Description	Login	Service Configuration Studio URLs
Data Storage	Central storage space for alarm data, archived data and event data from one or more Service Engine. You can find further information in relation to this in the **Data Storage** (on page 76) section.		*https://[IIoT Services URL]:9443/data-storage* **Example:** *https://mydomain.local:9443/data-storage*
Data Modelling	Service for the central administration and storage of data in the zenon software platform. Your own data models can also be created and administered. You can find further information in relation to this in the **Data Modelling** (on page 86) section.		*https://[IIoT Services URL]:9443/data-modelling* **Example:** *https://mydomain.local:9443/data-modelling*

Hint

You can make advanced settings for the IIoT Services via the configuration files. However, this is only necessary in exceptional cases.

3.5.5 Monitor and restart services

Where you can monitor and restart the services of the IIoT Services depends on the installation option selected for the IIoT Services.

DOCKER

In Docker, each service of the IIoT Services runs **as a separate container**. This applies for both the **Docker on Windows** and **Docker on Linux** installation options.

In both cases, you can stop and restart containers using the command line interface (**CLI**).

Which command line application you use depends on the operating system:

- Windows: **PowerShell**
- Linux: **Bash**

You required to open the command line with administrator privileges.

Description	Command
List all running containers (without stopped ones)	docker ps
List all available containers (including stopped ones)	docker ps -a
Stop a particular container	docker stop <container-name or container ID>
Stop all running Docker containers	docker stop $(docker ps -a -q)
Start a particular container (the container must be stopped)	docker start <container-name or container ID>
Restart a particular container	docker restart <container-name or container ID>

DOCKER ON WINDOWS

In **Docker Desktop for Windows**, the **Dashboard** also offers you a graphical user interface for managing containers. Here, you can visually monitor the status of individual containers and restart them with a mouse click.

WINDOWS-NATIVE

In this installation option, the services of the IIoT Services run **as Windows services**. You can use the **Services** service manager to monitor the services and restart them, if necessary.

Info

Automatic restart of services:

Services of the IIoT Services automatically restart under certain circumstances.

A distinction needs to be made here between two different cases:

- During the initial start, it can happen that due to delays, individual services restart themselves. This is normal system behavior as long as not all services completely restart.
- If an already installed service restarts itself during ongoing operation, this can be a sign of malfunction. In this case, the LOG messages contain further information.

This applies for all installation options of the IIoT Services.

3.6 Certificates

The entire communication between IIoT Services and clients is encrypted with certificates for security reasons.

Clients for the IIoT Services are, for example:

- Internal IIoT Services services
- zenon applications (e.g. Service Engine)
- Third-party applications that are connected via the IIoT API
- Browser access to the web interfaces of the IIoT Services

Each client connection to the IIoT Services must be encrypted. It is not possible to establish an unencrypted connection.

The IIoT Services need the following certificates:

- For HTTPS connections: a central HTTPS Server Certificate
- For Certificate Bundles (CB): each CB has an individual Client Certificate

You have the option to integrate your own HTTPS certificates.

3.6.1 Terminology

The following attributes and names are used for certificates in the IIoT Services:

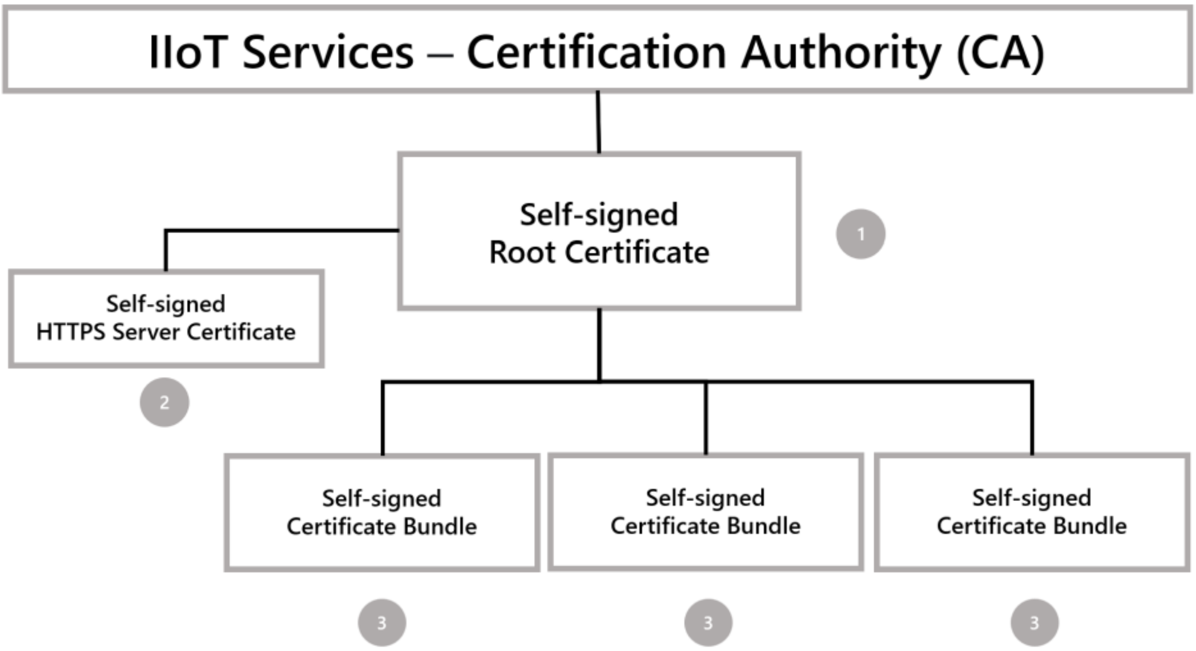
Term	Definition
(Digital) Certificate	All connections are encrypted with digital certificates in the IIoT Services. Certificates are issued and singed by a Certification Authority (CA). The complete key pair for a certificate consists of a Private Key and a Public Key.
Third-party Certificate	Certificate that has not been created and signed by IIoT Services.
Certificate Expiration	Certificates are issued with a defined period of validity. After the period of validity expires, the certificate can no longer be used to establish an encrypted connection.
Certificate Holder	The holder of a certificate. The holder is the only one who may have access to the Private Key.
Certification Authority (CA)	A certification authority creates and signs certificates. The term is often used synonymously with Trusted Third-party Certification Authority.
CA-signed Certificate	A certificate that has been issued and signed by a Certification Authority (CA). The term is often used synonymously for certificates that have been created by a Trusted Third-party Certification Authority.
HTTPS (Server) Certificate	The HTTPS server uses this central certificate to encrypt connections to HTTPS clients.
Private Key	The private part of the key pair for a digital certificate. The Private Key must provide effective protection against unauthorized access. Only the Certificate Holder may have access to the Private Key. If unauthorized persons gain possession of the Private Key, the certificate is compromised and must be replaced.
Public Key	The public part of the key pair. The Public Key can be published without any restrictions.
Root Certificate	Other certificates are derived from this certificate. A trust relationship to IIoT Services Root Certificate applies for all certificates derived from it.
(IIoT Services) Self-signed Certificate	Certificate that has been issued and signed by the IIoT Services. No client has a trust relationship to this certificate by default. The trust relationships must be configured manually for each client.
Trust	The basis of an encrypted connection is a trust relationship between the

Term	Definition
	client and the certificate of the server. The client must be able to verify the issuer of the certificate and trust the issuer.
Trusted Third-party Certificate	Certificate that has been issued and signed by a Trusted Third-party Certification Authority. All clients with common operating systems have a trust relationship to this certificate by default.
Trusted Third-party Certification Authority	A Certification Authority (CA) whose Root Certificate is already preinstalled in common operating systems.

3.6.2 Certificate hierarchy

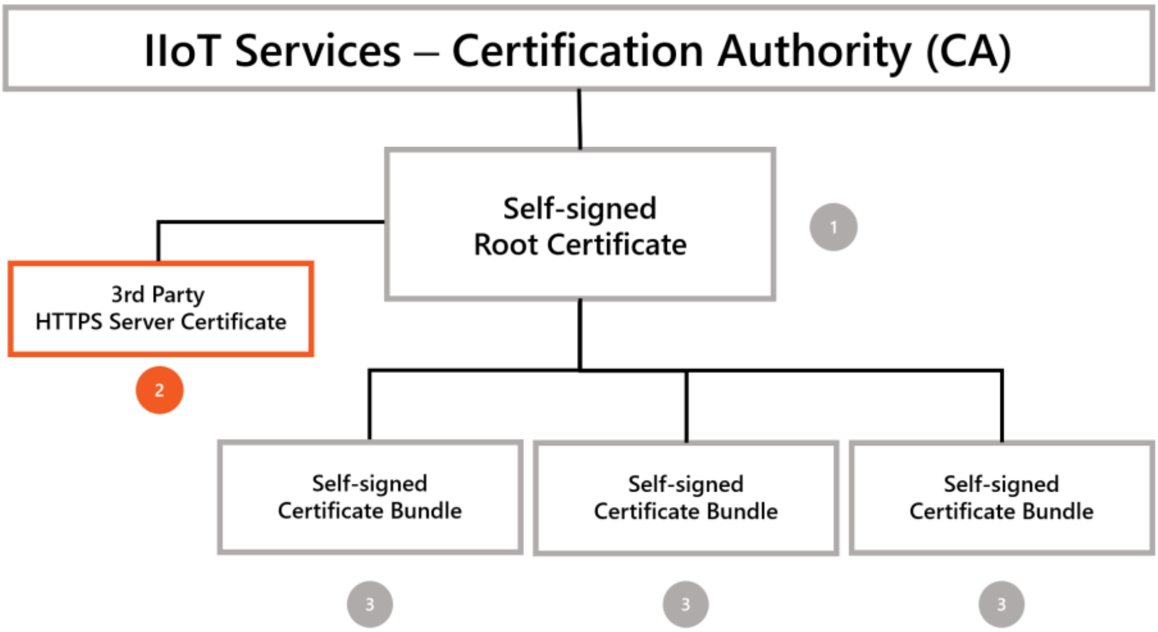
The IIoT Services have their own Certification Authority (CA) integrated. It is independent of third parties to secure the communication. The certificate hierarchy depends on the configuration selected.

DEFAULT: SELF-SIGNED CERTIFICATES FOR HTTPS AND CB


IIoT Services – Certification Authority (CA)
Self-signed Root Certificate
1
Self-signed HTTPS Server Certificate
2
Self-signed Certificate Bundle
Self-signed Certificate Bundle
Self-signed Certificate Bundle
3
3
3

IIoT Services create all required certificates themselves by default. Both the "HTTPS Server Certificate" (2) and the "Certificate Bundles" (3) are derived from the same Root Certificate (1).

OPTIONAL: THIRD-PARTY CERTIFICATE FOR HTTPS AND SELF-SIGNED CERTIFICATES FOR CB


IIoT Services – Certification Authority (CA)
Self-signed Root Certificate
1
3rd Party HTTPS Server Certificate
2
Self-signed Certificate Bundle
Self-signed Certificate Bundle
Self-signed Certificate Bundle
3
3
3

You replace the preconfigured "Self-signed HTTPS Server Certificate" with an imported "Third-party HTTPS Server Certificate" (2). The other certificates (1,3) remain unchanged.

3.6.3 Certificate files

The IIoT Services use the following certificate files:

	IIoTServices.pfx	Imported certificate file (*.pfx or *.p12)	ca.crt	IIoTServices.crt
Label	IIoT Services Self-signed HTTPS Server Certificate	Third-party HTTPS Server Certificate	IIoT Services Self-signed Root Certificate	Option A) IIoT Services Self-signed HTTPS Server Certificate (by default) Option B) Third-party HTTPS Server Certificate (optional)
Content	Public Key Private Key	Public Key Private Key	Public Key	Public Key
Certification Authority (CA)	IIoT Services Root CA - [FQDN]	Third-party CA	IIoT Services Root CA - [FQDN]	Option A) IIoT Services Root CA - [FQDN] Option B) Third-party CA
Period of validity	*5 years* Renewal via **Platform Configuration**.	Period of validity as defined by Third Party Certification Authority. Renewal via Third Party Certification Authority.	*30 years* No renewal possible.	Option A) *5 years*. Renewal via **Platform Configuration**. Option B) Period of validity as defined by Third Party

	IIoTServices.pfx	Imported certificate file (*.pfx or *.p12)	ca.crt	IIoTServices.crt
				Certification Authority. Renewal via Third Party Certification Authority.
GUI	Download in Service Configuration Studio: **Service Configuration Studio\Platform Configuration\Certificates** **<u>Important</u>**: This certificate file can only be downloaded directly after the certificate has been created. It cannot be downloaded at a later point in time.	Import in Service Configuration Studio: **Service Configuration Studio\Platform Configuration\Certificates**	Download in Service Configuration Studio: **Service Configuration Studio\Certificate Management\Certificates**	Download in Service Configuration Studio: **Service Configuration Studio\Platform Configuration\Certificates**
Example:	You are analyzing data from the IIoT Services in a test environment using a web-based third-party application.	You replace the preconfigured IIoT Services Self-signed HTTPS Certificate with the Third-party HTTPS Certificate in a productive environment.	Configuration of a trust relationship between client computers and the IIoT Services	Reading out the entire specification of the HTTPS Server Certificate used by the IIoT Services.

	IIoTServices.pfx	Imported certificate file (*.pfx or *.p12)	ca.crt	IIoTServices.crt
	The web interface of the third-party application is to be secured using the same "IIoT Services Self-signed HTTPS Certificate".	This ensures that IIoT Services HTTPS communication is encrypted with the Third-party HTTPS Certificate.	Self-signed HTTPS Certificate in a test environment.	
Example:Configuration	In the third-party application, configure the *IIoTServices.pfx* file as the certificate to be used.	Import the certificate file into the IIoT Services.	Import ca.crt onto the client computer in the certificate manager of the operating system.	Open the properties of the certificate. Just double-click on *IIoTServices.crt* under Windows.

3.6.4 HTTPS certificate options

In the IIoT Services, you can choose between the following certificates for the HTTPS certificate:

- IIoT Services Self-signed HTTPS Certificate (by default)
- Third-party HTTPS Certificate (optional)

There are different options for third-party certificates:

- Trusted Third-party HTTPS Certificates: These certificates are created by a recognized external certification authority. These are certification authorities such as Let's Encrypt, GoDaddy or VeriSign.
- Third-party Certificates: These certificates are created by an unspecified Certification Authority outside of the IIoT Services. This is typically the case if you are operating your own certificate infrastructure in your company.

It is possible to operate your own certificate infrastructure in many different forms and requires that the administrator possess the appropriate knowledge. The IIoT Services Help therefore only documents the use of Trusted Third-party HTTPS Certificates.

There are basic differences between the certificates:

	IIoT Services Self-signed HTTPS certificate	Trusted third-party HTTPS certificate
Certification Authority (CA)	IIoT Services Root CA - [FQDN]	Trusted Third-party Certification Authority
Configuration of the certificate	This certificate is created automatically during the installation of IIoT Services. No manual configurations are required for the IIoT Services.	This certificate is created by an external Certification Authority – i.e. outside of IIoT Services. These are typically well-known certification authorities such as Let's Encrypt, GoDaddy or VeriSign. The administrator manually imports the certificate file into the IIoT Services.
Client's trust relationship	Must be configured by the administrator manually or with group policies for each client. Reason: The IIoT Services Root Certificate is not preinstalled in the operating system.	Is provided automatically. Reason: The Root Certificates of these certificate authorities are preinstalled in common operating systems.

	IIoT Services Self-signed HTTPS certificate	Trusted third-party HTTPS certificate
Certificate renewal	The administrator can create a new HTTPS Certificate in Service Configuration Studio.	Same as the initial configuration.
Recommendation	This certificate is only recommended for small-scale test environments. Reason: The trust relationship to the certificate must be set up separately for each client. This requires a lot of time and effort in environments with many clients and/or a heterogeneous system landscape.	This certificate is recommended for test environments and productive environments of all sizes. This includes heterogeneous system landscapes with different client operating systems.

3.6.5 Trust relationships

All clients require trust relationships for connections to the IIoT Services.

IIoT Services support two connection types:

- Connections with Certificate Bundle (CB)
- Connections with HTTPS

The trust relationship must be configured separately for each connection type.

The following applies for connections with Certificate Bundle:

- Only certain zenon applications and services require a CB.
- Each client requires its own Certificate Bundle with an individual Certificate.
- Each Certificate is based on the IIoT Services Self-signed Root Certificate.

Certificate Bundles are configured using the **IIoT Services Connection Wizard**.

The following applies for connections with HTTPS:

- All IIoT Services clients can establish a HTTPS connection
  Important: This also applies to zenon applications and services that connect with CB.
- Each HTTPS client requires a trust relationship to the HTTPS Certificate used.
- IIoT Services use a central HTTPS Server Certificate.
- A IIoT Services Self-signed HTTPS Certificate is preinstalled by default.
- The preinstalled certificate can be replaced optionally by a Third-party HTTPS Certificate.

How you must configure the trust relationship depends on the HTTPS Certificate used.

3.6.5.1 IIoT Services Connection Wizard

From version 12, the **Service Node Configuration Tool** is no longer supported and is also not included in the setup:

- The connections of zenon components are configured with the **IIoT Services Connection Wizard**.

> **Information**
>
> The **Service Node Configuration Tool (SNCT)** was used in previous versions of zenon up to and including 11.2 for the generation of Certificate Bundles (CB) for clients.

3.6.5.2 HTTPS trust relationship

It is only possible to establish an HTTPS connection if there is a trust relationship between the client and the HTTPS Certificate of the IIoT Services.

HTTPS CLIENTS

Each client connected to the IIoT Services requires a trust relationship with the HTTPS Certificate.

You must set up trust relationships, for example, for the following client computers:

- Clients that access Service Configuration Studio with a browser
- Clients that access the IIoT API with a Third-party Application
- zenon components and applications that connect with the IIoT Services
- The host computer on which the IIoT Services are installed.

Frequently-used HTTPS certificates by the IIoT Services are:

- IIoT Services Self-signed HTTPS Server Certificate
- Trusted Third-party HTTPS Server Certificate

The configuration of the trust relationship depends on the certificate selected.

3.6.5.2.1Configure trust relationship

The configuration of a trust relationship depends on the certificate selected.

SELF-SIGNED HTTPS SERVER CERTIFICATE

You must configure trust relationships for all client computers.

To configure a client:

1. Load the ca.crt certificate file for the IIoT Services Self-signed Root Certificate from this web interface:
   Service Configuration Studio\Certificate Management\Certificates
2. Import the certificate file manually into the certificate store of the respective operating system. In the Windows operating system, you must, for example, import the certificate file into the *Trusted root Certification Authorities* folder via the certificate store.
   Note: You can automate this procedure by using group policies.

You have thus configured the trust relationship.

TRUSTED THIRD-PARTY HTTPS SERVER CERTIFICATE

In general, you do not have to manually configure any trust relationship for this certificate.

Reason: Root Certificates for Trusted Third-party Certification Authorities are already preinstalled in common operating systems. Thus a trust relationship exists by default.

3.6.5.2.2 Trust relationship with multiple certificates (Docker only)

There is the possibility of installing additional trust certificates for the IIoT Services. This is necessary, for example, if the services have to communicate with third-party systems that use self-signed certificates. An example of this is communication between the **Identity Service** and **Identity Provider** (e.g. OpenLDAP or Keycloak).

INSTALLATION

In Windows, the IIoT Services automatically use the certificates of the certificate store to verify certificates. In a Docker environment, these additional certificates that have to be trusted must be stored manually in the services.

Carry out the following steps in order to add additional certificates to the trusted certificates of the container:

1. Create a folder and place the certificates there (with the folder name "*certs*" for example).

2. Include the certificates in each Docker service that is to trust the certificate. Carry out this inclusion for each service.

Add a row in the **volumes** section of the respective service in the *docker-compose.yml* file.

   a) Configure the path and the folder in which your certificates are located. This can be an absolute or relative path entered.
   For example: - *./certs/:/certs/*

1. Restart the container.

Example

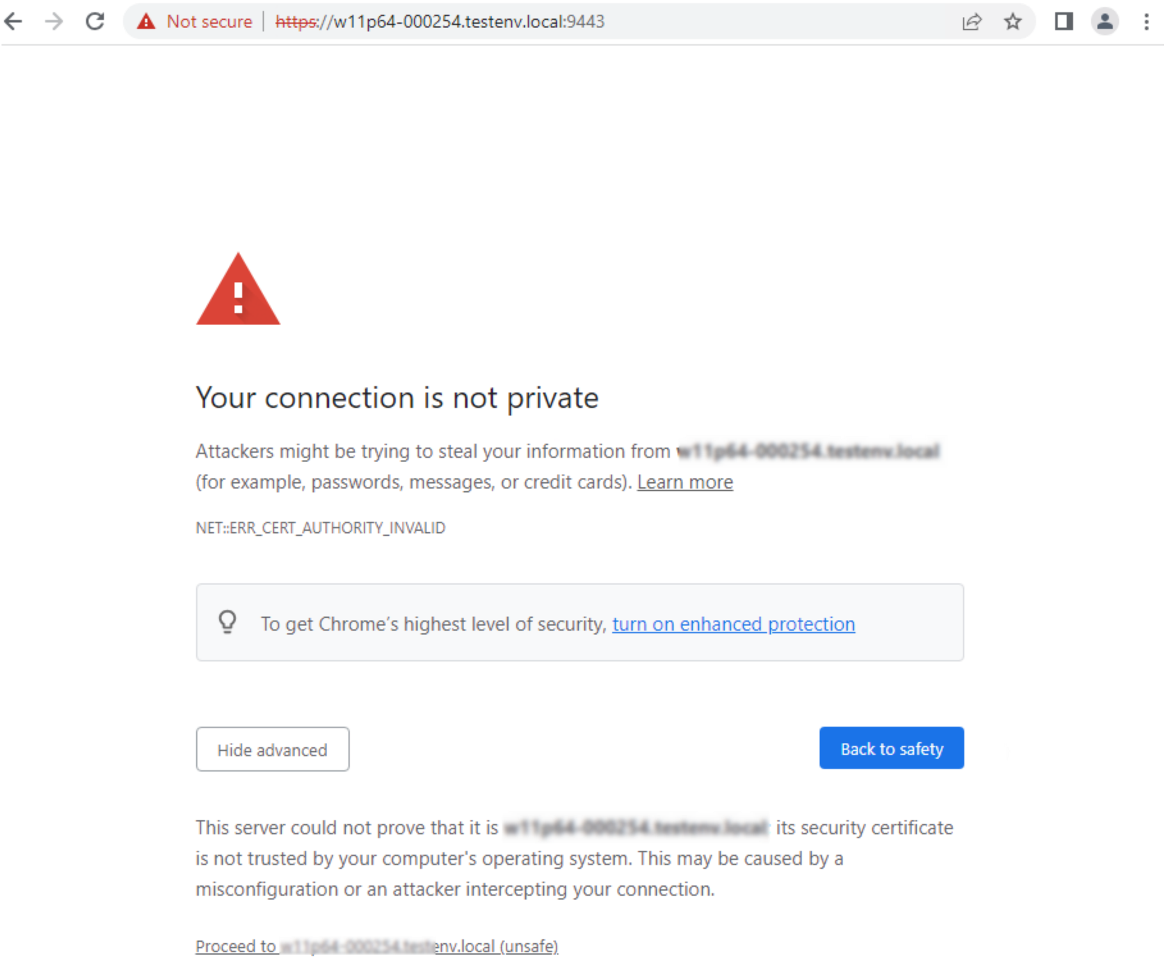
```
identity-service:
 ...
 networks:
 - iiot-services-network
 volumes:
 - iiot-services-data:/var/iiot-services-data/
 - ./certs/:/certs/ <-- add this line
```

The following is applicable here:

- The certificates are loaded each time the container is started.
- The newly-added certificates are loaded each time the container is started.
- Certificates that are removed from the folder are not uninstalled during operation.
- Deletion and recreation of the respective container resets all previously-installed certificates and only then installs the certificates currently present in the folder.

3.6.5.3 HTTPS certificate warnings

You may receive an HTTPS certificate warning when establishing an HTTPS connection between a client and IIoT Services.



Not secure | ~~https~~://w11p64-000254.testenv.local:9443

Your connection is not private

Attackers might be trying to steal your information from [redacted] (for example, passwords, messages, or credit cards). Learn more

NET::ERR_CERT_AUTHORITY_INVALID

To get Chrome's highest level of security, turn on enhanced protection

Hide advanced

Back to safety

This server could not prove that it is [redacted] its security certificate is not trusted by your computer's operating system. This may be caused by a misconfiguration or an attacker intercepting your connection.

Proceed to [redacted]env.local (unsafe)

The following applies for certificate warnings:

- They make the user aware that there is no trust relationship between the client and the HTTPS Certificate of the HTTPS connection.
- You must then check the certificate used by this HTTPS connection.

The certificate and thus the identity of the HTTPS server must be clearly verified. This prevents attacks over the HTTPS connection like Man-in-the-middle attacks.

The following applications issue certificate warnings:

- Browser: When you attempt to open a IIoT Services web interface.
- Engineering Studio during configuration with the **IIoT Services Connection Wizard**.

The user can manually establish the HTTPS connection after a positive check of the certificate warning.

The following applications do not issue certificate warnings:

- Service Engine: When you attempt to connect to IIoT Services via HTTPS.
- Report Engine: When you attempt to connect to IIoT Services via HTTPS.

The user cannot establish these HTTPS connections manually. In these cases, the trust relationship must be configured before establishing the connection.

Note: In the **Diagnosis Viewer**, you can find logs of failed connection attempts of zenon applications.

The following applies for certificate warnings when accessing via the IIoT API:

- User access via Service Configuration Studio: The browser issues a certificate warning.
- Client access to the IIoT API for a third-party application: The third-party application decides here how it handles HTTPS connections. The third-party application can also issue certificate warnings if configured accordingly.

Third-party applications are generally to be configured in such a way that there is a trust relationship to the HTTPS Certificate.

To avoid certificate warnings:

- Make sure that a trust relationship has been established with the HTTPS Certificate of the IIoT Services for all clients before establishing the connection.

You should not receive any more certificate warnings after successful initialization of IIoT Services and the correct configuration of all clients.

3.6.5.4 HTTPS certificate check

The basic principle is: After a HTTPS certificate warning, the HTTPS Certificate must be checked by the user. How the certificate check is actually done depends on the context of the certificate check.

CERTIFICATE WARNINGS DURING INSTALLATION

Certificate warnings are unavoidable during initialization of the IIoT Services. At this stage, you need not and cannot verify the certificate.

CERTIFICATE WARNINGS DURING OPERATION

In a fully configured IIoT Services environment, all required HTTPS trust relationships should be established. Therefore there should not be any HTTPS certificate warnings during operation. Certificate warnings in this context are thus unavoidable and must be checked very carefully.

To check the certificate:

- Preparation: Save the *IIoTServcies.crt* certificate file during the installation of IIoT Services. You have thus made sure that you always have access to the HTTPS Server Certificate configured for IIoT Services even in the case of a compromised HTTPS connection. This certificate should be used for the HTTPS connection.
- When the certificate warning is displayed: extract the certificate shown. This certificate is actually used for the HTTPS connection.
- Compare the two certificates with each other.

Both certificates must match. If this is not the case, the HTTPS connection has been compromised.

3.6.6 Period of validity

The validity of certificates is checked before the connection is established. A connection is then only established if the certificate is valid at the time of connection.

The period of validity of IIoT Services self-signed certificates is:

- Root Certificate (CA): *30 years*
- HTTPS Server Certificate: *5 years*
- Client Certificates for Certificate Bundles (CB): *5 years*

The period of validity of third-party certificates is:

- Freely definable by the respective Certification Authority.

The administrator must continuously monitor the terms of all certificates. Certificates must be replaced in time before the end of the period of validity.

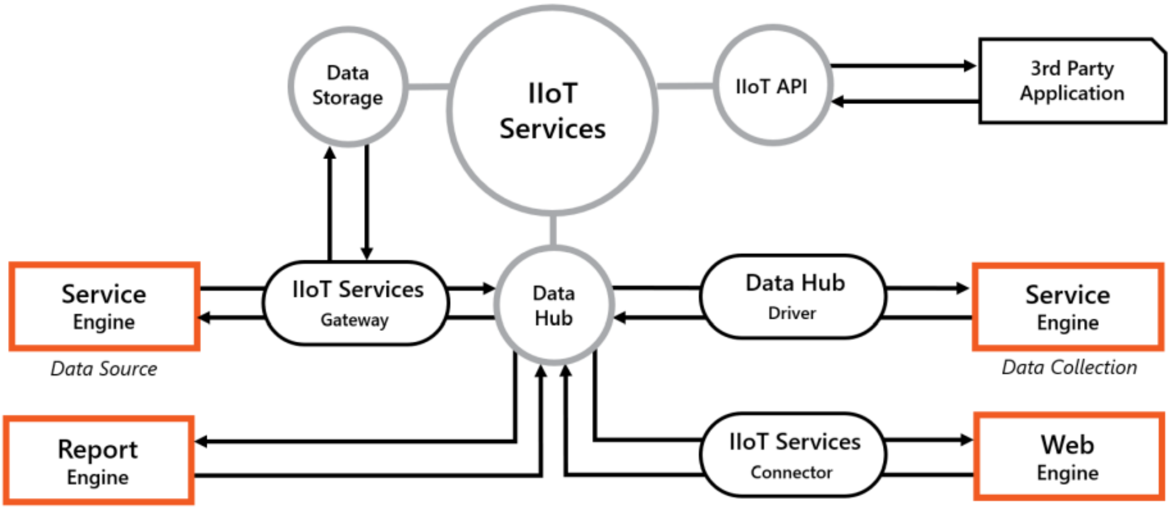
MONITOR THE PERIOD OF VALIDITY

To monitor the periods of validity of certificates:

Certificate	Where the period of validity is displayed
IIoT Services Root Certificate	In the properties of the ca.crt certificate file.
HTTPS Server Certificate	In the properties of the *IIoTServices.crt* certificate file.
Client Certificates for Certificate Bundles (CB)	**Service Configuration Studio web interface:** - Service Configuration Studio\Certificate Management\Certificates Each CB contains an individual Client Certificate

Certificate	Where the period of validity is displayed
	and must therefore be monitored separately. Hint: You can sort the column view by the period of validity of the certificates.

4 Possible use cases



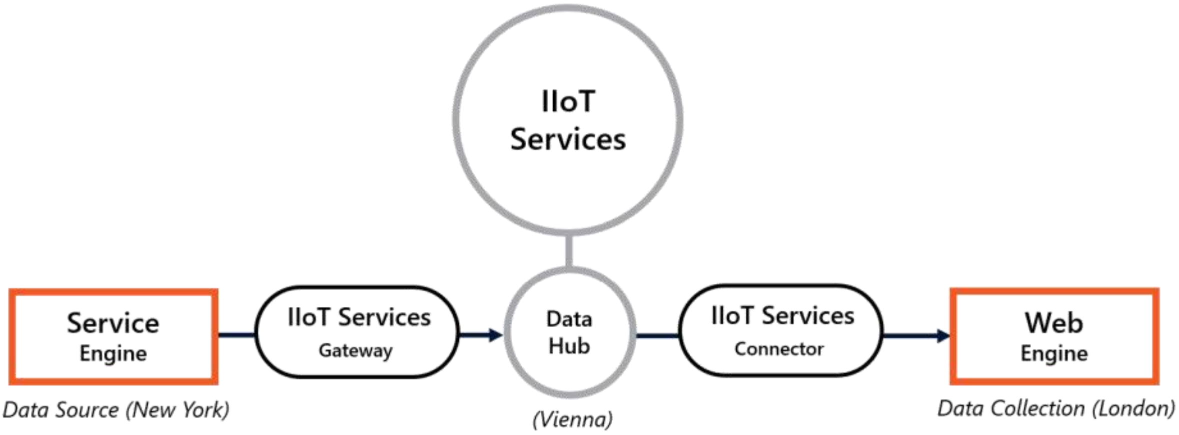
IIoT Services: Networking of zenon applications, zenon services and third-party applications.

This section describes supported use cases for the IIoT Services.

<u>**The IIoT Services can be used to connect the following services to one another:**</u>

- Select zenon applications (Service Engine for example)
- Selected zenon services (**Data Storage** for example)
- Deployment of zenon project data to devices (for example, computers, virtual machines, Raspberries or Linux).
- Third-party applications (via the IIoT API)

4.1 Service Engine > Web Engine: Remote control



IIoT Services allow the Web Engine to access process screens of a Service Engine.

Web Engine provides process screens of a Service Engine for access via browser. Using IIoT Services, the **Web Engine** can be operated at a different site than Service Engine.

AREA OF APPLICATION

You can use Web Engine as:

- Remote visualization of a Service Engine
- Remote control of a Service Engine

SUPPORTED DATA ACTIONS

You can carry out the following data actions in Service Engine by means of the **Web Engine**.

Supported data action	Variable access authorization*
Read alarms	*Read only*
Acknowledge alarms	*Read-write*
Comment on alarms	*Read-write*
Set causes of alarms	*Read-write*
Variables - read**	*Read only*
Variables - write**	*Read-write*
Archive data* - read**	*Read only*

Supported data action	Variable access authorization*
Read events	*Read only*****
Comment on events	*Read and write*

* * required access authorization in Service Engine (data source).

** simple variable type (no structure variables, no arrays).

*** Only as a curve in Extended Trend.

**** No access authorization is required for system events.

4.2 Report Engine > Service Engine: Create data predictions

AREA OF APPLICATION

Data predictions attempt to derive the future development of variables from recorded variable values of the past.

Examples of this are:

- Prediction of consumption figures
  Predictions can be made, for example, on how much electricity will probably be consumed.
- Prediction of material consumption:
  Predictions can be made, for example, about when an inventory will probably be depleted.
- Anomaly detection:
  Identifies unusual deviations between predicted and actual variable values. This can point to defects in production.

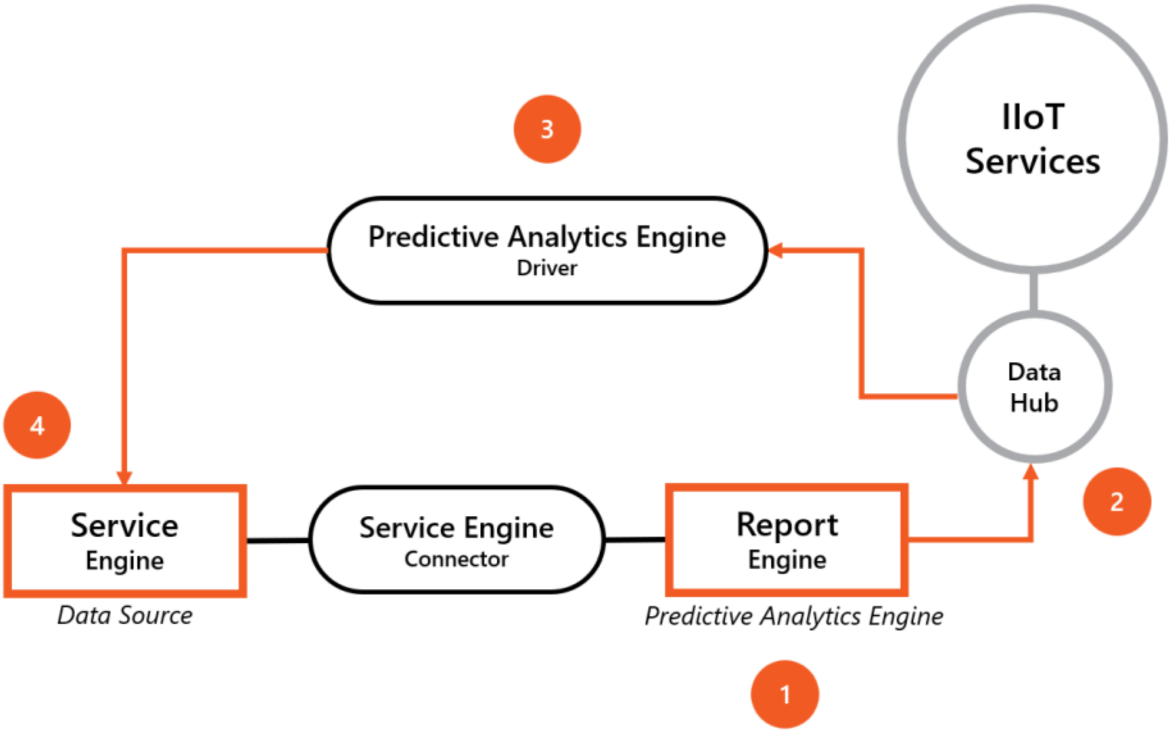
All data predictions are based on historical variable values.

PREDICTION MODEL

This use case requires that a prediction model for data predictions already exists in Report Engine. How you can create a prediction model for your data is documented in detail in the Help for Report Engine.

There are two cases how the Service Engine can be used for data predictions.

4.2.1 Case A) Passive reception of data predictions



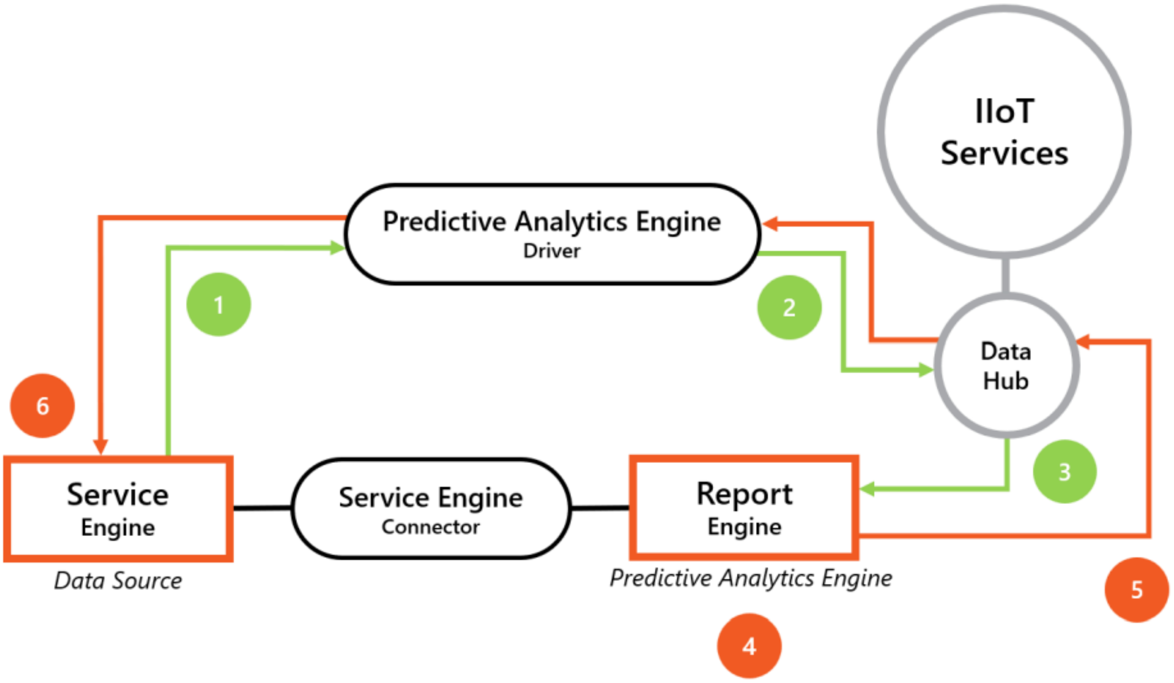
Service Engine is in this case a mere receiver of data predictions.

<u>The procedure is as follows:</u>

1. Report Engine creates a data prediction.
2. The data prediction is transmitted from Report Engine to **Data Hub** .
3. **Predictive Analytics Engine Driver** retrieves the data prediction from the **Data Hub** and forwards it to Service Engine.
4. Service Engine works with the data prediction.

In this case, Report Engine decides when a data prediction is created.

4.2.2 Case B) Active request of data predictions



In this case, Service Engine actively requests a data prediction from Report Engine.

<u>**The procedure is as follows:**</u>

1. Service Engine triggers the creation of a data prediction.
2. This request is sent to **Data Hub** with **Predictive Analytics Engine Driver**.
3. **Data Hub** transmits the prediction request to Report Engine.
4. Report Engine creates a data prediction based on the parameters of the request.
5. The data prediction is sent to Service Engine with **Data Hub** and **Predictive Analytics Engine Driver**.
6. Service Engine works with the data prediction.

In this case, Service Engine decides when a data prediction is created.

5 Installation

The zenon software platform has the following release cycles:

- From version 12:
  Major releases: Annual releases for all zenon applications including IIoT Services and **IIoT Services Gateway** (for example, zenon 12).
  Additions are available via build update. No minor releases (including setup) are offered.
- Up to and including version 11.2:
  Minor releases: Quarterly releases for the IIoT Services and the **IIoT Services Gateway** (for example, IIoT Services version 11.1)

The installation and configuration of the **IIoT Services Gateway** depends on the installed version of zenon.

Information on how to install the IIoT Services can be found in the Help in the Installation and Updates section in the IIoT Services chapter.

5.1 Compatibility

The IIoT Services typically connect zenon applications and third-party applications to one another. It must be ensured that all connected applications are compatible with one another.

The following applies for a IIoT Services installation:

- Install the most recent version of the IIoT Services.
- The IIoT Services version must be at least just as high as the version of the connected zenon applications.
- All connected zenon applications must be compatible with the installed IIoT Services version.

Information

Backward compatibility:

The IIoT Services support components of the zenon software platform from version 10 and higher.

5.1.1 Renaming of Service Grid to IIoT Services

With version 12, the **Service Grid** and its Services and components were renamed to **IIoT Services**. The graphical user interface of the **IIoT Services** and apps (programs) and the help have been amended accordingly.

Previous versions up to and including 11.2	From version 12
Service Grid	**IIoT Services**
Service Grid Hub Controller	**Certificate Management**
Service Grid API	**IIoT API**
Service Grid Persistence	**Persistence Service**
Service Grid Studio	**Service Configuration Studio**
Service Grid Proxy	**Proxy Service**
Service Grid Gateway	**IIoT Services Gateway**
Service Grid Egress Connector	**Data Hub Driver**
Service Grid Ingress Connector	**IIoT Services Gateway** components.

5.1.2 Compatibility of version 12 with previous versions

With version 12, the connection between Engineering Studio project configurations to IIoT Services and communication between the individual services was simplified.

For existing project configurations in Engineering Studio, it is expressly recommended that the connections to the IIoT Services and the configuration for individual services are reconfigured.

You can find detailed information on the configuration in the **IIoT Services - configuration in Engineering Studio** section.

5.1.3 Compatibility of version 11.2 with previous versions

From version 11.2 and higher, the individual services of IIoT Services are addressed using a central URL. The addressing of services was by means of port numbers in previous versions. The default port number for IIoT Services is *9443*. This port number can be amended by means of configuration.

If the central IIoT Services URL is entered in a web browser, Service Configuration Studio starts.

COMPATIBILITY NOTE - IIOT SERVICES 11.2

The following is applicable in order to work seamlessly with a zenon installation with IIoT Services:

- Service Engine or Engineering Studio in version 11 communicate with IIoT Services version 11.2 or higher:
  Build *111398* or higher for Service Engine or Engineering Studio must be installed.
  The following configurations must also be updated:
  - **Service Node Configuration Tool**
    When entering a **Connection** setting, add the */hub-controller* sub-path to the central URL.
    Example: *https://hostname.local/hub-controller*
    Attention: From version 12, the **Service Node Configuration Tool** has been replaced with the **IIoT Services Connection Wizard**.
  - Configuration in Engineering Studio
    It is necessary to specify a subpath for configurations in Engineering Studio. The same URL is used for **Identity Service** and **Data Storage**. Entering the central URL is sufficient. The URL of the corresponding properties must be configured as URL + port.
    Examples:
    URL for **Identity Service** (**Network** property group, **Identity Service**, property: **URL**)
    URL for **Data Storage** (**Network** property group, **Data Storage**, property: **URL**)
    hostname.local:9443

5.1.4 IIoT Services update

You can migrate a IIoT Services installation to a higher version with little effort.

How to perform an update:

1. Back up the existing **Persistence Instance** (on page 232).
2. Stop the IIoT Services.
3. Reinstall the IIoT Services in accordance with the installation option.
   - IIoT Services (Docker): The *.env* file for the new version must be configured with the necessary values. Afterwards, the IIoT Services are initialized with the *docker-compose.yml*.
   - IIoT Services (Windows-native): Carry out the setup of the new version.
4. Restart the IIoT Services.

You have now updated the IIoT Services.

Hint

Existing certificate bundles of an old IIoT Services version are usually compatible with the new version and do not need to be issued again.

The **Persistence Instance** with all configurations of the IIoT Services is migrated automatically.

The backup of the **Persistence Instance** is a security precaution. This means that a restore is possible in the event of an error.

5.1.5 Update MongoDB

From version 12, the update of MongoDB has been made much simpler for the user. MongoDB persistence instances can thus be updated to the most recent version that is used by the IIoT Services. Subsequent change is not possible.

Please also note the **Backup and restore – persistence instance** (on page 232) section in order to back up your data before an update.

UPDATE FOR WINDOWS (NATIVE)

The updating of MongoDB for Windows operating systems is fully integrated into the setup. No additional manual steps are necessary.

UPDATE FOR DOCKER

The following requirements are necessary for updating MongoDB in a Docker environment:

- The tool **CopaData.ServiceGrid.Tools.PersistenceManagementCli.exe** is installed on machines running Docker Desktop for Windows.
- The installation is done by running *PersistenceManagementCli.x64.msi*.
  The data are stored in the following folder:
  %programfiles%\zenon\zenon Platform 12\IIoT Services\PersistenceManagementCli.
- The **MongoDB Command Line Database Tools** (on page 233) are installed.
- The PATH environment variable has been extended with the path to the MongoDB Command Line database tools (see previous step), e.g.:
  C:\Tools\mongodb-database-tools-windows-x86_64-100.7.0\bin
- The current version of the IIoT Services is installed and running.
- The .ENV file with the current settings and the docker-compose .YML file for the new version are available in their own Windows folder.

- Port *27017* is available on the computer for connecting to the MongoDB database.

RUN UPDATE

In the Docker environment, do the following:

1. Open an elevated PowerShell.
2. Navigate to the storage location of the CLI, e.g. (default path): *%programfiles%\zenon\zenon Platform 12\IIoT Services\PersistenceManagementCli*.
3. Enter the following command:
   CopaData.ServiceGrid.Tools.PersistenceManagementCli.exe docker upgrade

- The tool starts and guides you through the update process step by step. Necessary parameters are queried. The update process is continued after the necessary parameters are entered. In addition, information and a log are visualized directly in the tool.

5.1.6 Login to the Identity Service after an update or upgrade

After an update or upgrade of the IIoT Services version, it may happen that logging in to the Identity Service in the web browser is no longer possible.

In this case, clear the cache of your web browser. In addition, you can call up the login screen in a private window of the web browser.

5.1.7 IIoT Services Gateway

The **IIoT Services Gateway** connects IIoT Services to zenon applications. It also ensures compatibility between different release versions.

The following is applicable for third-party applications:

The compatibility of the IIoT API with previous versions generally remains unchanged. The API is constantly being expanded. Third-party applications that have already been implemented are compatible with subsequent versions of the API. Thus it is not strictly necessary to adapt third-party applications for a new IIoT Services release.

Information

Generally speaking, you should always use the version of **IIoT Services Gateway** that corresponds to the installed version of the IIoT Services.

5.1.7.1 Installation

For the connection between zenon applications and IIoT Services, the appropriate version of the **IIoT Services Gateway** must be selected and installed.

The following applications use the **IIoT Services Gateway**:

- Service Engine
- Engineering Studio
- Report Engine
- Reporting Studio

Important: You must always execute both installers (x86 and x64) on each client. This way, you ensure that these clients can connect to the IIoT Services.

> 👍 **Tip**
>
> Check the installation:
>
> Under **Apps and features**, the Windows operating system shows a separate entry for each installed version of the **IIoT Services Gateway**.

5.1.7.2 Configuration

PRIOR CONSIDERATIONS

Several versions of the **IIoT Services Gateway** can be installed on a computer at the same time. The system cannot use these versions at the same time however. Only one version of the **IIoT Services Gateway** can ever be centrally configured and used.

These processes can install a **IIoT Services Gateway**:

- Installation of zenon applications via the platform setup.
- Build update of installed zenon applications.
- Installation of the **IIoT Services Gateway** via two separate installers (x64 and x86).

In practice, several versions of the **IIoT Services Gateway** are typically installed on a computer at the same time.

> ### Hint
>
> Use several versions alternately:
>
> You can use several versions of the **IIoT Services Gateway** alternately on one computer. For each change, you must configure the respective required version of the **IIoT Services Gateway** manually in *zenon6.ini*.

DEFAULT CONFIGURATION

By default, zenon applications always connect to IIoT Services via the most-recently-installed version of the **IIoT Services Gateway**.

The default configuration covers the usual application purposes and therefore does not generally need to be amended manually by the user.

MANUAL CONFIGURATION

Manual configuration of the **IIoT Services Gateway** by the user is only required in a few cases.

You can use manual configuration to stipulate to the system which version of the **IIoT Services Gateway** zenon applications connect to the IIoT Services.

General requirements:

- Several versions of the **IIoT Services Gateway** are installed on the computer.

Reasons for manual configuration:

- The version of zenon used does not support the most recent version of the **IIoT Services Gateway**.
- The connection to a IIoT Services must be established in an older version than the highest installed version of the **IIoT Services Gateway**.

Manual configuration can, for example, be necessary in the following cases:

- Subsequent downgrade of a zenon installation
- Parallel installations of different zenon versions on one computer
- Connection from zenon applications to different versions of the IIoT Services

👍 Hint

Restart applications and services:

You must restart the following after manual configuration of the **IIoT Services Gateway**:

- All zenon applications connected to the IIoT Services
- The Windows service for the Report Engine service node (if you are using Report Engine)

The new configuration is only effective after restarting these applications and services.

5.1.7.2.1 Configuration in zenon6.ini

The **IIoT Services Gateway** is configured centrally in the *%cd_system%\zenon6.ini* file. This setting is applicable for all zenon applications installed on the computer.

The default configuration is as follows:

[ServiceGridGateway]

Version=LAST

In this configuration, zenon applications connect to the most recent version of the **IIoT Services Gateway** that is installed on the computer.

Example of configuration for version 11:

[ServiceGridGateway]

Version=11_0

In this example, zenon applications connect to **IIoT Services Gateway 11**.

SYNTAX

The syntax for manual configuration of the version is "MM_N". The first two figures "MM" define the version number of the major release. The last figure "N" defines the minor release.

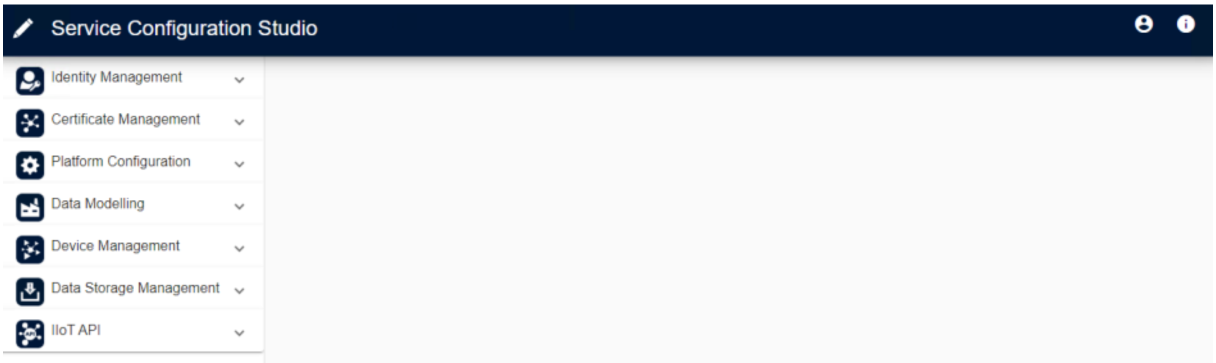
Hint

Configure the figure for the minor release:

The last figure must always be given, including for major releases, such as **IIoT Services Gateway** 11 for example. In this case, you must configure the value "*11_0*".

6 Service Configuration Studio

The Service Configuration Studio is the central administration interface of the IIoT Services.



Valid for the Service Configuration Studio:

- The configuration interface is called up by entering a URL.
  - All services can be called up in a configuration interface. In this case, navigation contains each service as a main node.
  - Each service can be called up as a separate configuration. In this case, the navigation contains only the main node and the individual service subnodes.
- The user must register before using the Service Configuration Studio.
  - Login is via **Identity Service** (on page 102).
  - The logged-in user must have the appropriate authorization to configure the particular service.
- Only services can be configured, for which the logged-in user has the appropriate authorizations.

Information

You can find an overview of the URLs from individual configuration web pages in the **Addresses and URLs in the Service Configuration Studio** (on page 22) section.

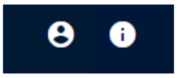
6.1 User interface

The Service Configuration Studio user interface consists of different areas. View and structure of the configuration depend on the service.

Element	Description
Header (on page 62)	Toolbar with user-specific actions.
Main nodes	Tree view of the IIoT Services with configuration options for each service. The subnodes subordinate to the main nodes can be displayed as expanded or collapsed with a mouse click. Individual nodes are expanded or collapsed by clicking on the node. ▶ Arrow down: Expands the node view ▶ Arrow up: Reduces the node view The display of the main node depends on the width of the browser window. ▶ If this window is wide enough, a symbol and the service name or the name of the subnode will be displayed for each service. ▶ If the window width is very small, only the service or its subnodes will be displayed.
Show/hide main nodes	The width of the main node is adjusted by clicking on the button: ▶ <: Reduces the display of the node and its subnodes to symbol view ▶ >: Expands the display of the node and its subnodes to symbol view + node names.

Element	Description
Subnodes	By clicking on the subnode, the configuration options from the particular service is displayed in the configuration area.
Configuration area	Options for configuring the service. Depending on the service, lists from existing configurations will also be displayed in this area. These lists can be sorted and filtered. - By clicking on the existing project configuration content, this can be changed in configuration dialogs. - By clicking on the appropriate buttons, new configuration content can be created and configured in configuration dialogs.
Configuration dialog	Depending on the configuration's content, your own configuration dialogs will be displayed on the right side.

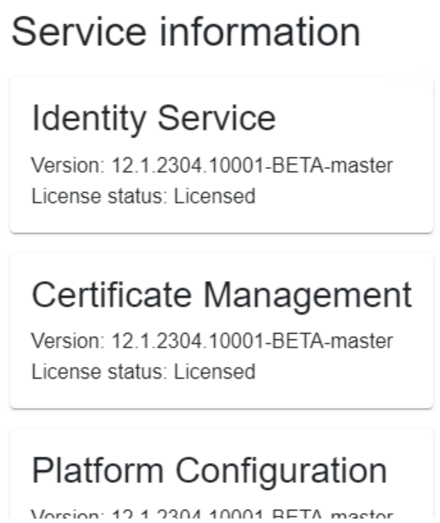
6.1.1 Header



Button	Description
User	User-specific actions: - *User Profile* Shows information about the user who is currently logged in. **Note:** You can find detailed information in the **Identity Service** (on page 102) section. - *Logout* Logs the current user out of Service Configuration Studio. The login dialog is shown in the browser after logout.
Information	Visualizes the licensed system services in a dialog.

DIALOG - SERVICE INFORMATION

THE INFORMATION BUTTON IS VISUALIZED



Service information

Identity Service

Version: 12.1.2304.10001-BETA-master

License status: Licensed

Certificate Management

Version: 12.1.2304.10001-BETA-master

License status: Licensed

Platform Configuration

Version: 12.1.2304.10001-BETA-master

This dialog visualizes information about the individual IIoT Services.

Note: This dialog can be closed with a click.

Option	Description
Name of the service	Name of the installed service.
Version	Version of the installed service.
License status	License status of the installed service.

7 Certificate Management

The **Certificate Management** node contains two components:

- **Certificate Management**: Is responsible for the access permissions of the individual services. It decides which Services get access and passes this information on to the **Data Hub**. In order to allow zenon applications access to **Certificate Management**, the user must use the **IIoT Services Connection Wizard** to connect these to the **Certificate Management** and allow individual credentials to be generated for each Service. This ensures that only authorized Services can provide and consume data.
- **Data Hub**: Is the actual data distributor.
  In coordination with the **Certificate Management**, this decides which Services may connect

with it and send or receive data. The **Data Hub** distributes the notifications and guarantees the delivery of system-critical events to the recipients.

> **Information**
>
> In the case of integration projects, each of the subordinate projects can communicate with another Data Hub. However, several projects can also access the same Data Hub.

DATA HUB

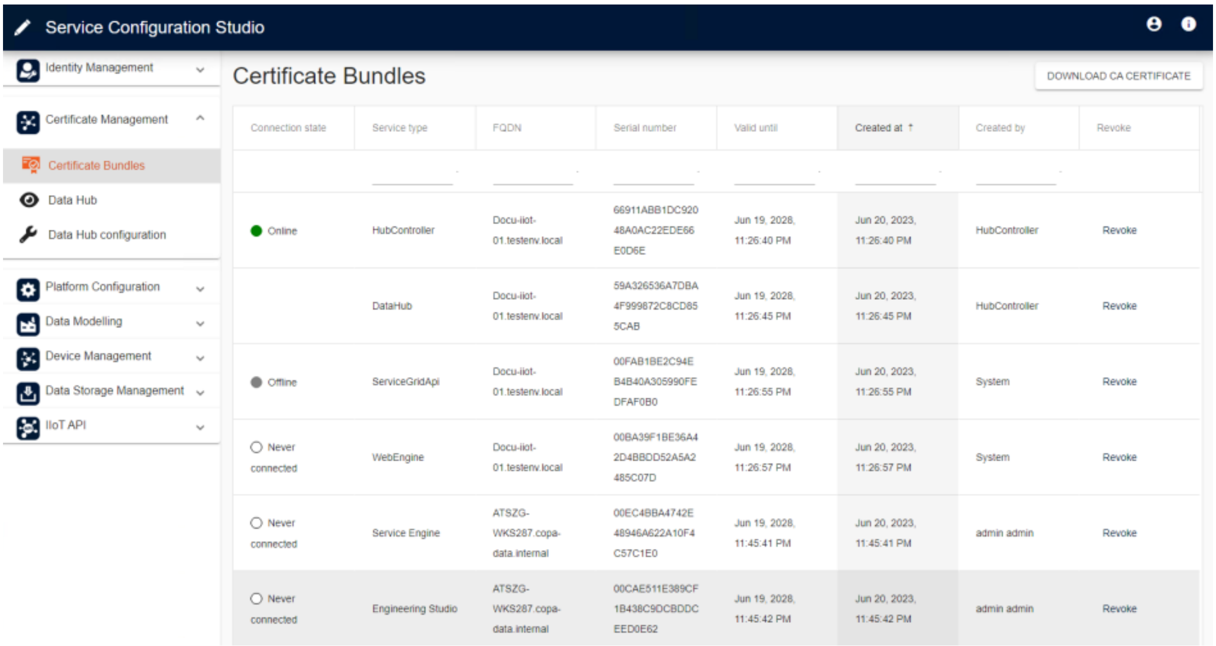
As the central data transfer hub, it receives information of services and distributes it. The **Data Hub** is always installed on the same computer as the **Certificate Management** by default. A separate Certificate Bundle is needed for communication with **Certificate Management**.

Administration depends on the installation option:

- IIoT Services (Windows native): Administration in Service Configuration Studio in the Certificate Management (on page 222) node
- IIoT Services (Docker): Administration with configuration files (on page 219)

With a standard installation, it is not necessary to change settings for the **Data Hub**. IIoT Services configure all required settings by default.

7.1 Certificate Management



Certificate Bundles

Connection state	Service type	FQDN	Serial number	Valid until	Created at ↑	Created by	Revoke
● Online	HubController	Docu-iiot-01.testenv.local	66911ABB1DC92048A0AC22EDE66E0D6E	Jun 19, 2028, 11:26:40 PM	Jun 20, 2023, 11:26:40 PM	HubController	Revoke
	DataHub	Docu-iiot-01.testenv.local	59A326536A7DBA4F999872C8CD855CAB	Jun 19, 2028, 11:26:45 PM	Jun 20, 2023, 11:26:45 PM	HubController	Revoke
● Offline	ServiceGridApi	Docu-iiot-01.testenv.local	00FAB1BE2C94EB4B40A305990FEDFAF0B0	Jun 19, 2028, 11:26:55 PM	Jun 20, 2023, 11:26:55 PM	System	Revoke
○ Never connected	WebEngine	Docu-iiot-01.testenv.local	00BA39F1BE36A42D4BBDD52A5A2485C07D	Jun 19, 2028, 11:26:57 PM	Jun 20, 2023, 11:26:57 PM	System	Revoke
○ Never connected	Service Engine	ATSZG-WKS287.copa-data.internal	00EC4BBA4742E48946A622A10F4C57C1E0	Jun 19, 2028, 11:45:41 PM	Jun 20, 2023, 11:45:41 PM	admin admin	Revoke
○ Never connected	Engineering Studio	ATSZG-WKS287.copa-data.internal	00CAE511E389CF1B438C9DCBDDCEED0E62	Jun 19, 2028, 11:45:42 PM	Jun 20, 2023, 11:45:42 PM	admin admin	Revoke

Configuration for Certificate Management is integrated into Service Configuration Studio.

Certificate Management takes on the task of access verification for the **Data Hub**. In order to access a **Data Hub**, Services require credentials and **Certificate Bundles** from the **Certificate Management**. This verification is initiated by the **Data Hub**.

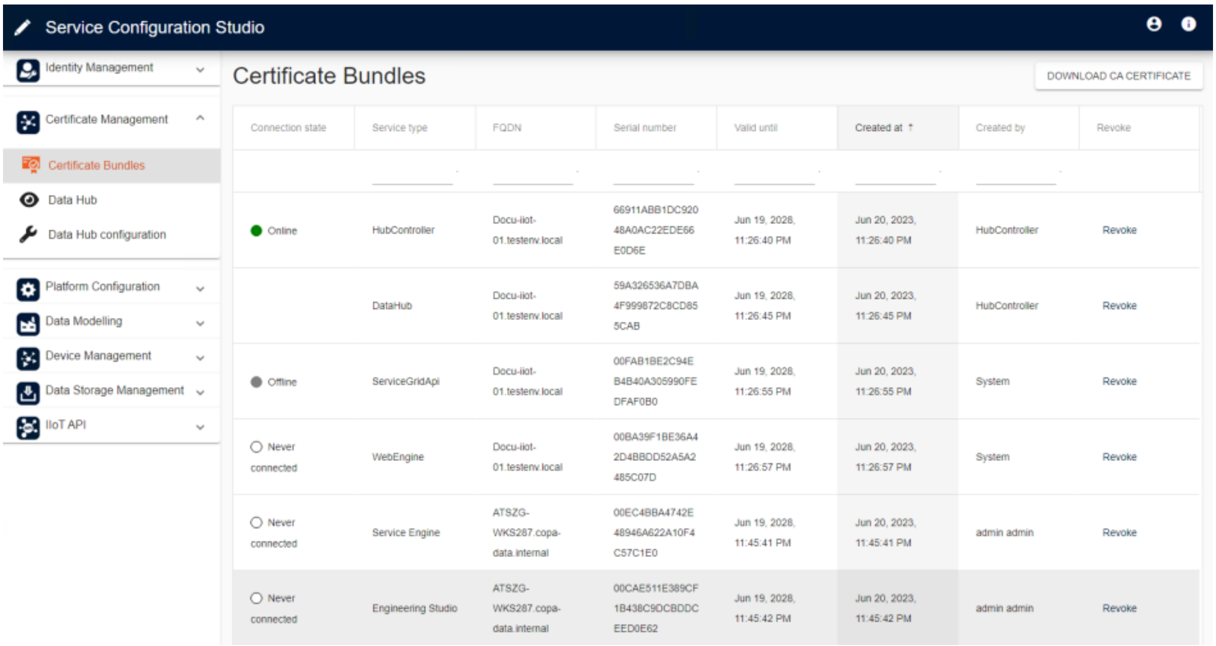
General procedure:

1. The client receives the **Certificate Bundle** and the credentials contained therein from the **Service Node Configuration Tool**.
2. The client connects to the **Data Hub**.
3. The **Data Hub** checks the access permission by querying the **Certificate Management**.
4. Depending on the result, the **Data Hub** either provides access to the client or declines it.

The tabs that are shown by Certificate Management depend on the variant installed:

IIoT Services (Docker)	IIoT Services (Windows native)
Tab: ▶ Certificates	**Tabs:** ▶ Data Hub ▶ Data Hub configuration ▶ Certificates

7.2 Certificate bundles



Certificate Bundles

Connection state	Service type	FQDN	Serial number	Valid until	Created at ↑	Created by	Revoke
● Online	HubController	Docu-iiot-01.testenv.local	66911ABB1DC92048A0AC22EDE66E0D6E	Jun 19, 2028, 11:26:40 PM	Jun 20, 2023, 11:26:40 PM	HubController	Revoke
	DataHub	Docu-iiot-01.testenv.local	59A326536A7DBA4F999872C8CD855CAB	Jun 19, 2028, 11:26:45 PM	Jun 20, 2023, 11:26:45 PM	HubController	Revoke
● Offline	ServiceGridApi	Docu-iiot-01.testenv.local	00FAB1BE2C94EB4B40A305990FEDFAF0B0	Jun 19, 2028, 11:26:55 PM	Jun 20, 2023, 11:26:55 PM	System	Revoke
○ Never connected	WebEngine	Docu-iiot-01.testenv.local	00BA39F1BE36A42D4BBDD52A5A2485C07D	Jun 19, 2028, 11:26:57 PM	Jun 20, 2023, 11:26:57 PM	System	Revoke
○ Never connected	Service Engine	ATSZG-WKS287.copa-data.internal	00EC4BBA4742E48946A622A10F4C57C1E0	Jun 19, 2028, 11:45:41 PM	Jun 20, 2023, 11:45:41 PM	admin admin	Revoke
○ Never connected	Engineering Studio	ATSZG-WKS287.copa-data.internal	00CAE511E389CF1B438C9DCBDDCEED0E62	Jun 19, 2028, 11:45:42 PM	Jun 20, 2023, 11:45:42 PM	admin admin	Revoke

This subpage shows all connections in IIoT Services set up with Certificate Bundles (CB).

In addition, you can download the self-signed root certificate of IIoT Services using **DOWNLOAD CA CERTIFICATE**. You require this certificate file to establish a HTTPS trust relationship (on page 40).

CERTIFICATE BUNDLES

Column	Description
Connection state	Shows the state of the connection between services and **Certificate Management**: - *Online*: This service is currently connected. - *Disconnected*: This service is currently not connected. - *Never connected*: This service has never been connected before - *Connection lost*: The connection has been interrupted with this service. If you move the mouse pointer over an entry, a pop-up (mouseover) also shows this information: - *Connection state changed*: Time stamp when the service last amended a connection state. No state is displayed for services that do not establish a direct connection to **Certificate Management**.
Service type	The service type is an internal name of the service.
FQDN	Fully-qualified domain name of the computer to which the certificate bundle is assigned.
Serial number	The serial number of the certificate.
Valid until	Date and time until which the certificate is valid and can be used by the respective service.
Created at	Date and time when the certificate was created.
Created by	The service or the user that creates the connection. **Note:** If the user that created the Certificate Bundle is deleted from the IDS, then the user ID of the user will be displayed here.
Revoke	With this button you can revoke the certificate bundle for the selected service.

Column	Description
	After selecting **Revoke**, the following is displayed: - The time (UTC) of the **Revoke** - The user who executed the **Revoke** A **Revoke** is only necessary if the certificate bundle is potentially compromised (on page 67). Normally, you continue to use the certificate bundle created during installation.

> ⚠ **Attention**
>
> The **Revoke** of a certificate bundle interrupts the communication between services. Before each **Revoke**, check and make sure if this intervention is appropriate and necessary for the respective service (on page 67).

7.2.1 Revoke of certificate bundles

Each connection between a service and **Certificate Management** requires its own certificate bundle. Each certificate bundle can be revoked separately using Revoke (on page 65).

Certificate bundles should only be revoked for specific reasons.

REASONS FOR A REVOKE

As soon as there are any indications that a certificate bundle could be compromised, it should be revoked.

Possible reasons for a revoke:

- The certificate including the private key has been stolen.
- Secret certificate information such as the private key is publicly available.

A Revoke has the following consequences:

- The respective service can no longer connect to IIoT Services.
- The exchange of data between the affected service and IIoT Services has been interrupted.

To restore functionality, you must generate a new certificate bundle. The exact procedure depends on the chosen installation option for IIoT Services.

⚠ Attention

The certificate bundle for **Certificate Management** should not be revoked using **Revoke**. If the certificate bundle for **Certificate Management** is thought to be compromised, a complete re-installation of the IIoT Services is required.

7.2.2 Generate new certificate bundles (Docker)

For the IIoT Services (Docker) installation option, there are several different ways to generate new certificate bundles for affected services.

OPTION A) WINDOWS-BASED SERVICES

This option applies for the following services under Windows:

- Service Engine
- Engineering Studio
- Report Engine
- Web Engine (on Windows)

To generate a new certificate bundle:

1. Revoke the certificate bundle for the selected service using **Revoke** in <NAME_SRVICE_GRID_STUDIO>.
   a) To do this, go to the **Certificate Bundles** subnode in the **Certificate Management** main node.
   A list of all certificate bundles is shown.
   a) In the **Revoke** column, click on the **Revoke** button in the entry for the corresponding bundle.
2. Use the **IIoT Services Connection Wizard** to generate a new certificate bundle for the connection between the service and IIoT Services.
3. Restart the service.

In this way, you have secured the communication for the selected service with a new certificate bundle.

OPTION B) DOCKER-BASED SERVICES

This option applies for the following services in Docker:

- All IIoT Services

- Web Engine (in Docker)

To generate new certificate bundles for a Docker-based service:

1. Revoke the certificate bundle for the selected service using **Revoke** in the web interface of **Certificate Management**.
2. Stop all services of the IIoT Services with this Docker command:
   docker-compose down
3. Delete the volume of the affected service (see table below) with this Docker command:
   docker volume rm <volume>
4. Start all services of the IIoT Services with this Docker command:
   docker-compose up
   When the service is started, deleted volumes will be created again and, if necessary, new certificate bundles will be created automatically.

In this way, you have secured the communication for the selected service with a new certificate bundle.

7.2.3 Generate new certificate bundles (Windows native)

For the IIoT Services (Windows native) installation option, you can generate new certificate bundles for affected services in the same way.

1. Revoke the certificate bundle for the selected service using **Revoke** in <NAME_SRVICE_GRID_STUDIO>.
   a) To do this, go to the **Certificate Bundles** subnode in the **Certificate Management** main node.
      A list of all certificate bundles is shown.

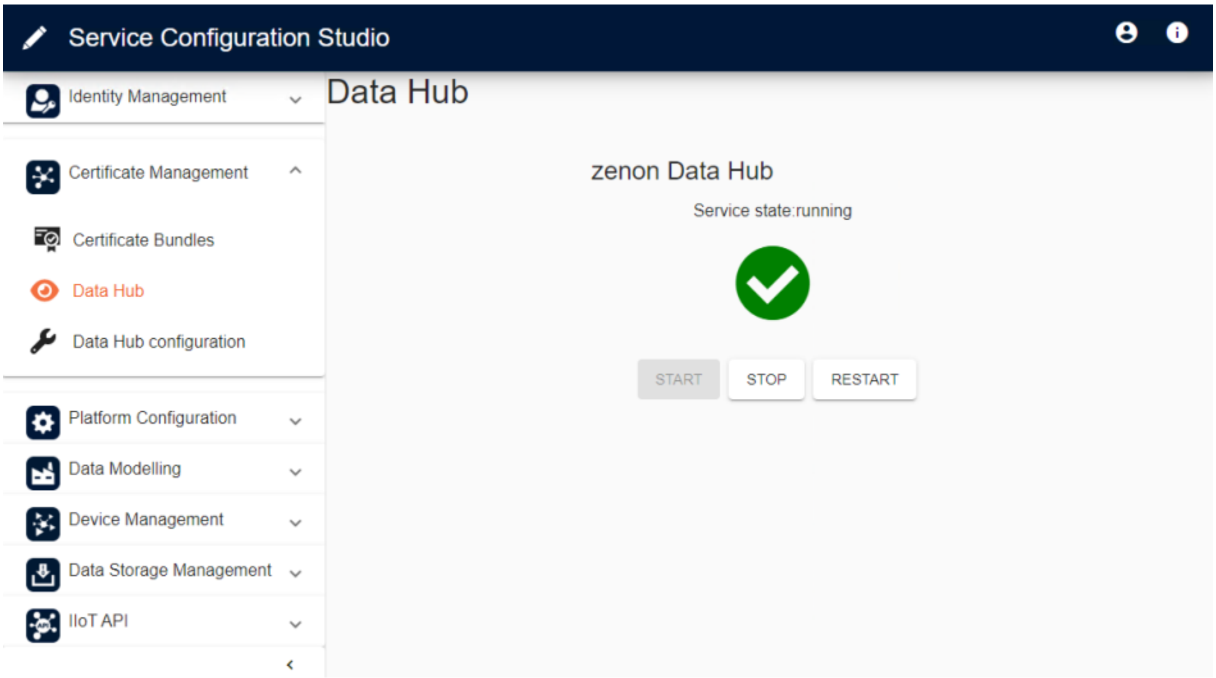
      In the **Revoke** column, click on the **Revoke** button in the entry for the corresponding bundle.

1. Use the **IIoT Services Connection Wizard** to establish the connection between the service and IIoT Services.
2. Restart the service.

In this way, you have secured the communication for the selected service with a new certificate bundle.

7.3 Data Hub

Here you can start and stop the **Data Hub**. It is the central communication hub in the IIoT Services.

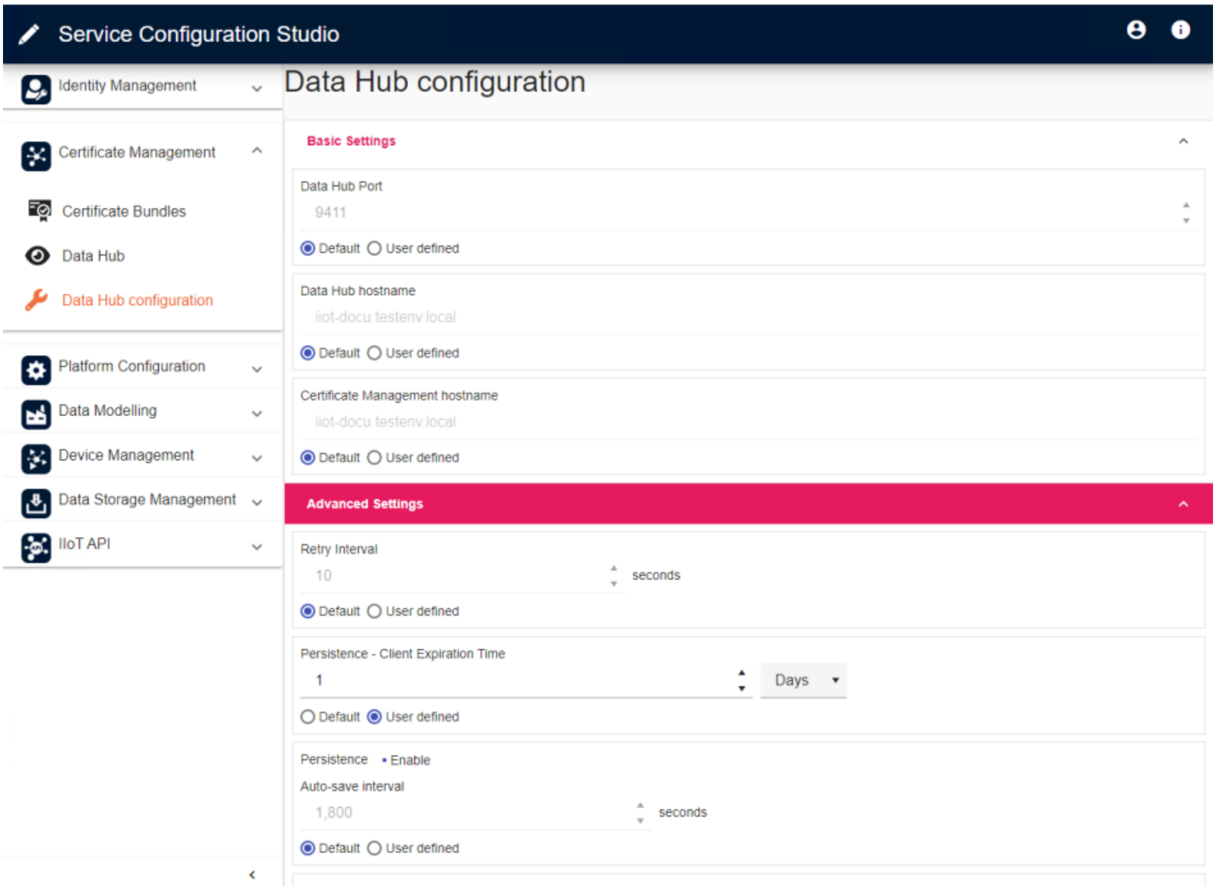


To start, stop or restart the service, click on the respective button.
A symbol shows the current status.

Option	Description
Symbol	Shows whether the service is running or stopped. ▸ Green check mark: Service is running. ▸ Red X: Service is stopped.
Start	Starts the **Data Hub**.
Stop	Stops the **Data Hub**. A confirmation prompt is displayed before it is stopped.
Restart	Restarts the **Data Hub**. A confirmation prompt is displayed before it is restarted.

7.3.1 Data Hub configuration

Here you can configure the **Data Hub**. The configuration is divided into **Basic Settings** and **Advanced Settings**.



Data Hub configuration

Basic Settings

Data Hub Port: 9411
● Default ○ User defined

Data Hub hostname: iiot-docu.testenv.local
● Default ○ User defined

Certificate Management hostname: iiot-docu.testenv.local
● Default ○ User defined

Advanced Settings

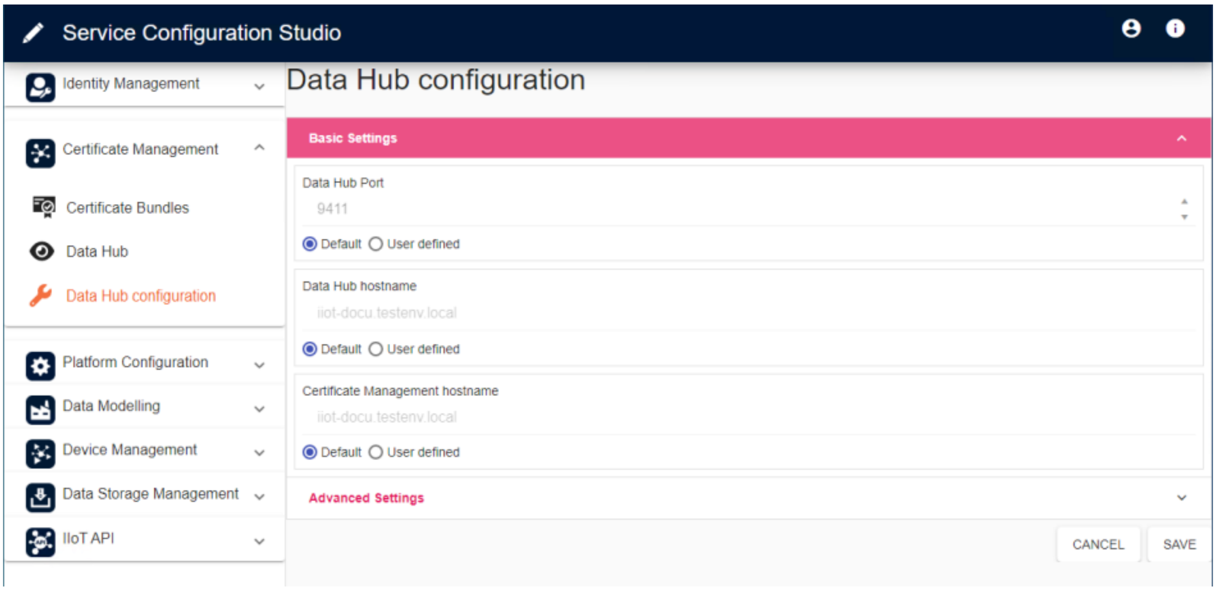
Retry Interval: 10 seconds
● Default ○ User defined

Persistence - Client Expiration Time: 1 Days
○ Default ● User defined

Persistence • Enable
Auto-save interval: 1,800 seconds
● Default ○ User defined

7.3.1.1 Basic Settings

Basic settings for communicating with **Data Hub**.



Data Hub configuration

Basic Settings

Data Hub Port: 9411
● Default ○ User defined

Data Hub hostname: iiot-docu.testenv.local
● Default ○ User defined

Certificate Management hostname: iiot-docu.testenv.local
● Default ○ User defined

Option	Description
Data Hub Port	Port number of the **Data Hub**. - **Default**: the default port number is used. Default: *9411* - **User defined**: A user-defined port number is used. Configuration in the input field. Input in the field or configuration by means of the arrow keys. Permitted range: *1* to *65535*. Default: *Default*
Data Hub hostname	Address of the **Data Hub**. - **Default**: The default URL is used. This is the same as the Fully Qualified Domain Name of the **Certificate Management**. - **User defined**: A user-defined URL is used. Configuration in the input field. Default: *Default*
Certificate Management hostname	Address of the **Certificate Management**. - **Default**: The default URL is used.

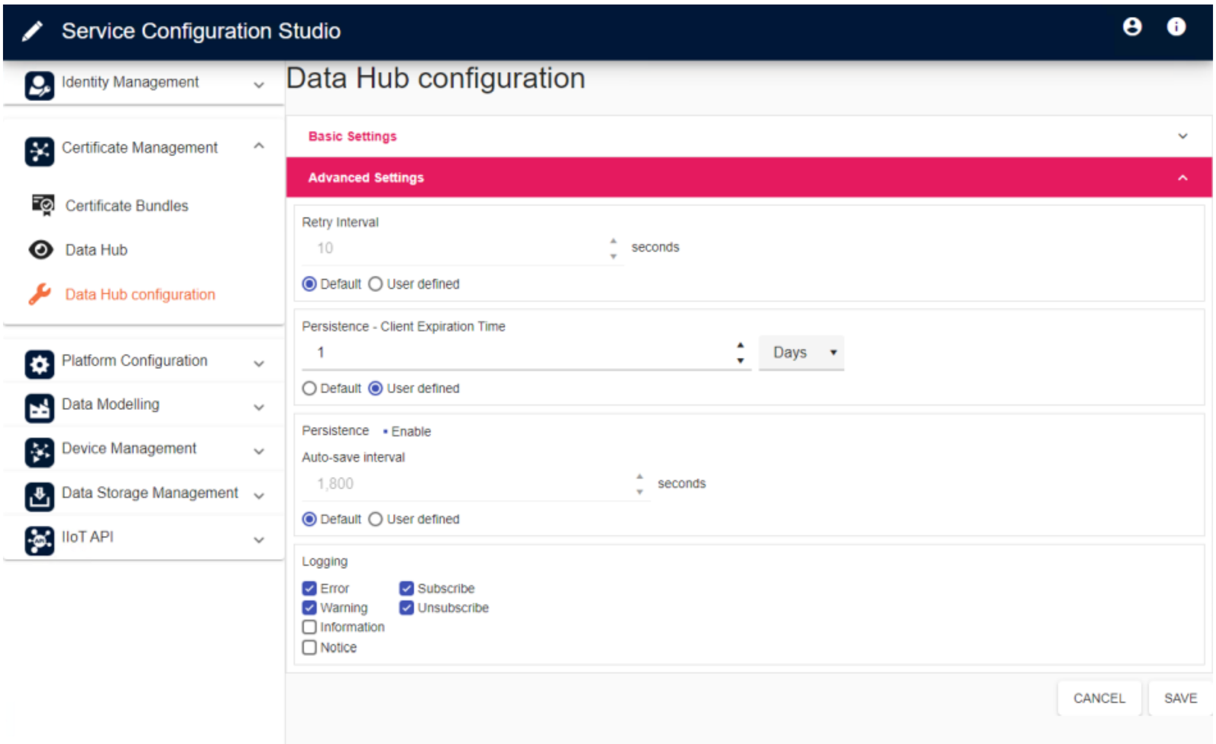
Option	Description
	- **User defined**: A user-defined URL is used. Configuration in the input field. Default: *Default*
Advanced Settings	Clicking on the button opens the advanced settings for the **Data Hub**.

NAVIGATION BAR

Option	Description
Cancel	Discards all settings and closes the dialog.
Save	Saves the current configuration on the **Certificate Management**. **Note:** If the **Certificate Management** and the **Data Hub** are running on the same computer, the configuration file of the **Data Hub** will also be updated. However, it is necessary to restart the **Data Hub**. The restart can be executed immediately in the dialog by the user.

7.3.1.2 Advanced Settings

Additional settings for communication with the **Data Hub** as well as client management and logging.



Data Hub configuration

Retry Interval: 10 seconds
● Default ○ User defined

Persistence - Client Expiration Time: 1 Days
○ Default ● User defined

Persistence • Enable
Auto-save interval: 1,800 seconds
● Default ○ User defined

Logging
☑ Error ☑ Subscribe
☑ Warning ☑ Unsubscribe
□ Information
□ Notice

Option	Description
Advanced Settings	Clicking on the button closes the advanced settings for the **Data Hub**.
Retry Interval	Time interval in seconds until an outgoing message is resent: - **Default**: The default time interval is used. - **User defined**: A user-defined time interval is used. Input in the field, or configuration using the arrow keys. Default: *10 s*
Persistence Client Expiration Time	Allows clients to be deleted automatically if they do not reconnect within a certain period of time. Setting in days of how long the Service Hub will wait for reconnection. - **Default**: The default time interval is used. - **User defined**: A user-defined time interval is used. Input in the field, or configuration using the arrow

Option	Description
	keys. Default: *1Day*
Persistence	Saving of persistent message data on a data storage device. Thus, information is stored for all messages including subscriptions, current messages and stored messages. The data are therefore available again after a restart. **Enable**: - *Active*: The data are saved. Configure the preferred time interval for automatic saving. - *Inactive*: The data are not saved. **Persistence Autosave Interval**: Interval for automatic saving. - **Default**: The default time interval is used. - **User defined**: A user-defined time interval in seconds is used. Input in the field, or configuration using the arrow keys. Default: *1800 s*
Logging	Settings for the logging of the **Service Node** actions on the **Data Hub**. Selection of the logging level using checkboxes. Possible levels: - *Error*: Error - *Warning*: Warnings - *Information*: Information - *Notice*: Notices - *Subscribe*: Subscription of Topics via Services - *Unsubscribe*: Unsubscription of Topics via Services **Note:** The log files can be displayed in the **Diagnosis Viewer** in the **Certificate Management** area. You can also find further information in relation to this in the **Additional modules in the Diagnosis Viewer** (on page 242) chapter.

NAVIGATION BAR

Option	Description
Cancel	Discards all settings and closes the dialog.
Save	Saves the current configuration on the **Certificate Management**. **Note:** If the **Certificate Management** and the **Data Hub** are running on the same computer, the configuration file of the **Data Hub** will also be updated. However, it is necessary to restart the **Data Hub**. The restart can be executed immediately in the dialog by the user.

7.3.2 Storage location

The **Hub Controllers** data is stored in the following save location:

- *%CD_SYSTEM%/ServiceGrid/HubController*

MONGODB CONFIGURATION FILE

The name of the

The *mongod.cfg* configuration file is saved in the following save location:

- *%CD_SYSTEM%/ServiceGrid/Persistence*

UPDATE

In the event of an update, all data from version 10 will be amended to the current storage structure.

> **Information**
>
> Delete your browser cache after an update. As a result, it is ensured that, when logging in to the Identity Service, no outdated or saved login data is used.

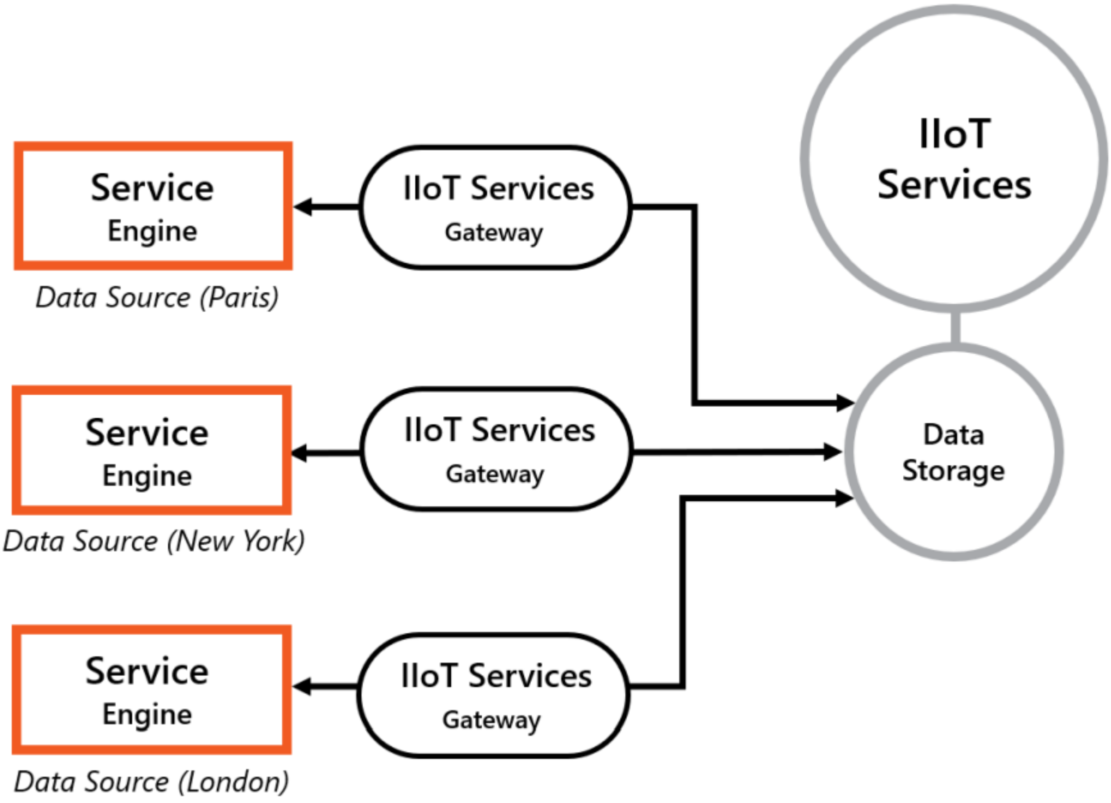
8 Data Storage

Data Storage enables you to store data centrally via IIoT and make it available for other applications. You administer **Data Storage** in Service Configuration Studio.

You can find notes on configuration in Engineering Studio in the **IIoT Services - configuration in Engineering Studio** section.

8.1 Evacuate data centrally

You can use **Data Storage** to evacuate alarm data, archive data and event data from several Service Engine instances into IIoT Services, and to read it back again. Archive data is historical variable values. The complete process runs transparently for the user. No user input is necessary regardless of the configuration.



You can store the data of multiple Service Engine instances centrally in Data Storage.
Every Service Engine can read back its own data from Data Storage.

Important: Each Service Engine can only read back its own data!

AREA OF APPLICATION

<u>You can use Data Storage as:</u>

- Central evacuation location for data from Service Engine instances.
- Central data source for reading back data in Service Engine instances.

The data is deleted in Service Engine (data source) once it has been successfully evacuated to **Data Storage**.

SUPPORTED DATA ACTIONS

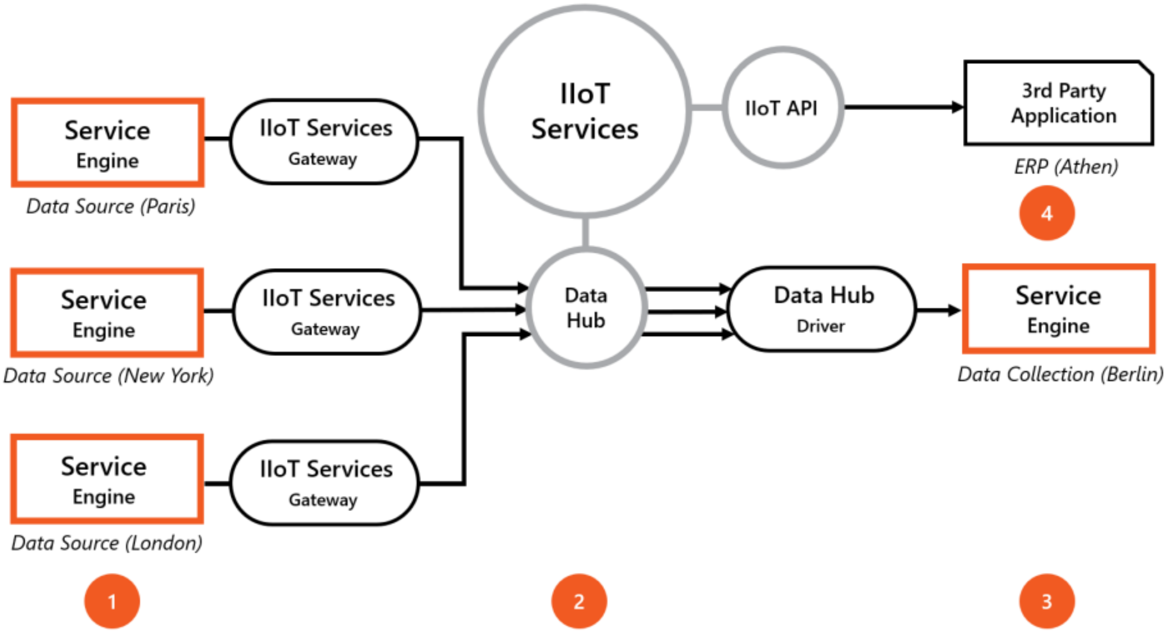
When transferring data from Service Engine to **Data Storage**, data is transferred and saved in unchanged form. This also applies to reading back the data from Data Storage to Service Engine.

Supported data types	Variable access authorization*
Alarms	*None*
Archive data	*Read only*
Events	*None*

*required access authorization in Service Engine (data source).

8.2 Provide evacuated data

You can use the IIoT API to query archive data from **Data Storage** and provide it to third-party applications. Archive data is historical variable values.



Archive data from Data Storage can be queried using the IIoT API.

AREA OF APPLICATION

<u>**You can use this use case for the following scenarios:**</u>

- Reading archive data from Data Storage
- Provision of archive data for third-party applications

SUPPORTED DATA ACTIONS

When transferring data from **Data Storage** to the IIoT API , archive data is transferred in unchanged form. Access to Data Storage is read only. A **Data Storage** query using the IIoT API therefore has no effect on the data in Data Storage.

Supported data types	Variable access authorization*
Archive data	*Read only*

*required access authorization in Service Engine (data source).

8.3 Administration in Service Configuration Studio

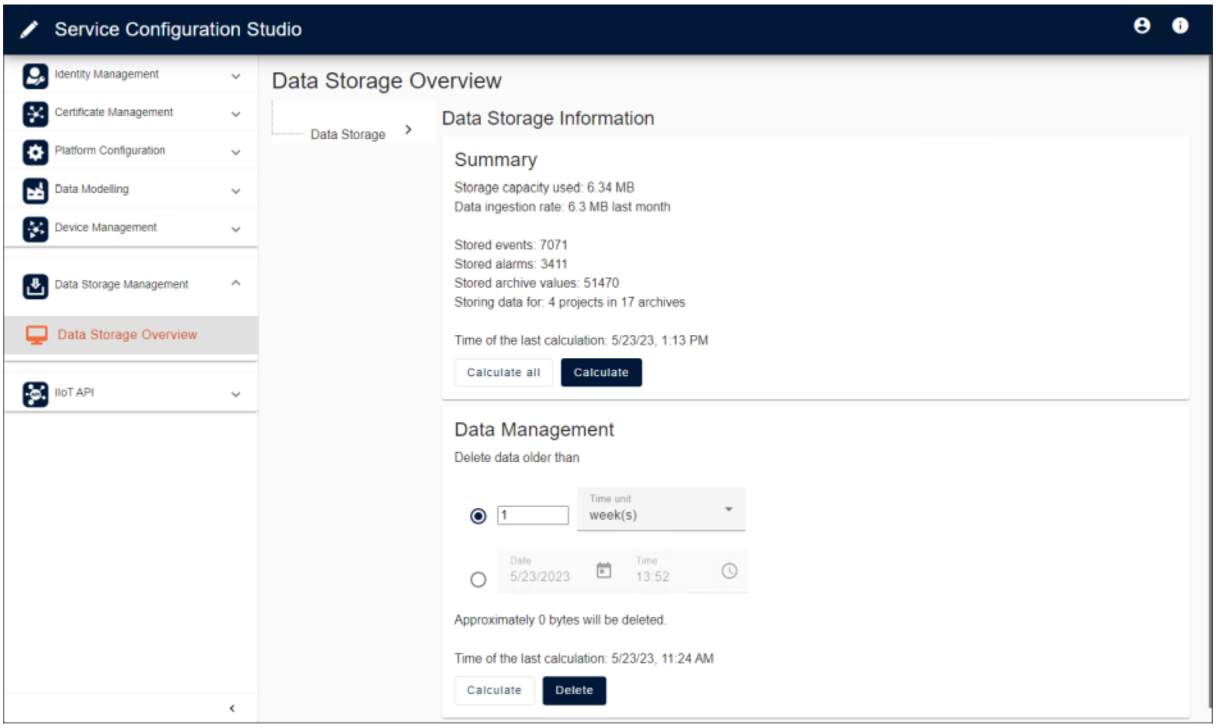
The service for **Data Storage** in the Service Configuration Studio allows the following:

- Display of database statistics
- Display of project statistics
- Manage data saved in **Data Storage**

The start takes place via:

- Service Configuration Studio
  or:

- as external HTML site: *https://[FQDN]:9443/data-storage*
  Example: *https://iiot-docu-v8.testenv.local:9443/data-storage*



Data Storage Overview

Data Storage Information

Summary

Storage capacity used: 6.34 MB
Data ingestion rate: 6.3 MB last month

Stored events: 7071
Stored alarms: 3411
Stored archive values: 51470
Storing data for: 4 projects in 17 archives

Time of the last calculation: 5/23/23, 1:13 PM

Data Management

Delete data older than

● 1 Time unit: week(s)

○ Date: 5/23/2023 Time: 13:52

Approximately 0 bytes will be deleted.

Time of the last calculation: 5/23/23, 11:24 AM

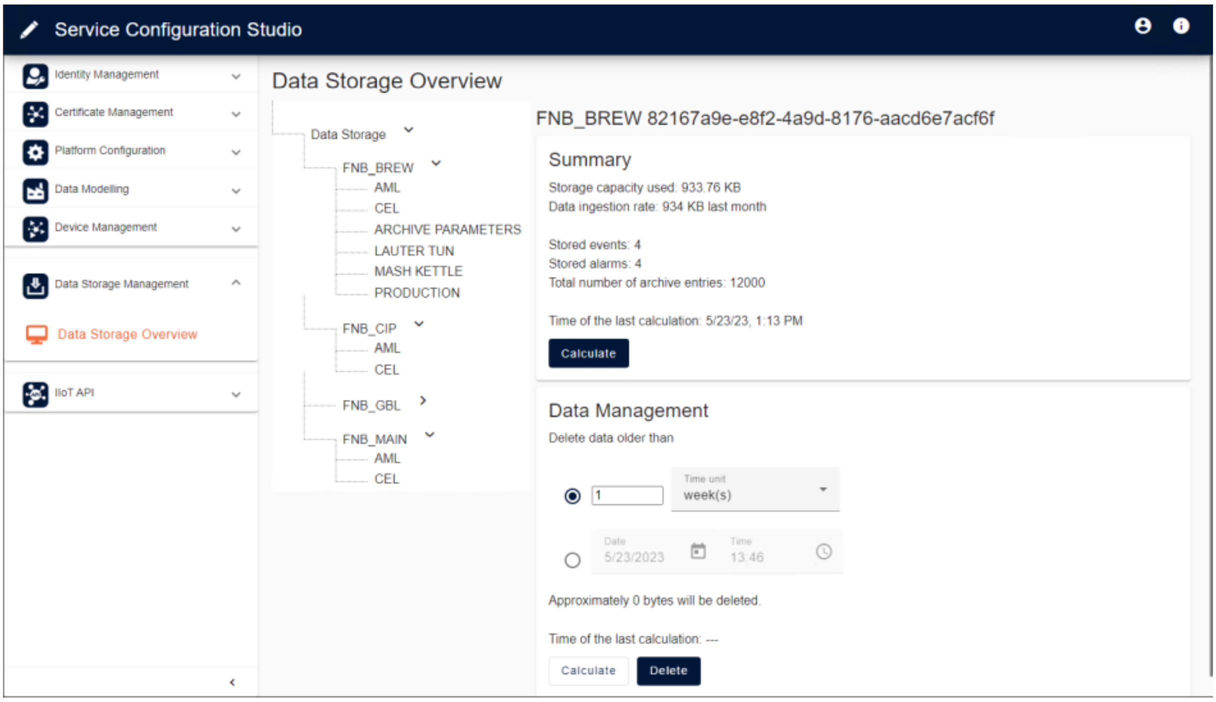
8.3.1 Data Storage Overview

You can get an overview of the saved data in the Overview tab.

Areas:

- Summary (on page 81): Saved objects
- Data Management (on page 83): Deletion of objects

- Service Engine (on page 84): Objects for each linked Service Engine



FNB_BREW 82167a9e-e8f2-4a9d-8176-aacd6e7acf6f

Summary

Storage capacity used: 933.76 KB
Data ingestion rate: 934 KB last month

Stored events: 4
Stored alarms: 4
Total number of archive entries: 12000

Time of the last calculation: 5/23/23, 1:13 PM

Calculate

Data Management

Delete data older than

● 1 Time unit week(s)

○ Date 5/23/2023 Time 13:46

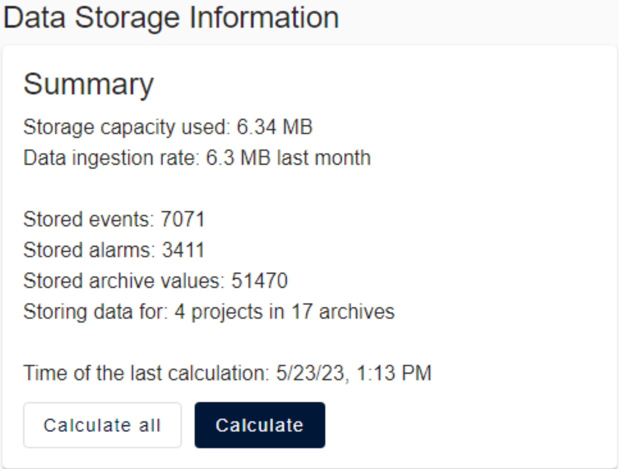
Approximately 0 bytes will be deleted.

Time of the last calculation: ---

Calculate Delete

8.3.1.1 Summary

You get an overview of all saved information in this section.



Data Storage Information

Summary

Storage capacity used: 6.34 MB
Data ingestion rate: 6.3 MB last month

Stored events: 7071
Stored alarms: 3411
Stored archive values: 51470
Storing data for: 4 projects in 17 archives

Time of the last calculation: 5/23/23, 1:13 PM

Calculate all Calculate

Display	Description
Storage capacity used:	Amount of storage used. This includes data storage and index storage.
Data ingestion rate:	Data ingestion per month.
Stored events:	Saved events.

Display	Description
Stored alarms:	Saved alarms.
Stored archive values:	Stored archive values.
Storing data for:	Number of saved projects and archives.
Time of the last calculation:	Time of the last calculation.
Calculate	Starts calculation for the Service Engine project page. **Note:** With large amounts of data, the calculation can take a very long time.
Calculate all	Starts calculation for all Service Engine project pages. **Note:** With large amounts of data, the calculation can take a very long time.

CYCLICAL CALCULATION OF THE STATISTICS DATA

Depending on the size of the database, the calculation of the statistics can take some time. For this reason, the calculation is carried out cyclically via a service.
The start time and calculation time can be set using *appsettings.json* with *ManagementCacheConfiguration* .

Parameter:

IntervalHours: Setting for the hour intervals in which the automatic statistics calculation is to be carried out.

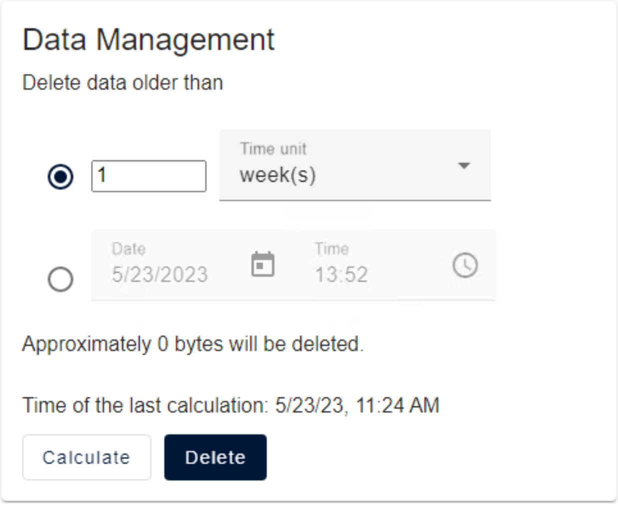
StartCalculationTime: Start of the first statistics calculation.

Example:

```
"ManagementCacheConfiguration": {

"IntervalHours": 12,

"StartCalculationTime": "02:00:00"

}
```

8.3.1.2 Data management

You administer the objects in **Data Storage** in this section. You can calculate the size of the storage occupied and delete data.



Data Management

Delete data older than

● 1 Time unit: week(s)

○ Date: 5/23/2023 Time: 13:52

Approximately 0 bytes will be deleted.

Time of the last calculation: 5/23/23, 11:24 AM

Option	Description
Input field	Entry of the numeric value for **Time Unit**. Configures the time period for the action to be executed.
Time Unit	Time unit from numerical field. Select from drop-down list: - *Hour(s)* - *Day(s)* - *Week(s)* - *Month(s)* - *Years()*
Date/Time	Timestamp, up to which data is deleted.
Time of last calculation:	Time of the last calculation. Indication of the amount of data to be deleted.
Calculate	Calculates the extent of data to be deleted. The basis is the set time filter. **Note:** With large amounts of data, the calculation can take a very long time.
Delete	Deletes data selected on the basis of the filter. Data from all subnodes is included.

Option	Description
	Note: With large amounts of data, deletion can take a very long time.

DELETE DATA

The time period for the deletion of data can be set in two ways.

- Data that is older than a certain time period. This is selected by means of configuration in the **Time Unit** option.
  This time period can be defined in complete hours, days, weeks months or years in the past. Smaller units are ignored in each case. 1 full hour thus means: Minutes and seconds are not taken into account.
- Data that is older than a certain timestamp. They are selected using the **Date/Time** option.

Example:

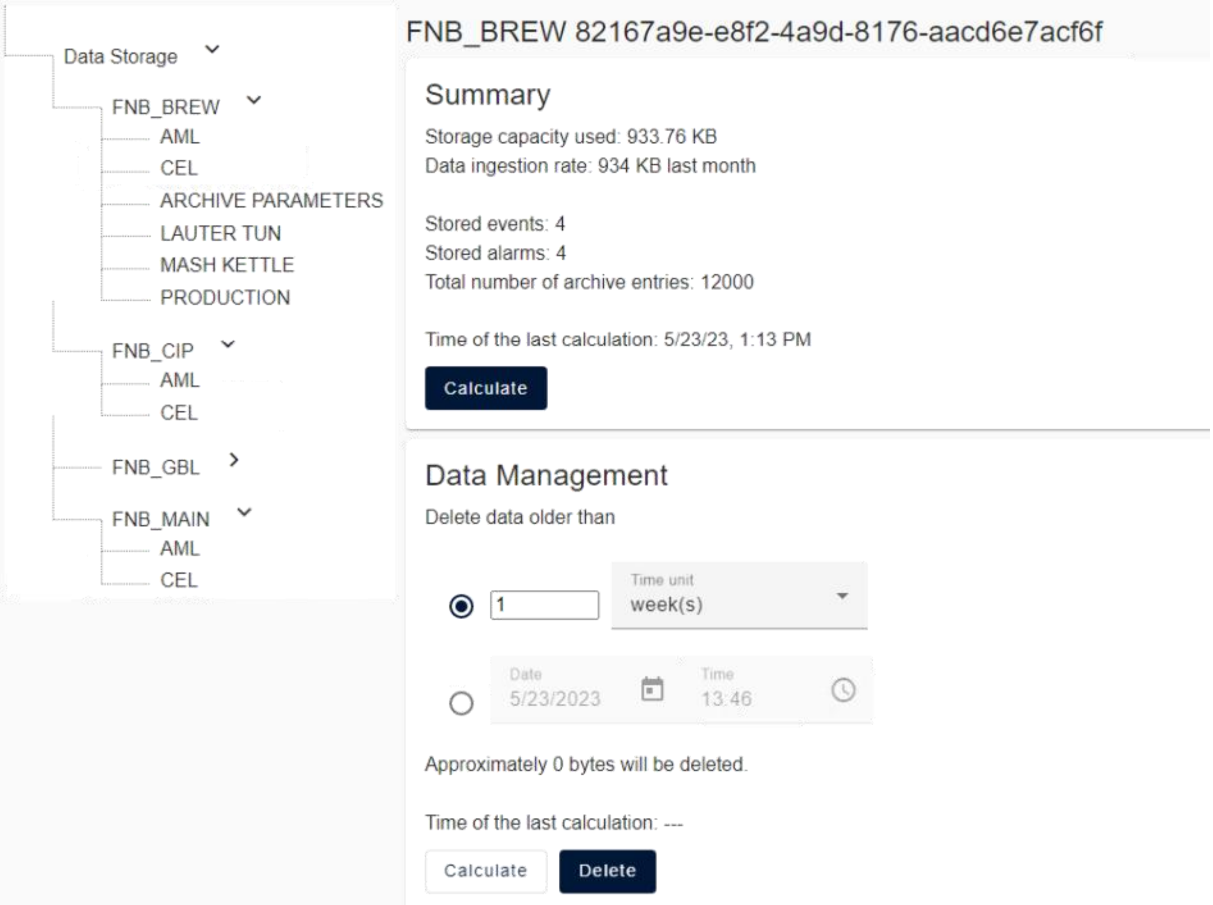
- The time filter is set to: Delete data that is older than 1 hour.
- Current time on deletion: 13:30:23
- All data from before 12:00:00 is deleted.

8.3.2 Linked Service Engine

Each linked Service Engine is represented by its own node.
In the subnodes, data is displayed for:

- Archive
- Lot archive
- AML

- CEL



FNB_BREW 82167a9e-e8f2-4a9d-8176-aacd6e7acf6f

Summary

Storage capacity used: 933.76 KB
Data ingestion rate: 934 KB last month

Stored events: 4
Stored alarms: 4
Total number of archive entries: 12000

Time of the last calculation: 5/23/23, 1:13 PM

Calculate

Data Management

Delete data older than

● 1 Time unit week(s)

○ Date 5/23/2023 Time 13:46

Approximately 0 bytes will be deleted.

Time of the last calculation: ---

Calculate Delete

Display	Description
Storage capacity used:	Amount of storage used. This includes data storage and index storage.
Data ingestion rate:	Data growth per month.
Stored events:	Stored events for CEL and **Continuous export**.
Stored alarms:	Stored alarms for AML and **Continuous export**.
Stored archive values:	Stored archive values.
Time of the last calculation:	Time of the last calculation.
Calculate	Starts calculation for the Service Engine project page. **Note:** With large amounts of data, the calculation can take a very long time.

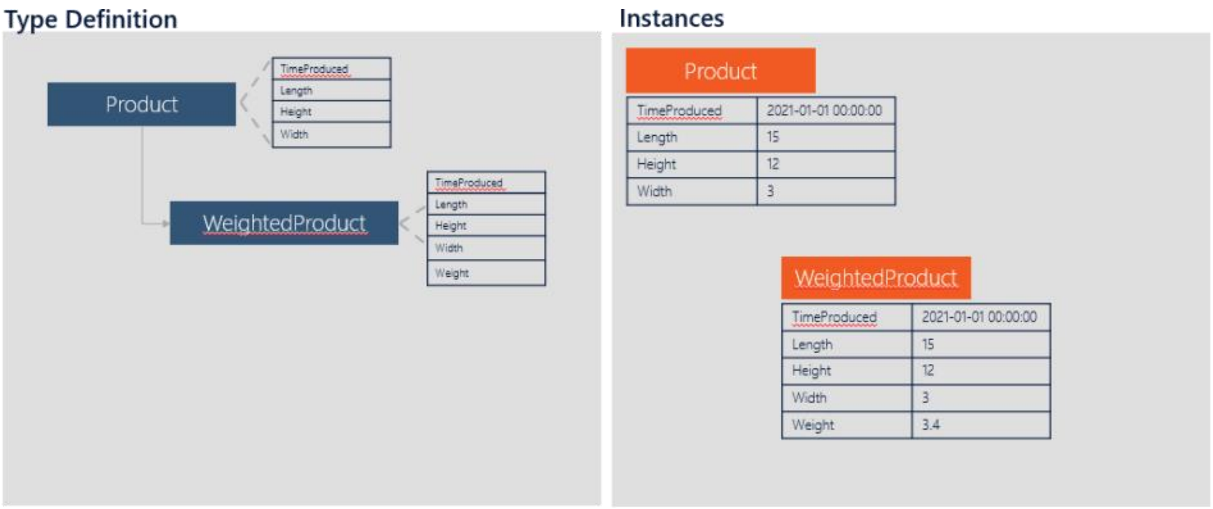
9 Data Modelling

Data Modelling is a central service of the zenon Software Platform. It offers a central repository for all types of structured data both of the zenon Software Platform as well as custom data models. The data is available via a **GraphQL** (on page 98) interface.

zenon Data Modelling is based on a relational model, similar to an Entity-Relationship model. Modular data models are defined in **Construction Kit Libraries**.

EXAMPLE

Example of a scheme of a data model and the data contained therein:



Product	
TimeProduced	2021-01-01 00:00:00
Length	15
Height	12
Width	3

WeightedProduct	
TimeProduced	2021-01-01 00:00:00
Length	15
Height	12
Width	3
Weight	3.4

Scheme for data model

Management of the data models:

- Support of tenants
- Import and removal of models
- Validation of models
- Administration of versions

Operative use of data models:

- Comprehensive **GraphQL** interface, which enables modifying and querying data and querying metadata.
- **GraphQL Editor** with autocomplete and integrated scheme documentation
- Online scheme update when changing the data model
- Validation of **GraphQL** mutations to scheme breaches

Note: zenon Data Modelling is still in development. Over time, there will be more and more data and functionality available.

9.1 Terminology for COPA-DATA Data Modelling

In this documentation, Data Modelling labels are used in English regardless of the display language.

Label	Description
Association	Relationship between entities of a certain type. It has incoming/outgoing multiplicity and a description.
Attribute	A property of an entity with a value. It has a simple data type and a description.
Construction Kit (CK)	The Data Model, i.e. all metadata together within a **Tenant**. Metadata consists of, for example, type definitions, tags etc. The Data Model is made up of different **Construction Kit Libraries**.
Construction Kit Library (CKL)	A logically-coherent scheme, which has its own metadata, and can be added to a **Construction Kit** by means of import.
Entity	Instance of a defined **Type**.
Namespace	A prefix for the types within a **Construction Kit Library**. **Functions:** ▸ Makes naming conflicts less likely ▸ Enables better identification of **Types**
Tag	Semantic information that can be added to the definition of **Attribute**s or **Association**s. It is for further categorization and classification.
Tenant	Highest organization level in the **Data Modelling service**. Each **Tenant** has its own **Construction Kit**.
Type	Definition of objects through names, inheritance, **Associations** and **Attributes**.

9.2 Construction Kit

The **Construction Kit** is all metadata together in a **Tenant** of Data Modelling. The **Construction Kit** is made up of different **Construction Kit Libraries**.

The following is configured in the **Construction Kit**:

- **Types**: Logical objects in the data model. On the basis of the defined **Types**, corresponding instances (**Entities**) can be generated.
- **Attributes**: The properties that the respective **Types** have are configured. **Attributes** are assigned to **Types** and have a certain data type. Instances of the respective **Types** can be used to set the values of the corresponding attributes **Attributes**.
- **Associations**: Stipulate which instances of different types can be linked to one another.
- **Inheritance**: Configures the inheritance structure of the defined **Types**.
- **Association Tags**: When configuring **Associations**, defined **Association Tags** can be linked.
- **Attribute Tags**: When configuring **Attributes**, defined **Attribute Tags** can be linked.

Construction Kit Libraries can exhibit dependencies on one another with:

- Inheritance: Reference to basic **Type** that has been configured in another **Construction Kit Library**.
- **Associations**: Reference from associated **Type** that has been configured in another **Construction Kit Library**.
- **Attribute Tag**: Use of an **Attribute Tag** that has been configured in another **Construction Kit Library**.
- **Association Tag**: Use of an **Association Tag** that has been configured in another **Construction Kit Library**.

9.2.1 Construction Kit Libraries

A **Construction Kit Library** (**CKL**) is a logical scheme for a certain area of application. It contains the **Types**, including **Asscociations**, **Attributes** and **Tags** that are required for a certain usage or solution.

Construction Kit Libraries can be imported into the **Construction Kit** of a **Tenant**. Imported **Construction Kit Libraries** can also be deleted. However, this is only possible if no other **Construction Kit Libraries** are based on them.
Attention: When a **Construction Kit Library** is deleted, the definitions and their instances are deleted.

Construction Kit Libraries are versioned; a new version of a **Construction Kit Library** must always contain all definitions contained in the previous versions (principle: grow only).

Construction Kit Libraries can have dependencies. You thus configure the other **Construction Kit Libraries** on which they are based. To do this, you derive your **Types** from **Types** configured there or create **Associations** to the **Types** configured there.

Metadata of a Construction Kit Library:

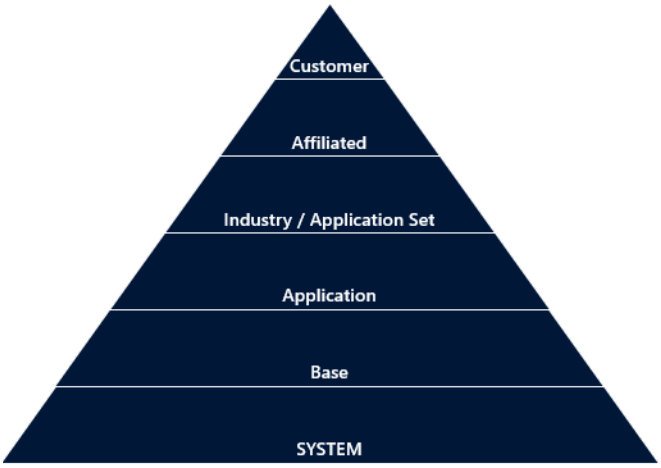
- unique ID
- Version
- Display name for display in the user interface
- unique Namespace
- Assignment to a Layer
- Dependencies on other **CKLs**:
  - GUID
  - Version

9.2.2 Construction Kit layer

Each **Construction Kit** consists of **Layer**s. Each **Construction Kit Library** configures an assigned layer.

Construction Kits serve different purposes and are developed by different groups of people. Layers offer a simple possibility for structuring.

A **Construction Kit Library** can be dependent on other **Construction Kit Libraries** that are configured on the same layer or a layer below.

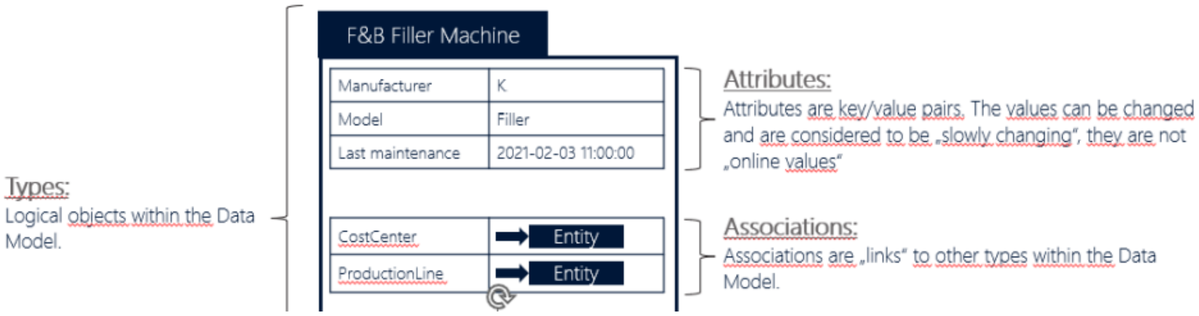


Display of all available layers.

Only the company COPA-DATA can create **Construction Kit Libraries** for the **System**, **Base** and **Application** layers.

9.2.3 Types

A **Type** is uniquely identifiable from the namespace of the **Construction Kit Library** + name (**Construction Kit ID**). **Types** have **Attributes** and **Associations** to other **Types**. They can be derived from other **Types** (simple inheritance).



Manufacturer	K
Model	Filler
Last maintenance	2021-02-03 11:00:00

CostCenter	Entity
ProductionLine	Entity

Types:
Logical objects within the Data Model.

Attributes:
Attributes are key/value pairs. The values can be changed and are considered to be „slowly changing", they are not „online values"

Associations:
Associations are „links" to other types within the Data Model.

Example of a Type with Attribute and Associations.

9.2.3.1 Attributes

Attributes are configured to types. A significant part of the definition is the name of the **Attribute**.

The Name:

- Must be unique in the inheritance tree of the **Type**
- Must not be identical to the name of an **Association**

A **Value Type** must also be given with the definition.
The following are available:

- *Integer*
- *String*
- *DateTime*
- *StringArray*
- *Boolean*
- *Double*
- *IntArray*
- *Binary*

9.2.3.2 Associations

Associations are configured to **Types**. This type is the **Source Type** for the **Association**. A significant part of the definition is the name of the **Association**.

The Name:

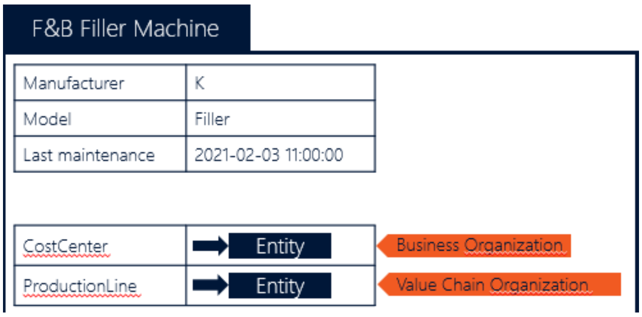
- Must be unique in the inheritance tree of the **Type**
- Must not be identical to the name of an **Attribute**

Further components for the definition of an **Association** are:

- **Target Type**: associated **Type**
- **Multiplizität**: Can be present in the form *N* (as many as desired) or *ZeroOrOne* (zero or one). A distinction is made between two types:
  - **Source** multiplicity: Denotes how many instances of the **Source Type** can be linked for this **Association**.
  - **Target** multiplicity: Denotes how many instances of the **Target Type** can be linked for this **Association**.

9.2.4 TAGs

Attribute TAGs and **Association TAGs** add additional information to the definitions of **Attributes** and **Associations**. These **TAGs** are configured within a **Construction Kit Library**. Defined TAGs are available for the definition of **Attributes** and **Associations**. They serve to categorize and classify **Attributes** and **Associations**.



F&B Filler Machine

Manufacturer	K
Model	Filler
Last maintenance	2021-02-03 11:00:00

CostCenter	➡ Entity	Business Organization
ProductionLine	➡ Entity	Value Chain Organization

Example of a Type with the Associations "Cost Center" and "Production Line" and the assigned Association TAGs "Business Organization" and "Value Chain Organization".

9.3 Tenants

A **Tenant** (mandant) is a logical separation (instance) of all the data that concerns Data Modelling. It comprises all metadata and data of a **Construction Kit**.

The following is applicable here:

- Each instance of the **Data Modelling Service** supports several Tenants.
- The default Tenant "*zenon*" is created during initialization.
- Each Tenant contains its own **Construction Kit**.

9.4 Data access

For program access to Data Modelling data, COPA-DATA provides a GraphQL interface. GraphQL is a language for data queries and data manipulation. You can find details in relation to this in the freely-available open-source documentation.

The **GraphQL Editor** (on page 98) in the Data Modelling service offers a simple graphical possibility to detect the **GraphQL** scheme, and to query and amend data.

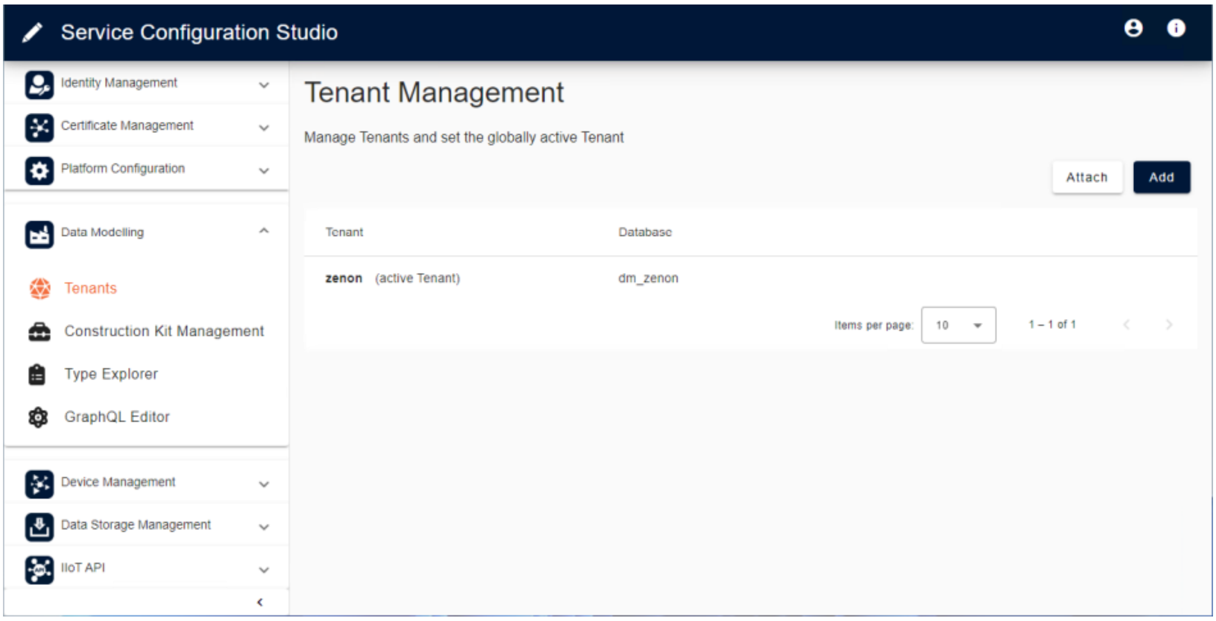
9.5 Configuration and display

Data Modelling is configured and queried in Service Configuration Studio.
Note: Activate your license before configuration in Service Configuration Studio.

To configure and use Data Modelling:

1. Open the Service Configuration Studio.
2. Click on the **Data Modelling** entry.
3. Open the desired tab:
   - **Tenants**: Creation and administration of the tenants.
   - **Construction Kit Management**: Administration of **Construction Kit Libraries**.
   - **Type Explorer**: Overview of all **Types** of installed **Construction Kit Libararies**.

- **GraphQL Editor**: Starts the **GraphQL Editor** (on page 98).



Tenant Management

Manage Tenants and set the globally active Tenant

Tenant	Database
zenon (active Tenant)	dm_zenon

Items per page: 10 1 – 1 of 1

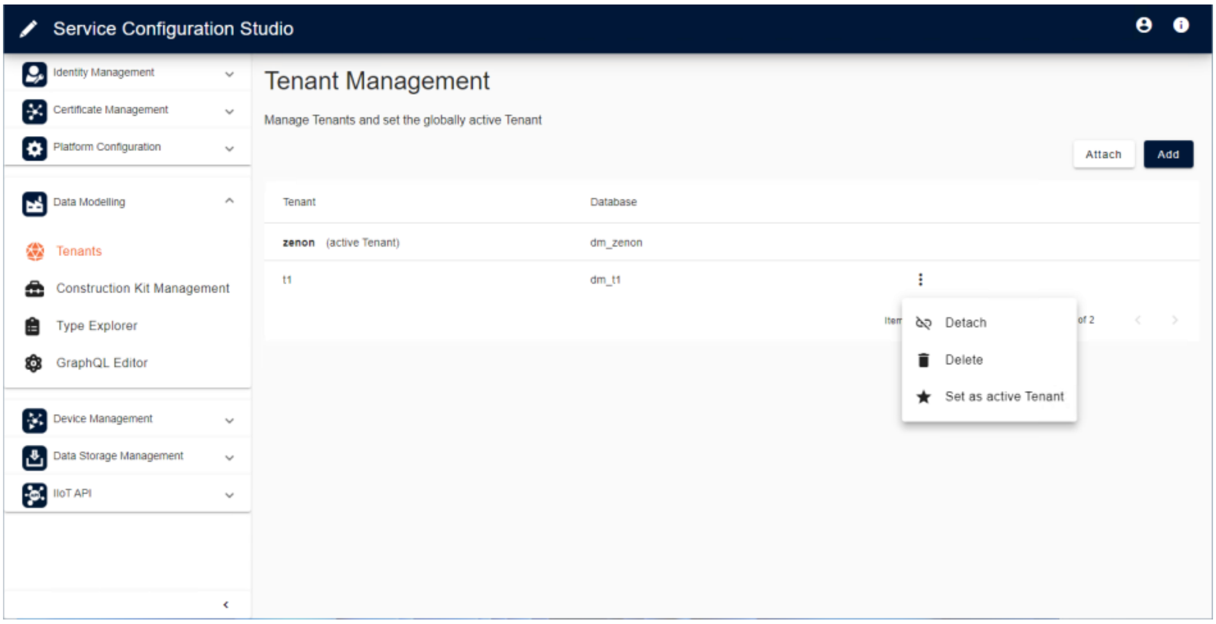
9.5.1 Tenants

You administer and configure **Tenants** here. A **Tenant** (mandant) is a logical separation (instance) of all the data that concerns Data Modelling. It comprises all metadata and data in a **Construction Kit**.

Each instance of the **Data Modelling Service** supports several **Tenants**. The **Tenant** "*zenon*", created by the system and available from the start, is used by default by all applications that access Data Modelling.

Recommendation: Use the default **Tenant** in the current version.

The selected (active) **Tenant** determines the **Tenant** in which all operations in the user interface of Data Modelling take place.



Tenant Management

Manage Tenants and set the globally active Tenant

Tenant	Database
zenon (active Tenant)	dm_zenon
t1	dm_t1

CREATE TENANT

<u>To create a</u> **Tenant**<u>:</u>

1. Click on the **Add** button.

   The dialog to enter the ID is displayed.

2. Enter the desired **Tenant ID**.

   Rules:

   - Must start with a letter.
   - Must only contain letters and numbers.
   - Maximum length 50 characters.

3. Click on the **Save** button.

4. The new **Tenant** is saved and displayed.

ATTACH TENANT

<u>To attach a</u> **Tenant**<u>:</u>

1. Click on the **Attach** button.
   The dialog for the selection of a **Tenant ID** is shown. Pre-existing **Tenants** without a connection are listed.

2. Select the desired **Tenant** or enter the desired **Tenant ID**.

3. Click on the **Save** button.
4. The **Tenant** is attached and displayed.

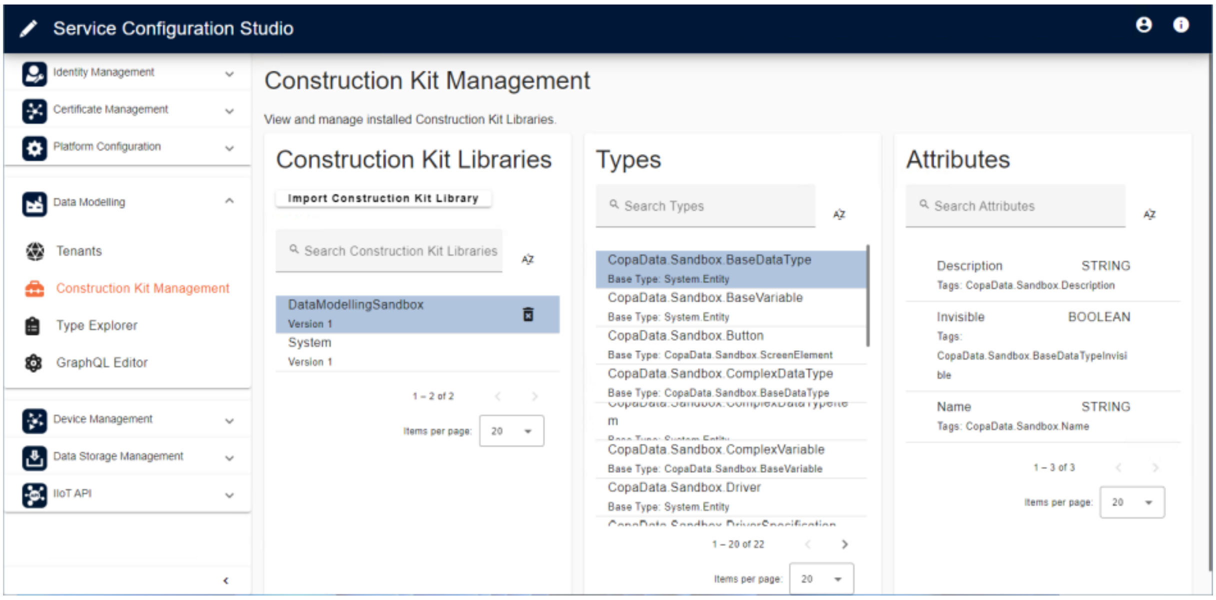
ADMINISTER TENANT

To administer a Tenant:

1. Click on the **Tenant**.
2. Click on the button with the three dots. This is only available for **Tenants** that are not active. The context menu is displayed.
3. Select the desired action:
   - **Detach**: The **Tenant** is detached and removed from the list.
   - **Delete**: The **Tenant** is deleted from the system.
   - **Set as active tenant**: The **Tenant** is set as an active **Tenant**. It is always only one **Tenant** that can take on this role.

9.5.2 Construction Kit Management

You manage your **Construction Kit Libraries** for the selected **Tenant** in this tab. Display of the **Types** and **Attributes** of the selected **Construction Kit Library**.


Service Configuration Studio
Identity Management
Certificate Management
Platform Configuration
Data Modelling
Tenants
Construction Kit Management
Type Explorer
GraphQL Editor
Device Management
Data Storage Management
IIoT API
Construction Kit Management
View and manage installed Construction Kit Libraries.
Construction Kit Libraries
Import Construction Kit Library
Search Construction Kit Libraries
DataModellingSandbox
Version 1
System
Version 1
1 – 2 of 2
Items per page: 20
Types
Search Types
CopaData.Sandbox.BaseDataType
Base Type: System.Entity
CopaData.Sandbox.BaseVariable
Base Type: System.Entity
CopaData.Sandbox.Button
Base Type: CopaData.Sandbox.ScreenElement
CopaData.Sandbox.ComplexDataType
Base Type: CopaData.Sandbox.BaseDataType
CopaData.Sandbox.ComplexVariable
Base Type: CopaData.Sandbox.BaseVariable
CopaData.Sandbox.Driver
Base Type: System.Entity
1 – 20 of 22
Items per page: 20
Attributes
Search Attributes
Description STRING
Tags: CopaData.Sandbox.Description
Invisible BOOLEAN
Tags:
CopaData.Sandbox.BaseDataTypeInvisible
Name STRING
Tags: CopaData.Sandbox.Name
1 – 3 of 3
Items per page: 20

CONSTRUCTION KIT LIBRARY MANAGEMENT DIALOG

Administration and display of the installed **Construction Kit Libraries**.

Column	Description
Construction Kit Libraries	Display of the installed **Construction Kit Libraries**. **<u>Administration of the display:</u>** - To filter the display: Enter the desired term into the text field. - To sort: Click on the **AZ** symbol. **<u>To add a library:</u>** - Click on the **Import Construction Kit Library** button. - Select the desired **Library**. - Click on the **Import** button. **<u>To remove a library:</u>** - Select the desired **Library**. Only **CKLs** that are not based on further **CKLs** can be deleted. - Click on the Trashcan symbol. **Attention:** The deletion of a **Construktion Kit Library** is a significant action. The scheme of the **Construction Kit Library** and all data contained in this scheme is deleted irreversibly.
Types	Shows all **Types** that are present in the selected **Construction Kit Library**.
Attributes	Shows all **Attributes** that have been configured in the selected **Type**.

9.5.3 Type Explorer

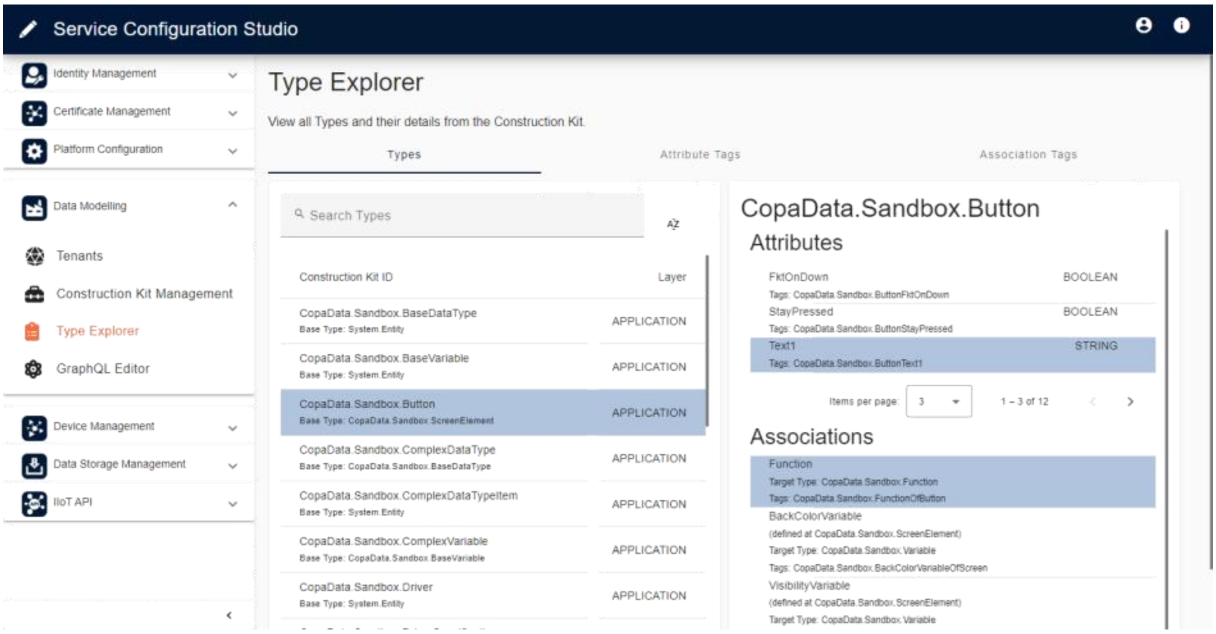
View of the **Types** and their details for all installed **Construction Kit Libraries** of the selected **Tenant**. Details are shown for:

- **Types**
- **Attribute Tags**
- **Association Tags**

The display can be filtered and sorted alphabetically.

TYPES

Display of all **Types** and the attendant linked **Attributes** and **Associations**.



Type Explorer

View all Types and their details from the Construction Kit.

Construction Kit ID	Layer
CopaData.Sandbox.BaseDataType Base Type: System.Entity	APPLICATION
CopaData.Sandbox.BaseVariable Base Type: System.Entity	APPLICATION
CopaData.Sandbox.Button Base Type: CopaData.Sandbox.ScreenElement	APPLICATION
CopaData.Sandbox.ComplexDataType Base Type: CopaData.Sandbox.BaseDataType	APPLICATION
CopaData.Sandbox.ComplexDataTypeItem Base Type: System.Entity	APPLICATION
CopaData.Sandbox.ComplexVariable Base Type: CopaData.Sandbox.BaseVariable	APPLICATION
CopaData.Sandbox.Driver Base Type: System.Entity	APPLICATION

CopaData.Sandbox.Button

Attributes

FktOnDown Tags: CopaData.Sandbox.ButtonFktOnDown	BOOLEAN
StayPressed Tags: CopaData.Sandbox.ButtonStayPressed	BOOLEAN
Text1 Tags: CopaData.Sandbox.ButtonText1	STRING

Items per page: 3 1 – 3 of 12

Associations

Function
Target Type: CopaData.Sandbox.Function
Tags: CopaData.Sandbox.FunctionOfButton

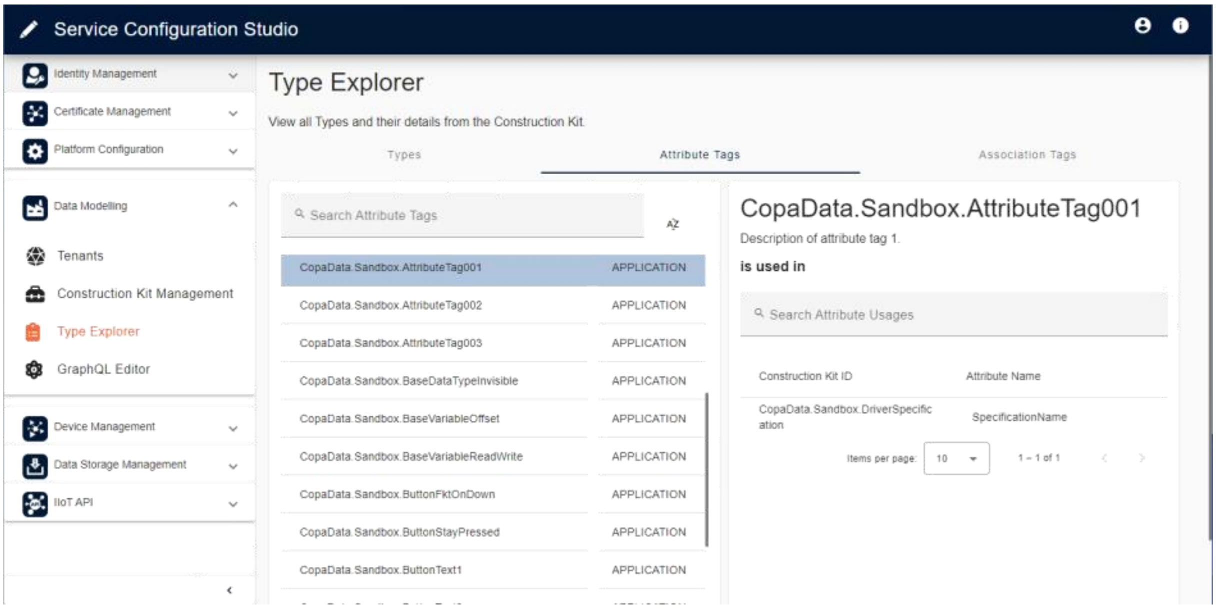
BackColorVariable
(defined at CopaData.Sandbox.ScreenElement)
Target Type: CopaData.Sandbox.Variable
Tags: CopaData.Sandbox.BackColorVariableOfScreen

VisibilityVariable
(defined at CopaData.Sandbox.ScreenElement)
Target Type: CopaData.Sandbox.Variable

Select a **Type** in order to display the respective linked **Attributes** and **Associations**.

ATTRIBUTE TAGS

Overview of where **Attribute Tags** are used.



Type Explorer

View all Types and their details from the Construction Kit.

CopaData.Sandbox.AttributeTag001	APPLICATION
CopaData.Sandbox.AttributeTag002	APPLICATION
CopaData.Sandbox.AttributeTag003	APPLICATION
CopaData.Sandbox.BaseDataTypeInvisible	APPLICATION
CopaData.Sandbox.BaseVariableOffset	APPLICATION
CopaData.Sandbox.BaseVariableReadWrite	APPLICATION
CopaData.Sandbox.ButtonFktOnDown	APPLICATION
CopaData.Sandbox.ButtonStayPressed	APPLICATION
CopaData.Sandbox.ButtonText1	APPLICATION

CopaData.Sandbox.AttributeTag001

Description of attribute tag 1.

is used in

Search Attribute Usages

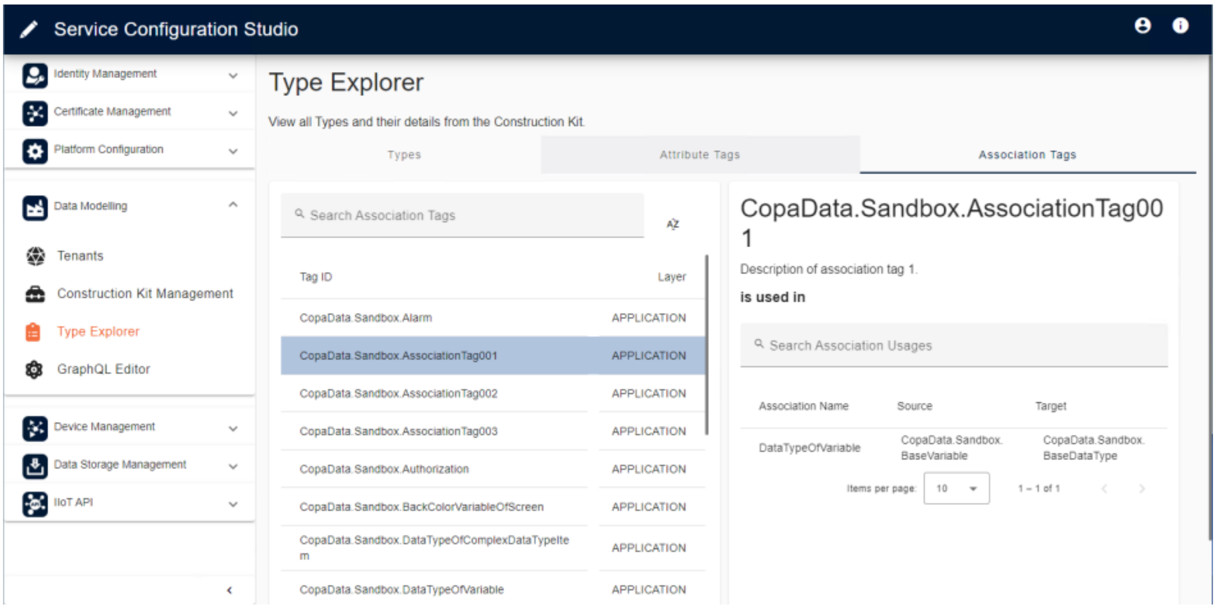
Construction Kit ID	Attribute Name
CopaData.Sandbox.DriverSpecification	SpecificationName

Items per page: 10 1 – 1 of 1

Select an **Attribute Tag** in order to display where it is used.

ASSOCIATION TAGS

Overview of where **Association Tags** are used.



Type Explorer

View all Types and their details from the Construction Kit.

Tag ID	Layer
CopaData.Sandbox.Alarm	APPLICATION
CopaData.Sandbox.AssociationTag001	APPLICATION
CopaData.Sandbox.AssociationTag002	APPLICATION
CopaData.Sandbox.AssociationTag003	APPLICATION
CopaData.Sandbox.Authorization	APPLICATION
CopaData.Sandbox.BackColorVariableOfScreen	APPLICATION
CopaData.Sandbox.DataTypeOfComplexDataTypeItem	APPLICATION
CopaData.Sandbox.DataTypeOfVariable	APPLICATION

CopaData.Sandbox.AssociationTag001

Description of association tag 1.

is used in

Association Name	Source	Target
DataTypeOfVariable	CopaData.Sandbox.BaseVariable	CopaData.Sandbox.BaseDataType

Select an **Association Tag** in order to display where it is used.

9.5.4 GraphQL Editor

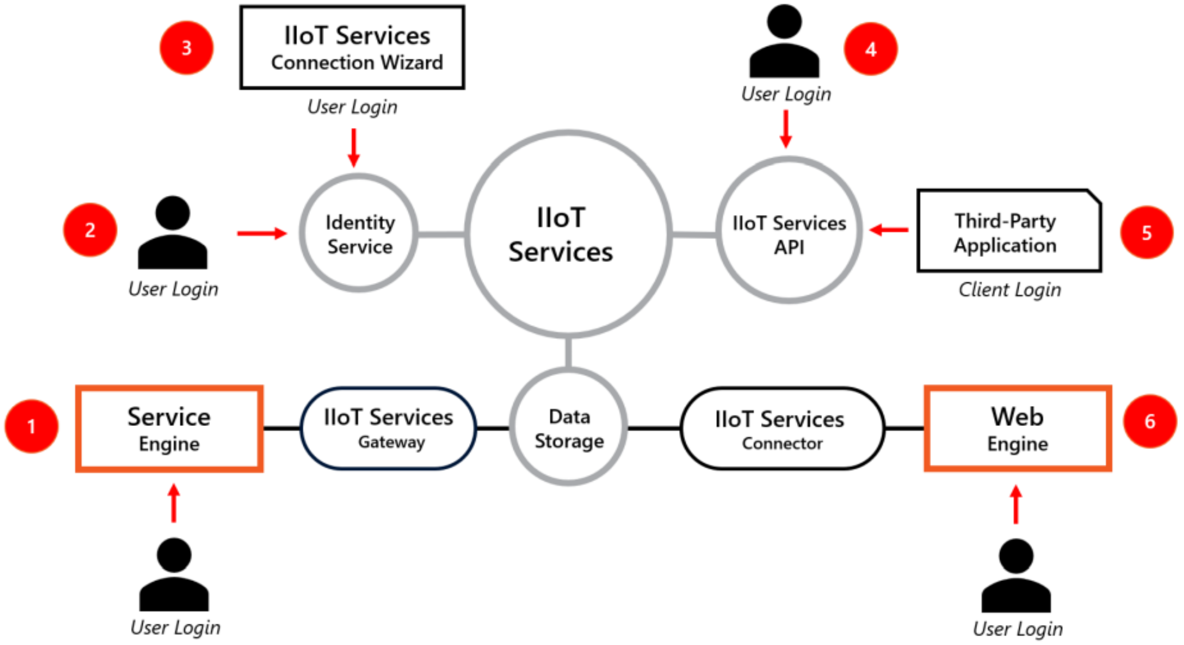
Click on the tab to start the **GraphQL Editor** for the selected **Tenant** in a new window or a new tab. Queries, **Mutations** and **Subscriptions** can be created with the editor.

A **GraphQL scheme** contains **Queries** and **Mutations** for all **Types** and metadata of the **Construction Kit**. The **GraphQL scheme** can be viewed using the **GraphQL Editor**. In the **GraphQL Editor**, you can find the accompanying **GraphQL scheme** documentation in **Docs** on the right border.

Information

the **GraphQL Editor** corresponds to the **GraphQL Interface** of the Report Engine in terms of appearance and design of the user interface. However, both differ in their functionality. Both use a **GraphQL Playground**.

10 Identity Service: Central authentication service



The Identity Service supports client/user logins for IIoT Services and applications connected with it.

The **Identity Service** is the central authentication service of IIoT Services. In addition, **Identity Service** also supports the authentication of selected zenon applications as well as third-party applications.

AREA OF APPLICATION

Logged-in users or clients can access IIoT Services resources and applications connected with it.

The resources that are available depend on the respective application:

Login to:	Login type	Input of user credentials	Available resources	Two-factor authentication
1. Service Engine	User login	Manual input via the GUI of Service Engine on the local computer of Service Engine.	The resources of Service Engine.	Not supported.
2. Identity Service web interface	User login	Manual input via browser in **Identity Service** web interface.	All web interfaces of IIoT Services.	Supported.
3. Service Engine Connection Wizard	Identity Service web interface	Manual input via browser in **Identity Service** web interface.		Supported. Applies to **Web login (Service Grid 10.4 and later)**
4. IIoT API (as the user in test operation)	User login	<u>**Two-step authorization:**</u> 1. Manual input via browser in **Identity Service** web interface. 2. Authorization via the **Authorize** button in the web interface of the IIoT API.	All resources in IIoT Services accessible for the IIoT API.	Supported.
5. IIoT API	Client login	Automated client login of an	All resources in IIoT Services	Not supported.

Login to:	Login type	Input of user credentials	Available resources	Two-factor authentication
(as the client in productive operation)		appropriately-configured third-party application to the IIoT API.	accessible for the IIoT API.	**Note:** As a matter of principle, two-factor authentication is not suitable for client logins.
6. Web Engine	User login	Manual input via browser in Web Engine. Network access is sufficient. Local login on the computer of Web Engine is not necessary.	The resources of Service Engine that are connected to the web engine.	Not supported.

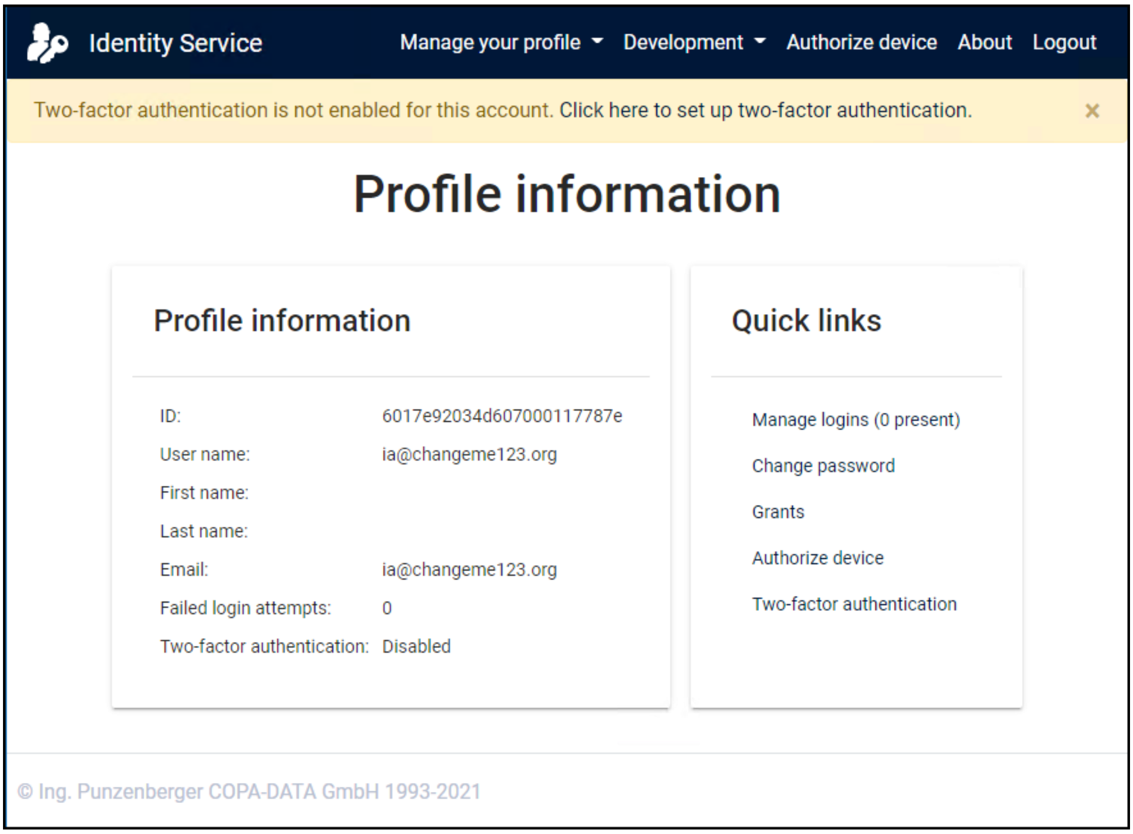
10.1 External identity providers

Identity Service supports the integration of external identity providers such as RADIUS or Azure AD.

Please note the following when using external identity providers:

- There are functional differences between the various providers. You can find more detailed information in relation to this in the **Identity Providers (for external logins)** (on page 146) section.
- Service Engine has more restrictive requirements on user names than **Identity Service**. You can find detailed information in relation to this in the **IIoT Services - configuration in Engineering Studio** section in the chapter **Compatibility table: User names**.

10.2 Identity Service



Two-factor authentication is not enabled for this account. Click here to set up two-factor authentication.

Profile information

Profile information

ID:	6017e92034d607000117787e
User name:	ia@changeme123.org
First name:	
Last name:	
Email:	ia@changeme123.org
Failed login attempts:	0
Two-factor authentication:	Disabled

Quick links

- Manage logins (0 present)
- Change password
- Grants
- Authorize device
- Two-factor authentication

© Ing. Punzenberger COPA-DATA GmbH 1993-2021

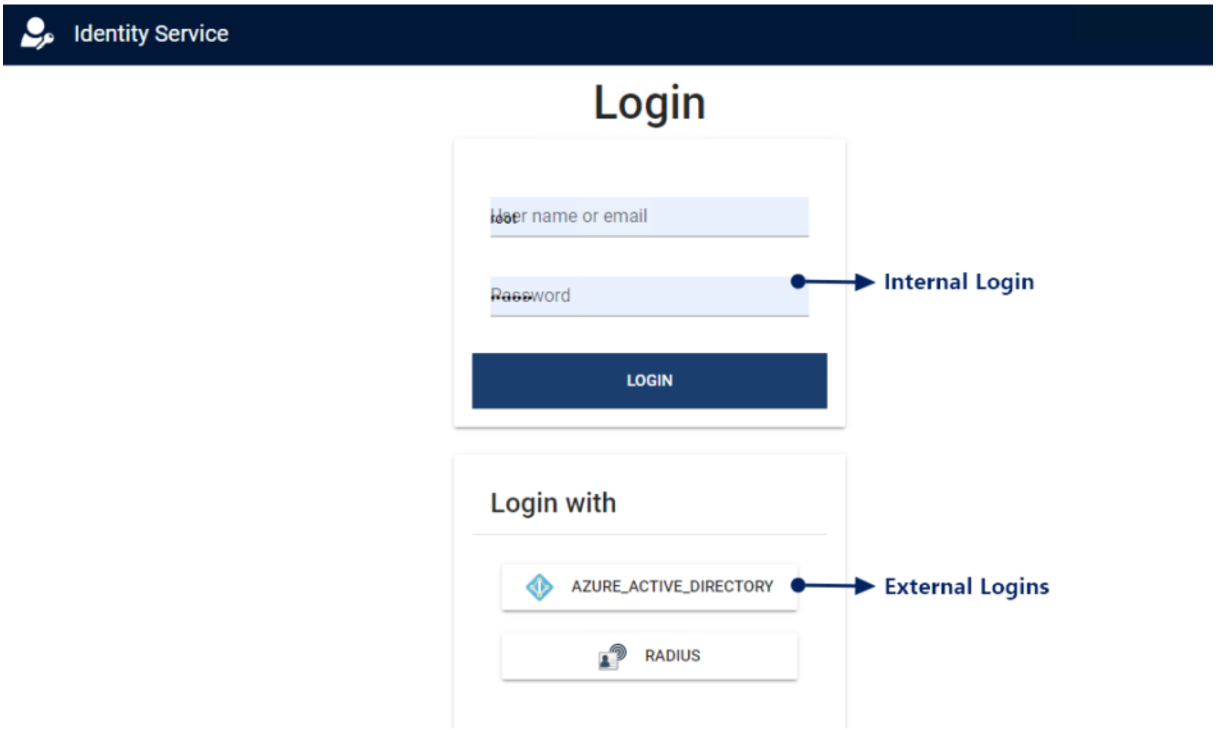
With the Identity Service in Service Configuration Studio, every logged-in user can change settings for their own account.

The **Identity Service** is the service for the central authentication of users and applications in IIoT Services.

The **Identity Service** in Service Configuration Studio supports the following functionality:

- Every user of IIoT Services can log in.
- Every user can make basic settings **for their own user account**.

10.2.1 Login



By default, users log in via an internal login in the Identity Service in Service Configuration Studio. You can also optionally configure the connection with external logins.

Users can use the login in the web interface of **Identity Service** to centrally authenticate themselves for all web interfaces of IIoT Services.

A distinction needs to be made between the login options:

- Internal login: Direct log in to a user account in **Identity Service**.
- External logins: Log in via an external **Identity Provider**.

Each time users log in, they can choose between all login options with which their user account in **Identity Service** is linked at the time of logging in. The differences between internal login and external login are documented separately (on page 106).

PLATFORM USER FOR ZENON

You can use the user authentication via **Identity Service** not only for IIoT Services but also for other services of the zenon software platform. You can find details in relation to this in the **Identity Service: central authentication service** (on page 99) section.

Attention

Lock after failed logins

Five consecutive failed attempts to log in to a user account will result in **Identity Service** blocking the respective user account for security reasons.

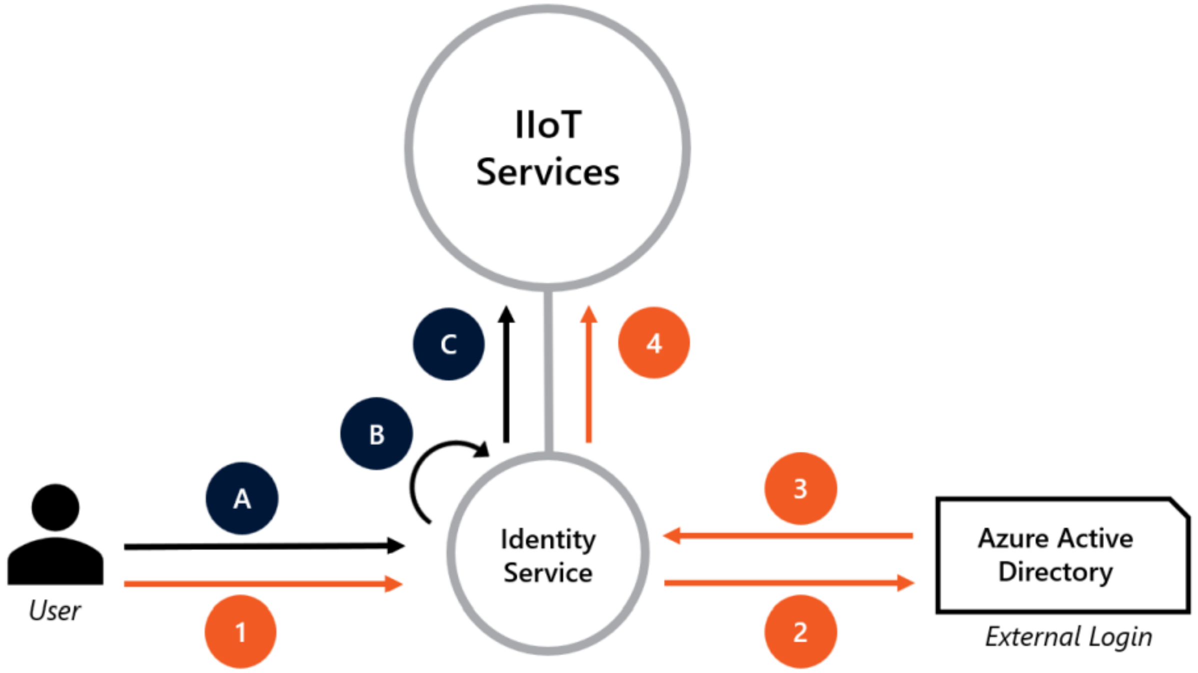
Each failed attempt to log in will be counted. The block is only temporary.

Attempts to log in include for example:

- Entering a password
- Two-factor authentication via an authenticator app
- Entering a recovery code for two-factor authentication

The temporary block of a user account lasts 5 minutes. After this time has elapsed, it is possible to log in again.

10.2.1.1 Logins: Internal vs. External



In the case of internal login (A,B,C), users log in directly to Identity Service. In the case of external login (1,2,3,4), users log in to Identity Service via an external identity provider.

Identity Service supports several user login options. The table below contains definitions and short names of the different login options.

Short name	Description	Administration
Internal login	Direct login to **Identity Service** (without external login). The integrated user administration of **Identity Service** verifies the user's login credentials. This is the default setting in **Identity Service**.	In Service Configuration Studio in the **Identity Management** node via the following menu item: ▸ **Users** These are the **internal user accounts** of **Identity Service**. **Note:** Each external login is assigned an internal user account. These assigned user accounts are also displayed here.
External login	Log in to **Identity Service** via an external login. An external **Identity Provider** (for instance, **Azure Active Directory**) verifies the user's login credentials for **Identity Service**. External logins must be configured manually.	Using the administration tools of the respective **Identity Provider**. **Example:** User accounts in **Azure Active Directory** are administered via the **Azure Portal**. From the point of view of **Identity Service**, these are **external user accounts**.
Login	If the login to **Identity Service** is not specified further, it always refers to both options (**internal login** and **external login**).	See specific logins.

Hint:

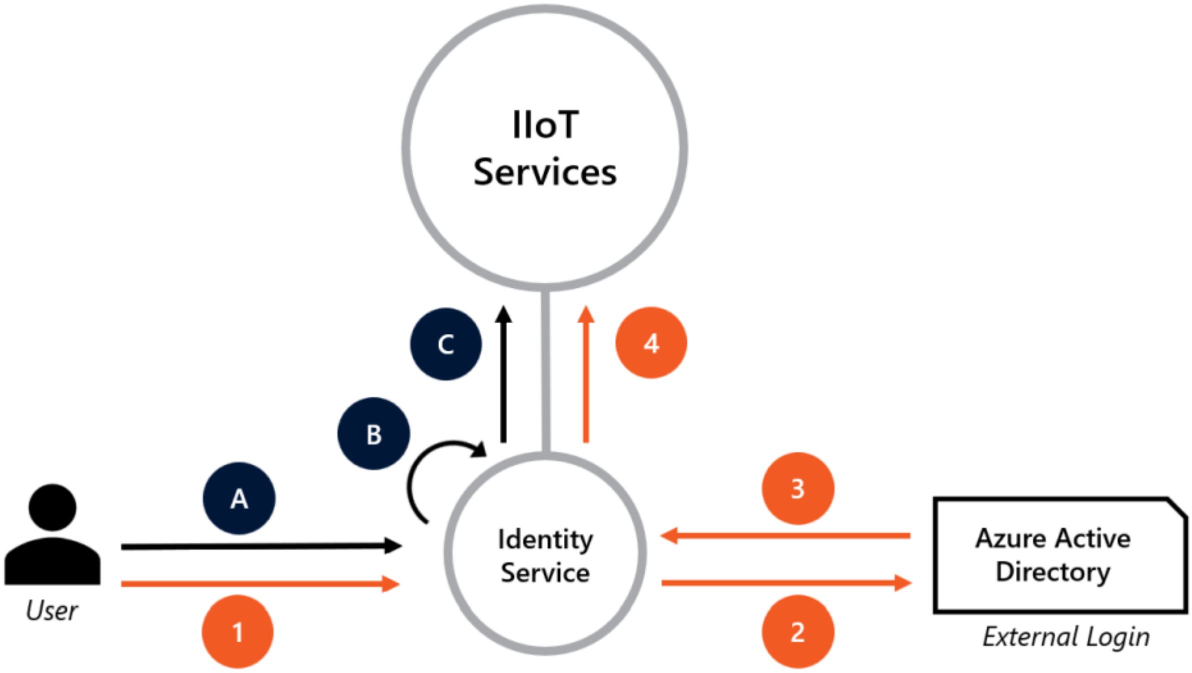
Login processes in detail

The login processes to **Identity Service** are documented in detail for each login option:

- External login (on page 146): Login via external identity provider (for instance, Microsoft Active Directory)
- Internal login (on page 127): Direct login (without external login)

Each time users log in, they can choose between all login options with which their user account in **Identity Service** is linked at the time of logging in.

10.2.1.2 External login



Every user that logs in via an external login (1,2,3,4) can alternatively also log in via an internal login (A,B,C) to Identity Service.

Logging in to the web interface of **Identity Service** with the user account of an external **Identity Provider** requires that the **external user account** is linked with an **internal user account**. There are two possibilities when using **External logins**.

- The user already has an account in the Identity Service:
  The user can **manually** link their internal user account with **External logins** (on page 112). External logins are provided by the Administrator.
- The user does not have an account in Identity Service yet:
  An internal user is **automatically** created and linked in **Identity Service** when logging in via an external **Identity Provider** for the first time. The user can only log in by default via **External login** with the user's login credentials. However, users can use **Change password** in **Identity Service** at any time to create access themselves for logging in directly to **Identity Service**.

Every user who logs in via an external login can alternatively log in via an internal login (or create this access themselves at any time).

> ⚠ **Attention**
>
> **External login: Correct way to block user access**
>
> To block a user account for an **external login** for **Identity Service**, you must deactivate the user account in **Identity Management**.
>
> It is not enough just to deactivate the account via the external **Identity Provider**.

10.2.1.3 User login via external login

Carry out the following steps to log in to an external provider:

1. Open the login page in the **Identity Service**.
   To do this, enter the corresponding URL in a web browser:
   For example *https://mycomputer.testenv.local:9443/identity-service/*
2. Click on the button of the desired identity provider.
3. You will then be redirected in the web interface to the login page of the external identity provider.
   Note: The method of redirection depends on the identity provider selected (see the table).
4. Authenticate yourself by entering your user credentials.
5. You are sent back to **Identity Service**.

REDIRECTION FOR EXTERNAL LOGINS

The type of forwarding on login and the entry of user credentials depends on the respective identity provider.

Identity provider	Redirection	User credentials
Azure Active Directory	To external website of **Azure Active Directory**: *https://login.microsoftonline.com*	Input to an external web page of the identity provider.
Microsoft Active Directory	To internal website of **Identity Service**.	Input to an internal web page of the Identity Service.
OpenLDAP	To internal website of **Identity Service**.	Input to an internal web page of the Identity Service.
RADIUS	To internal website of **Identity Service**.	Input to an internal web page of the Identity Service.
OpenID Connect	To external website of **OpenID Connect**: *https://myopenidconnect.com*	Input to an external web page of the identity provider.
Keycloak	To external website of **Keycloak**: *https://my-keycloak.com/auth/realms/master*	Input to an external web page of the identity provider.

10.2.2 User role: Identity Administrator

ROLE DEFINITION

The *Identity Administrator* is the only user role that can log into both the **Identity Service** (as a normal user) and **Identity Management** (as an administrator). This user role can also be assigned to several users. An Identity Administrator can administrate, in **Identity Management**, all settings that concern other **User**s, **Groups**, **Clients** and external **Identity Provider**s.

INITIAL CONFIGURATION

The initial user account in **Identity Service** is automatically assigned the role of *Identity Administrator*.

Proceed as follows:

1. Open the web interface for **Identity Service**.
2. If a user has not been defined yet, open an input screen where you can define a user.

3. Enter the desired **User name**.
4. Enter the desired password under **New Password**.
   Important: The password must meet the minimum password requirements.
5. Confirm the selected password under **Confirm new password**.
6. Close the process by clicking on the **Create** button.

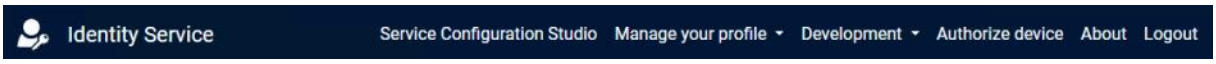
You have thus created the initial user account in **Identity Service**.

> **Tip**
>
> You can further configure the user account you created in **Identity Management** under **Users**.
>
> There you can also configure the following data fields:
>
> - **First name**
> - **Last name**
> - **Email**
> - **Description**
>
> You can overwrite existing default settings.

10.2.3Identity Service - user interface

In the web-based interface, information and configurations for the last logged-in user are provided.

HEADER

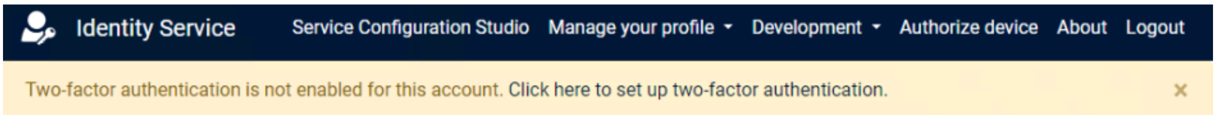


In the header, there are linkings for the administration of functionality of the logged-in user profile for the logged-in user. In addition, under the navigation bar, the activation of **Two-factor authentication** is offered if this has not already been activated.

Option	Description
Service Configuration Studio	Linking to Service Configuration Studio for the administration of all installed IIoT Services. Login to Service Configuration Studio is carried out with the same user that is already logged in to the Identity Service.

Option	Description
Manage your profile (on page 111)	User interface for the administration of the logged-in user's own user profile. Select from drop-down list: - *Profile information* Visualizes the information about the current logged-in user. In addition, links to further configuration options are offered. - *External logins* Shows an overview of the external logins for the logged-in user. In addition, the external logins are visualized with a link. - *Change password* Opens the dialog to change the password for the current logged-in user. - Application grants - *Two-factor authentication* Opens the configuration to set parameters for two-factor authentication for the logged-in user.
Development	Select from drop-down list: - *API documentation* Opens the **Identity Service** API documentation. - *Diagnostics* Opens a view with technical information for the logged-in user. - *OAuth2 discovery document* Opens a new tab in the web browser with information for OAuth authentication.
Authorize device (on page 125)	Opens the dialog for authentication on devices or software components that do not have a graphical user interface.
About	Shows the license status and the version of **Identity Service**.
Logout	Logs out the current logged-in user. The login page is displayed in the browser.

CURRENT STATE OF TWO-FACTOR AUTHENTICATION



Two-factor authentication is not enabled for this account. Click here to set up two-factor authentication.

This view shows the information under the menu bar if two-factor authentication has been deactivated. Click on the text "Click here to set up two-factor authentication" to activate two-factor authentication and the corresponding configuration is shown.

The display of the notice is deactivated by clicking on the **X** button.

10.2.3.1 Manage your profile

In this menu item, the logged in user can manage their own user profile.

10.2.3.1.1 Profile information

Displays information about the user currently logged in

PROFILE INFORMATION

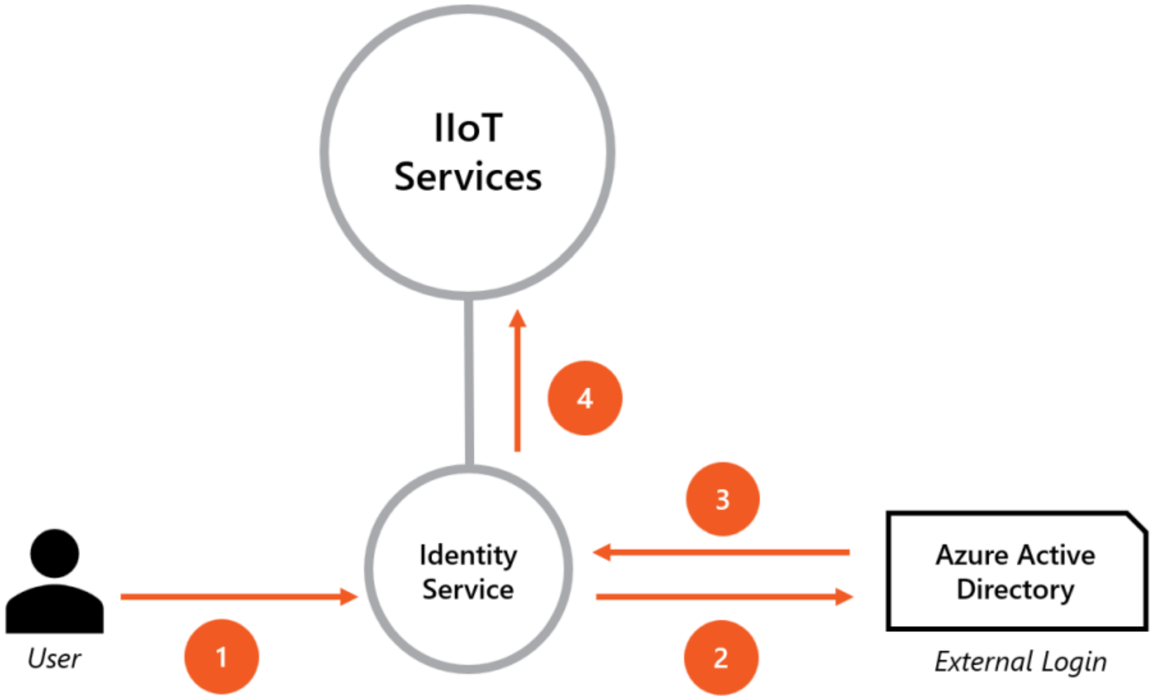
Option	Description
ID	Unique ID of the current logged-in user. This is automatically issued when the user is created.
User name	Configured user name of the logged-in user.
First name	The first name of the user.
Last Name	The last name of the user.
Email	The email address of the user.
Failed login attempts	Number of failed login attempts for the current logged-in user.
Two-factor authentication	Information about the type of two-factor authentication for the logged-in user.

QUICK LINKS

Option	Description
Manage logins (n present)	Shows an overview of the external logins for the logged-in user. In addition, the external logins are visualized with a link.

Option	Description
Change password	Opens the dialog to change the password for the current logged-in user.
Grants	
Authorize device	Opens the dialog for the authentication of devices. This is necessary if the authentication of devices is not possible on the client directly.
Two-factor authentication	Opens the configuration to set parameters for two-factor authentication for the logged-in user.

10.2.3.1.2 External logins



In the case of external login, users log in to Identity Service via a user account of an external identity provider.

Here users can link their existing user account from **Identity Service** with an external login.

EXAMPLE: USE DOMAIN USER ACCOUNT FOR IDENTITY SERVICE

<u>Users already have two different user accounts:</u>

- A user account in **Identity Service**. Users use this user account to log in to IIoT Services.
- A domain user account in **Microsoft Active Directory**. Users use this account to log in to their workstation in the domain.

Problem:
Users must remember separate user names and separate passwords for IIoT Services and the domains.

Solution:
Users link their **Identity Service** user account with their user account from **Microsoft Active Directory**. This way users can log in to both systems with the User Credentials from **Microsoft Active Directory**

LINKING OF EXTERNAL LOGINS

The following applies for the linking of external logins with an existing user account:

- You as the user can only link external logins if the respective**Identity Provider**s have already been configured in **Identity Management** (on page 146) by the administrator.
- Each user also needs their own user account in **Identity Service**.
- Despite an existing link to an external login, you can still log in via the internal login. During each login, you have the choice to log in via the internal login or an external login.

A link to an external login can be removed at any time. The internal login is nevertheless always possible.

10.2.3.1.3 Change password (for internal login only)

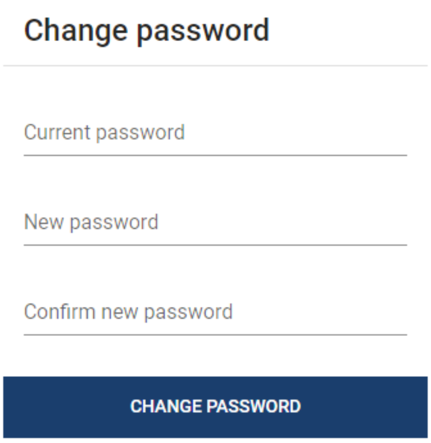
Change password allows users to change the password for their user account in **Identity Service** (internal login).

- A password change in **Identity Service** does not affect the user account of an external identity provider (external login).
- If you are logged in via an external login, you can use a password change in **Identity Service** to gain access via an internal login (provided it does not exist yet) (on page 106).

Passwords for external logins can only be changed directly via the external **Identity Provider**.

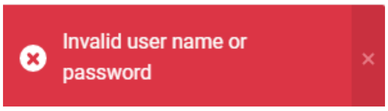
CHANGE PASSWORD

In this dialog, you change the password for the current logged-in user.



Option	Description
Current password	The current password for the user.
New password	The new password for the user.
Donfirm new password	Confirmation of the new password, as already entered in the **New password** option.

Incorrect entries are visualized with a corresponding warning notice in red.



Invalid user name or password

Information

Password changes always required the minimum requirements for a password (on page 178).

10.2.3.1.4 Application grants

Here, the user can find an overview of the applications with which they are logged into IIoT Services. This is the case, for example, if external client applications are connected via OAuth 2.0. These authorizations can be removed by the user at any time by clicking on the **Revoke Access** button.

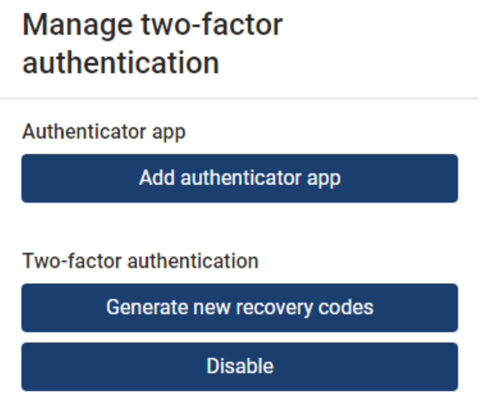
10.2.3.1.5 Two-factor authentication

The optional two-factor authentication increases the security of a user login to **Identity Service**. The combination of two independent factors effectively secures login processes.

TWO-FACTOR AUTHENTICATION WITH IDENTITY SERVICE

The following applies for two-factor authentication in the Identity Service:

- It applies for all login procedures to **Identity Service**.
- It doesn't matter whether the user logs in via an internal login or an external login (on page 104).
- Each user configures the two-factor authentication for their own user account in the web interface of **Identity Service**.
- The administrator can disable two-factor authentication for individual users (on page 128) in **Identity Management**.
- Configuration of two-factor authentication is only possible if two-factor authentication has also been activated for the user.


Manage two-factor authentication

Authenticator app

Two-factor authentication

Configuration dialog with two-factor authentication activated.

Before you activate two-factor authentication in the Identity Service, you must check the compatibility for the planned use case (on page 117).

TWO-FACTOR AUTHENTICATION WITH AN EXTERNAL IDENTITY PROVIDER

Some external identity providers offer their own two-factor authentication. This is completely independent of **Identity Service**. The two-factor authentication of an external identity provider is configured in the administration tools of the external identity provider.

Tip

Client logins work with Client ID and Secret and are usually automated. Manual entry of login credentials by a human user is not expected here. Therefore, client logins do not support two-factor authentication.

10.2.3.1.6 Terminology

The following terms are used in this chapter:

Term	Description	Practical application
Authenticator App	An app for two-factor authentication that is installed on an authenticator device.	This is usually a mobile device app. Suitable apps are, for example, Microsoft Authenticator (Windows Phone, Android, iOS) or Google Authenticator (Android, iOS).
Authenticator Device	The device on which the authenticator app is installed.	This is usually a mobile device, generally a smartphone.
Authenticator Key	This static key is generated in **Identity Service** and is linked long term to a specific user account in **Identity Service**. All verification keys generated by the authenticator app are based on this authenticator key.	The authenticator key is transferred from the web interface of **Identity Service** to the authenticator app via a QR code or manually. You need this key to configure a user account in the authenticator app of a mobile device.
Recovery Code	If two-factor authentication is enabled, you can log in using your user credentials and a recovery code. Recovery codes are only used if you do not have access to an authenticator device. It is a fallback mechanism.	Recovery codes are automatically generated in various configurations of **Identity Service** and are then displayed once. ▶ Recovery codes cannot be displayed again afterwards. ▶ Each recovery code can only be used once.
Verification Code	Verification codes are one-time passwords. They are generated	You need verification codes for different scenarios:

Term	Description	Practical application
	dynamically in the authenticator app and are valid each time for only 30 seconds. All verification codes are based on an authenticator key.	- First-time enabling of two-factor authentication - Re-enabling of a disabled two-factor authentication - Adding an additional authenticator app - Each user login (if two-factor authentication has been enabled)

10.2.3.1.7 Use of two-factor authentication

Two-factor authentication is only supported for certain use cases in IIoT Services.

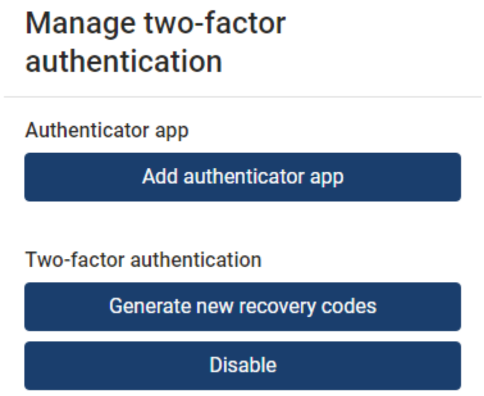
You should therefore only use two-factor authentication for a user account if the use case you require is compatible with two-factor authentication. This must be checked for each user account and for each use case.

Important restrictions of two-factor authentication: Users cannot log in to Service Engine.

You can find all information on compatibility of the different login variants in the **Identity Service: central authentication service** (on page 99) chapter.

10.2.3.1.8 Manage two-factor authentication

You configure two-factor authentication in this dialog.


Manage two-factor authentication
Authenticator app
Add authenticator app
Two-factor authentication
Generate new recovery codes
Disable

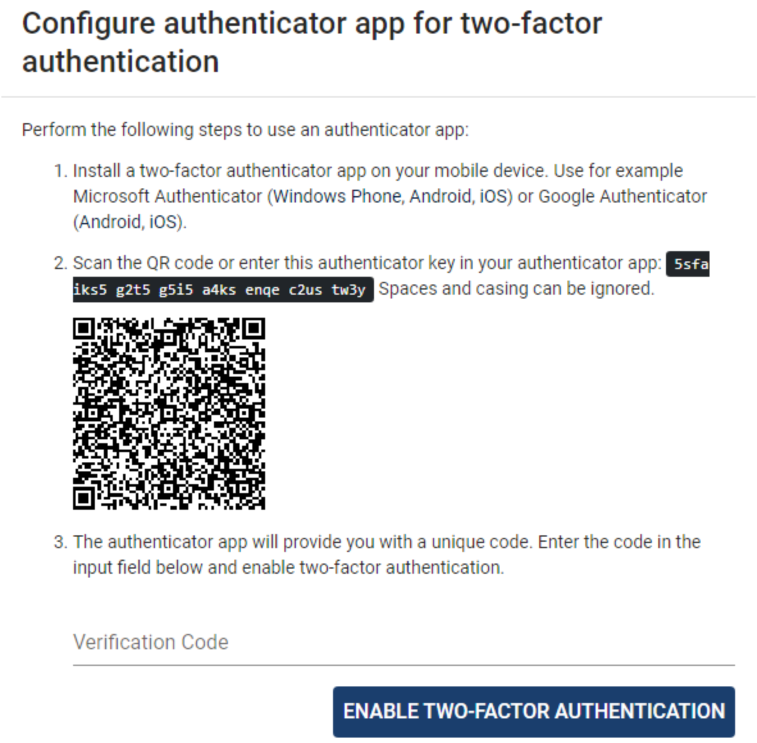
Option	Description
Add authenticator app	Opens the dialog to set up an additional two-factor authenticator app.
Generate new recovery codes	Generates new recovery codes for a user account in **Identity Service**. Previous recovery codes are thus invalid. **Note:** Only visible if two-factor authentication has been activated.
Disable	Deactivates two-factor authentication. At the same time, the earlier authenticator key loses its validity.

Enable Two-factor authenticaton

The setup of the authenticator app comprises the following actions:

- It automatically generates a new authenticator key.
- The authenticator key is registered in an authenticator app.
- Two-factor authentication is enabled for the **Identity Service** user account.

SETTING PARAMETERS



Configure authenticator app for two-factor authentication

Perform the following steps to use an authenticator app:

1. Install a two-factor authenticator app on your mobile device. Use for example Microsoft Authenticator (Windows Phone, Android, iOS) or Google Authenticator (Android, iOS).
2. Scan the QR code or enter this authenticator key in your authenticator app: `5sfa iks5 g2t5 g5i5 a4ks enqe c2us tw3y` Spaces and casing can be ignored.
3. The authenticator app will provide you with a unique code. Enter the code in the input field below and enable two-factor authentication.

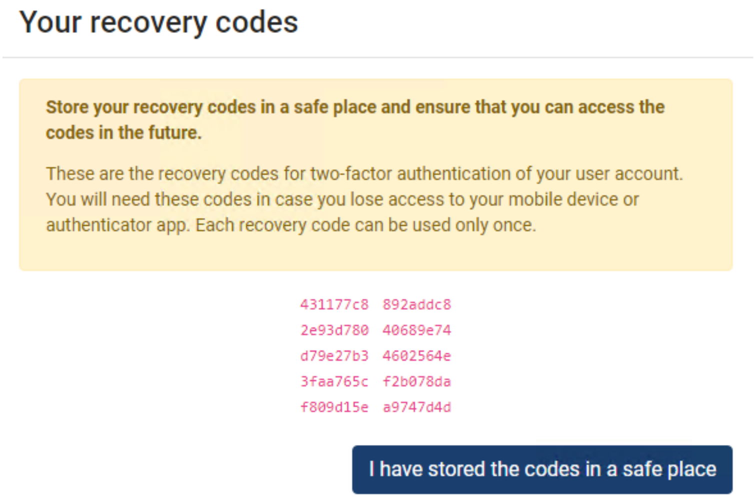
Carry out the following steps to activate two-factor authentication:

1. Go in the web interface of **Identity Service** to the **Manage your profile\Two-factor authentication.** subpage
2. Click on the **Enable Two-factor authentication** button. This opens a configuration dialog.
3. Install an authenticator app on your mobile device. Suitable apps are, for example, Microsoft Authenticator (Windows Phone, Android, iOS) or Google Authenticator (Android, iOS).
4. Scan the QR code in the web interface using the app. Alternatively, you can also enter the key manually.
   Note: This way you transfer the authenticator key from **Identity Service** to the app.
5. The app then automatically generates verification codes. Each code is valid for 30 seconds.
6. Enter the currently valid verification code from the app in the appropriate input field in the web interface.
7. Confirm your configuration with the **Enable Two-factor Authentication** button.
8. Save the displayed recovery codes in a safe place.
9. Confirm that the recovery codes have been saved with the **I have stored the codes in a safe place** button.

You have thus enabled two-factor authentication for your user account and have configured the first authenticator device with the first authenticator app.

Recovery codes

Recovery codes allow you to also log in to **Identity Service** without an authenticator app. This is an emergency mechanism in case you can no longer log in via the authenticator app.



Your recovery codes

Store your recovery codes in a safe place and ensure that you can access the codes in the future.

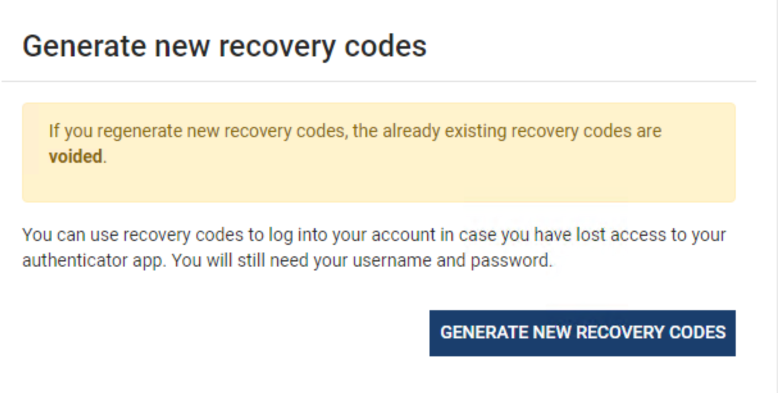
These are the recovery codes for two-factor authentication of your user account. You will need these codes in case you lose access to your mobile device or authenticator app. Each recovery code can be used only once.

```
431177c8  892addc8
2e93d780  40689e74
d79e27b3  4602564e
3faa765c  f2b078da
f809d15e  a9747d4d
```

The following is applicable for recovery codes:

- Only one set of recovery codes is valid for each user account in **Identity Service**.
- New recovery codes automatically overwrite older recovery codes. Older recovery codes are thus invalid immediately.
- The system will show you newly generated recovery codes once. It is not possible to display existing recovery codes again afterwards.
- Certain configurations in the Identity Service automatically generate new recovery codes.
- As a logged-in user, you can regenerate Recovery Codes at any time.

Recommendation: When you are shown recovery codes, you should always save them immediately.



Generate new recovery codes

If you regenerate new recovery codes, the already existing recovery codes are **voided**.

You can use recovery codes to log into your account in case you have lost access to your authenticator app. You will still need your username and password.

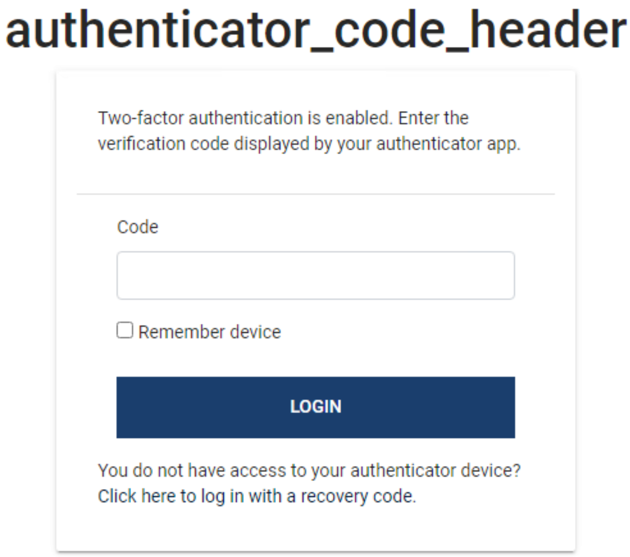
Confirmation dialog when generating new recovery codes.

10.2.3.1.9 User login - authenticator code

If two-factor authentication is enabled, the user logs in to the Identity Service in two steps:

1. Manual entry of user name and password
2. Entry of a verification code

The verification code is generated by an appropriately configured authenticator app. You must read off this code in the app and then manually enter it in the appropriate input field in **Identity Service**.



authenticator_code_header

Two-factor authentication is enabled. Enter the verification code displayed by your authenticator app.

Code

☐ Remember device

You do not have access to your authenticator device? Click here to log in with a recovery code.

CHECKBOX: REMEMBER DEVICE

If the **Remember device** checkbox is activated, you can also log in to a certain device without entering a verification code.

The following applies here:

- In this context, a device is a specific browser installation.
- If you are using several browsers on the same computer, each browser is considered to be a separate device.
- You can enable **Remember device** on every device through which you can log in to **Identity Service**.
- You can use **Forget device** to remove (on page 123) the **Remember device** again.

You should only use the **Remember device** function on devices which are protected against unauthorized access.

10.2.3.1.10 Add authenticator app

This option is only shown if two-factor authentication is enabled.

You can use **Add authenticator app** to configure additional authenticator apps for a user account. The GUI of this configuration dialog is identical with the dialog of **Enable Two-factor authentication** (on page 118).

With "Add authenticator app", there are the following differences in how it works:

1. No new authenticator key is created.
2. All recovery codes, however, are newly generated. Therefore, all previously generated recovery codes lose their validity.
3. Two-factor authentication has already been enabled.

You can set up and simultaneously use as many additional authenticator apps as you want using this configuration dialog. You can thus use several authenticator devices, for example.

Note: Since all apps then work with the same authenticator key, they simultaneously display identical verification codes.

10.2.3.1.11Disable

This option is only shown if two-factor authentication is enabled.

You can thus deactivate two-factor authentication.

This has the following consequences:

- The existing authenticator key is deleted.
- All authenticator apps that have been set up with this authenticator key lose access permanently.
- Two-factor authentication is disabled for your user account.

Afterwards, the only way you can log in is via user name and password in **Identity Service**.

The following applies for the deleted authenticator key:

- The deleted authenticator key cannot be restored.
- To reactivate two-factor authentication, you must set up two-factor authentication again. In the process, a new authenticator key is created.

Once the authenticator key has been newly created, you must configure each connected authenticator app again.

10.2.3.1.12 Forget device

This option is only shown on devices on which you have activated the **Remember device** checkbox for user login (on page 121).

The following applies here:

- If the **Remember device** option is enabled, you do not have to enter a verification code during user login via a certain device.
- You can use **Forget device** to disable **Remember device**. You must once again enter a verification code each time during user login.

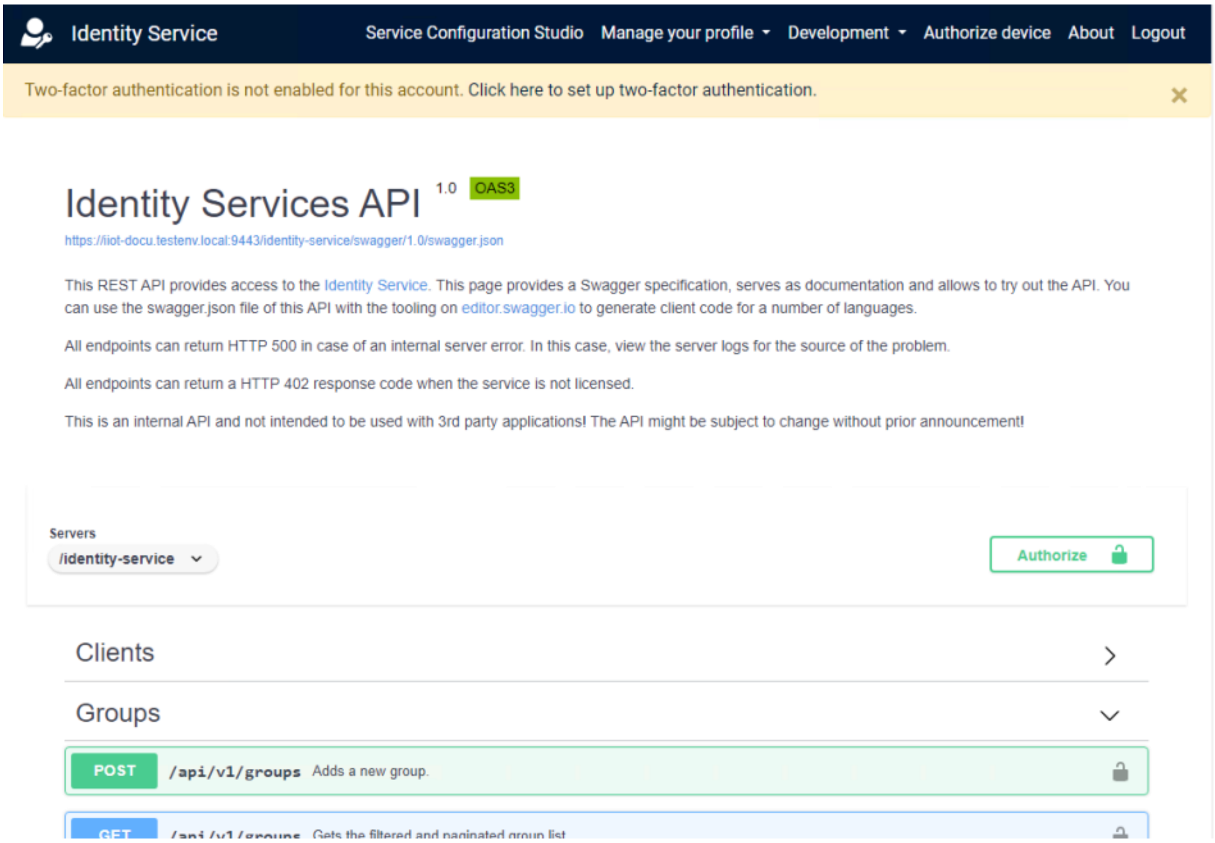
The **Forget device** option is only offered on devices on which you originally performed **Remember device**.

> **Tip**
>
> Forget several devices:
>
> If you have selected the **Remember device** option for several devices, you must log in separately via each device and perform **Forget device**.

10.2.3.2 Development

In this section, you can find technical information for the logged-in user.

10.2.3.2.1 API documentation



Two-factor authentication is not enabled for this account. Click here to set up two-factor authentication.

Identity Services API 1.0 OAS3

https://iiot-docu.testenv.local:9443/identity-service/swagger/1.0/swagger.json

This REST API provides access to the Identity Service. This page provides a Swagger specification, serves as documentation and allows to try out the API. You can use the swagger.json file of this API with the tooling on editor.swagger.io to generate client code for a number of languages.

All endpoints can return HTTP 500 in case of an internal server error. In this case, view the server logs for the source of the problem.

All endpoints can return a HTTP 402 response code when the service is not licensed.

This is an internal API and not intended to be used with 3rd party applications! The API might be subject to change without prior announcement!

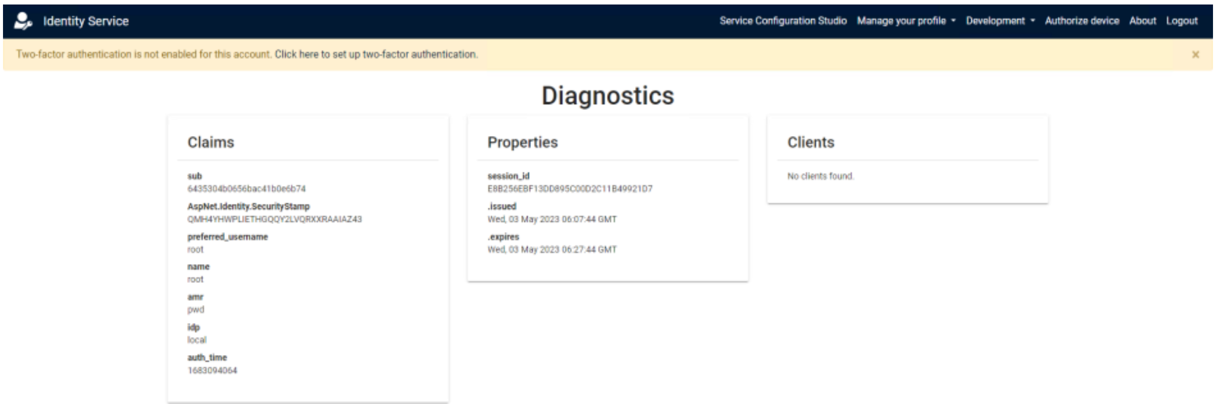
POST /api/v1/groups Adds a new group.

GET /api/v1/groups Gets the filtered and paginated group list

The **Identity Service** Swagger documentation is opened in this window.

The **Identity Service** API is an internal API. This API is continually enhanced and adjusted.

10.2.3.2.2 Diagnostics



Two-factor authentication is not enabled for this account. Click here to set up two-factor authentication.

Diagnostics

Claims

sub
6435304b0656bac41b0e6b74
AspNet.Identity.SecurityStamp
QMH4YHWPLIETHGQQY2LVQRXXRAAIAZ43
preferred_username
root
name
root
amr
pwd
idp
local
auth_time
1683094064

Properties

session_id
E8B256EBF13DD895C00D2C11B49921D7
.issued
Wed, 03 May 2023 06:07:44 GMT
.expires
Wed, 03 May 2023 06:27:44 GMT

Clients

No clients found.

In this view, you can find technical information about the logged-in user. This information is helpful, for example, if external applications from third-party providers are connected to IIoT Services.

10.2.3.2.3 OAuth2 discovery document

In this section, you can find information on OAuth authentication.

The OpenID configuration is a standard method for OpenID Connect providers to publish their authorization and token end points together with other configuration details.

This information is always visualized in a new tab in the web browser.

10.2.3.3 Authorize device

This menu item is for application scenarios in which applications or devices are to authenticate with the **Identity Service** without a graphical user interface. Here, it may be necessary to implement this via the Device Flow.

To authorize a Device , you must enter its **User code**. The code is displayed on the device. By entering the codes and confirming by clicking on the **SUBMIT** button, the device is authorized for communication with IIoT Services.

Note: You can find the **User code** on the device or in its user interface.

10.2.3.4 About

This dialog shows the **Identity Service** license status and version information.

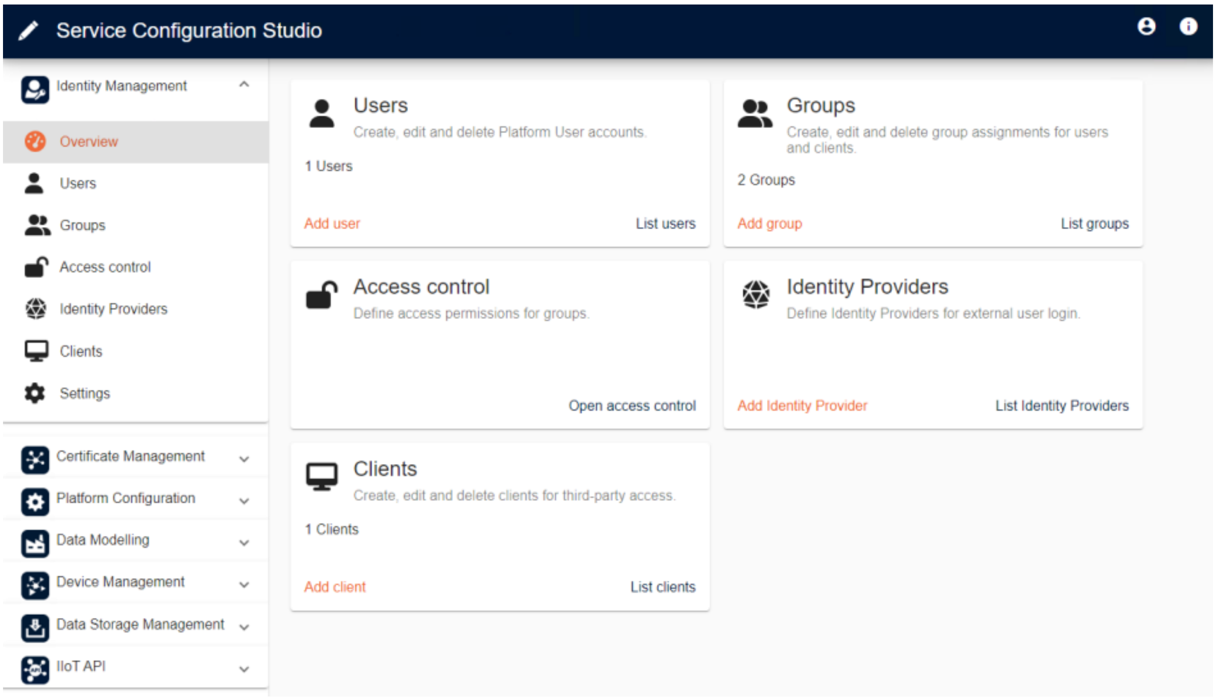
10.2.3.5 Logout

Logout allows you to log out centrally (Single Sign Out).

This concerns the following services:

- **Identity Service** web interface
- **Identity Management** web interface

10.3 Identity Management



Users

Create, edit and delete Platform User accounts.

1 Users

Add user | List users

Groups

Create, edit and delete group assignments for users and clients.

2 Groups

Add group | List groups

Access control

Define access permissions for groups.

Open access control

Identity Providers

Define Identity Providers for external user login.

Add Identity Provider | List Identity Providers

Clients

Create, edit and delete clients for third-party access.

1 Clients

Add client | List clients

Via the homepage (overview) of Identity Management, you can navigate to all the subpages of this service.

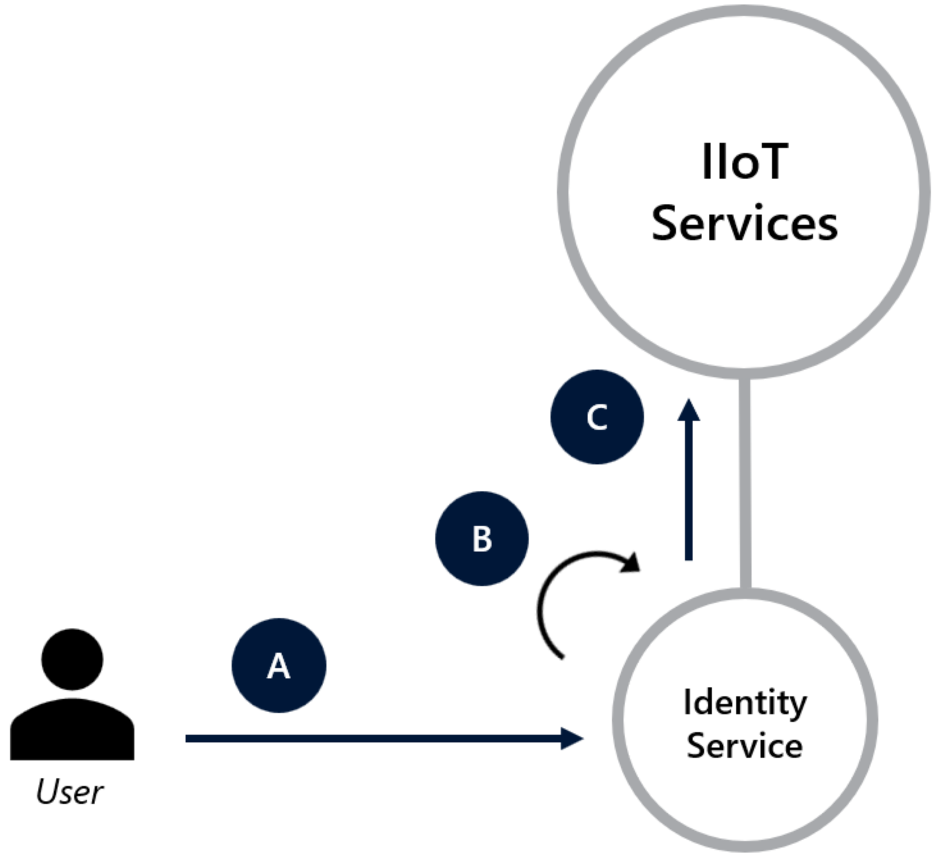
Identity Management is the web interface for the comprehensive administration of **Identity Service**.

The following applies for Identity Management:

- Logged-in users can administer **all user accounts** of the Identity Service.
- An admin user is created on initial installation.
- Only users that belong to the *Administrators* group can log in to the web interface of **Identity Management**.
- The initial user created in the IIoT Services automatically belongs to the *Administrators* group.

Other users can subsequently be granted permission to log in to **Identity Management**.

10.3.1 Users (internal login)



Via internal login, users log in to Identity Service as follows:
A. The user enters their internal login credentials in Identity Service.
B. Identity Service compares the credentials with the existing user accounts.
C. The authenticated user has access to IIoT Services.

Here you can centrally administer all internal user accounts of **Identity Service**. Every user who wants to log in to the web interfaces of IIoT Services needs a user account in the **Identity Service**.

Using the login information defined here, the user authenticates themself directly to **Identity Service** (**internal login**).

Hint

External login: You can also optionally link user accounts of **Identity Service** with **external logins** (on page 112).

10.3.1.1 Create new user

You create a new User for **Identity Service** with the **ADD** button.
An **internal user** is thus created. This User can log in via the internal login.

The detail view contains these options:

Option	Description
First name	The first name of the user.
Last name	The last name of the user.
User name	The user name of the user. The user name cannot be changed afterwards.
Email	The email address of the user.
Password	The password for the user.
Confirm password	You reenter the password here.
Require password change on next login	If you activate this checkbox, the user must change their password the next time they log in.
Description	You can add a description here.

10.3.1.2 Edit user

EDIT

<u>You can edit an existing user as follows:</u>

1. Click on the ... symbol in the user's line
2. Click on Edit.

A window with the following properties then opens:

Option	Description
First name	The first name of the user.
Last name	The last name of the user.
Email	The email address of the user.
Checkbox **Active**	By deactivating the checkbox, the user is

Option	Description
	deactivated. Every new user is active by default.
Description	Text field for a description.

Note: You cannot edit the **User name** afterwards.

CHANGE PASSWORD

You can reset a user’s password as follows:

1. Click on the **...** symbol in the user's line
2. Click on **Change password**.

A window with the following properties then opens:

Option	Description
Password	The password for the user.
Confirm password	You reenter the password here.
Require password change on next login	If you activate this checkbox, the user must change their password the next time they log in.

DISABLE TWO-FACTOR AUTHENTICATION

This menu item is only displayed if the respective user has enabled two-factor authentication for their user account.

This option is intended for the following use case:

- The user has enabled two-factor authentication for their user account.
- The user does not have access to their authenticator device and therefore cannot generate a verification code.
- The user does not have access to their recovery codes.

If the administrator disables two-factor authentication for the user concerned, the user can log in again to Identity Service.

You as the administrator can disable the two-factor authentication of a user as follows:

1. Click on the **...** symbol in the user's line
2. Click on the **Disable two-factor authentication** menu item.
3. Confirm your configuration by clicking on the **Disable two-factor authentication** button.

You have thus disabled the two-factor authentication for this user. The user can log in again without a verification code.

> **Tip**
>
> Re-enable two-factor authentication:
>
> Every user logged in to **Identity Service** has the option to re-enable two-factor authentication (on page 115) for their user account at any time.

10.3.1.3 Deactivate users

Newly created users of the **Identity Service** are by default in the *Active* status.

You can deactivate a user as follows:

1. Go to **Users** in the **Identity Management**.
2. Click on the **...** symbol next to the user.
3. Select the **Edit** option.
   A pop-up then opens with the editable properties of this user.
4. Uncheck the **Active** checkbox.
5. Confirm the action by clicking on the **Submit** button.

You have now deactivated the user.

> **Info**
>
> Deactivated users cannot:
>
> - Log in directly to the **Identity Service**
> - Log in to the **Identity Service** via external logins
> - Be assigned to groups
>
> Deactivated users are hidden by default in the overview table of **Identity Management\Users**. You can also display deactivated users in the table using the **Show inactive user accounts** option.

10.3.1.4 Activate users

You can reactivate a deactivated user as follows:

1. Go to **Users** in the **Identity Management**.
   Deactivated users are not displayed by default.
2. Click on the **...** button in the header of the table.
3. Activate the **Show inactive user accounts** checkbox.
   The table now displays deactivated users too.
4. Click on the **...** button next to the deactivated user in the header of the table.
5. Select the **Edit** option.
   A pop-up then opens with the editable properties of this user.
6. Activate the **Active** checkbox
7. Confirm the action with **Submit**.

You have now reactivated the user.

10.3.1.5 Pre-defined users

No users are pre-configured for IIoT Services by default. You must create the initial user when you log in to the web interface of the **Identity Service** for the first time (on page 108).

10.3.2 Groups

You administer groups for the **Identity Service** in the **Groups** menu item.

Groups can contain either **Users**s or **Clients**.

- The **Users** tab contains the users who can log into IIoT Services
- The **Clients** tab contains the applications that can log into IIoT Services

The main page of **Groups** shows an overview in table form. You can do the following on this page:

- Create new groups
- Edit groups
- Administer group affiliations (for users and clients)
- Delete existing groups

When a group is newly created or edited, the corresponding detail window opens.

> **⚠Attention**
>
> User groups of **external** Identity Providers **cannot** be administered in **Groups**.
>
> You can only administer user groups of external Identity Providers with the administration tools of the respective Identity Provider.

10.3.2.1 Administer groups

CREATE GROUP

<u>**You create a new group as follows:**</u>

- Click on the **Create Group** button
- Enter the **Group name** and the **Description** into the detail window
- Click on the **Add** button.

The group is thus created.

EDIT GROUPS

<u>**Proceed as follows to edit an existing group:**</u>

- Click on the **...** button of the desired group
- Select the *Edit* option
- You can now edit the properties **Group name** and **Description** .
  Note: The **Group name** must be unique.
- Confirm your changes by clicking on the **Submit** button

Your changes are thus saved.

DELETE GROUP

<u>**Delete a group as follows:**</u>

- Click on the **...** button for the group
- Select the **Delete** option
  A corresponding confirmation dialog will now open.
- Enter the name of the group to be deleted under **Enter group name**. This is a request for confirmation to prevent unintended deletion of the group.
  Note: The deletion of a group cannot be undone.
- Confirm by clicking on the **Delete** button.

Note: The assigned user accounts of a deleted group are retained. They are not deleted with the group.

10.3.2.2 Administer users in groups

In this menu item, you can add or remove existing users to or from groups.

ADDING USERS TO GROUPS

Proceed as follows:

1. Select a group and highlight it by clicking the mouse.
2. Switch to the **Users** tab.
3. Click on the **Add user** button.
4. In the **Add users** dialog, select one or more users from the list.
   Note: You can filter the list by user names.
5. Apply the selected users by clicking on the **Add** button.

The selected users thus become members of the corresponding group.

REMOVE USERS FROM A GROUP

You only remove the user from the group at this point. The user's account continues to exist regardless of group affiliation.

Proceed as follows:

1. Highlight the group with a mouse click.
2. Switch to the **Users** tab.
3. Select the user and click on the **...** button for the user.
4. Click on **Remove**.
5. Confirm the request for confirmation again with **Remove**.

The user is thus deleted from the group.

> **Information**
>
> The last-assigned user of the pre-defined **Administrators** group cannot be deleted from the group.

10.3.2.3 Administering clients in groups

The group affiliation of clients is administered in exactly the same way as group affiliation of users (on page 133).

10.3.2.4 Pre-defined groups

The *Administrators* system group is pre-configured for IIoT Services.

The following applies for the Administrators system group:

- Members of this group can administer IIoT Services in Service Configuration Studio in the **Identity Management**.
- The *Administrators* system group cannot be deleted or renamed.
- At least one user must always be assigned to the group. This is ensured by corresponding protective functions in IIoT Services. The last user cannot be deleted from the group.

It is thus ensured that an administrator has access to IIoT Services at all times.

10.3.3 Access control

You administer authorizations for **Groups** in **Access control**.

- Authorizations are issued via resources and roles.
- Only the combination of resource and role results in the authorization. It is not possible to assign a resource without a role.

The authorizations result from the group affiliations (**Groups**) of users (**User**) and applications (**Clients**).

BASIC PROCEDURE

The basic principle of assigning an authorization is as follows:

1. You select an **Group**
2. You add a resource to the **Group**.
3. You assign a role to the resource.

You have now assigned an authorization to the group.

10.3.3.1 Add resources

Resources in IIoT Services are, for example:

- Self-created Service Engine projects
- Pre-defined resources such as **Infrastructure**

You add resources as follows:

1. Select a group.
2. Click on the **Add resources** button.

3. Select the checkboxes for the desired resources from the list.
4. Confirm your selection by clicking on the **Add** button.

You have thus assigned resources to a **Group**. You must now assign each resource at least one role.

> **⚠Attention**
>
> The selection of a resource is only saved if a role has been assigned. A resource without role assignment is not saved.

10.3.3.2 Adding roles to the resource

You must add at least one role after adding a resource.

You add a role as follows:

1. Click on the **...** button in the resource.
2. Select **Manage roles** from the menu.
3. Highlight the checkbox of the desired role.
4. Apply the change by clicking on the **Submit** button.

You have thus assigned the role. Now the authorization – the combination of resource and role – is active.

10.3.3.3 Removing resources

You remove a resource as follows:

1. Please select a group.
2. Click on the **...** button for the resource that you want to remove.
3. Select the **Remove resource** option in the context menu.
4. Confirm your selection by clicking on the **Remove** button.

You have thus removed the resource from the group.

> **Information**
>
> You cannot remove the **Infrastructure** resource from the pre-defined **Service Grid Administator** group.

10.3.3.4 Pre-defined resources and roles

The following resources with the corresponding roles have been pre-configured in IIoT Services:

Pre-defined resources	Pre-defined roles	Description
Infrastructure	*Identity Administrator*	Only users with the *Identity Administrator* role have access to **Identity Management**. This assignment between role and resource is pre-defined and cannot be unassigned. The role is assigned by default to the *Administrators* group.
Infrastructure	*Certificate Management Administrator*	Only users with this role have access to **Certificate Management**. The role is assigned by default to the *Administrators* group.
Infrastructure	*Platform Configuration Administrator*	Only users with this role have access to **Platform Configuration**. The role is assigned by default to the *Administrators* group.
Infrastructure	*Device Management Administrator*	Only users with this role have access to **Device Management**. The role is assigned by default to the *Administrators* group.
Infrastructure	*Device Management Agent*	System role for device agents. Is not envisaged as a user role.
Infrastructure	*Service Connection Commissioner*	This role can be used for the connection of a Service Engine to the **IIot Services Connection Wizard** and for the registration of the **Device Agent** with **Device Management**.
Infrastructure	*Data Storage Administrator*	This role can manage and fully administer **Data Storage Management** in Service Configuration Studio. This role is necessary to clean up data or to delete it.

Pre-defined resources	Pre-defined roles	Description
Infrastructure	*Data Modelling Administrator*	This role serves as administrator of **Data Modelling** in Service Configuration Studio. In this role, **Tenants** for example, are administered in **Data Modelling** and **Construction Kit Libraries** are imported or deleted.
Infrastructure	*Service Connection Commisoner*	With this role, it is only possible for a client to call up new or current Certificate Bundles. All device agents communicate in this role. Old agents get the new role through migration.
Infrastructure	*<NAME_SERVICE_AGENT> Client*	With this role, the Service Engine user administration can be used with the **Identity Service** of IIoT Services. This role can only read users and groups. Configuration of the **Identity Service** cannot be edited by this role.
Report Engine Server	*IIoT API – Reporting Read*	This role can, in this resource, query all reports on this Report Engine Server via the **IIoT API**.
Report	*IIoT API – Reporting Read*	Users in this role can, in this resource, query the report or the reports precisely via the **IIoT API**. The following is applicable here: ▶ Each report is a subordination of precisely one Report Engine Server. ▶ Each Report Engine Server has 0 - n subordinate reports.
"zenon project name"	*IIoT API – Data Read*	Allows read access to enabled variables of the project via the IIoT API.
"zenon project name"	*IIoT API – Data Write*	Allows write access to enabled variables of the project via the IIoT API.
"zenon Projektname"	*IIoT API – Acknowledge Alarms*	Allows confirmation of alarm messages for this project via the IIoT API.

Pre-defined resources	Pre-defined roles	Description
"zenon project name"	*Data Storage – Data Read*	Allows data to be read from **Data Storage**.
"zenon project name"	*Data Storage – Data Write*	Allows data to be written from **Data Storage**.

ZENON PROJECT AS A RESOURCE

A zenon project is only available as a resource if the following conditions are met:

- The project has been configured for IIoT Services using Engineering Studio and the **IIoT Services Connection Wizard**.
- Service Engine must be started.
- There must be a network connection with IIoT Services.

If even only one condition is not met, you will not have access to the project via IIoT Services.

10.3.4Report permissions

Here you can configure the availability and the access permissions for reports and Custom Report Items via the IIoT API.

REPORT ENGINES MENU

Option	Description
Search	Searches the existing instances of Report Engine. String input. The search starts when the first character is entered. All instances of Report Engine are found that contain the string. The list is limited to search hits.
Report Engines	List of available instances of Report Engine.
Detail display	Displays all the properties of the selected Report Engine and allows you to configure them.
Assign group permissions Symbol	Clicking on the symbol opens the dialog Dialog (on page 144) for assigning group permissions. Only available if an instance of Report Engine has been selected.

10.3.4.1 Report

Option	Description
Search	Searches the available items for name and alias. String input. The search starts when the first character is entered. All items are found that contain the string. The list of items is limited to the items found during the search.
Last update time	Displays the date and the time of the last update of the table, such as: Last time the table was displayed, last time the content was updated, last time the content was saved.
Update symbol	Clicking on the button updates the data in the table for the IIoT Services.
List of reports	List of available reports. Their names and properties are displayed. Users can: ▸ Edit an alias ▸ Publish or withdraw individual reports
Name	Name of the report.
Description	Description of the report. Clicking on the description opens a window that displays the complete description.
Alias	Name of the alias. This can be changed for the respective action via the symbol. Permitted characters: Alphanumeric characters and underscores **Note:** The assigned alias must be unique throughout all configured Report Engine instances for each item type.
Published	Displays whether the report has been published.
Actions	Available actions:

Option	Description
	- Edit an item: Allows you to Change an alias (on page 142). - Publish an item: Opens the Publishing dialog (on page 143) of an item.
Navigation	Enables you to navigate in the list. - **|<**: Displays the first page. - **<**: Displays the previous page. - **Number**: Displays the current page and selection of another page. - **>**: Displays the next page. - **>|**: Displays the last page. - **Items per page**: Configuration of the lines per page to be displayed. Select from drop-down list. - **Items displayed**: Shows how many items are currently being displayed and how many in total are available. Filters are taken into account. - **Update** symbol: Clicking on the button updates the displayed data with the current data of the system.

10.3.4.2 SQL element

List of the available SQL elements. The SQL functionality of Report Engine can thus be made available via the IIoT API in the form of Stored Procedures and User-defined Functions. These can be checked here and provided with an alias as well as published and withdrawn.

The following options are available:

Option	Description
Search	Searches the available items for name and alias. String input. The search starts when the first character is entered. All items are found that contain the string. The

Option	Description
	list of items is limited to the items found during the search.
Last update time	Displays the date and the time of the last update of the table, such as: Last time the table was displayed, last time the content was updated, last time the content was saved.
Update symbol	Clicking on the button updates the data in the table for the IIoT Services.
List of SQL elements	List of the available SQL elements. Their names and properties are displayed. Users can: - Edit an alias - Publish or withdraw individual SQL elements
Name	Name of the SQL element.
Description	Description of the SQL element. Clicking on the description opens a window that displays the complete description.
Alias	Name of the alias. This can be changed for the respective action via the symbol. Permitted characters: Alphanumeric characters and underscores **Note:** The assigned alias must be unique throughout all configured Report Engine instances for each item type.
Published	Displays whether the SQL element has been published.
Actions	Available actions: - Edit an item: Allows you to Change an alias (on page 142). - Publish an item: Opens the Publishing dialog (on page 143) of an item.
Navigation	Enables you to navigate in the list. - **|<**: Displays the first page.

Option	Description
	- **<**: Displays the previous page. - **Number**: Displays the current page and selection of another page. - **>**: Displays the next page. - **>|**: Displays the last page. - **Items per page**: Configuration of the lines per page to be displayed. Select from drop-down list. - **Items displayed**: Shows how many items are currently being displayed and how many in total are available. Filters are taken into account. - **Update** symbol: Clicking on the button updates the displayed data with the current data of the system.

10.3.4.3 Edit an alias

Aliases can be created, edited and deleted directly in the lists or in the Publishing dialog (on page 143).

CREATING OR EDITING AN ALIAS

To edit an alias:

1. Click on the **Pencil** symbol in the Actions column.

   The input field opens in the Alias column.

2. Enter the desired alias.
   Note: The assigned alias must be unique for all configured instances of Report Engine for each element type.

   Click on the **Apply** symbol (check mark):
   To leave the field without making changes, click on the **Discard** symbol.

3. The alias is entered into the list.

The alias has thus been edited.

DELETE AN ALIAS

To delete an alias:

1. Click on the **Pencil** symbol in the Actions column.

   The input field opens in the Alias column. An **X** symbol is displayed next to the alias.

2. Click on the **X**.

   The text is deleted.

3. Click on the **Apply** symbol (check mark):
   To leave the field without making changes, click on the **Discard** symbol.

The alias has thus been deleted.

10.3.4.4 Publish

This dialog is opened if you click on the **Publish** button of an element of a Report Engine.

Published means that the respective element can be reached via the IIoT API using the selected alias.

To publish or withdraw the publishing:

1. Select the desired item.

2. Configure the options of the detail view.
   These are different for individual items and parameters.

   For parameters, you receive a list of available items in the detail view. Select the desired items by means of the checkboxes. Clicking on the button affects all the selected parameters.

3. Click on the **Publish selected** or **Unpublish selected** button.

ITEM DIALOG

Option	Description
Tree view Additional filter	Project filter. Selection of an item from the tree view. Detailed configuration takes place in the detail window. **Symbols:** ‣ Red triangle: There is a problem in this node. ‣ Yellow triangle: There is a problem in the nesting of this node.
Detail view	Configuration of the selected item from the tree view.
Alias	Alias of the item.

Option	Description
	Note: The assigned alias must be unique throughout all configured Report Engines for each item type.
Published	Displays the status of the item via a checkbox: - *active*: published - *inactive*: unpublished
Unpublish	Clicking on the button opens the confirmation dialog. If this is confirmed, the selected, already published item is reset to unpublished. Thus, the item is no longer accessible via the REST interface.
Publish	Clicking on the button opens the confirmation dialog. If this is confirmed, the selected item is published.

10.3.4.5 Assign permissions

Here you can configure which item is to be available in which permissions group. You can configure or copy assignments.

To assign an item to a permissions group:

1. Select the permissions group in the group list.
2. Highlight the items you would like to allow in the tree view.
3. Press the **Save** button.

To copy an assignment from one permissions group to another:

1. Click on the **Copy assignment from group to group** button.

   The dialog to assign groups is opened.
2. Select the source group.
3. Select the target group.
4. Click on **Copy**.

   The rights granted in the source group are copied to the target group.

ASSIGNMENT OF PERMISSIONS GROUPS TO ITEMS DIALOG

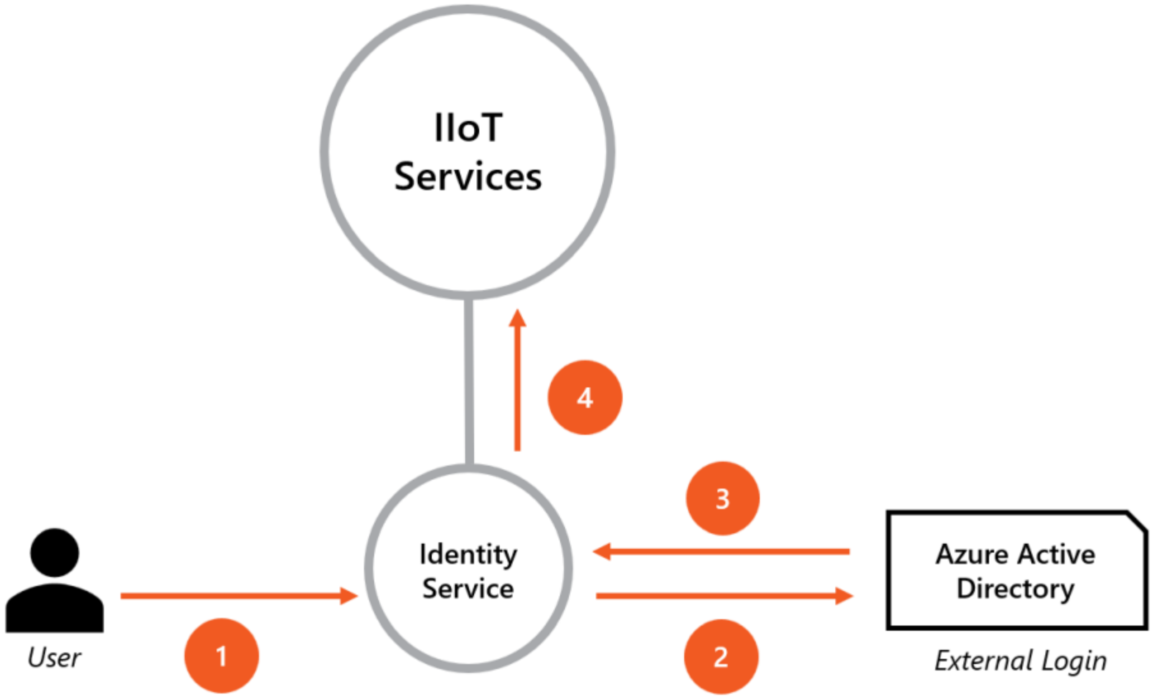
Option	Description
Groups	Displays all the available permissions groups.
Search	Search in the group list. String input. The search starts when the first character is entered. All permissions groups are found that contain the string. The list of groups is limited to the groups found during the search.
Items	Displays all the available items.
Search (Groups)	Search for item name in all items of Report Engine. String input. The search starts when the first character is entered. All items are found that contain the string. The list of items is limited to the items found during the search.
Search (Items)	Search for item type in all items of Report Engine. Select from drop-down list. All items are found that match the type. The list of items is limited to the items found during the search.
List of items	Displays the available items. When an entry is made in the search window, the display is limited to the relevant hits.
Copy assignment from group to group	Opens the dialog to copy permissions from one group to another permissions group.
Save	Clicking on the button saves the configuration.

TRANSFER OF RIGHTS FROM GROUP TO GROUP DIALOG

Option	Description
Search	Search in the group list. String input. The search starts when the first character is entered. All permissions groups are found that contain the string. The list of groups is limited to the groups found during the search.
List of groups	Displays all the available source groups.
Search	Search for item name in all items of Report Engine.

Option	Description
	String input. The search starts when the first character is entered. All items are found that contain the string. The list of items is limited to the items found during the search.
List of groups	Displays all the available target groups.
Copy	Transfers rights from the source group to the target group

10.3.5 Identity providers (for external logins)



The process of logging in a user via an external login is as follows:
1. The user clicks on an external login on the login page in the web interface of Identity Service (here: Azure Active Directory).
2. Identity Service redirects the user to the website of Azure Active Directory. The user enters their credentials there.
3. The external identity provider compares the credentials with its user accounts and reports the result to Identity Service.
4. The Identity Service grants the authenticated user access to IIoT Services.

In the **Identity Providers** menu item, you can define external identity providers for **External logins** (on page 112). This therefore allows you to integrate IIoT Services into an existing authentication infrastructure.

Tip

How a user name is entered during user login depends on the identity provider.

Example: Microsoft Active Directory as external identity provider

- Full user name:
  john.doe@microsoft.com
- Login to web interface of **Identity Service**:
  john.doe@microsoft.com
- Login to Service Engine:
  john.doe

You can find detailed information in relation to this in the **IIoT Services - configuration in Engineering Studio** section in the chapter **Compatibility table: User names**.

10.3.5.1 Azure Active Directory

Option	Description	Example
Provider Alias	Is needed for unique identification. The **Provider Alias** is also used for the login page of the Identity Service. There, the user can select which Identity Provider they want to use for login.	*my-azure-ad*
Provider description	Optional description of the Identity Provider.	
Tenant ID	A unique identifier that you must look up in your Azure subscription.	*e5fa83b7-d261-48f0-8728-524d9b949d52*
Authority	URL at which the OpenID Connect Provider can be contacted.	*https://login.microsoftonline.com*
Client ID	Enter the **Client ID** defined in the Azure Active Directory. This is required by Identity Service to	*07c8b99b-fe15-2587-8e58-c1dc85d4fe17*

Option	Description	Example
	authenticate external users. **Note:** The **Client ID** must be registered in the Azure Active Directory.	
Client Secret	**Client Secret** that is generated in Azure Active Directory and has been assigned to the **Client ID**. **Note:** Azure Active Directory always creates **Client ID** and **Client Secret** in pairs. Identity Service uses this specific combination of pairs to authenticate external users.	*85NZrL17WRBWVbyn.FDuz_tZ4Yt55..341*

Tip

Configuration in Azure Active Directory

The configuration dialog displays – adjusted to your system – the following hint:

```
Please enter the following redirect URL(s) to your external identity provider:
https://mycomputer.mydomain.com:9443/identity-service/auth/my-azure-ad/signin-callback
```

You must save your individual URLs in Azure Portal as valid Redirect URL(s).

Redirect URL after upgrade

As of Version 10.1, Identity Service uses different Redirect URLs for authentication with **Azure Active Directory**.

After an upgrade of an older IIoT Services version 10.0 (or older), you must thus reconfigure the updated Redirect URLs of Identity Service in Azure Portal.

10.3.5.2 Microsoft Active Directory

There are the following options for configuration with **Microsoft Active Directory**:

Option	Description	Example
Provider Alias	Is used for unique identification if several Identity Provider s have	

Option	Description	Example
	been configured. The **Provider Alias** is also used for the login page of the Identity Service. There, the user can select which Identity Provider they want to use for login.	
Provider description	Optional description of the Identity Provider.	
Hostname	The Microsoft Active Directory domain controller.	
Port	Port number via which the Microsoft Active Directory is accessible on the host.	*636 (TLS active)* *389 (TLS inactive)*
Apply TLS encryption	*Active*: The communication is TLS encrypted. *Inactive*: Communication is not encrypted. Requirements: - The matching **Port** must be selected. - The server must provide a matching server certificate. Default: *active*	*active*

10.3.5.3 OpenLDAP

There are the following options for configuration with **OpenLDAP**:

Option	Description	Example
Provider Alias	Is used for unique identification if several Identity Provider s have been configured. The **Provider Alias** is also used for the login page of the Identity	

Option	Description	Example
	Service. There, the user can select which Identity Provider they want to use for login.	
Provider description	Optional description of the Identity Provider.	
Hostname	Address of the authentication source.	*openldap.myserver.com*
Port	Port number for the connection.	*636 (TLS active)* *389 (TLS inactive)*
Apply TLS encryption	*Active*: The communication is TLS encrypted. *Inactive*: Communication is not encrypted. Requirements: - The matching **Port** must be selected. - The server must provide a matching server certificate. Default: *active*	*active*

10.3.5.4 RADIUS

RADIUS implementation in the **Identity Service** supports:

- Client login for applications via IIoT Services
- User login via Service Engine and Web Engine
- User login via the Service Configuration Studio of the Identity Service

Option	Description
Provider Alias	Is needed for unique identification. The **Provider Alias** is also used for the login page of the Identity Service. There, the user can select which Identity Provider they want to use for login.

Option	Description
	Example: *my-radius*
Provider description	Optional description of the Identity Provider.
Apply group assignments from Identity Provider	If the checkbox is checked, group memberships are applied to users when logging in.
Hostname	The hostname of the RADIUS server. **Example:** *myradius.mydomain.com*
Port	Port number of the RADIUS server. Default: *1812* **Note:** This option is only visible after a **host name** has been configured.
Shared Secret	The **Shared Secret** is used for the encrypted transfer of the user password when logging in to the RADIUS Server. **Example:** *85NZrL17WRBWVbyn.FDuz_tZ4Yt55..341* (as defined on the Radius server) **Note:** This option is only visible after a **host name** has been configured.
Authentication type	The following authentication methods are supported: - *PAP* - *CHAP* The use of *CHAP* is recommended. Default: *CHAP* **Note:** This option is only visible after a **host name** has been configured.
Clear all	All current and unsaved parameter settings are reset to the default value.
Add new connection	Adds the option to configure a new fallback connection for the configuration dialog.
Add	Applies the configuration and closes the dialog. The configuration is available as a new radius connection for login.

CONNECTIONS

The first connection for **RADIUS** is the **Primary connection**. This must be configured to use **RADIUS**.

You can also optionally create one or several **Fallback connections**. If the **Primary connection** does not respond to a request, the system attempts to connect via a **Fallback connection**. **Fallback connections** are used in the configured order.

10.3.5.4.1 Get group assignments via RADIUS

The **Identity Service** can apply group assignments of users from the **RADIUS** server.

To do this, the following requirements must be fulfilled:

1. The **Apply group assignments from Identity Provider** checkbox must be activated in the configuration of **RADIUS** as the **Identity Provider**.
2. You can only apply group assignments of the user from RADIUS if the groups to be assigned to already exist in the **Identity Service**.
3. For the purpose of correct assignment, group names may not contain a ',' (comma) or an '@' (at sign).
4. The RADIUS server must be configured in such a way that, upon the successful authentication of users, it returns a list of group names to the **Identity Service**. This list represents the groups to which the user should be added automatically in the Identity Service.
5. The list of group names must be transferred from the RADIUS server during login using **Vendor Specific Attributes (VSA)**. To do this, the RADIUS server must be configured accordingly.

The RADIUS protocol uses Vendor Specific Attributes to transfer additional information during user login.

SUPPORTED 'VENDOR SPECIFIC ATTRIBUTES': TRANSFER OF GROUP ASSIGNMENTS

The following VSA are supported for RADIUS as the identity provider:

- **Vendor ID:** *22050*
- **Attribute ID:** *1*
- **Format of group names:** List with one or more group names (separated by a comma).
  Examples: *Group1* or *Administrator,Group2,Group3*

You can find further information on VSAs in the documentation of your RADIUS server.

RULES FOR GROUP ASSIGNMENT

The following rules apply for automatic group assignment via RADIUS:

- Groups from RADIUS are automatically assigned groups with the same name from **Identity Service**. This means that the group names in **Identity Service** and the group names transferred from the **RADIUS** server must match exactly. This includes the case of the group names.
- The group assignments of users for **Identity Service** are automatically taken from the RADIUS server each time during login. The group assignments are valid for the user as long as they are logged in to RADIUS.

As long as the user is logged in via RADIUS, the group assignments manually defined in **Identity Management** for **Identity Service** do not apply.

Hint: Avoid manual group assignments

If RADIUS has been configured in **Identity Service** for obtaining group assignments, then you should only use the groups defined on the RADIUS server.

Reason: Manual group assignments of RADIUS users in **Identity Service** are automatically written over with the group assignments from the RADIUS server when the user logs in.

Therefore, it is not possible to simultaneously use manually configured (in **Identity Service**) and automatically obtained (via RADIUS) group assignments.

10.3.5.5 OpenID Connect

Option	Description	Example
Provider Alias	Is needed for unique identification. The **Provider Alias** is also used for the login page of the Identity Service. There, the user can select which Identity Provider they want to use for login.	
Provider description	Optional description of the Identity Provider.	
Authority	URL at which the OpenID Connect Provider can be	*https://myopenidconnect.com*

Option	Description	Example
	contacted.	
Client ID	Enter the **Client ID** defined in the OpenID Connect Provider. This is required by Identity Service to authenticate external users. **Note:** The **Client ID** must be registered in the OpenID Connect Identity Provider .	**For example:** *07c8b99b-fe15-2587-8e58-c1dc85d4fe17* The format of the Client ID depends on the OpenID Connect Provider used.
Client Secret	**Client Secret** that is generated inOpenID Connect and has been assigned to the OpenID Connect **Client ID**. **Note:** OpenID Connect always creates **Client ID** and **Client Secret** in pairs. Identity Service uses this specific combination of pairs to authenticate external users.	**For example:** *85NZrL17WRBWVbyn.FDuz_tZ4Yt55..341* The format of the Client Secret depends on the OpenID Connect Provider used.

Tip

Configuration in OpenID Connect

The configuration dialog displays – adjusted to your system – the following hint:

Please enter the following redirect URL(s) to your external identity provider:

https://mycomputer.mydomain.com:9443/identity-service/auth/my-openid-connect/signin-callback

https://mycomputer.mydomain.com:9443/identity-service/auth/my-openid-connect/signout-callback

You must save your individual URLs in the administration tool of the OpenID Connect Providers as valid Redirect URL(s).

10.3.5.6 Keycloak

Option	Description	Example
Provider Alias	Is needed for unique identification. The **Provider Alias** is also used for the login page of the Identity Service. There, the user can select which Identity Provider they want to use for login.	
Provider description	Optional description of the Identity Provider.	
Apply group assignments from Identity Provider	If the checkbox is checked, group memberships are applied to users when logging in.	
Authority	The URL which provides Keycloak for user authentication.	*https://my-keycloak.com/auth/realms/master*
Client ID	Enter the **Client ID** defined in the Keycloak. This is required by Identity Service to authenticate external users. **Note:** The **Client ID** must be registered in the Keycloak.	*identity-service-client* Note: The Client ID is defined by the administrator.
Client Secret	**Client Secret** that is generated in Keycloak and has been assigned to the **Client ID**. **Note: Client ID** and **Client Secret** are always required in pairs. **Identity Service** uses this specific combination of pairs to authenticate external users.	*07c8b99b-fe15-2587-8e58-c1dc85d4fe17*

Tip

Configuration in Keycloak

The configuration dialog displays – adjusted to your system – the following hint:

Please enter the following redirect URL(s) to your external identity provider:

https://mycomputer.mydomain.com:9443/identity-service/auth/my-keycloak/signin-callback

https://mycomputer.mydomain.com:9443/identity-service/auth/my-keycloak/signout-callback

You must save your individual URLs in the administration tool of Keycloak as valid Redirect URL(s).

10.3.5.6.1 Keycloak server: Client configuration

You must configure a client for IIoT Services in the Keycloak server.

The client needs specific settings for IIoT Services:

- **Access Type** = *confidential*
  Note: This settings activates the **Credentials** tab. You require **Client ID** and **Client Secret**.
- **Implicit Flow Enabled** = *ON*
- **Standard Flow Enabled** = *ON*
- **Direct Access Grants** = *ON*

You use this client to integrate Keycloak as the external Identity Provider.

10.3.5.6.2 Get group assignments via Keycloak

Identity Service can automatically apply group assignments of users from the Keycloak server.

Keycloak and Identity Service display groups differently:

- The Keycloak server supports group hierarchies. Groups can be nested.
- **Identity Service** does **not** support group hierarchies. It uses a flat group structure. All groups are located on the same hierarchic level.

The hierarchic group structures of Keycloak are therefore automatically converted by **Identity Service** into flat group structures. This can be relevant for security (see example (on page 160)).

The following rules apply for the automatic group assignment of users:

- Group assignments from Keycloak are not applied for all users of **Identity Service**. They are only applied for users that logged in via Keycloak.
- The group assignments valid for a user are automatically applied for **Identity Service** each time this user logs in via Keycloak.
- Source groups from Keycloak are assigned target groups of the same name in **Identity Service**. A requirement for this is that the group already exists in **Identity Service**.
- The required target groups in **Identity Service** must be created manually in **Identity Management**. If a target group is missing, group memberships will not be applied by Keycloak in this respect.
- This means that the group names created manually in **Identity Service** and the group names transferred from the Keycloak server must **match exactly**. This includes the case of the group names.
- Group assignments of Keycloak permanently overwrite conflicting group assignments in **Identity Service**. They are also maintained in **Identity Service** after the user logs out of Keycloak.

Therefore, it is generally recommended to administer group memberships only in Keycloak (on page 163).

Hint: Group administrators in Identity Service

The administrative users in IIoT Services are part of the *Administrators* group in the **Identity Service**. At least one user of the **Identity Service** must be part of the *Administrators* group at all times to ensure administrative access to IIoT Services.

<u>This is ensured by the following protective functions:</u>

- In **Identity Management**, the last user cannot be removed manually from the *Administrators* group in **Identity Service**.
- The group assignments automatically obatined from Keycloak also cannot remove the last user from the *Administrators* group.

This ensures that administrative access to IIoT Services is also guaranteed when group assignments are obtained via Keycloak.

10.3.5.6.3 Configuration

CONFIGURATION OF THE KEYCLOAK SERVER

<u>You must configure a protocol mapper in the Keycloak server :</u>

1. Go to the administrator console of the Keycloak server used by you.
2. Go to **Clients** -> **Identity-Server** -> **Mappers**.
3. Create a new protocol mapper by clicking **Create**.
4. Configure the protocol mapper with the following settings:

Property	Necessary configuration
Protocol	*openid-connect* (value is preset)
Name	Enter a name of your choice. Note: This is an internal name for Keycloak. It is not relevant for **Identity Service**.
Mapper Type	*Group Membership*
Token Claim Name	*Group*
Full group path	*On* or *Off* This setting defines whether, in the case of hierarchically nested groups, the server sends the full group path to **Identity Service** or only the group names. **Example:** - Group name: *Administrators* - Group path: *London/Administrators* This setting determines which group assignments can be generated (see example (on page 160)).
Add to ID token	Any configuration.
Add to access token	Any configuration.
Add to userinfo	*On*

You have thus finished configuring the Keycloak server for transferring group assignments to **Identity Service**.

CONFIGURATION OF KEYCLOAK IN IDENTITY SERVICE

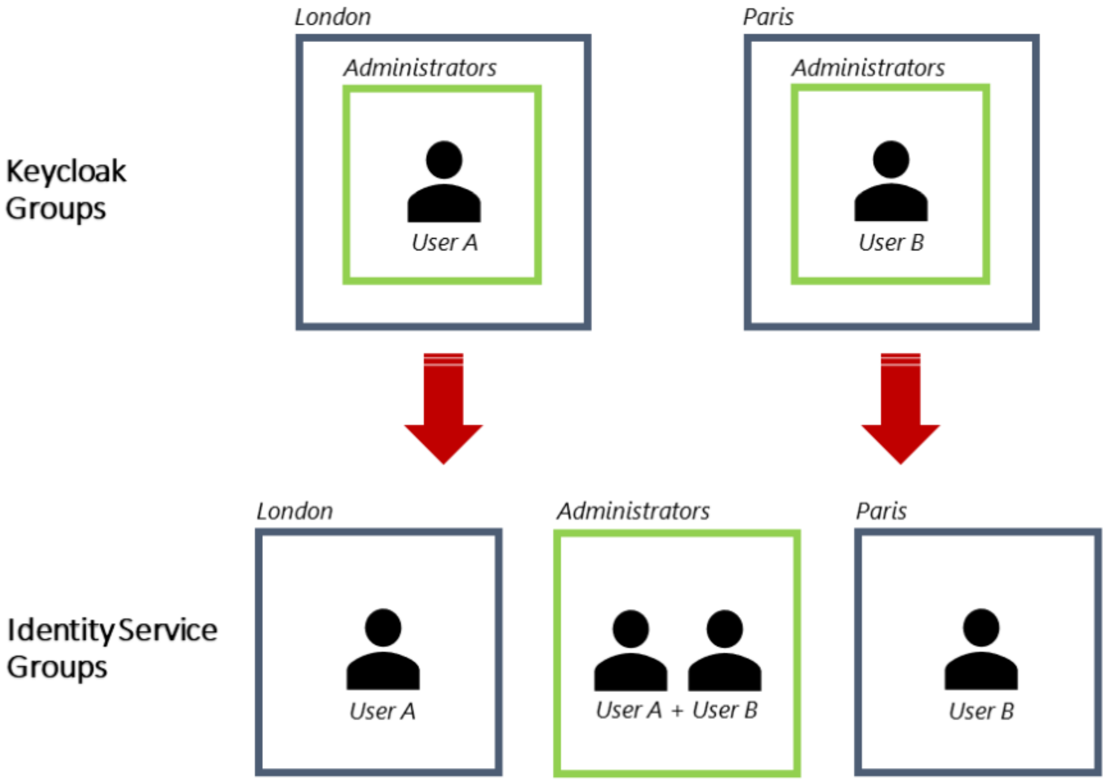
To configure Identity Service:

1. **Identity Management** -> **Identity Providers** -> **Keycloak**:
   The **Apply group assignments from Identity Provider** checkbox must be activated in the configuration.

2. A target group must be defined in **Identity Management** for each source group from the Keycloak server. If the target group is missing, the group assignments of the source group will not be applied.

Identity Management has now been configured for applying group assignments from the Keycloak server.

10.3.5.6.4 Example: Apply group assignments



Example of the "Full group path -> On" configuration: Identity Service converts the hierarchic group structures of Keycloak into flat group structures.

Case description:

In the Keycloak server, separate administrator groups (*Paris/Administrators* and *London/Administrators*) are defined for the Paris site and the London site.

- The hierarchic arrangement into the parent groups *Paris* and *London* differentiates the *Administrators* subgroups of the same name. There are two separate subgroups with the same group name.
- How the two group structures are transferred to **Identity Service** depends on the *Full group path* setting selected in the protocol mapper.

The configuration options in Keycloak are as follows:

Keycloak: Group assignments	Keycloak: Configuration	Shared group information	Identity Service: Group assignments

Paris/Administrators;
London/Administrators
	Full group path -> *On*
	(see diagram)
		Paris/Administrators;
		London/Administrators
			Paris;
			London;
			Administrators

Paris/Administrators; *London/Administrators*	**Full group path** -> *Off*	*Administrators;* *Administrators*	*Administrators*

The following applies for the applying of group assignments in this example:

- The corresponding group assignment can be made in **Identity Service** for each of the listed Keycloak group assignments.
- The *Full group path* setting in Keycloak determines which group names are transferred.
- The hierarchic group structures of Keycloak are converted by **Identity Service** into flat group structures.

The *Administrators* subgroups which are separated in Keycloak are combined into one group with the same name in **Identity Service**.

⚠ Attention: Exclude security risks

> The different representations of group structures in Keycloak and **Identity Service** can – depending on the application – pose a security risk. You should already plan how you are going to apply group assignments later on in Identity Service (on page 163) when creating the group structures in Keycloak.

10.3.5.6.5 Best practice: Avoid security risks

Keycloak and **Identity Service** use different group architectures and thus have a different security architecture in this respect.

When group assignments are obtained from Keycloak, the following could be relevant for security:

- The hierarchic group structures from Keycloak are converted by **Identity Service** into flat group structures.
- Group assignments configured manually in **Identity Management** are not synchronized with Keycloak.
- A target group must be defined manually in **Identity Management** for each source group in Keycloak. If the target group is missing, the group memberships of the source group will not be synchronized.
- Several subgroups of the same name in the Keycloak group hierarchy are combined by **Identity Service** into one group (see the example in the separate topic (on page 160)).
- Keycloak overwrites conflicting group assignments in **Identity Service** (see the example in the box).

This system behavior must be taken into consideration during configuration to exclude any possible security risks in advance.

Example: Conflicting group assignments

Initial situation:

- *User A* and *User B* are both defined in **Identity Service** as members of the *Administrators* group.
- In Keycloak only *User B* is defined as a member of the *Administrators* group.

User A thus has different group assignments for Keycloak and **Identity Service**.

When User A logs in to the web interface of Identity Service:

- The group assignment from Keycloak overwrites the group assignment in **Identity Service**.
- *User A* is removed from the *Administrators* group in **Identity Service**.

User A thus permanently loses the right to log in to the web interface of **Identity Management**

The following procedure is recommended if you want to automatically obtain group assignments for Identity Service via Keycloak:

- Manage group assignments of users not in **Identity Management** but only in Keycloak.
  Reason: This way it is ensured that a group assigment defined manually in **Identity Management** cannot be overwritten by a group assignment obtained automatically from Keycloak.
- In Keycloak only use group names that are unique even without a hierarchic relationship.
  Reason: This ensures that the group assignment of users can be applied unchanged in **Identity Service**.
- In general, do not use *Administrators* as a group name in Keycloak.
  Reason: This ensures that Keycloak users are not accidentally allocated to the *Administrators* group in **Identity Service**.

In general, the following applies: The different group architectures of Keycloak and **Identity Service** must be taken into account when planning a group structure.

10.3.6 Clients

In the **Clients** menu item, you manage client access to IIoT Services.

Example: Third-party applications use the client to connect to IIoT Services via the IIoT API.

CLIENTS

The table shows the existing client definitions. Only clients that have been created manually are shown there by default.

> **Tip**
>
> System clients are created and managed automatically by IIoT Services. They cannot be edited and are hidden in the table by default. You can display the system clients with the **Show system clients** checkbox.

TABLE

The table contains the following columns:

- **Client ID**
- **Client name**
- **Redirect URLs**
- **Grant types**
- **Allowed scopes**

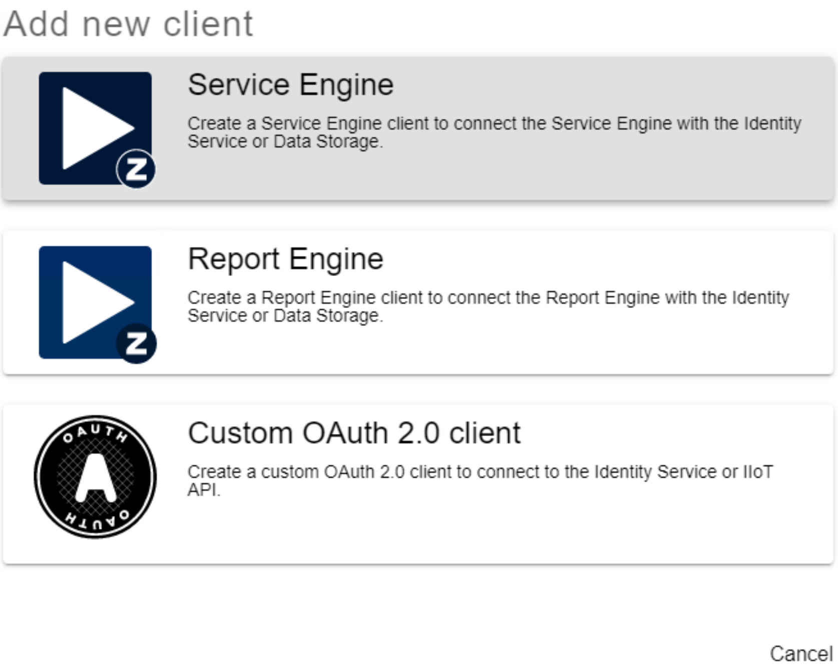
Each table column offers the option to filter and sort the displayed **Clients**.

You can do the following in the table:

- Add clients (**Add** button, opens the detail view)
- Edit clients (**Edit** button, opens the detail view)
- Delete clients (**Delete** button)

10.3.6.1 Add new client

In this dialog, you select the type of client that you want to create.



Add new client

Service Engine

Create a Service Engine client to connect the Service Engine with the Identity Service or Data Storage.

Report Engine

Create a Report Engine client to connect the Report Engine with the Identity Service or Data Storage.

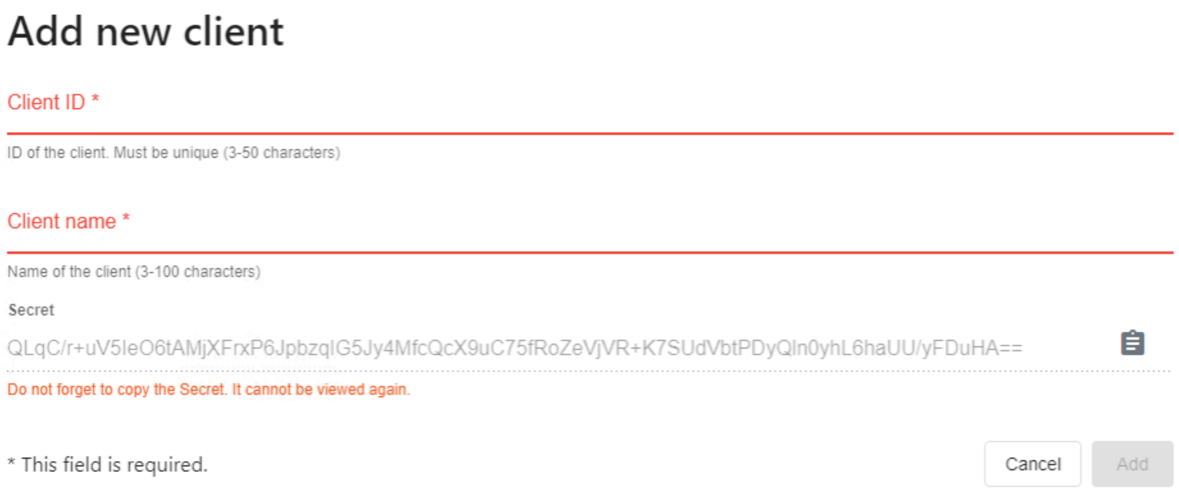
Custom OAuth 2.0 client

Create a custom OAuth 2.0 client to connect to the Identity Service or IIoT API.

Option	Description
Service Engine	Opens the configuration dialog for a new Service Engine client.
Report Engine	Opens the configuration dialog for a new <REPORT_ENGINE> client.
Custom OAuth 2.0 client	Opens the configuration dialog for a new OAuth client.
Cancel	Closes the selection dialog.

10.3.6.1.1 Service Engine

In this dialog, you configure a new client for logging in to Service Engine.



Add new client

Client ID *

ID of the client. Must be unique (3-50 characters)

Client name *

Name of the client (3-100 characters)

Secret

QLqC/r+uV5leO6tAMjXFrxP6JpbzqlG5Jy4MfcQcX9uC75fRoZeVjVR+K7SUdVbtPDyQIn0yhL6haUU/yFDuHA==

Do not forget to copy the Secret. It cannot be viewed again.

* This field is required.

Option	Description
Client ID	Unique ID of the client. The ID must be unique and can contain numbers, letters, and special characters. If there is already a client with the same ID, the client will not be created and a corresponding warning will be displayed. The required length is at least 3 characters and no more than 50 characters.
Client name	User-defined name of the client. The name must contain at least 3 characters and may not be longer than 50 characters.
Secret	The **Secret** is created automatically by the system. The client application uses the Client ID and the Secret to authenticate itself with the **Identity Service** successfully. **Attention:** The Secret is only displayed when the entry is initially created. If the entry is edited again at a later time, it is no longer displayed. Ensure that you can also access the Secret later (by saving it as a text file, for example). You need the Secret to configure the client application accordingly. If needed, you can use the dialog to have a new Secret generated.

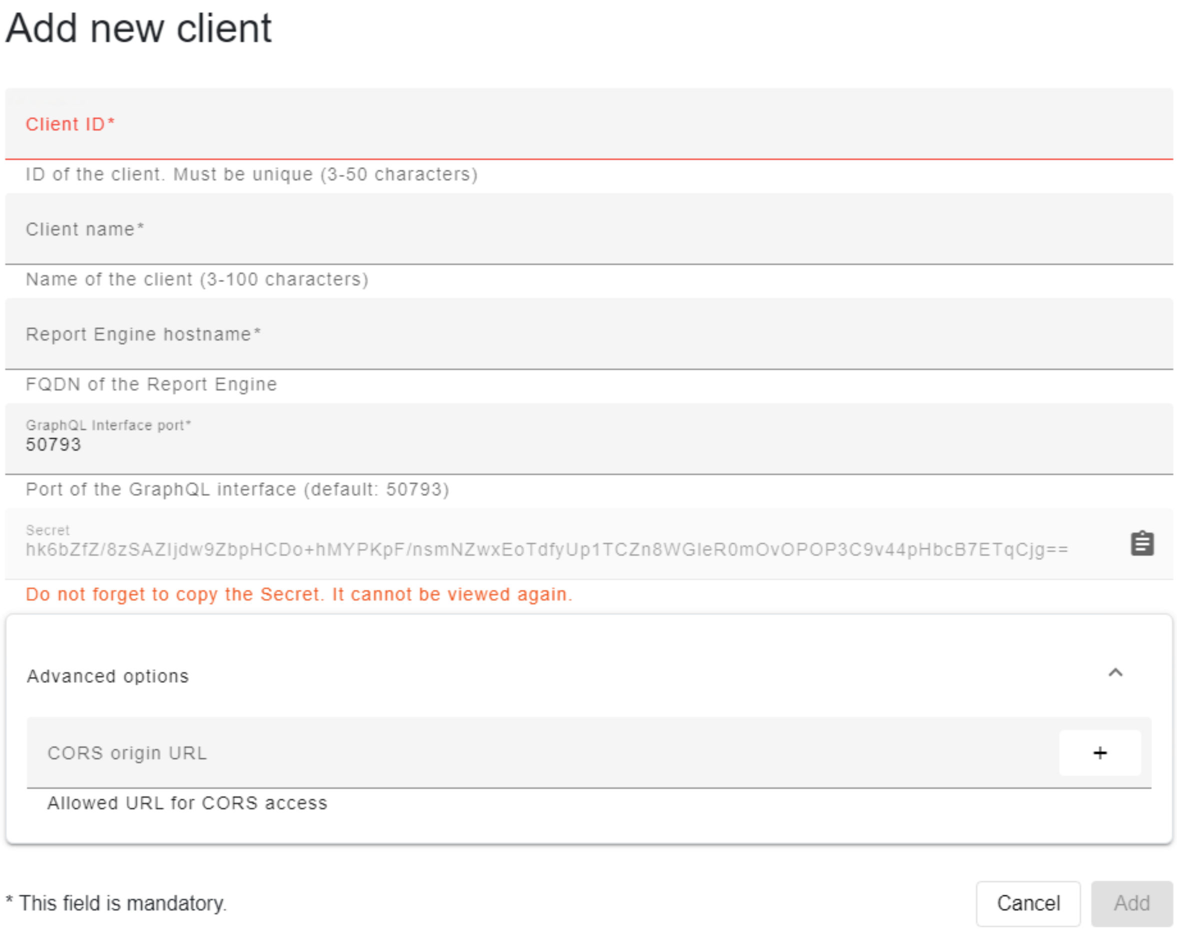
Option	Description
	Note: only available if an existing client is changed. This symbol is not available during the initial configuration of a client.
Renew Secret (symbol)	Generates a new Secret for the client. Each click generates a new Secret. **Note:** This button is only available when editing an already existing client.
Copy to clipboard (symbol)	Copies the Secret to the clipboard.

NAVIGATION

Button	Description
Cancel	Discards all inputs and closes the dialog.
Add	Closes the dialog. The client is created with the current configuration. **Note:** This button is only active if the dialog has been configured with valid inputs.

10.3.6.1.2 Report Engine

In this dialog, you configure a client for Report Engine login for the Identity Service.



Add new client

Client ID*

ID of the client. Must be unique (3-50 characters)

Client name*

Name of the client (3-100 characters)

Report Engine hostname*

FQDN of the Report Engine

GraphQL Interface port*
50793

Port of the GraphQL interface (default: 50793)

Secret
hk6bZfZ/8zSAZIjdw9ZbpHCDo+hMYPKpF/nsmNZwxEoTdfyUp1TCZn8WGIeR0mOvOPOP3C9v44pHbcB7ETqCjg==

Do not forget to copy the Secret. It cannot be viewed again.

Advanced options

CORS origin URL

Allowed URL for CORS access

* This field is mandatory.

Option	Description
Client ID	Unique ID of the client. The ID must be unique and can contain numbers, letters, and special characters. If there is already a client with the same ID, the client will not be created and a corresponding warning will be displayed. The required length is at least 3 characters and no more than 50 characters.
Client name	User-defined name of the client. The name must contain at least 3 characters and may not be longer than 50 characters.
Report Engine hostname	FQDN of the Report Engine that the client will use for the

Option	Description
	login to the Identity Service.
GraphQL Interface port	Port number for communication to the GraphQL interface. Default: *50793*
Secret	The **Secret** is created automatically by the system. The client application uses the Client ID and the Secret to authenticate itself with the **Identity Service** successfully. **Attention:** The Secret is only displayed when the entry is initially created. If the entry is edited again at a later time, it is no longer displayed. Ensure that you can also access the Secret later (by saving it as a text file, for example). You need the Secret to configure the client application accordingly. If needed, you can use the dialog to have a new Secret generated. **Note:** only available if an existing client is changed. This symbol is not available during the initial configuration of a client.
Renew Secret (symbol)	Generates a new Secret for the client. Each click generates a new Secret. **Note:** This button is only available when editing an already existing client.
Copy to clipboard (symbol)	Copies the Secret to the clipboard.

ADVANCED OPTIONS

Option	Description
CORS origin URL	URL of the host or domain used to access the **GraphQL Interface**. **Example:** *https://mywebsite.com* or for on-premises deployment, *https://[FQDN]:12345* This is the only allowed CORS source to access the **Identity Service**..
+	Adds a new entry for the configuration of a new CORS URL.

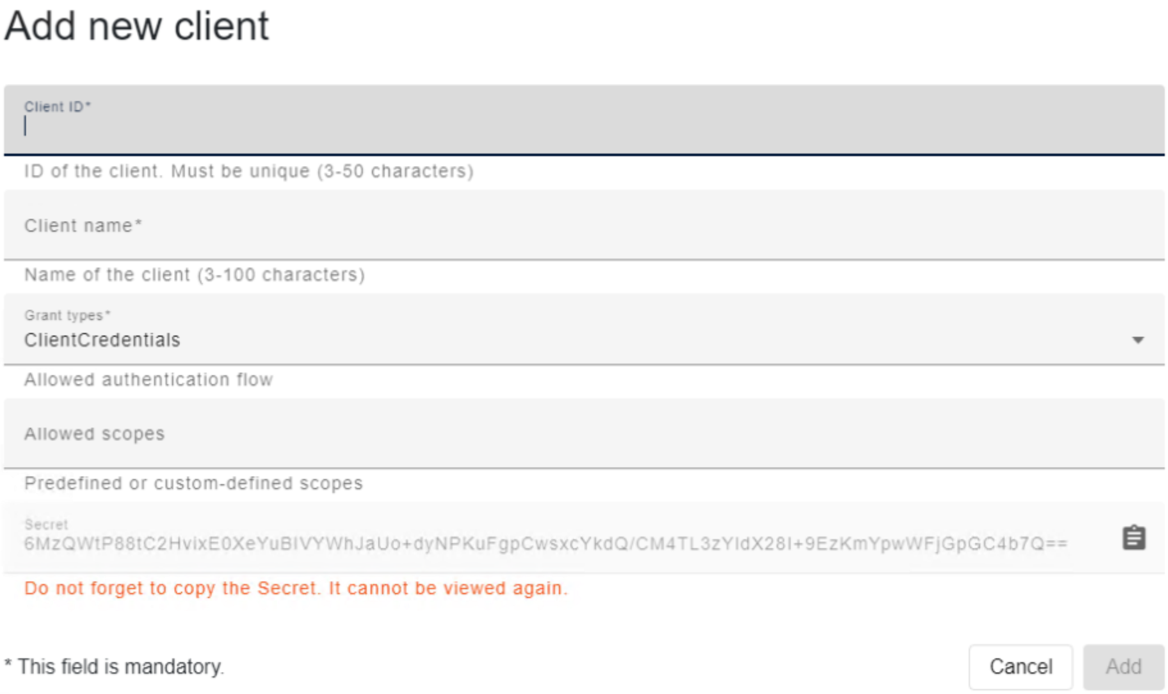
Option	Description
x	Deletes the configuration of the entry.

NAVIGATION

Button	Description
Cancel	Discards all inputs and closes the dialog.
Add	Closes the dialog. The client is created with the current configuration. **Note:** This button is only active if the dialog has been configured with valid inputs.

10.3.6.1.3 Custom OAuth 2.0 client

In this dialog, you configure a client for logging in via OAuth.



Add new client

Client ID*

ID of the client. Must be unique (3-50 characters)

Client name*

Name of the client (3-100 characters)

Grant types*
ClientCredentials

Allowed authentication flow

Allowed scopes

Predefined or custom-defined scopes

Secret
6MzQWtP88tC2HvixE0XeYuBIVYWhJaUo+dyNPKuFgpCwsxcYkdQ/CM4TL3zYIdX28I+9EzKmYpwWFjGpGC4b7Q==

Do not forget to copy the Secret. It cannot be viewed again.

* This field is mandatory.

Option	Description
Client ID	Unique ID of the client. The ID must be unique and can contain numbers, letters, and special characters. If there is already a client with the same ID, the client will

Option	Description
	not be created and a corresponding warning will be displayed. The required length is at least 3 characters and no more than 50 characters.
Client name	User-defined name of the client. The name must contain at least 3 characters and may not be longer than 50 characters.
Grant types	Client authentication method. Select from drop-down list: - *Implicit* - *Code* - *ClientCredential* - *RessourceOwnwerPassword* - *DeviceFlow* - *CodeAndDeveLogin* Default: *ClientCredentials* You can find a detailed overview in the **Grant types** (on page 176) chapter.
Allowed scopes	Configure the access for access tokens. The scope defines which data a client application can access on behalf of the user. Selection from drop-down list; multiple selection is possible: - *openid* - *profile* - *email* - *role* - *groups* - *identityAPI.read_only* - *identityAPI.full_access* - *iiotServicesAPI*

Option	Description
	- *certificateManagementAPI* - *platformConfigurationAPI* - *dataStorageAPI* - *graphQLInterface* - *deviceManagementAPI* - *dataModelingAPI* - *offline_access* You can find a detailed overview of the pre-configured scopes in the **Allowed scopes** chapter.
PCKE requred	Only visible if one of the following scopes has been selected in the **Allowed scopes** option:*Code*, *CodeAndDeviceLogin*
Client secret required	Only visible if one of the following scopes has been selected in the **Allowed scopes** option:*Code*, *DeviceFlow*, *CodeAndDeviceLogin*
Redirect URL	Defines the URLs to which the **Identity Service** can forward the user after successful authentication. This is a security feature. Only visible if one of the following scopes has been selected in the **Allowed scopes** option:*Code*, *CodeAndDeviceLogin*
Secret	The **Secret** is created automatically by the system. The client application uses the Client ID and the Secret to authenticate itself with the **Identity Service** successfully. **Attention:** The Secret is only displayed when the entry is initially created. If the entry is edited again at a later time, it is no longer displayed. Ensure that you can also access the Secret later (by saving it as a text file, for example). You need the Secret to configure the client application accordingly. If needed, you can use the dialog to have a new Secret generated. **Note:** only available if an existing client is changed. This symbol is not available during the initial configuration of

Option	Description
	a client.
Renew Secret (symbol)	Generates a new Secret for the client. Each click generates a new Secret. **Note:** This button is only available when editing an already existing client.
Copy to clipboard (symbol)	Copies the Secret to the clipboard.
Allow access tokens via browser	Here you stipulate whether access tokens are permitted for browser access or not. **Hint:** This option must be activated for browser access or Webview. Only visible if one of the following scopes has been selected in the **Allowed scopes** option:*Device Flow*, *CodeAndDeviceLogin*

ADVANCED OPTIONS

The display of this option depends on the configuration of the **Allowed scopes** option.

Option	Description
Allowed URL for CORS access	URL of the host or domain used to access the **GraphQL Interface**. **Example:** *https://mywebsite.com* or for on-premises deployment, *https://[FQDN]:12345* This is the only allowed CORS source to access the **Identity Service**.. Only visible if one of the following scopes has been selected in the **Allowed scopes** option:*Implicit*, *Code*, *DeviceFlow*, *CodeAndDeviceLogin*
Logout notification Type	This defines the type and manner in which a client signs out the user from **Identity Service** in accordance with OpenID Connect. There are three options in the drop-down menu: - *None* (default setting) - *Frontchannel* - *Backchannel*

Option	Description
	Only visible if one of the following scopes has been selected in the **Allowed scopes** option:*Implicit*, *Code*, *DeviceFlow*, *CodeAndDeviceLogin*
Allowed post logout redirect URL	Defines the URLs to which the **Identity Service** can forward the user after being logged out successfully. This is a security feature. Only visible if one of the following scopes has been selected in the **Allowed scopes** option:*Implicit*, *Code*, *DeviceFlow*, *CodeAndDeviceLogin*
+	Adds a new entry for the configuration of an additional option.
x	Deletes the configuration of the entry.

NAVIGATION

Button	Description
Cancel	Discards all inputs and closes the dialog.
Add	Closes the dialog. The client is created with the current configuration. **Note:** This button is only active if the dialog has been configured with valid inputs.

10.3.6.2 Allowed scopes

Allowed scopes define the access for access tokens. The scope defines which data a client application can access on behalf of the user.

A) SELF-DEFINED SCOPES

You can also define your own scopes. To do this, enter the name of the scope into the text field directly and confirm the process with the **Enter** key.

B) PRE-DEFINED SCOPES

Important: The pre-defined **Allowed scopes** only work in conjunction with certain **Grant types**. Only use permitted combinations of **Allowed scopes** and **Grant types** (on page 177).

Parameters	User context*	Description
openid	yes	You need this scope if you want to log in via OpenID. The *openid* scope shows that the application uses OpenID to verify the identity of the user.
profile	yes	The application can request the user's profile information with the *profile* scope.
email	yes	The application can request the user's email address with the *email* scope.
offline_access	yes	The application can request refresh tokens (on page 176) with the *offline_access* scope. **Note:** The client application can only access IIoT Services online. The name *offline_access* for the scope is misleading in this respect.
iiotServicesAPI	No	The application can access the IIoT API with the *iiotServicesAPI* scope. The application can thus query variables from < NAME_SERVICE_ENGINE> (on page 211), for example.
identityAPI. full_access	No	The *identityAPI.full_access* scope is reserved for system-related purposes.
identityAPI.read_only	No	The *identityAPI.read_only* scope is reserved for system-related purposes.
dataStorageAPI	No	The application may access the data storage with the *dataStorageAPI* scope. The scope is required for the client definition of a Service Engine.
role	yes	
groups	yes	With the *groups* scope, the application can query group dependencies for the logged-in user.
certificateManagementAPI	No	The *certificateManagementAPI* scope is reserved for system-related purposes.
platformConfigur	No	The *platformConfigurationAPI* scope is reserved for

Parameters	User context*	Description
ationAPI		system-related purposes.
graphQLInterface		
deviceManagementAPI	No	The *deviceManagementAPI* scope is reserved for system-related purposes.
dataModellingAPI	No	The *dataModellingAPI* scope is reserved for system-related purposes.

***Grant type** with user context is required

10.3.6.2.1 Refresh tokens and offline_access

An OAuth 2.0 client application can request Refresh Token. The application can remain logged in on a lasting basis with Refresh Token.

TIME VALIDITY OF ACCESS TOKENS

Access tokens for a client application always have a time limit. The application logs out automatically after this time period. The client application can avoid this automatic logout if it requests a new access token by means of a refresh token.

CONFIGURATION OF THE CLIENT

To configure a client application for the Refresh Token request:

1. Create the client in **Identity Management**.
2. Enter the following for the **Allowed scopes** option: *offline_access*.

The client application can thus request Refresh Token.

10.3.6.3 Grant types

You stipulate which authentication flow the client uses in the **Grant types**.

Important: The **Grant types** only work in conjunction with certain **Allowed scopes**. Only use permitted combinations of **Allowed scopes** and **Grant types** (on page 177).

Grant type	User context	Description
Implicit	yes	Interactive authentication: Is required if the client implements the authentication using a browser.

Grant type	User context	Description
Code	yes	Authorization code according to *OAuth 2.0* specification.
ClientCredentials	No	Authentication of the client by means of **Client ID** and **Secret**.
ResourceOwnerPassword	yes	Authentication via user name and password. They are also sent in the authentication request.
DeviceFlow	yes	The *DeviceFlow* option can also be used to enable devices without a browser or with otherwise limited input possibilities to request an Access Token. Is also used for the client definition in Service Engine or Report Engine.
CodeAndPkce	yes	Combination: *Code* and *Pkce*. Pkce is the abbreviation for *Proof Key for Code Exchange*. The use of Pkce increases security in certain cases.

10.3.6.4 Permitted combinations: "Allowed scopes" and "Grant types"

A functional client configuration needs **Allowed scopes** and **Grant types** to suit one another. During configuration, it is necessary to check whether the selected **Grant type** has a user context or not.

Hint

You can find out whether a **Grant type** has a user context from the corresponding table (on page 176).

GRANT TYPES WITH USER CONTEXT

You can combine **Grant types** with a user context (such as *Implicit*, *Code*, *DeviceFlow* for example) with desired **Allowed scopes**.

You can also expressly combine **Grant types** that have user context with **Allowed scopes** that do not have user context.

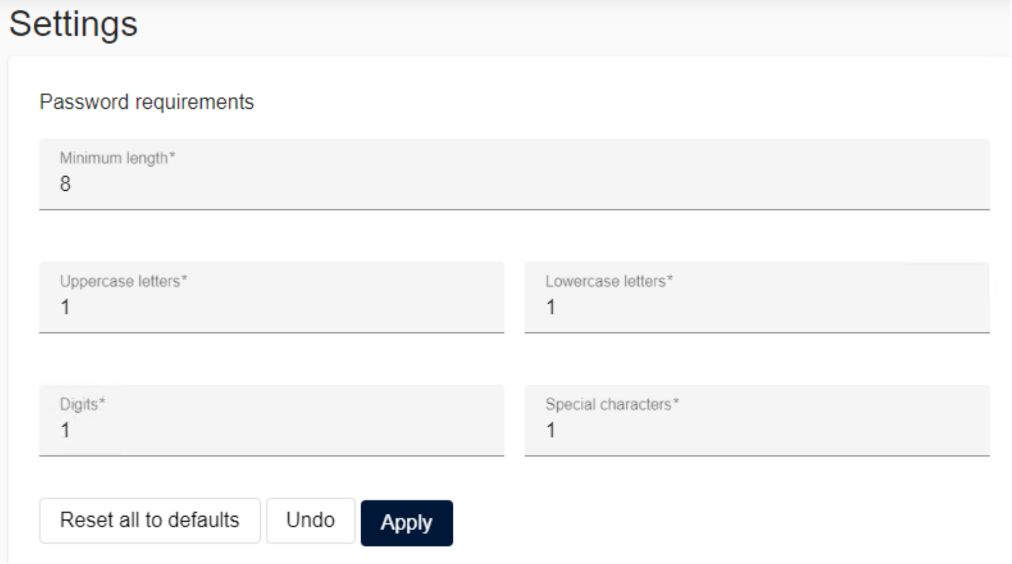
GRANT TYPES WITHOUT USER CONTEXT

You can only combine **Grant types** without user context (*ClientCredentials* for example) with **Allowed scopes** without user context (*serviceGridAPI*, *identityAPI.full_access* for example).

> ⚠**Attention**
>
> It is technically possible to configure a client with a non-permitted combination of **Allowed scopes** and **Grant types**. This leads to authorization problems when accessing the IIoT API however.

10.3.7 Settings

You can configure the password requirements in this dialog.



Settings

Password requirements

Minimum length* 8

Uppercase letters* 1

Lowercase letters* 1

Digits* 1

Special characters* 1

PASSWORD REQUIREMENTS

Option	Description
Minimum length	Minimum number of characters, numbers or special characters that a password must contain in order to be valid. Default: *8*
Uppercase letters	Minimum number of upper-case letters that a password must contain in order to be valid. Default: *1*

Option	Description
Lowercase letters	Minimum number of lower-case letters that a password must contain in order to be valid. Default: *1*
Digits	Minimum number of numbers that a password must contain in order to be valid. Default: *1*
Special characters	Minimum number of special characters that a password must contain in order to be valid. Default: *1*

BUTTONS

Buttons for the configuration of the minimum requirements for a password.

Button	Description
Reset all to defaults	Resets all the configurations for this dialog back to the respective default values. **Note:** Only active if at least one default value has been changed.
Undo	Resets all current and unsaved changes to the configuration back to the respective default value. **Note:** Only active if at least one default value has been changed.
Apply	Applies the current configuration for the minimum requirements for a password. **Note:** Only active if at least one default value has been changed.

Information

The password complexity is applicable to Identity Service users. Users for external providers are subject to the password stipulations of the respective provider.

10.3.8 Navigation bar

<u>**The buttons in the navigation bar offer the following options:**</u>

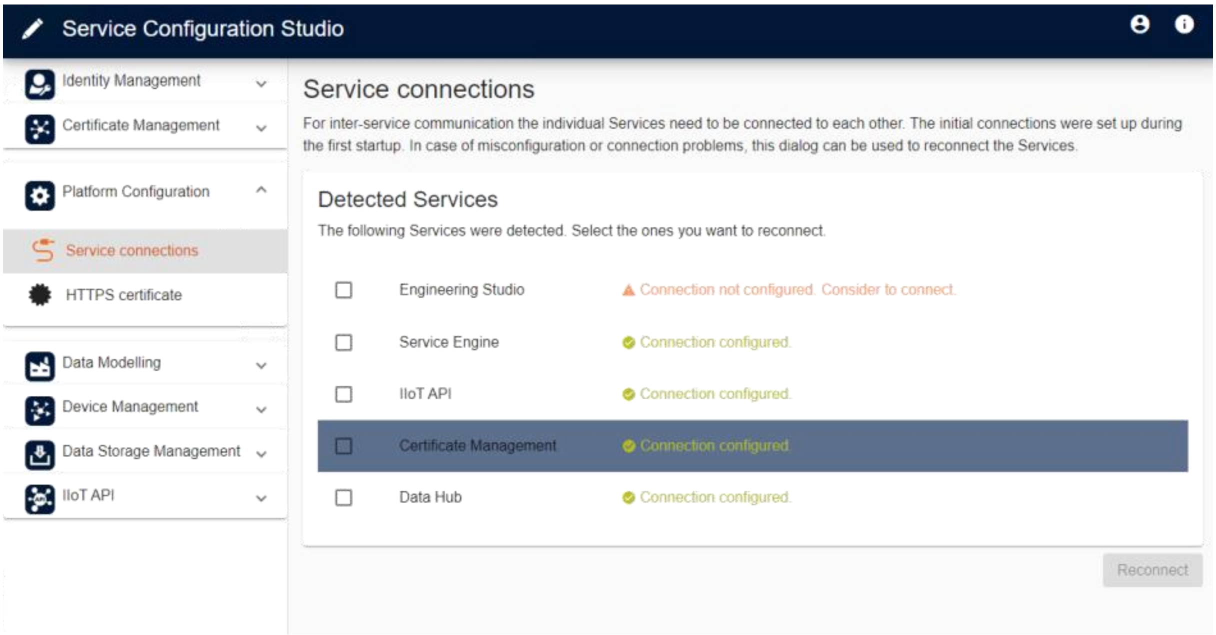
- **User profile**: Directs the logged-in user to their profil page in the **Identity Service** (on page 111).
- **Logout**: Logs out of **Identity Service** (on page 125)
- **About**: Displays the installed version of the **Identity Service** and the license status.

11 Platform configuration

The platform configuration supports the following configurations:

- Reconfiguration of existing connections to IIoT Services.
- Configuration of self-signed or externally-provided HTTPS certificates.

11.1 Connect components



Service connections

For inter-service communication the individual Services need to be connected to each other. The initial connections were set up during the first startup. In case of misconfiguration or connection problems, this dialog can be used to reconnect the Services.

Detected Services

The following Services were detected. Select the ones you want to reconnect.

☐	Engineering Studio	⚠ Connection not configured. Consider to connect.
☐	Service Engine	✓ Connection configured.
☐	IIoT API	✓ Connection configured.
☐	Certificate Management	✓ Connection configured.
☐	Data Hub	✓ Connection configured.

This page lists the services found that connect to the **Data Hub**. In the case of an installation on a Windows operating system, all zenon services that are found on the system and that can be configured for the connections to IIoT Services will also be shown.

⚠Attention

If a new connection for the **Data Hub** is configured, it may take up to a maximum of 60 seconds until the **Data Hub** has applied the changes and properly allows connections to other services again.

LIST OF CONNECTIONS FOUND.

All connections that were found during the initial installation of IIoT Services on the system are listed. If components of the zenon Software Platform are subsequently installed, the list is updated accordingly.

CONNECTION STATE

The state of each of the individual connections is shown in text and color.

- *Connection configured* (in green)
  Configured and valid connections.
- *Connection not configured. Consider to connect* (in orange)
  Connections that were found but were not connected.
- *Connection expired at [date of expiration]* (in red)
  Configured, valid connections whose validity will expire shortly. This display is shown three months before expiry.

RECONNECT BUTTON

This button reconnects existing connections.
Note: Only active if at least one connection has been selected.

LINK NEW CONNECTION

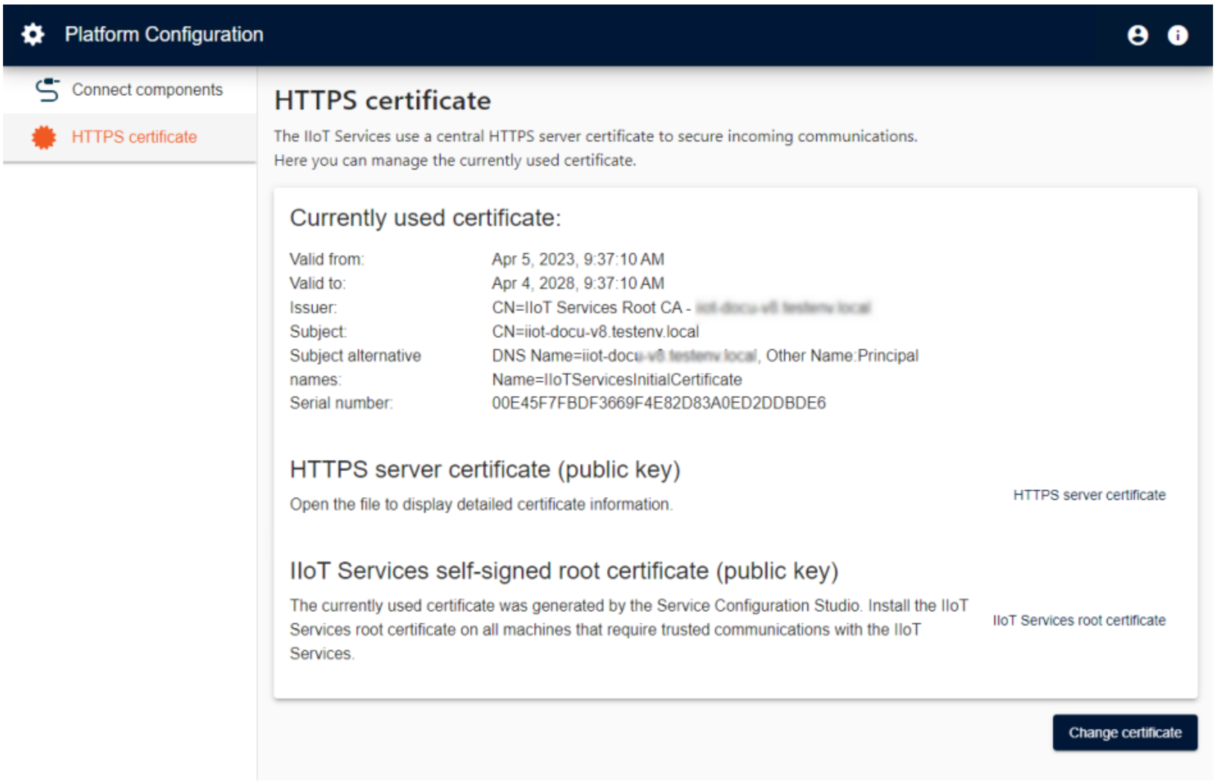
Carry out the following steps if you want to link a new connection.

1. Activate the checkbox for the desired connection.
   Note: Multiple selection is possible.

- Click on the **Reconnect** button.
  The connection is reconnected and the state amended accordingly.

11.2 HTTPS certificate

In this section, you can make changes to the configuration of the HTTPS certificate used.



HTTPS certificate

The IIoT Services use a central HTTPS server certificate to secure incoming communications.
Here you can manage the currently used certificate.

Currently used certificate:

Valid from:	Apr 5, 2023, 9:37:10 AM
Valid to:	Apr 4, 2028, 9:37:10 AM
Issuer:	CN=IIoT Services Root CA - [redacted]
Subject:	CN=iiot-docu-v8.testenv.local
Subject alternative names:	DNS Name=iiot-docu[redacted], Other Name:Principal Name=IIoTServicesInitialCertificate
Serial number:	00E45F7FBDF3669F4E82D83A0ED2DDBDE6

HTTPS server certificate (public key)

Open the file to display detailed certificate information.

IIoT Services self-signed root certificate (public key)

The currently used certificate was generated by the Service Configuration Studio. Install the IIoT Services root certificate on all machines that require trusted communications with the IIoT Services.

The initial certificate is created automatically the first time the **IIoT Services** are started.

HTTPS CERTIFICATE

The following actions are available for the HTTPS certificate:

- Create and use a new IIoT Services Self-signed HTTPS Certificate
- Import and use a Third-party HTTPS Certificate

  Certificate files for a third-party certificate must meet the following requirements:

  - File format: PFX and P12 are permitted
  - Contains private key and public key
  - Supports server authentication (1.3.6.1.5.5.7.3.1)
  - Certificate must be valid

A certificate with or without password protection can be used. If a certificate file is imported without password protection, you must leave the password field empty.

CURRENTLY USED CERTIFICATE

This section shows information about the HTTPS certificate currently being used.

Option	Description
Valid from:	Start date of the validity of the certificate.
Valid to:	End date of the validity of the certificate.
Issuer	Name of the certificate issuer, certification instance and FQDN. Default: *IIoT Services Root CA - [FQDN]*
Subject	FQDN of the certificate owner.
Subject alternative names	Alternative names of certificate holder, for example, additional domains.
Serial number	Unique number for identifying the certificate.

HTTPS SERVER CERTIFICATE (PUBLIC KEY)

You can download the server certificate in this section.

Button	Description
HTTPS server certificate	Downloads the HTTPS server certificate currently being used.

IIOT SERVICES SELF-SIGNED ROOT CERTIFICATE (PUBLIC KEY)

You can download the IIoT Services root certificate in this section.

This section is only displayed if the currently-used HTTPS certificate is a self-signed HTTPS certificate and was created by IIoT Services.

This certificate must be installed for successful trust with IIoT Services. You can find further information in relation to this in the **Trust** (on page 39) section and in the **Configure trust** (on page 41) chapter.

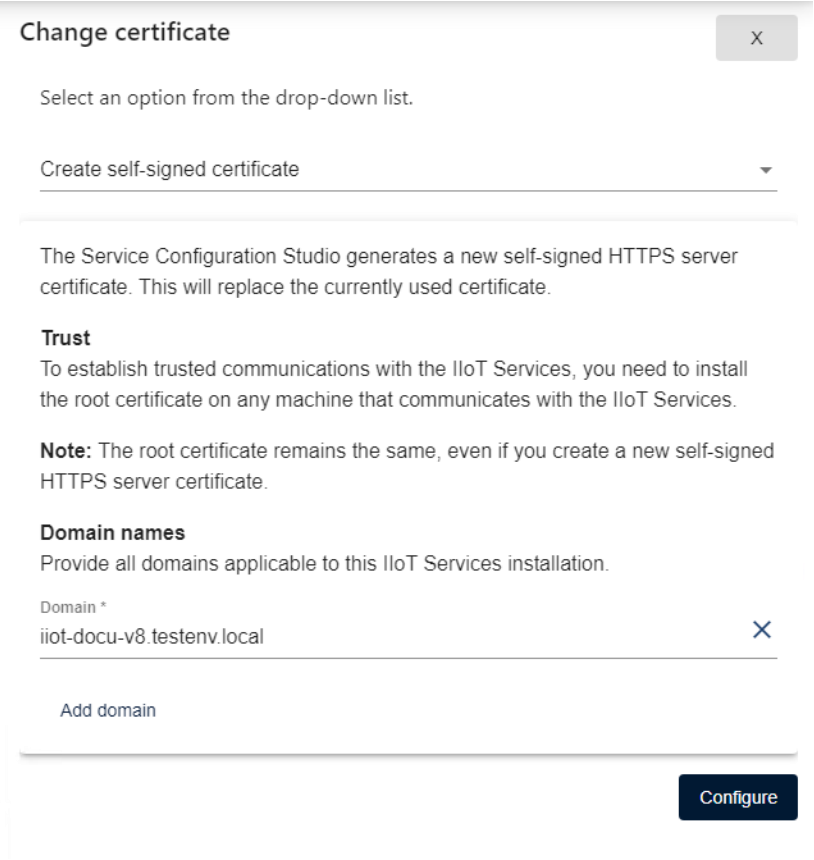
Button	Description
IIoT Services root certificate	Downloads the server certificate currently present.

CHANGE CERTIFICATE

Button to replace the certificate used. Click on Change Certificate to open the **dialog to configure certificates** (on page 184).

11.2.1 Change HTTPS certificate

In this area, you can change the HTTPS certificate used. A new certificate with amended parameters can be created or an existing certificate can be uploaded. The input fields depend on the option selected.



Change certificate

Select an option from the drop-down list.

Create self-signed certificate

The Service Configuration Studio generates a new self-signed HTTPS server certificate. This will replace the currently used certificate.

Trust
To establish trusted communications with the IIoT Services, you need to install the root certificate on any machine that communicates with the IIoT Services.

Note: The root certificate remains the same, even if you create a new self-signed HTTPS server certificate.

Domain names
Provide all domains applicable to this IIoT Services installation.

Domain *
iiot-docu-v8.testenv.local

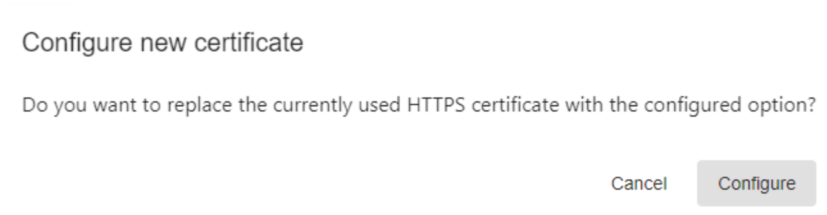
Option	Description
X	Closes the configuration dialog. **Caution:** all unsaved changes are lost.
Certificate type	Selection of the certificate type that is to be edited or uploaded - *Create self-signed certificate* Creates a new self-signed certificate - *Use custom certificate* Uploads an existing certificate. If the certificate used is encrypted, the corresponding password for successful use must be saved in the **Password (optional)** option.
Domain	This value is used as the certificate owner in the

Option	Description
	certificate to be created. Enter the domain name with which IIoT Services can be reached by other computers. **Note:** Only available if *Create self-signed certificate* is selected as the certificate type.
Add domain	Add an input field to configure an additional domain. **Note:** Only available if *Create self-signed certificate* is selected as the certificate type.
X	Removes the selected domain entry. **Note:** Only available if *Create self-signed certificate* is selected as the certificate type.
Import **Drag & Drop the certificate here or click to browse**	Field to upload an already existing certificate: - Clicking on this area opens the file selection dialog to upload an existing certificate file. - Dragging and dropping a certificate file in this area uploads the certificate. Please check the requirements for a certificate in the **Certificates** section. **Note:** Only available if *Use custom certificate* is selected as the certificate type.
Password (optional)	Entry of the certificate password if the uploaded certificate is encrypted with a password. The display is masked and validated. In the case of errors, this is displayed with a red bar under the password entry. **Note:** Only available if *Use custom certificate* is selected as the certificate type.
Eye icon	Shows the password for verification in plain text. **Note:** Only available if *Use custom certificate* is selected as the certificate type.
Configure	Applies the configured settings. With the *Create self-signed certificate* option: Creates the new HTTPS certificate with the parameters entered.

Option	Description
	With the *Use custom certificate* option: Applies the HTTPS certificate supplied. A dialog is shown when clicking on the button.

CONFIRMATION DIALOG

Dialog to apply or discard the new certificate settings.



Configure new certificate

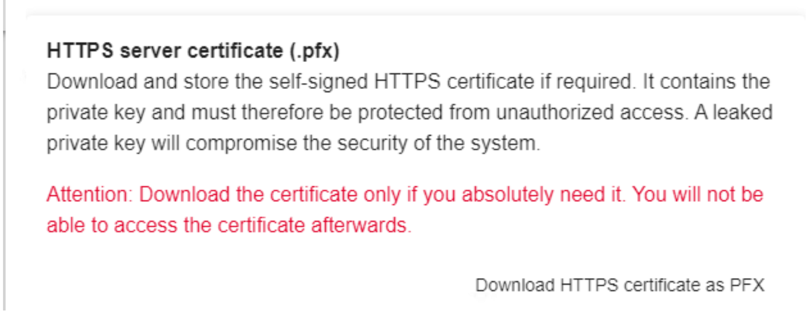
Do you want to replace the currently used HTTPS certificate with the configured option?

Option	Description
Cancel	Discards all changes and closes the dialog.
Configure	Applies settings and closes the dialog.

DIALOG - DOWNLOAD NEW CERTIFICATE

A dialog is also shown after changing a certificate and confirming this with Configure. In this dialog, you can download the newly-used certificate.

> **⚠Attention**
>
> Only download the newly-generated HTTPS certificate if you need it for your storage or data archive. This file contains the private certificate key and must therefore be treated with extreme caution due to security concerns. Loss or unintended publication of this file can lead to considerable security problems in IIoT Services and to potential attacks on your IIoT Services as well as your complete infrastructure.



HTTPS server certificate (.pfx)

Download and store the self-signed HTTPS certificate if required. It contains the private key and must therefore be protected from unauthorized access. A leaked private key will compromise the security of the system.

Attention: Download the certificate only if you absolutely need it. You will not be able to access the certificate afterwards.

Option	Description
Download HTTPS certificate as PFX	The newly-created certificate, including the private key, is downloaded.

11.2.2 Change "Issued by" for custom certificate

The issuer of the certificate is generated with *IIoT Services Root CA* by default. The suffix (= issued by) is created as with - *[FQDN]* by default.

This suffix can be configured in a Docker environment with the help of an environment variable for the **Platform Configuration Service**. This is not included in the docker compose YAML file by default.

Carry out the following configuration in the docker-compose.yaml file in order to configure the suffix using the environment variable:

1. Open the *docker-compose.yaml* file.
2. Add the following entry for the environment variable in the section for the platform configuration service:
   'PLATFORM_CONFIGURATION_Certificate__RootCertificateCustomText=*[desired text for issued by]***'**

Example

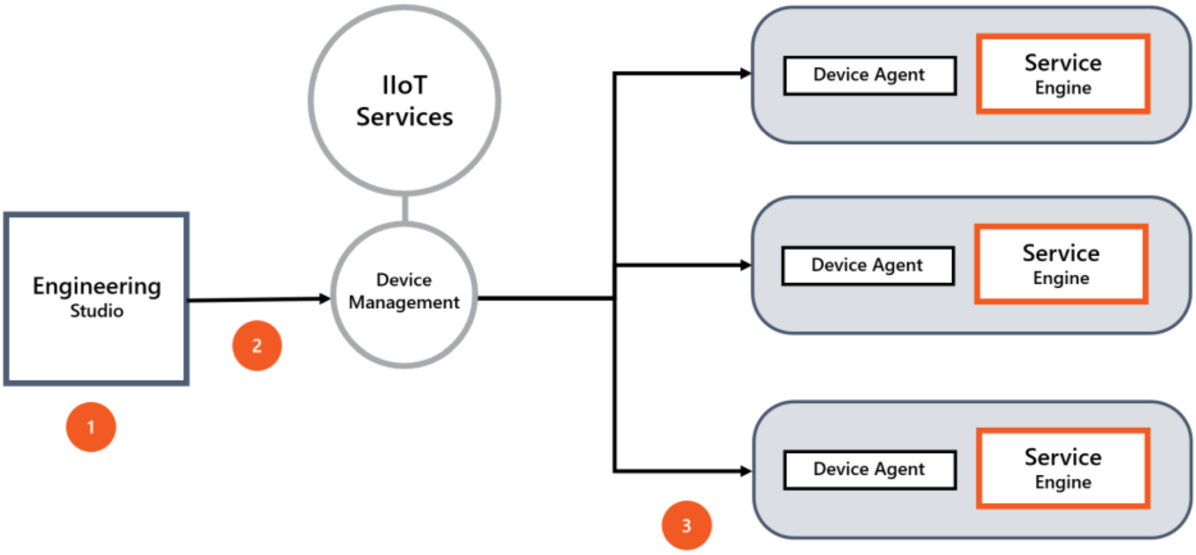
```
...
platform-configuration:
...
 -
'PLATFORM_CONFIGURATION_Certificate__RootCertificateCustomText=xycvxvc
```

APPSETTINGS FILE IN WINDOWS

Alternatively, in Windows, the suffix/issued by can also be amended with a custom definition in the Appsettings file:

```
{
 ...
 ,
 "CertificateOptions": {
 "RootCertificateCustomText": "user-defined text for Issued by]"
 }
}
```

12 Device Management: Monitoring and software deployment



You can use Device Management to centrally manage connected client computers.

Device Management is a service for monitoring and software deployment in the zenon Software Platform.

The following applies for Device Management:

- Each device requires an installed **Device Agent**.
- When executing a deployment task, a check is carried out to ensure that an appropriate Certificate Bundle has been configured on the device if the project has been configured for IIoT Services.
- The software deployment supports updates and configurations for Service Engine.
- The software deployment can be done manually or scheduled.

To deploy software using Device Management:

1. Create and configure a zenon project in Engineering Studio.
2. Upload the project from Engineering Studio to **Device Management**. This creates a deployable software package.
3. The software package can be deployed on any number of Service Engine instances.

Device Management also allows you to efficiently manage large environments.

12.1 Device Management

Device Management is used to supply Service Engine files for Service Engine on devices.

1. Create and configure a zenon project in Engineering Studio. These projects are prepared as software packages using a wizard and are transferred to **Device Management** of the IIoT Services for distribution.
2. The software package can be deployed on any number of Service Engine instances. You use the **Device Management** service in Service Configuration Studio to configure the deployment to the devices.
   This deployment can also be done scheduled.
   Device Management offers a user interface and an overview of the available devices and software packages as well as a configuration interface for the listing and planning of the deployment.
3. An appropriate service is registered on the devices to make the devices accessible to **Device Management**. Both Windows operating systems as well as Linux and Raspberry are supported as devices.
   Note: You can find a list of supported operating systems in the **Installation and updates** section in the **Linux** chapter.

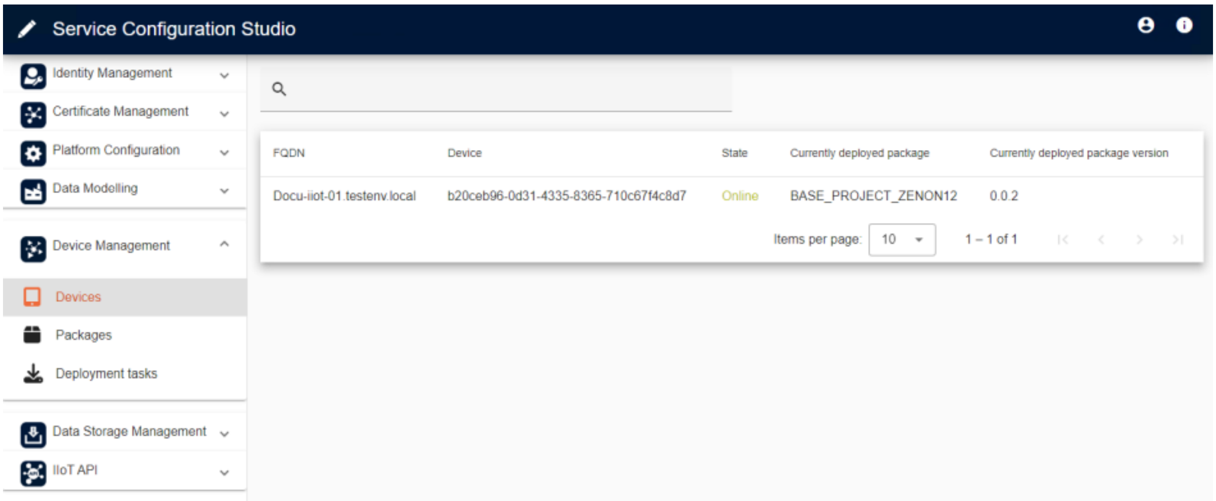
PROCEDURE

The following steps are carried out on the device when executing a **Deployment task**:

1. Software packages are transferred to a device with a **Deployment Task**. The project packages are saved on the device in the Default folder for zenon projects.
   - Windows default folder: *C:\Users\Public\Documents\zenon_Projects*.
   - Linux default folder: */etc/copa-data*
2. Any Service Engine running on the device is stopped.
3. The current deployment task project is installed on the device and set as a start project.
4. The Service Engine is started again.
   Note: If Service Engine had not yet been started before execution of the deployment task, Service Engine is started after successful transfer.

12.1.1 Devices

List of all registered devices for **Device Management** on the IIoT Services.



FQDN	Device	State	Currently deployed package	Currently deployed package version
Docu-iiot-01.testenv.local	b20ceb96-0d31-4335-8365-710c67f4c8d7	Online	BASE_PROJECT_ZENON12	0.0.2

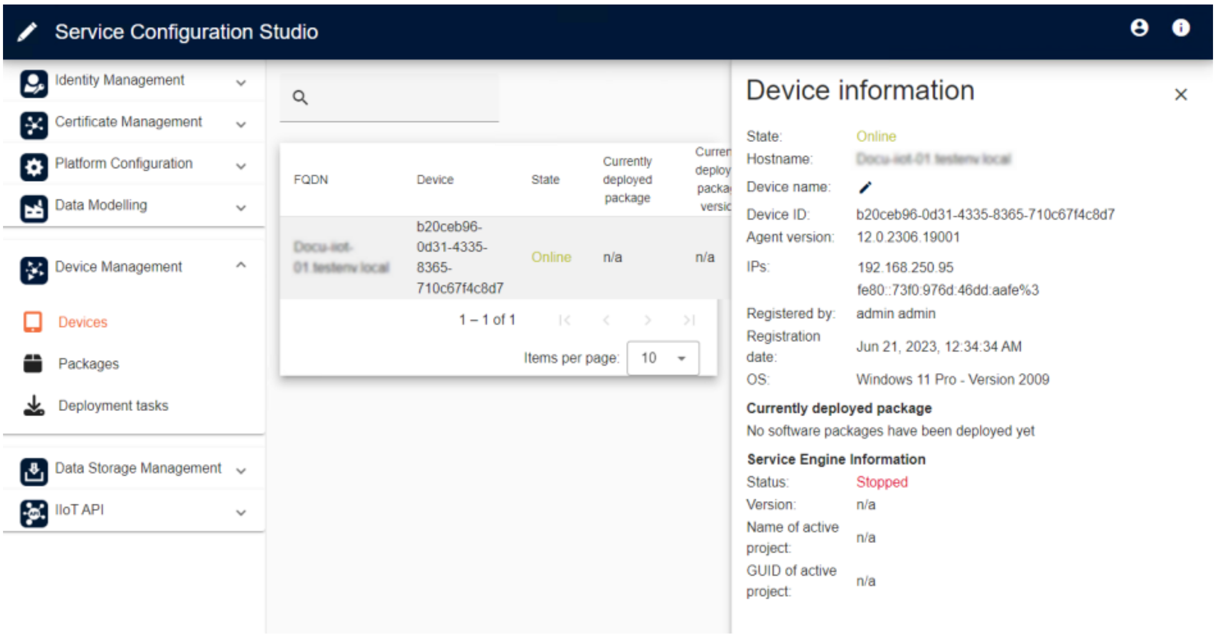
Items per page: 10 1 – 1 of 1

The items of this list can be sorted by clicking on the column heading.

Option	Description
FQDN	FQDN of the device. Is applied by the device when registering the **Device Agent**.
Device ID	Unique ID of the client on IIoT Services. Is created automatically on the device when registering the **Device Agent** .
State	State of the device. The state is displayed as colored text. - *Registered* (black) Device has been configured for **Device Management** but cannot currently receive any deployment tasks. - *Offline* (red) Device cannot be reached. - *Online* (green) Device can be reached; Service Engine service for **Device Management** is running.
Currently deployed package	Name of the package that is currently installed on the device. **Note:** If no deployment task has been executed on the device yet, the entry *n/a* is shown.

Option	Description
Currently deployed package Version	Version of the package that is currently installed on the device. **Note:** If no deployment task has been executed on the device yet, the entry *n/a* is shown.

Clicking on the item opens a tab with detailed information.



FQDN	Device	State	Currently deployed package	Curren deploy packa versic
[redacted]	b20ceb96-0d31-4335-8365-710c67f4c8d7	Online	n/a	n/a

1 – 1 of 1

Items per page: 10

Device information

State:	Online
Hostname:	[redacted]
Device name:	
Device ID:	b20ceb96-0d31-4335-8365-710c67f4c8d7
Agent version:	12.0.2306.19001
IPs:	192.168.250.95 fe80::73f0:976d:46dd:aafe%3
Registered by:	admin admin
Registration date:	Jun 21, 2023, 12:34:34 AM
OS:	Windows 11 Pro - Version 2009

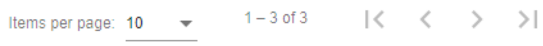
Currently deployed package

No software packages have been deployed yet

Service Engine Information

Status:	Stopped
Version:	n/a
Name of active project:	n/a
GUID of active project:	n/a

NAVIGATION AND STATUS BAR



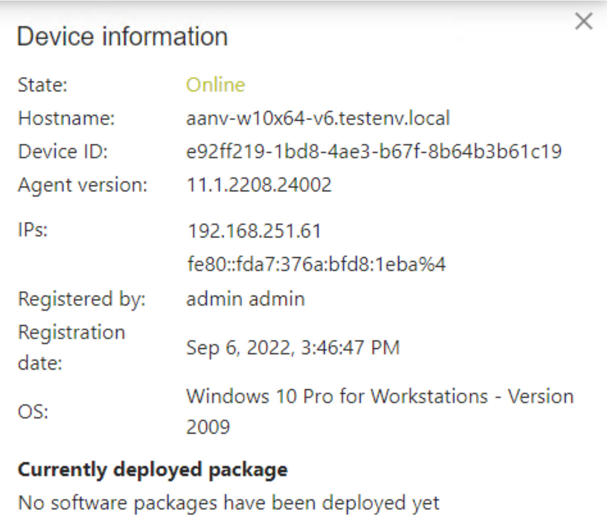
The status bar allows you to customize the view of the respective page and to navigate in the list view.

Option/symbol	Description
Items per page	Number of items shown per page. Selection from a drop-down list.
[a] - [b] of [c]	Sum of all available items and information on the items shown: - *[a]*: Number of the first item shown - *[b]*: Number of the last item shown - *[c]*: Sum of all items

Option/symbol	Description
Zur ersten Seite (|<)	Jumps to first page of the list view.
Vorherige Seite (<)	Jumps to previous page of the list view. **Note:** Not available on the first page.
Nächste Seite (>)	Jumps to the next page of the list view. **Note:** Not available on the last page.
Zur letzten Seite (>|)	Jumps to the last page of the list view.

12.1.1.1 Detail view - Devices

The detail view provides you with additional information about the device.



Device information

State:	Online
Hostname:	aanv-w10x64-v6.testenv.local
Device ID:	e92ff219-1bd8-4ae3-b67f-8b64b3b61c19
Agent version:	11.1.2208.24002
IPs:	192.168.251.61 fe80::fda7:376a:bfd8:1eba%4
Registered by:	admin admin
Registration date:	Sep 6, 2022, 3:46:47 PM
OS:	Windows 10 Pro for Workstations - Version 2009

Currently deployed package

No software packages have been deployed yet

Option	Description
State	State of the device. The state is displayed as colored text. ▸ *Registered* (black) Device has been configured for **Device Management** but cannot currently receive any deployment tasks. ▸ *Offline* (red) Device cannot be reached. ▸ *Online* (green)

Option	Description
	Device can be reached; Service Engine service for **Device Management** is running.
Hostname	FQDN of the device. Is applied by the device when registering the **Device Agent**.
Device ID	Unique ID of the client on IIoT Services. Is created automatically on the device when registering the **Device Agent** .
Agent version	Version of the **Device Agent** service that is installed on the device.
IPs	IP address of the client. Is applied by the device when registering the **Device Agent**.
Registered by	User name of the user who has registered the service on Device Management.
Registration date	Date of registration on Device Management with the current Device Management service.
OS	Operating system of the client. Is applied by the device when registering the **Device Agent**.

CURRENTLY-DEPLOYED PACKAGE

Option	Description
Currently deployed package	Information on the current software package provided: - Name of the software package which has been delivered most recently to the client. - If no software package has been transferred to the client yet, a corresponding message is displayed: *No software packages have been deployed yet*

SERVICE ENGINE INFORMATION

Option	Description
Status	State of the Service Engine on the device. The state is displayed as colored text. - *running* (green) Service Engine is installed and running on the device.

Option	Description
	- *stopped* (red) Service Engine is installed on the device but has not been started. - *No Service Engine installed* (red) No valid installation of Service Engine has been found on the device.
Version	Version of Service Engine on the device.
Name of active project	Name of the active project that is running in Service Engine on the device.
GUID of active project.	GUID of the project that is running in Service Engine on the device.

NAVIGATION - CLOSE DIALOG

Button	Description
X	Closes the detail view.

12.1.1.2 Configure device for Device Management (Windows)

In order for software packages to be deployed for a device, an appropriate IIoT Services service must be registered and running on the device. Name of the service: **CopaData.ServiceGrid.DeviceManagement.Agent**.

- The service must be registered on the device.
  The destination address of the Device Agent service must be configured during registration. After correct registration, the device is visible in **Device Management** on the **Devices** page.
  Note: Note that it may take some time until the newly-registered device is displayed in the device overview (on page 192).
- The service must run on the device.
  The service must be running on the device to deploy or deliver a software package.

INSTALLATION

Carry out the following steps in order to install and register the service:

1. Ensure that there is appropriate trust between the device and IIoT Services.

2. Register the required service.
   - Go to directory *C:\Program Files\Common Files\COPA-DATA\ServiceGridCli\12_0>*.
   - Start the Windows **PowerShell** application as administrator.
   - Enter the following command:
     .\CopaData.ServiceGrid.DeviceManagement.CLI.exe setup-agent -u https://*[IIoT Services URL]:[IIoT Services port]*
     .\CopaData.ServiceGrid.DeviceManagement.Agent.exe setup-agent -u https://[IIoT Services URL]:[IIoT Services port]
     Example: *.\CopaData.ServiceGrid.DeviceManagement.CLI.exe setup-agent -u https://iiot-docu-v8.testenv.local:9443*
3. Authenticate yourself with the **Identity Service**.
   a) When authenticating via web browser:
      Enter user name and password if the web browser opens with the login page.
      Note: This step does not take place if you are already logged in to **Identity Service**.
   b) When authenticating on another client:
      Enter the following command during installation, if web access is not possible on the device on which you want to install the Device Agent:
      C:\Program Files\Common Files\COPA-DATA\ServiceGridCli\12_0>
      .\CopaData.ServiceGrid.DeviceManagement.CLI.exe setup-agent -u https://iiot-docu-v8.testenv.local:9443 --use-device-code
      Authorization is carried out by means of Identity Service in Service Configuration Studio (on page 197).
      The installation on the client will be completed after successful authorization.

   After successful registration, the device is listed in the overview of devices.
   Note: When updating the state in Device Management, there can be delays of several minutes.

> **Hint**
>
> If the *CopaData.ServiceGrid.DeviceManagement.CLI.exe* is started in the command line without parameters, an appropriate help text is displayed in the **Command Line Interface**.

12.1.1.3 Install Device Agent for Linux

The **Device Agent** is installed together with the IIoT command line interface.

Attention

Device Management is not available for Service Engine in Docker environments.

You can find detailed instructions for installation of the Service Engine on Linux systems in the **Service Engine on Linux** section.

Carry out the following steps to install the service for **Device Management** (= **Device Agent**) on a Linux device:

1. Update the list of the zenon software packages available. Execute the following command:
   sudo apt update
2. Install the command line interface on the Linux device. Execute the following command:
   sudo apt install iiot-cli
3. Carry out the installation of the service. To do this, enter the following command:
   a) If you have web access for authentication on the Linux device:
   iiot-cli setup-agent -u [URL to the IIoT Services]:Port
   Example: *iiot-cli setup-agent -u https://iiot-docu-v8.testenv.local:9443*

   b) If you do not have web access for authentication on the Linux device:
   iiot-cli setup-agent -u https://iiot-docu-v8.testenv.local:9443 --use-device-code

   The **--use-device-code** command tag is used for authorization via the Identity Service in Service Configuration Studio (on page 197).
   In the command line, the URL and code for this authorization is visualized.
   After successful authorization, the installation on the Linux device will be finished.
4. Check the status of the device agent:
   - Check the connection status of the device in the user interface of **Device Management** (on page 192) device administration in Service Configuration Studio. The Linux device must display the status Online.
   - On the Linux computer, the status of the device agent is checked with the following command:
     sudo systemctl status device-agent.service

Information

Ensure that the following language settings (**Locales**) are installed on your system:

- *en_US.UTF-8*

- *UTF-8/en_US.UTF-8*

- *UTF-8*

If these **Locales** are missing, it can happen that the **Device Agent** is closed with the following error message:

terminate called without an active exception Aborted

12.1.1.4 Authorization via Identity Service

When installing services on devices without web access, authentication can be carried out on another computer or device (a smartphone for example).

Carry out the following steps to authorize devices:

a) Open the URL as stated in the command line of the CLI.
Example: "*To login please go to 'https://iiot-docu-v8.testenv.local:9443/identity-service/device' and enter the code: '539083363' to login.*"

b) Log in to the Identity Service in Service Configuration Studio.

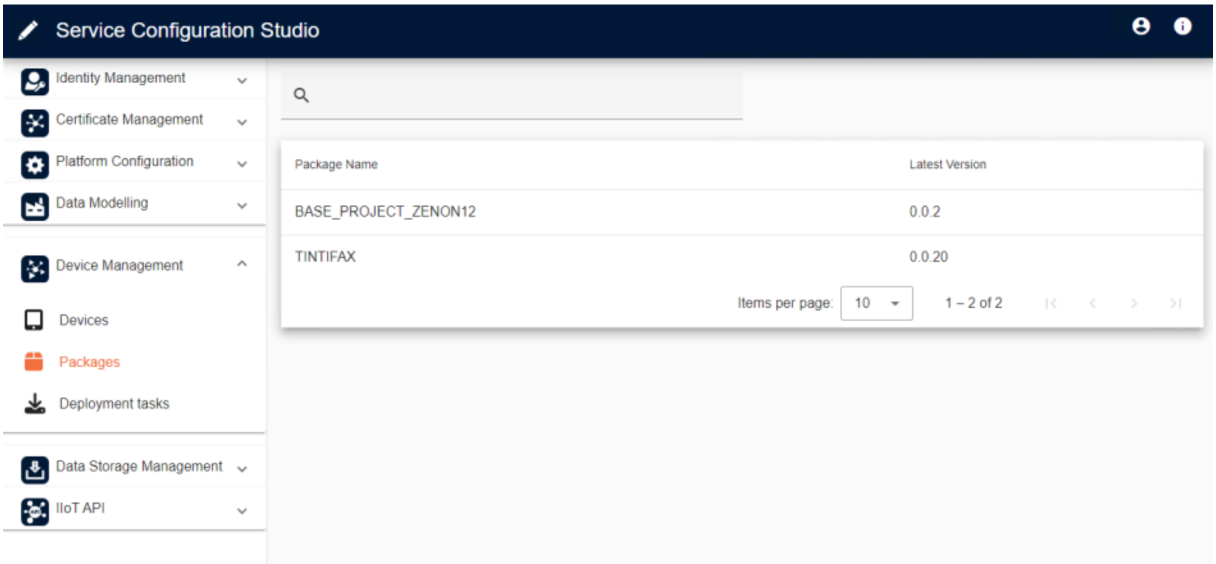
c) Enter the code, as stated in the command line interface, into the authorization dialog.

d) Confirm your input by clicking on the Send button.

e) After confirmation, the installation of the device agent on the device will be continued automatically.

12.1.2 Packages

List of all available software packages for **Device Management** on the IIoT Services.



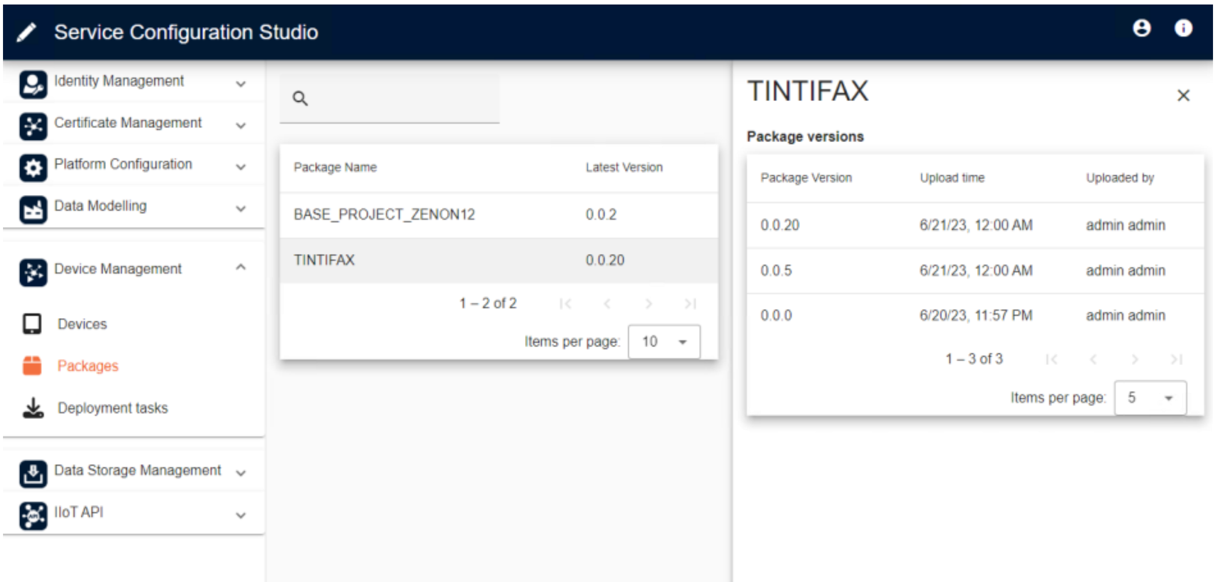
Package Name	Latest Version
BASE_PROJECT_ZENON12	0.0.2
TINTIFAX	0.0.20

The items of this list can be sorted by clicking on the column heading.

Option	Description
Package Name	Name of the software package.
Latest Version	Number of the latest version of the software package.

PROJECT - DETAILED INFORMATION

Clicking on the item opens a tab with detailed information.



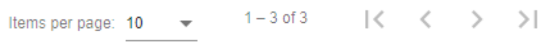
Package Name	Latest Version
BASE_PROJECT_ZENON12	0.0.2
TINTIFAX	0.0.20

Package versions

Package Version	Upload time	Uploaded by
0.0.20	6/21/23, 12:00 AM	admin admin
0.0.5	6/21/23, 12:00 AM	admin admin
0.0.0	6/20/23, 11:57 PM	admin admin

Option	Description
Package Version	Version of the zenon project. The version number is based on the **Versioning active** project property in Engineering Studio. This must be activated in order to be able to send valid package versions. You can find detailed information in relation to this in the **Project backup** section in the **Versioning** chapter.
Update time	Timestamp from when the package of the Engineering Studio was transferred with the wizard.
Uploaded by	**Identity Service** user who transferred the project of the Engineering Studio with the wizard.

NAVIGATION AND STATUS BAR



Items per page: 10

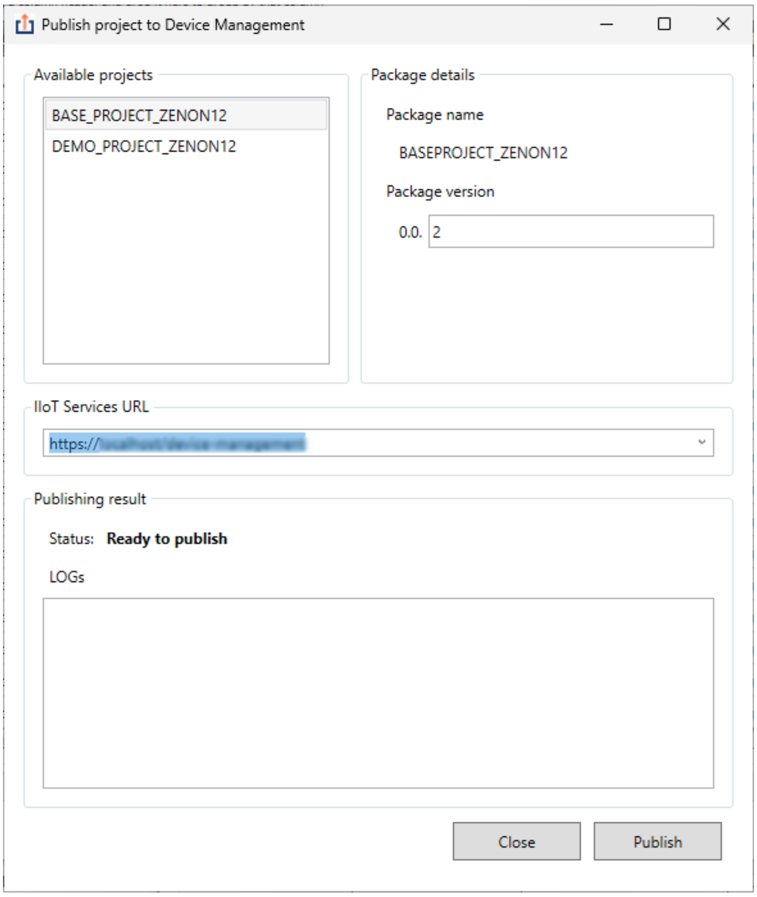
1 – 3 of 3

The status bar allows you to customize the view of the respective page and to navigate in the list view.

Option/symbol	Description
Items per page	Number of items shown per page. Selection from a drop-down list.
[a] - [b] of [c]	Sum of all available items and information on the items shown: - *[a]*: Number of the first item shown - *[b]*: Number of the last item shown - *[c]*: Sum of all items
Zur ersten Seite (|<)	Jumps to first page of the list view.
Vorherige Seite (<)	Jumps to previous page of the list view. **Note:** Not available on the first page.
Nächste Seite (>)	Jumps to the next page of the list view. **Note:** Not available on the last page.
Zur letzten Seite (>|)	Jumps to the last page of the list view.

12.1.2.1 Deploying zenon projects for Device Management

The deployment of packages for **Device Management** is implemented in Engineering Studio with a wizard.



BASE_PROJECT_ZENON12
DEMO_PROJECT_ZENON12

Package name
BASEPROJECT_ZENON12

Package version
0.0. 2

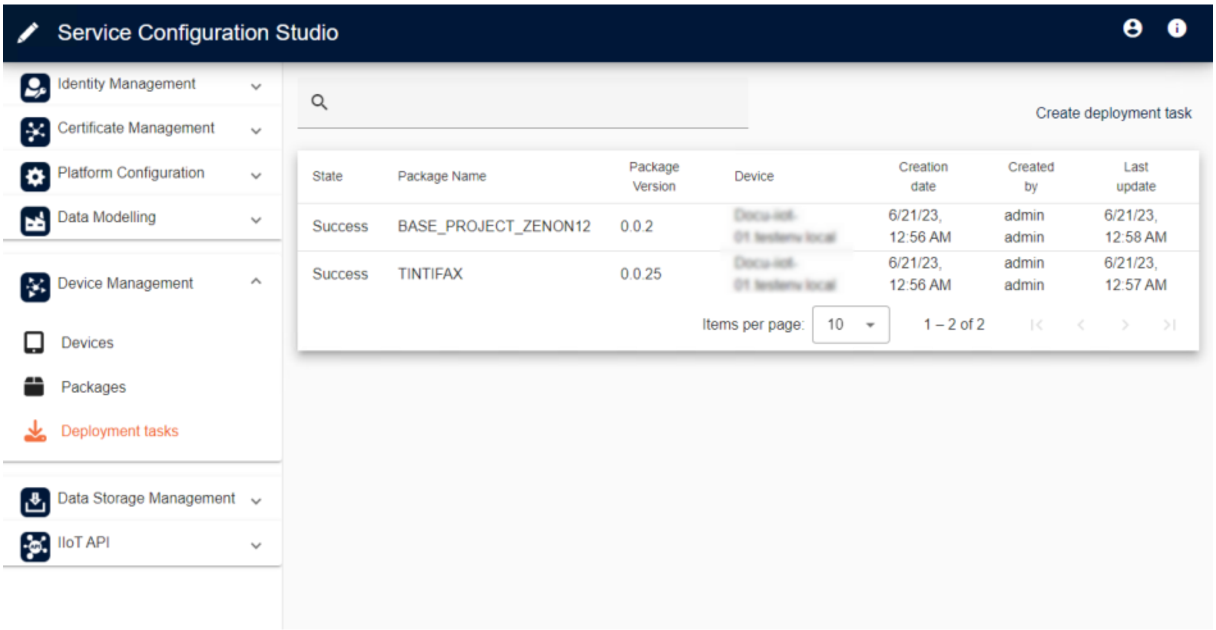
https://[redacted]

Status: **Ready to publish**

You can find detailed information for the transfer of a project from Engineering Studio in the **Device Management** section in the **IIoT Services - configuration in Engineering Studio** chapter.

12.1.3 Deployment task

List of all **Device Management** deployment tasks on IIoT Services.



State	Package Name	Package Version	Device	Creation date	Created by	Last update
Success	BASE_PROJECT_ZENON12	0.0.2	[redacted]	6/21/23, 12:56 AM	admin admin	6/21/23, 12:58 AM
Success	TINTIFAX	0.0.25	[redacted]	6/21/23, 12:56 AM	admin admin	6/21/23, 12:57 AM

BUTTON: CREATE DEPLOYMENT TASK

Opens the tab for configuring a new deployment task.

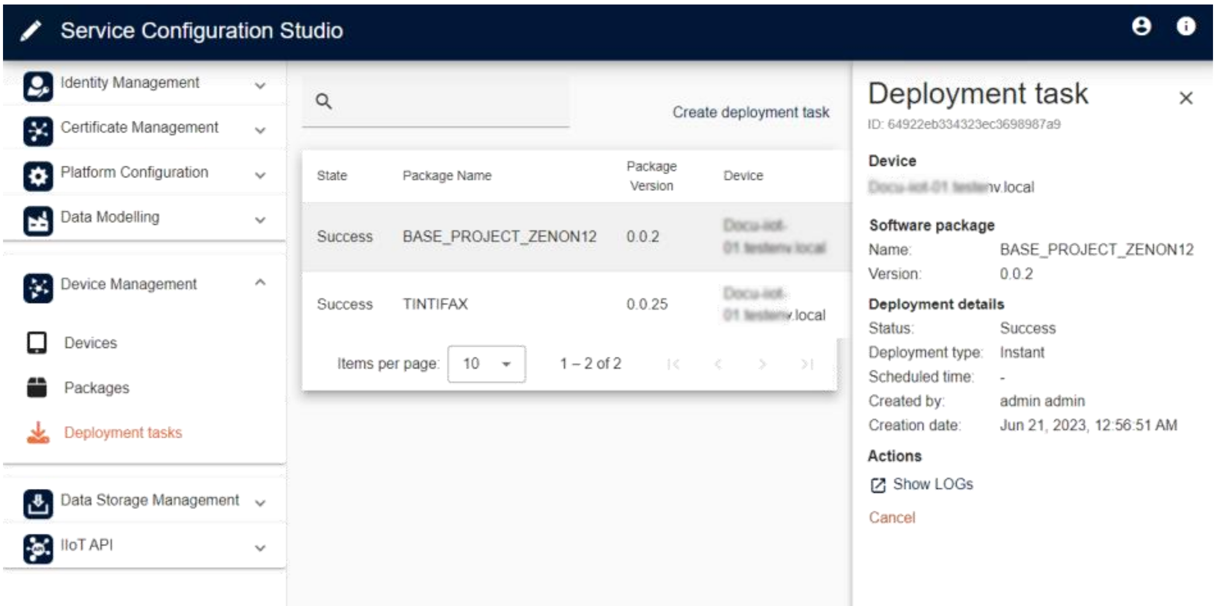
LIST: DEPLOYMENT TASKS

The items of this list can be sorted by clicking on the column heading.

Option	Description
State	Current state of the deployment task ▶ *Success* The deployment task could be successfully performed. The software package has been transferred to the device ▶ *Pending* The deployment task is planned but has not been successfully performed yet. ▶ *Cancelling* A planned deployment task which has not been performed yet (previous state was pending) is canceled. ▶ *Canceled*

Option	Description
	A previously planned deployment task has been successfully canceled.
Package Name	Name of the software package.
Package Version	Version of the software package which is delivered to the device by the deployment task.
Device	FQDN of the target device to which the software package is delivered.
Creation date	Timestamp of the creation of the deployment task.
Created by	User name of the user who has created the deployment task.
Last update	Timestamp of the latest update of the deployment task.

Clicking on the item opens a tab with detailed information.



State	Package Name	Package Version	Device
Success	BASE_PROJECT_ZENON12	0.0.2	[redacted]
Success	TINTIFAX	0.0.25	[redacted]v.local

Items per page: 10 1 – 2 of 2

Deployment task
ID: 64922eb334323ec3698987a9

Device
[redacted]v.local

Software package

Name:	BASE_PROJECT_ZENON12
Version:	0.0.2

Deployment details

Status:	Success
Deployment type:	Instant
Scheduled time:	-
Created by:	admin admin
Creation date:	Jun 21, 2023, 12:56:51 AM

Actions

Show LOGs

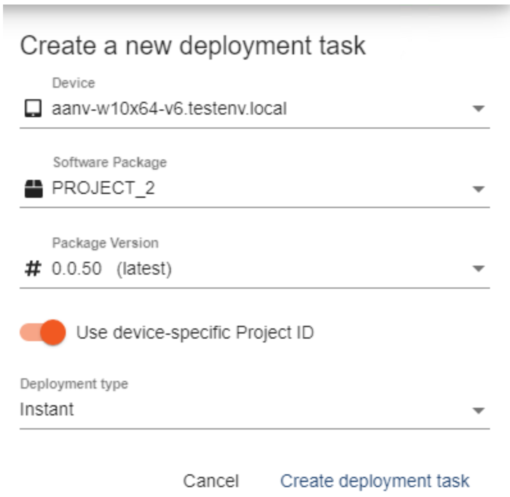
Cancel

12.1.3.1 Create deployment tasks

You can configure the transfer (=deployment) of software packages on this page. The options correspond to existing configurations:

- Selection of devices corresponds to the list on the **Device** page.
- Selection of the software packages to be deployed corresponds to the list on the **Software Package** page.

Clicking on the **Create deployment task** button opens the tab for configuring a deployment task.



Create a new deployment task

Device: aanv-w10x64-v6.testenv.local

Software Package: PROJECT_2

Package Version: 0.0.50 (latest)

Use device-specific Project ID

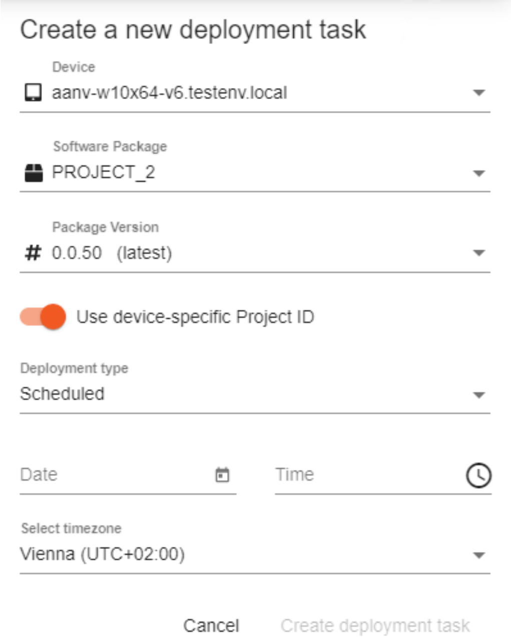
Deployment type: Instant

Option	Description
Device	List of all available clients. Select from drop-down list.
Sofware Package	List of all available software packages. Select from drop-down list. **Note:** Only available if a **Device** has been selected.
Package Version	List of versions of the selected software package. Select from drop-down list. The latest version is marked with the additional text **(latest)**. **Note:** Only available if a **Software Package** has been selected.
Use device-specific Project ID	Option for the automatic generation of a unique GUID for the deployment task. ▸ *Active:* Generates a new, unique GUID. This GUID is used as the project GUID. If the same project is distributed to several devices and these devices connect to IIoT Services, this GUID is used for unique identification. ▸ *Inactive*: No new, unique GUID is generated.
Deployment type	Type of deployment. Select from drop-down list. ▸ *Instant* Software package is transferred immediately to the selected devices.

Option	Description
	- *Scheduled* Software package is transferred based on a schedule. If this deployment type is selected, additional configuration options are displayed for configuring the schedule.

CONFIGURATION OPTIONS FOR THE “SCHEDULED” DEPLOYMENT TYPE

If the Scheduled deployment type is selected, additional options are displayed.



Create a new deployment task

Device: aanv-w10x64-v6.testenv.local

Software Package: PROJECT_2

Package Version: 0.0.50 (latest)

Use device-specific Project ID

Deployment type: Scheduled

Date

Time

Select timezone: Vienna (UTC+02:00)

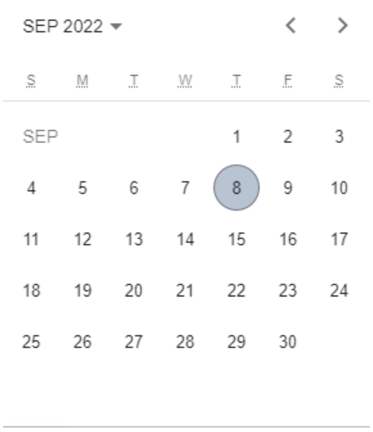
Option	Description
Date	Date on which the deployment task is to be performed. Selection in a calendar dialog.
Time	Time when the deployment task is to be performed. Selection in a clock dialog.
Select timezone	Selection of the time zone that applies for the date and time. The deployment task is performed on the device on the configured date at the configured time using the time zone selected here. Select from drop-down list.

NAVIGATION

Option	Description
Cancel	Discards the configurations and closes the configuration.
Create deployment task	Applies the configuration and creates a new deployment task. **Note:** This button is only available if the deployment task has been completely configured and all options have been configured.

12.1.3.1.1 Select date in the calendar display

You can configure the date on which the deployment task is to be performed in a calendar display. This view is opened automatically by the **Date** option. In the header of the dialog, you can click to switch between the month and year view.



SEP 2022 ▾ ‹ ›

S	M	T	W	T	F	S
SEP				1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	

Month view for selecting the date on which the deployment task is to be performed.

- The current date is displayed with a light blue circle.
- The date on which the task is to be performed is displayed with a dark blue circle.

Option	Description
Year view/month view	Depending on the current view, the years or months of the selected year are displayed. After selecting a year, the view automatically switches to the month view. - *Year view:* After selecting the year, all months of that year are displayed for selection.

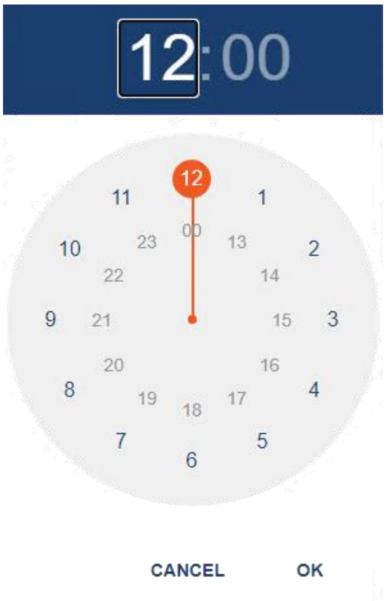
Option	Description
	- *Month view:* Calendar month view for selecting the execution date. Click to switch between the views.
Left arrow (symbol <)	Switches to the previous month in the month view.
Right arrow (symbol >)	Switches to the following month in the month view.

CLOSE DIALOG

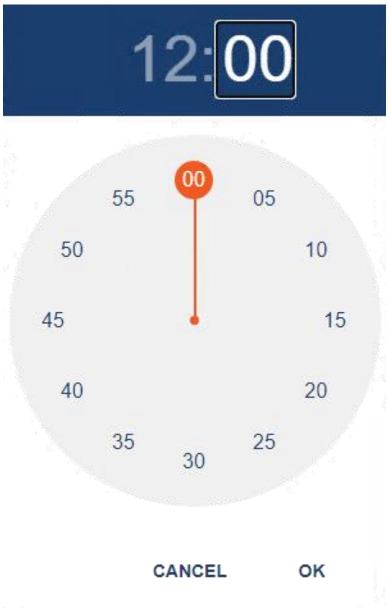
Option	Description
OK	Applies settings and closes the dialog.
Cancel	Discards all changes and closes the dialog.

12.1.3.1.2 Select time in the clock display

You can configure the time when the deployment task is to be performed in a clock display. This view is opened automatically by the **Time** option. In the header of the dialog, you can click to switch between the hours and minutes. If the hour is selected first, the display will automatically switch to the minute view.



Select the hour by clicking on the clock display. In the case of a 24-hour format, select the afternoon hours in the inner circle.



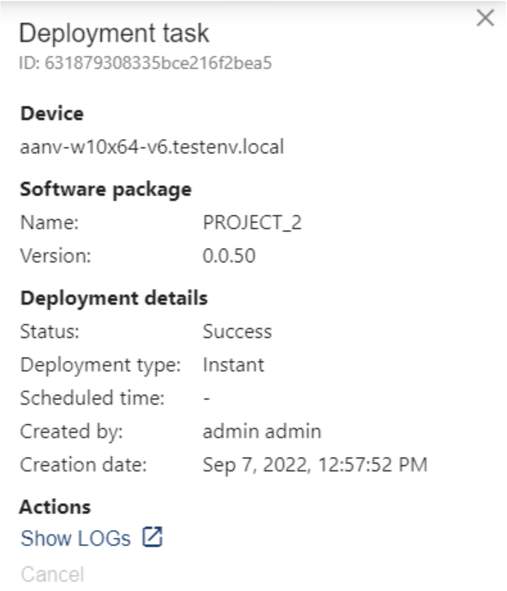
Select the minutes by clicking on the clock display.

CLOSE DIALOG

Option	Description
OK	Applies settings and closes the dialog.
Cancel	Discards all changes and closes the dialog.

12.1.3.2 Detail view - Deployment task

The detail view provides you with additional information about the the deployment task.



Deployment task
ID: 631879308335bce216f2bea5

Device
aanv-w10x64-v6.testenv.local

Software package

Name:	PROJECT_2
Version:	0.0.50

Deployment details

Status:	Success
Deployment type:	Instant
Scheduled time:	-
Created by:	admin admin
Creation date:	Sep 7, 2022, 12:57:52 PM

Actions
Show LOGs
Cancel

DEPLOYMENT TASK

The ID of the deployment task is shown as a header of the detail view.

DEVICE

ID of the target device to which the software package contained in the deployment task is delivered.

SOFTWARE PACKAGE

Option	Description
Name	Name of the software package delivered by the deployment task.
Version	Version number of the software package delivered by the deployment task.

DEPLOYMENT DETAILS

Option	Description
Status	Current state of the deployment task - *Success*

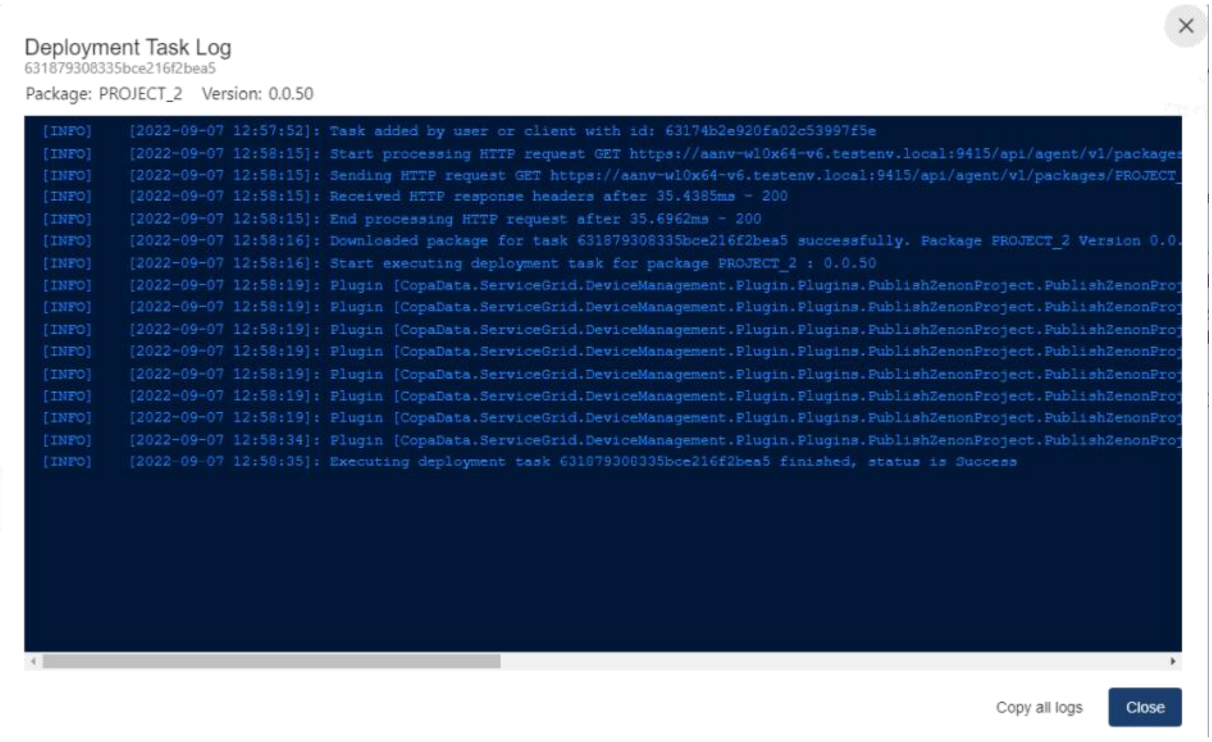
Option	Description
	The deployment task could be successfully performed. The software package has been transferred to the device - *Pending* The deployment task is planned but has not been successfully performed yet. - *Cancelling* A planned deployment task which has not been performed yet (previous state was pending) is canceled. - *Canceled* A previously planned deployment task has been successfully canceled.
Deployment type	Type of delivery: - *Instant*: The deployment task is performed immediately. The selected software package is delivered to the device immediately after the deployment task is created. - *Scheduled*: The deployment task is performed at a configured time.
Scheduled time	Timestamp of the planned execution of the deployment task. **Note:** This information is not available for the Instant deployment type. This is displayed with a hyphen (-)..
Created by	User name of the user who has created the deployment task.
Creation date	Timestamp of the creation of the deployment task.

ACTIONS

Option	Description
Show LOGs	Opens a new window with all LOG entries for the deployment task.
Cancel	This button has several uses: - *Creation of a new deployment task:* Closes the detail view.

Option	Description
	- *Editing of an already existing deployment task:* If an already configured deployment task with a schedule (*Scheduled*) is reopened, the pending deployment is canceled.

SHOW LOGS - DETAIL VIEW



Deployment Task Log

631879308335bce216f2bea5

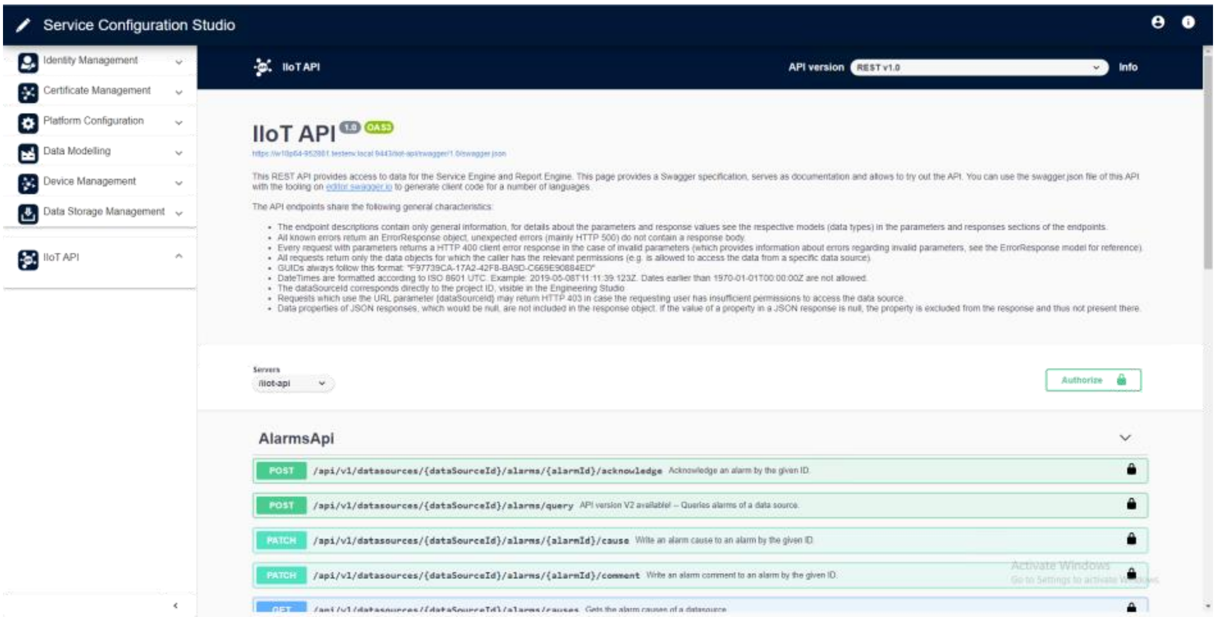
Package: PROJECT_2   Version: 0.0.50

```
[INFO]    [2022-09-07 12:57:52]: Task added by user or client with id: 63174b2e920fa02c53997f5e
[INFO]    [2022-09-07 12:58:15]: Start processing HTTP request GET https://aanv-w10x64-v6.testenv.local:9415/api/agent/v1/packages
[INFO]    [2022-09-07 12:58:15]: Sending HTTP request GET https://aanv-w10x64-v6.testenv.local:9415/api/agent/v1/packages/PROJECT_
[INFO]    [2022-09-07 12:58:15]: Received HTTP response headers after 35.4385ms - 200
[INFO]    [2022-09-07 12:58:15]: End processing HTTP request after 35.6962ms - 200
[INFO]    [2022-09-07 12:58:16]: Downloaded package for task 631879308335bce216f2bea5 successfully. Package PROJECT_2 Version 0.0.
[INFO]    [2022-09-07 12:58:16]: Start executing deployment task for package PROJECT_2 : 0.0.50
[INFO]    [2022-09-07 12:58:19]: Plugin [CopaData.ServiceGrid.DeviceManagement.Plugin.Plugins.PublishZenonProject.PublishZenonProj
[INFO]    [2022-09-07 12:58:19]: Plugin [CopaData.ServiceGrid.DeviceManagement.Plugin.Plugins.PublishZenonProject.PublishZenonProj
[INFO]    [2022-09-07 12:58:19]: Plugin [CopaData.ServiceGrid.DeviceManagement.Plugin.Plugins.PublishZenonProject.PublishZenonProj
[INFO]    [2022-09-07 12:58:19]: Plugin [CopaData.ServiceGrid.DeviceManagement.Plugin.Plugins.PublishZenonProject.PublishZenonProj
[INFO]    [2022-09-07 12:58:19]: Plugin [CopaData.ServiceGrid.DeviceManagement.Plugin.Plugins.PublishZenonProject.PublishZenonProj
[INFO]    [2022-09-07 12:58:19]: Plugin [CopaData.ServiceGrid.DeviceManagement.Plugin.Plugins.PublishZenonProject.PublishZenonProj
[INFO]    [2022-09-07 12:58:19]: Plugin [CopaData.ServiceGrid.DeviceManagement.Plugin.Plugins.PublishZenonProject.PublishZenonProj
[INFO]    [2022-09-07 12:58:34]: Plugin [CopaData.ServiceGrid.DeviceManagement.Plugin.Plugins.PublishZenonProject.PublishZenonProj
[INFO]    [2022-09-07 12:58:35]: Executing deployment task 631879308335bce216f2bea5 finished, status is Success
```

NAVIGATION

Button	Description
Copy all LOGs	Copies all LOG entries of the LOG window to the clipboard.
Close	Closes the LOG window.

13 IIoT API



IIoT API

https://w10p64-952881.testenv.local:9443/iiot-api/swagger/1.0/swagger.json

This REST API provides access to data for the Service Engine and Report Engine. This page provides a Swagger specification, serves as documentation and allows to try out the API. You can use the swagger.json file of this API with the tooling on editor.swagger.io to generate client code for a number of languages.

The API endpoints share the following general characteristics:

- The endpoint descriptions contain only general information, for details about the parameters and response values see the respective models (data types) in the parameters and responses sections of the endpoints.
- All known errors return an ErrorResponse object, unexpected errors (mainly HTTP 500) do not contain a response body.
- Every request with parameters returns a HTTP 400 client error response in the case of invalid parameters (which provides information about errors regarding invalid parameters, see the ErrorResponse model for reference).
- All requests return only the data objects for which the caller has the relevant permissions (e.g. is allowed to access the data from a specific data source).
- GUIDs always follow this format: "F97739CA-17A2-42F8-BA9D-C669E90884ED"
- DateTimes are formatted according to ISO 8601 UTC. Example: 2019-05-08T11:11:39.123Z. Dates earlier than 1970-01-01T00:00:00Z are not allowed.
- The dataSourceId corresponds directly to the project ID, visible in the Engineering Studio.
- Requests which use the URL parameter {dataSourceId} may return HTTP 403 in case the requesting user has insufficient permissions to access the data source.
- Data properties of JSON responses, which would be null, are not included in the response object. If the value of a property in a JSON response is null, the property is excluded from the response and thus not present there.

AlarmsApi

```
POST   /api/v1/datasources/{dataSourceId}/alarms/{alarmId}/acknowledge  Acknowledge an alarm by the given ID.
POST   /api/v1/datasources/{dataSourceId}/alarms/query  API version V2 available! -- Queries alarms of a data source.
PATCH  /api/v1/datasources/{dataSourceId}/alarms/{alarmId}/cause  Write an alarm cause to an alarm by the given ID.
PATCH  /api/v1/datasources/{dataSourceId}/alarms/{alarmId}/comment  Write an alarm comment to an alarm by the given ID.
GET    /api/v1/datasources/{dataSourceId}/alarms/causes  Gets the alarm causes of a datasource.
```

The IIoT API allows you to easily connect external **Clients** to the zenon Software Platform using the REST API.

<u>**Clients include for example:**</u>

- Mobile Apps
- Web applications
- Manufacturing Execution Systems (MES)
- Enterprise Resource Planning Systems (ERP)

You can integrate any application into your zenon network via IIoT Services. For clients to be able to access project data, for example, you must grant relevant permissions in **Access control** (on page 134).

FUNCTIONALITIES AND DOCUMENTATION

The IIoT API currently supports the following functionalities:

Service Engine

- Querying and writing of real-time data
- Querying of archive data
- Querying and confirmation of alarms including equipment groups (as JSON array) as well as resources label.
- Querying of chronological events including equipment groups (as JSON array) as well as resources label.

Report Engine

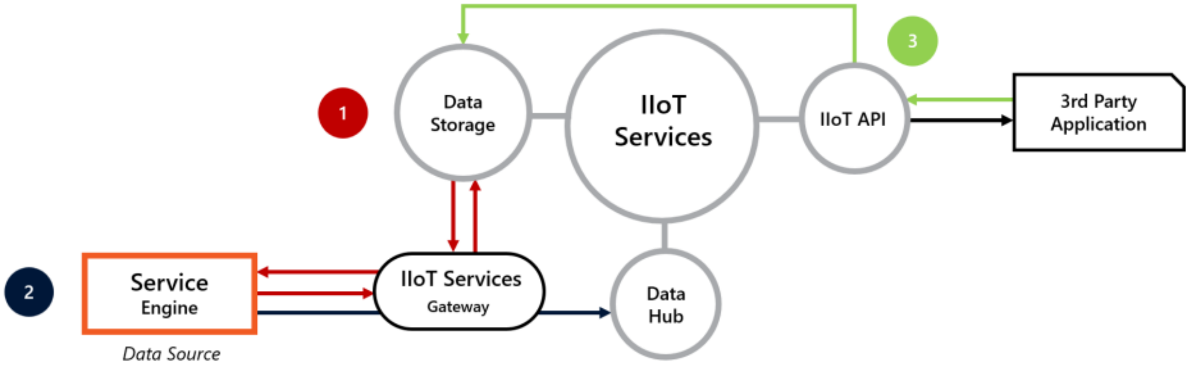
- Triggering and querying of Reports
- Triggering and querying of SQL elements

Data Storage

- Querying of archive data

All IIoT API functions are documented in detail in the Swagger help of Service Configuration Studio.

13.1 Service Engine - third-party application: Provide process data



Process data entered into the IIoT Services can be processed by any third-party application via the IIoT API.

The IIoT API allows the connection of third-party applications. Third-party applications are applications that are not part of the zenon software platform.

Third-party applications can process data from one or more Service Engine instances using the IIoT API.

AREA OF APPLICATION

Third-party applications include for example:

- Mobile Apps
- Web applications
- Manufacturing Execution Systems (MES)
- Enterprise Resource Planning Systems (ERP)

You can in principle connect any interfaceable application to the IIoT API.

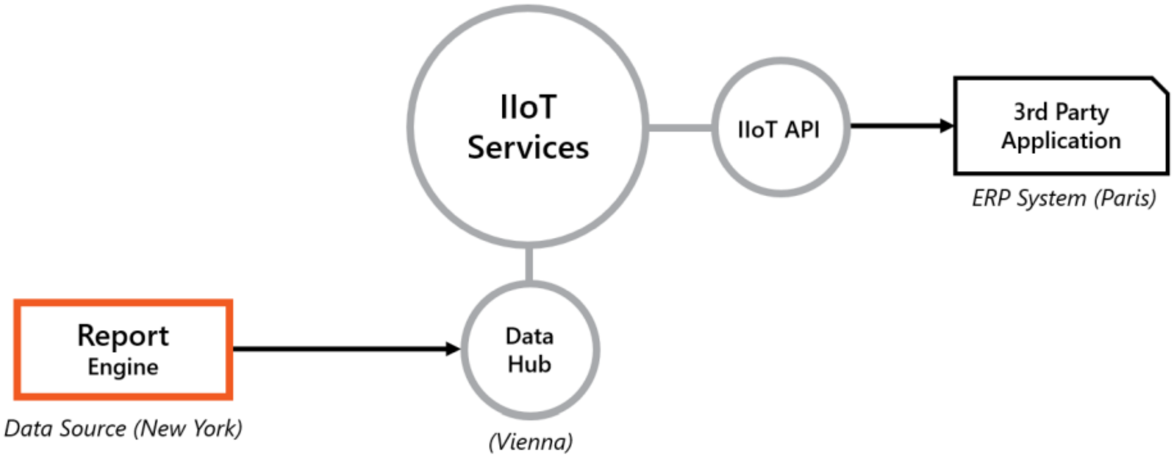
SUPPORTED DATA ACTIONS

The IIoT API supports the following data actions in Service Engine:

Supported data action	Variable access authorization*
Read alarms	*Read only*
Acknowledge alarms	*Read-write*
Comment on alarms	*Read-write*
Set causes of alarms	*Read-write*
Variables - read**	*Read only*
Variables - write**	*Read-write*
Read archive data	*Read only*
Read events	*Read-only****
Comment on events	*Read-write*

* Required access authorization in Service Engine (data source).
** * Simple variable type (no structure variables, no arrays).
*** No access authorization is required for system events.

13.2 Report Engine > third-party application: Provide report data



The IIoT API allows you to enable third-party applications to access the data of Report Engine.

Report Engine can provide report data for third-party applications via IIoT Services. Third-party applications access resources in IIoT Services via **REST Interface**.

AREA OF APPLICATION

The IIoT API allows you to access the data of Report Engine using any third-party application.

Third-party applications include for example:

- Mobile Apps
- Web applications
- Manufacturing Execution Systems (MES)
- Enterprise Resource Planning Systems (ERP)

You can in principle connect any interfaceable application to the IIoT API.

SUPPORTED DATA ACTIONS

The IIoT API allows you to perform various data actions in Report Engine.

Supported data actions:

- Reports - execute
- Reports - read results

- Database queries - execute
- Database queries - read results

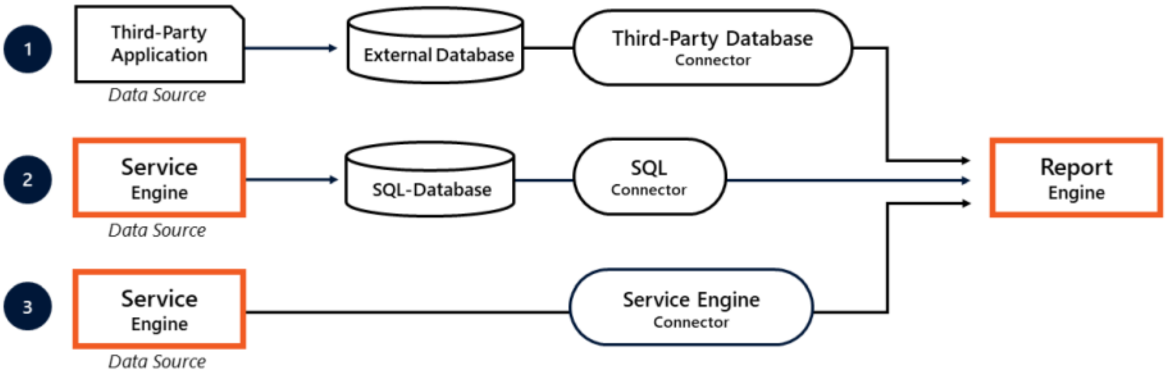
CONFIGURATIONS

The following configurations are required for reports and database queries:

- Reporting Studio: Configuration in the **Service Node Interface** dialog
- Service Configuration Studio: Configuration of the **Report Permissions**

It is thus ensured that IIoT Services can access Report Engine reports and databases.

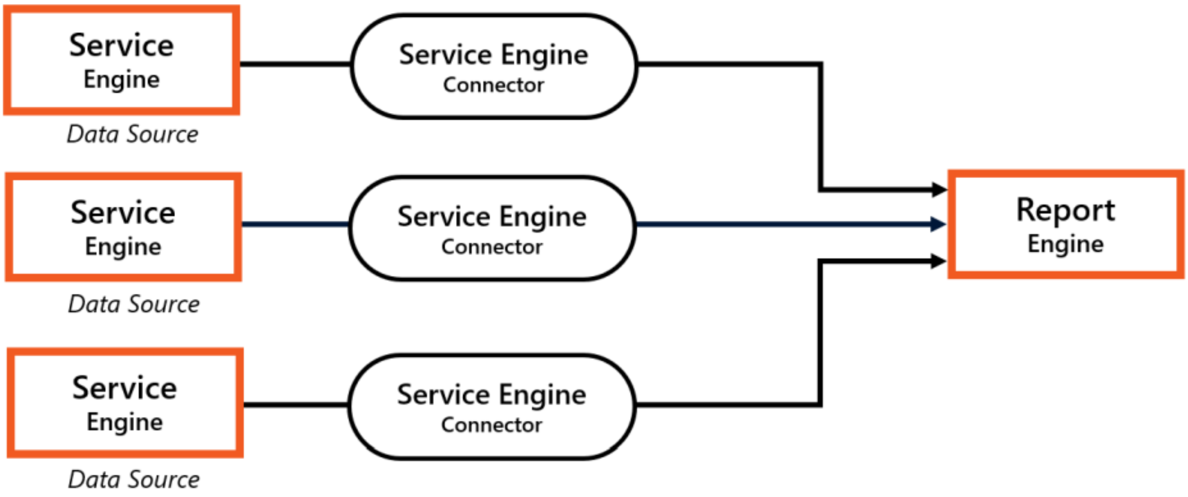
13.2.1 Data sources for Report Engine



Report Engine can evaluate the following data sources:
1) Data from third-party applications from an external database.
2) Process data of a Service Engine from a SQL database.
3) Process data of a Service Engine (without an intermediate database).

Report Engine can read and evaluate data from different sources. The connectors necessary for the connection depend on the particular use case.

13.2.2 Multiple Service Engine instances



Report Engine can collect and evaluate data from different Service Engine instances.

Report Engine allows you to collect data from different sources and evaluate it centrally. Thus, for instance, it is possible to connect multiple instances of Service Engine with Report Engine.

13.3 IIoT API

You can test the API using the interactive Swagger API documentation. The documentation contains all necessary information for use of the API.

The user must be logged on to **Identity Service** in order to use the interactive documentation.

The commands are summarized in groups. The respective commands in the list are shown or hidden by clicking on the arrow key.

13.4 Login to Swagger API documentation

Carry out the following steps to log in to the IIoT API:

- Click on the **Authorize** button.
- In the **Login** dialog process, the access token for access to the API is fetched and, in the case of HTTPS requests, sent to the API.
- You can now use the API.
- Clicking on **Logout** revokes the access permission again.

> **Hint**
>
> Activate – if not yet set – the checkbox for the **iiotServicesAPI** and **dataStorageAPI** options in the authorization dialog.

13.5 Status bits for variables

When variable values are read, all status bits are returned. In addition, the selected set of status bits is also output as a JSON object.

When variable values are written, only values are written. Time stamp or status are not passed along.

> **Information**
>
> You can find detailed information on this in the **Status processing** section.

13.6 Navigation bar

The navigation bar in the IIoT API **offers the following options**:

- **API Version**: Enables switching between the available API versions.
- **Info**: Displays the installed version of the IIoT API and the license status.

14 Configuration of the IIoT Services in the Engineering Studio

Information on the configuration and parameterization in the Engineering Studio for connections between <CD_PRODUCNTNAME> apps and IIoT Services can be found in the IIoT Services - Configuration in the Engineering Studio section.

15 Appendices

In this chapter, you can find further information on the IIoT Services.

15.1 Advanced configurations

You can also, optionally, adjust the IIoT Services by means of advanced configurations. For advanced configuration, edit the corresponding configuration files with a text editor. You only need advanced configuration in a few, very specific usage scenarios.

> **Tip**
>
> After making changes to configuration files, it is generally a good idea to restart the IIoT Services with all services. You thus ensure that all services work with the current configuration.

15.1.1 Installation options

Where you carry out advanced configurations depends on the IIoT Services installation option:

- IIoT Services (Docker):
  You can find the file called *docker-compose.yml* in the installation directory. Here you configure all services centrally using corresponding environment variables.
- IIoT Services (Windows native):
  You can find various *JSON-Dateien* in the *%CD_SYSTEM%\ServiceGrid\* directory. The JSON file that you have to configure for a particular service is documented in the respective chapters.

You configure the same settings for both installation options in principle. There are however platform-specific differences that are documented in the following chapters.

> ⚠ **Attention**
>
> Only configure the documented JSON files! You should not make any changes to JSON files that are not documented.

15.1.1.1 Environment variables (Docker)

In the IIoT Services (Docker) installation version, you edit the *docker-compose.yml.* file

In Docker, there is the particular feature that you cannot configure the settings in the container directly. You work with environment variables instead.

The IIoT Services check whether environment variables are set each time they are started:

- Environment variables set:
  The environment variables in *docker-compose.yml* overwrite the corresponding default settings in the Docker containers.
- No environment variables set:
  Start the IIoT Services with the default settings from the containers.

With self-configured environment variables in *docker-compose.yml*, you ensure that your settings are effective when the services are started.

15.1.1.2 JSON (Windows native)

In the IIoT Services (Windows native) installation version, you edit the JSON files of the services for advanced configuration. The IIoT Services load the current respective configuration each time services are restarted.

15.1.1.3 Example: Port change for a service

The ports for all services of the IIoT Services are pre-defined by default. In certain cases – if for example a certain port in a network is already assigned – it can be beneficial to define the port yourself.

15.1.1.3.1 Solution: Configure port change (Docker)

The central port number can be configured in the .env file with the **PORT=** entry. If this entry is empty, communication is via port *9443* by default.

If you want to make a change to the port number for the **Proxy Service** (on page 16), save the file with the changed settings and restart all services of the IIoT Services.

15.1.1.3.2 Solution: Configure port change (Windows native)

You configure the port for the **Proxy Service** in the *%CD_SYSTEM%\ServiceGrid\ProxyService.json* file. If the *ProxyService.json* file is not present on your system, you must create this file yourself.

RELEVANT SECTIONS IN THE CONFIGURATION FILE

The following sections in the file are relevant:

1. *HostingInformationConfiguration* section
   Here you define the domain names and the port under which the **Proxy Service** is contactable from outside.
2. *Kestrel* section
   Here you also define the port that the **Proxy Service** will use. The port must be the same as in *HostingInformationConfiguration*

THE CONFIGURATION IN DETAIL

You must change the port in both sections.

Code Sample

Default configuration (port 9443)

```
{
"HostingInformationConfiguration": {
"Uri": "https://[mycomputer.mydomain.com]:9443"
},
"Kestrel": {
"EndPoints": {
"Https": {
"Url": "https://*:9443"
}
}
}
}
```

Change both ports from *9443* to *1234*.

Code Sample

Amended configuration (port 1234)

```
{
"HostingInformationConfiguration": {
"Uri": "https://[mycomputer.mydomain.com]:1234"
},
"Kestrel": {
"EndPoints": {
"Https": {
 "Url": "https://*:1234"
}
}
}
}
```

Save the file with the amended settings and restart all IIoT Services.

Result: You have thus changed the port of the **Proxy Service** from *9443* to *1234*. The **Identity Management** is now also available via this port in the Service Configuration Studio.

15.1.2 Configurations for individual services

This chapter contains service-specific configuration details.

15.1.2.1 Data Hub

There are no service-specific configurations for the **Data Hub**.

15.1.2.2 Certificate Management

Where you configure **Certificate Management** depends on the installation method of the IIoT Services.

IIOT SERVICES (DOCKER)

You configure the environment variables in this file:

- *docker-compose.yml* file (in the installation directory)

Switch to this section in the file:

- *certificate-management:* section

- *environment:* subsection

IIOT SERVICES(WINDOWS NATIVE)

You configure this JSON file:

- *%CD_SYSTEM%\ServiceGrid\CertificateManagement.json*

If the file is missing in the installation directory, you create the *CertificateManagement.json* yourself.

15.1.2.2.1 DataHubConfig

You configure the connection between **Certificate Management** and the **Data Hub** here.

Environment variables (Docker)	JSON (Windows native)	Description	Permitted values

CERTIFICATEMANAGEMENT_DataHubConfig__Url=*NULL*	DataHubConfig"":
	{
	"Url": NULL}
		The default value is the FQDN of the system. You can optionally also configure a specific URL (*https://myhubcontroller.com* for example).	Optional value.
			Permitted value: Valid URL.
			Default:
			NULL
			(i.e. no URL is defined)

CERTIFICATEMANAGEMENT_DataHubConfig__Port=*9411*	"DataHubConfig":
	{
	"Port": 9411
	}
		You configure the port from the Data Hub here.
		The configuration is for the accessibility of the service from external networks.
			Mandatory value.
			Default:
			9411
			Permitted values:
			1 - 65535
			Recommended ports:
			1024 - 49151

CERTIFICATEMANAGEMENT_DataHubConfig__InternalDataHubUrl=*localhost*	DataHubConfig"": { "InternalDataHubUrl": localhost	The default value is *localhost*. You can optionally also configure a specific URL (*https://myhubcontroller.com* for example). **Note:** Bundles for internal IIoT Services, for	Optional value. Permitted value: Valid URL. Default: *NULL* (i.e. no URL is defined)

Environment variables (Docker)	JSON (Windows native)	Description	Permitted values
		example IIoT API, get their information from this entry. If this configuration is configured incorrectly, communication with the Data Hub may fail.	

CERTIFICATEMANAGEMENT_DataHubConfig__InternalDataHubPort=*9411*	"DataHubConfig":
	{
	"InternalDataHubPort": 9411
	}
		Here you can configure, as an option, the port from the Data Hub for internal communication.
		Note: Bundles for internal IIoT Services, for example IIoT API, get their information from this entry. If this configuration is configured incorrectly, communication with the Data Hub may fail.
			Mandatory value.
			Default:
			9411
			Permitted values:
			1 - 65535
			Recommended ports:
			1024 - 49151

15.1.2.3 Identity Service

Where you configure the **Identity Service** depends on the installation method of the IIoT Services.

IIOT SERVICES (DOCKER)

<u>**You configure the environment variables in this file:**</u>

- *docker-compose.yml* file (in the installation directory)

<u>**Switch to this section in the file:**</u>

- *identity-service:* section
- *environment:* subsection

IIOT SERVICES (WINDOWS NATIVE)

<u>**You configure this JSON file:**</u>

- *%CD_SYSTEM%\ServiceGrid\IdentityService.json*

15.1.2.3.1 Radius configuration

Here you can configure the Radius **Identity Provider**.

Environment variables (Docker)	JSON (Windows native)	Description	Permitted values

IDENTITYSERVICE_RadiusConfiguration__AuthenticationTimeoutInMs=3000	"RadiusConfiguration": {
	"AuthenticationTimeoutInMs": 3000
	}
		Timeout for requests to the Radius server.
		After the timeout expires for the **Primary connection**, the **Identity Service** attempts to log in via the configured **Fallback connections**.
			Mandatory value (in milliseconds).
			Default: *3000*

15.1.2.4 Identity Management

There are no service-specific configurations for the **Identity Management**.

15.1.2.5 IIoT API

Where you configure the IIoT API depends on the installation method of the IIoT Services.

IIOT SERVICES (DOCKER)

<u>**You configure environment variables in this file:**</u>

- *docker-compose.yml* file (in the installation directory)

<u>**Switch to this section in the file:**</u>

- *iiot-api:* section
- *environment:* subsection

IIOT SERVICES (WINDOWS NATIVE)

<u>**You configure this JSON file:**</u>

- *%CD_SYSTEM%\ServiceGrid\IIoTApi.json*

15.1.2.5.1 EnableSwaggerDocumentation

You can activate the Swagger documentation here.

The Swagger documentation is used by developers:

- As a complete interface specification
- To test certain IIoT Services functions
- As a reference for self-developed clients

The Swagger documentation is not activated by default.

Environment variables (Docker)	JSON (Windows native)	Description	Permitted values
IIOTAPI_EnableSwaggerDocumentation=true	"EnableSwaggerDocumentation": true	Determines whether the Swagger documentation is available.	Optional value. Permitted values: - *true* (activated) - *false* (deactivated) Default: *true*

15.1.2.5.2 SgIdentityConfiguration

You configure the connection between **IIoT API** and the **Identity Service** here.

Environment variables (Docker)	JSON (Windows native)	Description	Permitted values

IIOTAPI_SgIdentityConfiguration__RequireHttpsMetadata=true	"SgIdentityConfiguration":
	{
	"RequireHttpsMetadata": true
	}
		Defines whether HTTPS is needed for the Discovery Endpoint.	Optional value.
			- *true* (activated)
			- *false* (deactivated)
			Default value: *true*

15.1.2.5.3 SgApiConfiguration

These settings are relevant for the IIoT API.

Environment variables (Docker)	JSON (Windows native)	Description	Permitted values

IIOTAPI_SgApiConfiguration__VariablesCacheLifetime=300	"SgApiConfiguration":
	{
	"VariablesCacheLifetime": 300
	}
		Time period in seconds of how long variable values are stored in the cache.	- Minimum: *0* Values are not limited in time.
			- Maximum: *2147483647*
			Default: *300*

IIOTAPI_SgApiConfiguration__ReportMaxExecutionTime=1800	"SgApiConfiguration":
	{
	"ReportMaxExecutionTime": 1800
	}
		Time period in seconds of the maximum waiting period for the completion of a report in Report Engine.	- Minimum: *0*
			- Maximum: *2147483647*
			Default: *1800*

IIOTAPI_SgApiConfiguration__ReportCacheLifetime=86400	"SgApiConfiguration":
	{
	"ReportCacheLifetime": 86400
	}
		Time period in seconds of how long the result is available for a report in the IIoT API.
		It is discarded once the time has run out.
			- Minimum: *0*
			- Maximum: *2147483647*
			Default: *86400*
			Note: Must be greater than ReportMaxExecutionTime.

IIOTAPI_SgApiConfiguration__SqlElementMaxExecutionTime=1800	"SgApiConfiguration":
	{
	"SqlElementMaxExecutionTime": 1800
	}
		Time period in seconds of the maximum waiting period for the execution of a SQL element.	- Minimum: *0*
			- Maximum: *2147483647*
			Default: *1800*

Environment variables (Docker)	JSON (Windows native)	Description	Permitted values

IIOTAPI_SgApiConfiguration__SqlElementCacheLifetime=86400	"SgApiConfiguration":
	{
	"SqlElementCacheLifetime": 86400
	}
		Time period in seconds of how long the result is available for a SQL call in the IIoT API.
		It is discarded once the time has run out.
			- Minimum: *0*
			- Maximum: *2147483647*
			Default: *86400*
			Note: Must be greater than SqlElementMaxExecutionTime.

15.1.3 Configurations for several services (configured centrally)

Where you configure the settings for several services depends on the installation method of the IIoT Services.

IIOT SERVICES (DOCKER)

You configure these settings as environment variables in this file::

- *.env* file (in the installation directory)

IIOT SERVICES(WINDOWS NATIVE)

You configure this JSON file:

- *%CD_SYSTEM%/ServiceGrid/common.json*

15.1.3.1 Incorporating your own database

By default, the IIoT Services create a database as a Persistence Service and configure this database automatically. In this case, no further configuration is necessary.

You can, optionally, install your own MongoDB database and incorporate it into IIoT Services as a Persistence Service. You must configure this connection manually.

Environment variables (Docker)	JSON (Windows native)	Description	Sample values
Persistence_Uri=mongo	"SGSystemConfigurati	The URL for the	Mandatory

Environment variables (Docker)	JSON (Windows native)	Description	Sample values

db://mycomputer.mydomain:27017	on"
	{
	"DatabaseUri": "mongodb://mycomputer.mydomain.com:27017"
	}
		Persistence Service.
		Note: The Persistence Service is based on a MongoDB.
			value.
			Permitted values: Valid URL.
			Example value: *mongodb://[mycomputer.mydomain.com]:27017*

Persistence_Username=Admin	"SGSystemConfiguration"
	{
	"AdminUser": "Admin"
	}
		User name for the Persistence Service. Required if authentication via user name and password is activated.	Mandatory value.
			Permitted values: Desired strings (in accordance with MongoDB specification)
			Example value:
			mdb_user

Persistence_Password=	"SGSystemConfiguration" {"AdminUserPassword": "mdb_Changeme123!" }	The password for the administrator. Required if authentication via user name and password is activated.	Mandatory value. Permitted values: Must correspond to the password guidelines for IIoT Services. Example value: *mdb_Changeme123!*

15.1.4 Configurations for several services (configured decentrally)

There are different settings in the IIoT Services, which you have to set up in several services at the same time.

STATE GENERIC PREFIX

Because the settings documented in this chapter are not service-specific, we will use, for example, the generic placeholder in the environment variables *<SERVICE_PREFIX>_* for the service-specific prefix (*IIOTAPI_* for example).

For each setting in this chapter, we document the services for which it is relevant. You can then look up the specific information such as the **service prefix** (Docker) and the correct **JSON file** (Windows native) for the respective service.

LOOK UP SPECIFIC PREFIX

You can find the required specific information for the respective service here:

- Data Hub (on page 222)
- IIoT API (on page 225)
- Certificate Management (on page 222)
- Identity Service (on page 224)

With this information, you can easily create the specific configuration values for each service.

15.1.4.1 Kestrel

Kestrel is the standard web server for projects based on the ASP.NET core.

These settings are relevant for the following services:

- IIoT API
- Certificate Management
- Identity Service
- Identity Management

GENERAL CONFIGURATION

You can find the configuration of Kestrel at Microsoft:

- https://docs.microsoft.com/en-us/aspnet/core/fundamentals/servers/kestrel (https://docs.microsoft.com/en-us/aspnet/core/fundamentals/servers/kestrel)

SPECIFIC CONFIGURATION FOR IIOT SERVICES

You can look up the relevant configurations for the IIoT Services here:

- Port changes for services (internal and web interface) (on page 220)

15.1.5 Amendments when changing the host name

If a change is made to the host name of the computer on which IIoT Services are installed, the following adjustments must be made manually. These modifications must be carried out for both Windows operating systems as well as Docker installations.

1. Docker installations only:
   Changing the host name in the .env file.
2. Restart IIoT Services services.
3. Log in to the **Platform Configuration** with a IIoT Services administrator account.
   Note: A certificate warning might be displayed in the web browser because the host name of the certificate no longer matches with the new host name. If this is the case, the certificate warning must be ignored temporarily in order to access the Service Configuration Studio web interface.
4. Recreate the HTTPS certificate in the **Platform Configuration**.
5. Recreate the certificate bundles for IIoT Services services.
6. Restart IIoT Services services.
7. Renew all service connections (certificate bundles) for all connected zenon services.

   Update also the configurations of all connected zenon services, such as Service Engine, Engineering Studio or Report Engine.
8. Modify the configured IIoT Services URLs for the **Identity Service** and the **Data Storage** in the configuration of a Engineering Studio project or in Report Engine.
   You can find this parameterization, for example, in Engineering Studio in the **Network active** project properties group under the **Identity Service**properties.
   Background: When changing the URL, it may be necessary to create a new **Client Secret** in the **Identity Management**. When configuring a new URL, the existing **Client Secret** will be deleted in the input field of the respective option. If the previous **Secret** is no longer known, a new **Client Secret** must be created. This new **Secret** must be modified accordingly in the existing configurations.
   If the previously used **Client Secret** is to be used again for the configuration of a new URL, these changes are unnecessary.
9. When using applications from third-party providers:
   Adjustment of the IIoT Services URLs for Identity Service, Data Storage and IIoT API for all applications that connect to the IIoT Services.

15.2 IIoT API: Query historic variables

To query archive data from Service Engine via the IIoT API:

1. Create an archive in the zenon project.
2. Add variables. These variables must be released for the IIoT Services.
3. Open the Swagger API documentation of the IIoT API and authenticate yourself.
4. To retrieve a list of available archives and their variables, use the following endpoint: */api/v1/datasources/{dataSourceId}/archives*
5. To retrieve historic variable values, use this endpoint: */api/v1/datasources/{dataSourceId}/archives/{archiveId}/query*
6. Specify the desired **Datasource ID** (**Projekt-ID**), archive ID, time filter and variable filter.
7. Execute the query and check the result.

You have thus carried out the query.

15.3 Backup and Restore – Persistence Instance

You will find information in this section about backup, restore and updates of Persistence Instance of a MongoDB.

The following applies for the Persistence Instance:

- The Persistence Service should be backed up before every update of the IIoT Services. This is a precaution.
- A restore of the backup is only necessary in rare cases. This is the case, for instance, if a problem occurs during an update.

The Persistence Service is based on MongoDB. The CLI tools mongodump and mongorestore of the database manufacturer can be used for backup and restore. Both tools are described in the following chapters.

Info

Host operating system and backup folder:

The following applies for all paragraphs:

- You will execute the CLI tools locally on the host operating system where IIoT Services is installed natively or in Docker.
- The backup is stored in the host operating system in the *backups* folder.

The backup commands described create the *backups* folder relative to the folder path in which you are located during the command processing in PowerShell.

15.3.1 CLI tools: mongodump and mongorestore

The CLI tools mongodump and mongorestore allow you to back up and restore the Persistence Service via the command line.

For further information, see the documentation at *www.mongodb.com* (*https://www.mongodb.com/*).

Links to download:

- mongodump — MongoDB Database Tools (https://www.mongodb.com/docs/database-tools/mongodump/)
- mongorestore — MongoDB Database Tools (https://www.mongodb.com/docs/database-tools/mongorestore/)

START CLI TOOLS UNDER WINDOWS

A corresponding environment variable is automatically saved in the host operating system when CLI tools are installed under Windows.

Thus the following applies:

- You can basically start the CLI tools via PowerShell from any folder path. This requires that the environment variable is active.
- In some cases, the operating system must be restarted after installation for the environment variable to be active.

You can start the CLI tools at any time – regardless of the environment variable – via the installation path.

15.3.2 Determine user credentials

The commands contain the following placeholders:

- <*username*>: The user name for MongoDB
- <*password*>: The password for MongoDB

You must replace the placeholders in the commands with the individual user credentials for your system. Where the user credentials can be found depends on your IIoT Services installation option.

See the comparison in the table below:

Placeholder	IIoT Services (Windows native)	IIoT Services (Docker)
<username>	"*AdminUser*"	"*Persistence_Username*"

Placeholder	IIoT Services (Windows native)	IIoT Services (Docker)
<password>	*"AdminUserPassword"*	*"Persistence_Password"*
	Path to the user credentials: *C:\ProgramData\COPA-DATA\System\ServiceGrid\common.json*	**Path to the user credentials:** *.env* file in the installation directory of the IIoT Services.

15.3.3 Perform backup

These instructions basically work for all installation options of IIoT Services.

Please note the following:

- Information is provided at each configuration step on which installation of IIoT Services it refers to.
- You can use the configuration steps for IIoT Services (Docker on Windows) for IIoT Services (Docker on Linux) too. However, you have to change the folder paths for linux and use a Linux Shell.

The tools and backup commands used are basically the same under Linux and Windows.

PREPARATION (DOCKER)

In Docker you must open the database containers beforehand for access from the host operating system.

To do this, carry out the following steps:

1. Open an elevated PowerShell.
2. In PowerShell, go to the installation directory of the IIoT Services.
   cd C:\iiot-services
   Note: You have created this folder path yourself for the installation of the IIoT Services. It contains all IIoT Services configuration files such as *docker-compose.yml*.
3. Stop all containers:
   docker-compose down
4. Start all containers with an additional configuration file:
   docker-compose -f docker-compose.yml -f docker-compose.expose-db.yml up

Now you can access the database in the Docker containers via the Windows host system.

PERFORM BACKUP (WINDOWS NATIVE, DOCKER)

Follow these steps to use the command *mongodump* to back up the data of the Persistence Service:

1. Open an elevated PowerShell.
2. In PowerShell go to the directory path where the backup folder should be created.
3. Use the following command to create a backup folder in the selected directory path and back up the database there:
   mongodump --username='<username>' --password='<password>' --host='fqdn-clientname' --port=27017
   --archive='backups\IIoTServices.archive'
   Note: You must replace the <*username*> and <*password*> placeholders with the appropriate user credentials.

You have now backed up the data from your Persistence Service.

POSTPROCESSING (DOCKER)

You must restart all containers in Docker:

1. Stop all containers:
   docker-compose down
2. Restart the containers:
   docker-compose up

The database is thus protected again from access via the host operating system.

15.3.4Apply restore

These instructions basically work for all installation options for IIoT Services.

Please note the following:

- Each configuration step specifies for which installation version of IIoT Services this is valid.
- You can use the configuration steps for IIoT Services (Docker on Windows) for IIoT Services (Docker on Linux) too. However, you have to change the folder paths for linux and use a Linux Shell.

The tools and backup commands used are the same on Linux and Windows.

PREPARATION (DOCKER)

In Docker you must open the database containers beforehand for access from the host operating system. To do this, carry out the following steps:

1. Open an elevated PowerShell.

2. In PowerShell, go to the installation directory of the IIoT Services.
   cd C:\iiot-services
   Note: You have created this folder path yourself for the installation of the IIoT Services. It contains all IIoT Services configuration files such as *docker-compose.yml*.
3. Stop all containers:
   docker-compose down
4. Start all containers with an additional configuration file:
   docker-compose -f docker-compose.yml -f docker-compose.expose-db.yml up

Now you can access the database in the Docker containers via the Windows host system.

APPLY RESTORE (WINDOWS NATIVE, DOCKER)

Perform the following steps to restore the Persistence Service data using the mongorestore command:

1. Open a PowerShell.
2. Go to the directory path of the backup folder.
3. Perform the restore of the database:
   mongorestore --username='<username>' --password='<password>' --host='fqdn-clientname' --port=27017 --archive='backups\IIoTServices.archive' --drop

You have now restored the Persistence Service from the backup.

⚠ Attention

With the --*drop* argument, all existing data in Persistence Service are deleted by the restore and replaced with data from the backup.

👍 Tip

With the --*dryRun* argument, it is possible to simulate the restore of the data. Thereby, existing data of the Persistence Service are not overwritten.

POSTPROCESSING (DOCKER)

You must restart all containers in Docker:

1. Stop all containers:
   docker-compose down
2. Restart the containers:
   docker-compose up

The database is thus protected again from access via the host operating system.

15.3.5COPA-DATA command line tool

As an alternative to the database manufacturer's CLI tools, the **CopaData.ServiceGrid.Tools.PersistenceManagementCli.exe** is also available. This app is best suited for IIoT Services in a Docker environment. After a call without parameters, the app provides integrated step-by-step instructions directly in the command line call. For native Windows environments, the update process is also implemented in the setup.

UPDATE STEPS

The tool performs the following steps:

- Stops all IIoT Services, except for Persistence
- Export of MongoDB database.
- Stop of the running (old) Persistence.
- Update of MongoDB
- Start of the latest (new) Persistence for the current version.
- Import of the MongoDB database saved from the previous version.
- Start all IIoT Services and the current version of the Persistence Service.

These steps are visualized directly in the commandline app when they run in a Docker environment. If an interaction by the user is necessary, this is indicated accordingly by the tool. After entering the necessary parameters, the tool continues to run.

15.3.5.1 Docker environment update

The following requirements are necessary for updating MongoDB in a Docker environment:

- The tool **CopaData.ServiceGrid.Tools.PersistenceManagementCli.exe** is installed on machines running Docker Desktop for Windows.
- The installation is done by running *PersistenceManagementCli.x64.msi*.
  The data are stored in the following folder:
  %programfiles%\zenon\zenon Platform 12\IIoT Services\PersistenceManagementCli.
- The **MongoDB Command Line Database Tools** (on page 233) are installed.
- The PATH environment variable has been extended with the path to the MongoDB Command Line database tools (see previous step), e.g.:
  C:\Tools\mongodb-database-tools-windows-x86_64-100.7.0\bin
- The current version of the IIoT Services is installed and running.

- The .ENV file with the current settings and the docker-compose .YML file for the new version are available in their own Windows folder.
- Port *27017* is available on the computer for connecting to the MongoDB database.

RUN UPDATE

In the Docker environment, do the following:

1. Open an elevated PowerShell.
2. Navigate to the storage location of the CLI, e.g. (default path): *%programfiles%\zenon\zenon Platform 12\IIoT Services\PersistenceManagementCli*.
3. Enter the following command:
   CopaData.ServiceGrid.Tools.PersistenceManagementCli.exe docker upgrade
4. The tool starts and guides you through the update process step by step. Necessary parameters are queried. The update process is continued after the necessary parameters are entered. In addition, information and a log are visualized directly in the tool.

15.4 Docker commands

This section provides you with an overview of Docker commands.

Action	PowerShell command
List all running containers	docker ps
List all containers, including the ones already stopped	docker ps -a
Stop a particular container	docker stop <container-name or container ID>
Stop all running Docker containers*	docker stop $(docker ps -aq)
Start a stopped container	docker start <container-name or container ID>
Remove a particular container	docker rm <container-name or container ID>
Remove all containers*	docker rm $(docker ps -aq)
List all volumes	docker volume ls
Remove a particular volume	docker volume rm <volume-name>
Remove all used volumes*	docker volume rm $(docker volume ls -q)
List all images	docker image ls
Remove a particular image	docker image rm <image-name or image ID>

Action	PowerShell command
Remove all Docker images (containers must be stopped)*	docker image rm $(docker image ls -q)
With this command, you can reset the entire system.* **It removes:** - all stopped containers - all networks that are not used by at least one container - all **Dangling Images** - the build cache	docker system prune --all --volumes

* Commands only for test environments: **Remove** applies to all containers, volumes and images on the computer.

15.4.1 Completely delete installation

Sometimes it's a good idea on a test system to completely remove an existing IIoT Services installation and start from scratch.

To delete a IIoT Services installation:

1. Stop all running Docker containers with the following command:
   ```
   docker stop $(docker ps -aq)
   ```
2. Remove all running Docker containers with the following command:
   ```
   docker rm $(docker ps -aq)
   ```
3. Remove all volumes with the following command:
   ```
   docker volume rm $(docker volume ls -q)
   ```
4. Remove all Docker images with the following command:
   ```
   docker image rm $(docker image ls -q)
   ```

To check whether the IIoT Services installation has been deleted:

1. List all containers (including the stopped ones):
   ```
   docker ps -a
   ```
2. List all volumes:
   ```
   docker volume ls
   ```
3. List all container images:
   ```
   docker image ls
   ```

In all cases, an empty list should be displayed as the output.

> **Attention**
>
> These commands remove all Docker containers and images on the computer. Therefore, they are notsuitable for productive systems.

15.5 Log messages

Logs can be used to analyze the behavior of services of the IIoT Services and localize any problems.

The IIoT Services log the following events by default:

- Standard results (such as the successful start of a service).
- Error messages (such as a failure to establish a connection).
- Additional LOG modules for IIoT Services and Linux.
- LOG messages are sent to the **Diagnosis Server**.

IIOT SERVICES (WINDOWS NATIVE)

You can retrieve LOGs for the zenon services in IIoT Services via the GUI using the **Diagnosis Viewer**. LOGs for external services of IIoT Services, such as the Persistence Service, cannot be retrieved using the **Diagnosis Viewer**.

You can find further information on the **Diagnosis Viewer** in the Help in the **Diagnosis Viewer** section.

IIOT SERVICES (DOCKER)

The following possibilities are available for the LOGs under Docker:

1. **Diagnosis Viewer**:
   LOG messages can be analyzed using the zenon Diagnosis Server.
2. Analysis via Docker logs

   Under Docker, you can retrieve LOGs for all services of the IIoT Services.

   Carry out the following command to call up LOG messages for a service:

   docker logs <containername>

   Note: Replace the placeholder "<containername>" with the name of the service.

IIOT SERVICES (DOCKER ON WINDOWS)

You can also query the logs using the Docker Dashboard. Clicking on the container opens a properties window with the logs.

ACTIVATION OF ZENON LOGGING FOR DOCKER

zenon logging for Docker installations is not contactable from outside by default. Carry out the following steps to enable connections to the **zenon Logging Server**.

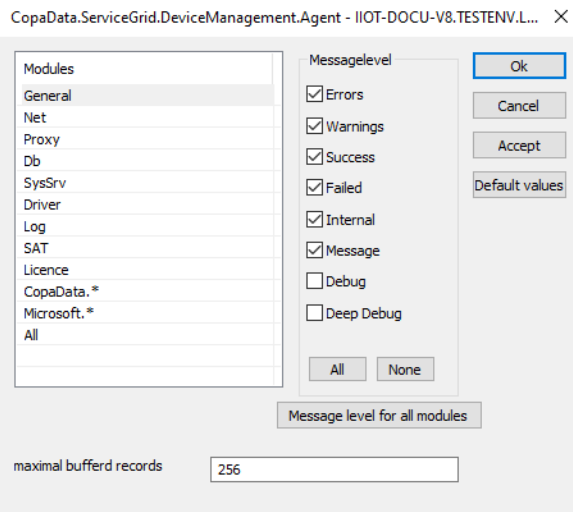
1. Stop the logging service. To do this, carry out the following Docker command in the Docker directory of IIoT Services (*c:\iiot-services*):
   docker compose stop zenon-logging-server
2. Restart the logging service with the port for the zenon Diagnosis Server enabled. To do this, carry out the following command:

   docker compose -f docker-compose.yml -f docker-compose.expose-logging.yml up zenon-logging-server -d
   Note: the port enabling is defined in the *docker-compose.expose-logging.yml* file.
3. Then open the connection to the LOG server in Docker in the Diagnosis Viewer.
   Note: In order to be able to establish the connection to the logging server in the Docker environment, the local logging service (**zenLogSrv**) must first be stopped on the computer. This is necessary because both LOG servers use the same port number.
4. Once you have finished your analysis, close the logging service in the Docker environment. To do this, execute the following command:
   docker compose stop zenon-logging-server
5. Restart the Diagnosis Server in the Docker environment without the port enabled. To do this, execute the following command:
   docker compose -f docker-compose.yml up zenon-logging-server -d
   Note: If necessary, restart the logging service (**zenLogSrv**) on the local computer that is running the **Diagnosis Viewer**.

> ⚠**Attention**
>
> For security reasons, it is recommended that port enabling for the zenon logging server in Docker is only activated temporarily for the duration of the analysis.

15.5.1 Additional modules in the Diagnosis Viewer

Services of the IIoT Services have additional entries for module configuration in the **Diagnosis Viewer**.


CopaData.ServiceGrid.DeviceManagement.Agent - IIOT-DOCU-V8.TESTENV.L...

Modules
- General
- Net
- Proxy
- Db
- SysSrv
- Driver
- Log
- SAT
- Licence
- CopaData.*
- Microsoft.*
- All

Messagelevel
- ☑ Errors
- ☑ Warnings
- ☑ Success
- ☑ Failed
- ☑ Internal
- ☑ Message
- □ Debug
- □ Deep Debug

maximal bufferd records: 256

ADDITIONAL LOG MODULES

In general, all zenon components have the same logging modules. However, IIoT Services uses a different technology from that of the zenon **Diagnosis Server**. For IIoT Services, the modules are configured on the basis of namespaces. As a result, the IIoT Services have additional modules that enable more detailed configuration of the Messagelevels.

The logging for IIoT Services uses the following, additional modules.

Module entry	Description
CopaData*	All zenon internal IIoT Services logging modules write their entries to this module.
Microsoft*	If, for example, Microsoft frameworks are used in one of the IIoT Services, they write your entries to this module.

MESSAGE LEVEL - MAPPING

All protocol levels that are "higher" than the minimum level are activated automatically in the process.

Example: If, in the **Diagnosis Viewer**, only *Messages* have been activated, the levels *Warning* and *Failed* are also active.

zenon message LOG level	IIoT Services LOG level
Errors	*Error*

zenon message LOG level	IIoT Services LOG level
Warning	*Warning*
Success	*Not mapped*
Failed	*Not mapped*
Internal	*Not mapped*
Message	*Information*
Debug	*Debug*
Deep Debug	*Trace*

15.6 Troubleshooting.

In this chapter, you can find information on troubleshooting in the IIoT Services.

15.6.1 Checklist: Check basic functions

If you work through this list, you can localize and rectify the most common configuration errors for the IIoT Services.

USER NAMES: MAXIMUM LENGTH FOR SERVICE ENGINE

You can also use user names from the **Identity Service** for logging in to Service Engine. However, this also means that the effective length of the user name is no more than 20 characters. This is a limitation of Service Engine.

How you calculate the effective length of a user name is documented separately.

HOSTNAME: CONTINUOUS LOWERCASE LETTERS

Upper-case letters in host names lead to authentication problems in the IIoT Services. Hostnames must **always** be written with continuous lowercase letters. This is also required if the hostname actually happens to contain uppercase letters.

Please refer to the following table with examples:

Actual hostname	Correct way of writing for the IIoT Services
MyComputer.mydomain.com	*mycomputer.mydomain.com*

Actual hostname	Correct way of writing for the IIoT Services
MYCOMPUTER.mydomain.com	*mycomputer.mydomain.com*
mycomputer.mydomain.com	*mycomputer.mydomain.com*

If you accidentally entered the host name in upper-case letters when setting up IIoT Services (Docker), you must do the following:

- Remove the incorrect installation completely (on page 239)
- Reinstall the IIoT Services (Docker) with the correct host name.

A host name written entirely in lower-case letters is required for proper functioning of the IIoT Services.

CHECK MINIMUM PASSWORD REQUIREMENTS

Unsuitable passwords lead to authentication problems with the IIoT Services. Only use suitable passwords that meet the minimum password requirements for IIoT Services.

RESTART SERVICES

If changes to the configuration are made in the IIoT Services, it may be necessary to restart individual services. By restarting all services of the IIoT Services (on page 30), you ensure in all cases that all services can access the current configuration.

After restarting, check all services to see if all relevant services are in the status *Running*. This is a basic requirement for operation of the IIoT Services.

CONNECTION STATUS IN CERTIFICATE MANAGEMENT

In the web interface of **Certificate Management**, check the connection status to the individual services.

These services must, as a minimum, be in connected state:

- **Certificate Management**
- **IIoT API**

Note: Due to reconfigurations or expired certificates, it is possible that there are several certificate entries for one service type. In this case, all you need is one certificate entry per service type in the status*connected*.

Depending on the application scenario, the following connections may also be required:

- **Engineering Studio**
- **Service Engine**
- **Report Engine**

CHECK LOG MESSAGES

The LOG messages (on page 240) provide extensive information that can be useful for troubleshooting for the IIoT Services.

CHECK CONNECTION TO SERVICE ENGINE

The following requirements must be met:

- The Service Engine must be configured for IIoT Services. Use the **IIoT Services Connection Wizard** for this.
- Service Engine must be started.
- The Service Engine license must include the **Data Hub Gateway**. The license must be activated.
- Service Engine must be connected to **Certificate Management** (see connection overview (on page 65)).

You can check the status of this connection in **Certificate Management**. If this connection is not provided, you can view the logs for Service Engine using the **DiagViewer**. Here you can get information on establishing connections.

15.6.2 Docker

DATA LOSS DURING UPDATE

In general, an update of Docker does not affect an existing IIoT Services installation. In a few cases, however, data loss can occur during the update.

Hint: Before updating the Docker version, back up the Persistence Instance of your IIoT Services installation (on page 232). That way, even if your Docker update fails, you can still restore the Persistence Instance.

DATA LOSS DURING SUBSEQUENT CHANGE TO WSL2 (DOCKER ON WINDOWS)

Docker Desktop for Windows offers the choice between *Hyper-V* and *WSL2* to execute Docker containers.

The subsequent change from *Hyper-V* to *WSL2* leads to a complete loss of existing IIoT Services installations. You must then reinitialize and set up the IIoT Services.

Hint

When installing **Docker Desktop for Windows**, select the *WSL2* option.

- **Settings\General\Use the WSL2 based engine**

Only install the IIoT Services afterwards.

15.6.3 HSTS problems during new installation

When the digital certificates of the services change due to a complete new installation of the IIoT Services, this can lead to problems. The web browser will not direct you to the website, nor will there be a certificate warning.

This behavior is caused by a browser function called **HSTS (HTTP Strict Transport Security)**.

Carry out the following steps to rectify the problem:

1. Open the **Certificate Management** in Service Configuration Studio: *https://[mycomputer.mydomain.com]:9443*
2. Download the **CA Certificate**.
3. Install the **CA Certificate**.
4. Restart the browser.

After restarting the browser, you should be able to access IIoT Services web pages via FQDN again (for example: *https://[mycomputer.mydomain.com]:9443*).

15.6.4 Identity Service

The **Identity Service** is the central authentication service of IIoT Services. The problems described below relate to the **Identity Service**.

ERROR MESSAGE: "UNAUTHORIZED CLIENT”

Clients can authenticate themselves using the **Identity Service**. With correct configuration, the clients can then use the IIoT API. The error message "*unauthorized client*” indicates that the client configuration in the **Identity Service** does not match the configuration in the client application.

Solution: Compare the client configuration in the **Identity Service** with the configuration in the client application.

Check in particular:

- **Client-ID**
- **Redirect-URL**
- **Allowed Scopes**
- **Grant types**
- **Secret**

Important: The **Identity Service** also checks the case of the configured URLs. This must match with the URL entered.

FORGOT PASSWORD. NO LONGER POSSIBLE TO LOG IN

In the case of a forgotten password, a difference has to be made between different user roles:

1. A user has forgotten their password and can no longer log in to the **Identity Service**.
   Solution: Contact the user with the **Identity Administrator** user role. This user can reset the password for each user separately in the **Identity Management**.
2. The user with the *Identity Administrator* user role has forgotten their password. They cannot log in to the **Identity Service** or the **Identity Management**.
   Solution: Contact zenon Support. Support can help you reset the user database.

> ⚠ **Attention**
>
> **Reset the user database**
>
> If you reset the user database, configurations for users, groups, clients and **Identity Provider** will be lost, for instance. After resetting the user database, you must reconfigure the IIoT Services to a large extent.

15.6.5 IIoT API: Error codes

In the event of a failed attempt to access the IIoT API, one of the following error codes will be output.

HTTP ERROR CODE 400

- Cause: Incorrect request of client application.
- Solution: Check whether the request is formulated correctly and meets the expected data model.

HTTP ERROR CODE 402

- Cause: The IIoT API does not have a valid license.

- Solution: Install a valid license. After installation, the license must be activated. It might be necessary to restart the IIoT Services.

HTTP ERROR CODE 403

- Cause: The **User** or **Client** does not have the necessary access permissions.
- Solution: Configure the access permissions for the **User** or **Client** in the **Identity Management**.

HTTP ERROR CODE 500

- Cause: The IIoT API had an internal error while processing a request. This error occurs repeatedly.
- Solution: Restart the IIoT API service. If the error still continues to occur, contact zenon Support.

ERROR CONNECTING [...] NO SUCH DEVICE OR ADDRESS

When using the Swagger API documentation, it can happen that the authorization of the user or the client to the IIoT API does not work because the address of the **Identity Service** cannot be resolved.

The error message is displayed in the authorization dialog as follows:

Error connecting to https://[mycomputer.myddomain.com]:9443/identity-service/.well-known/openid-configuration: No such device or address

This error can even occur if the naming resolution generally works and, for instance, if the Service Configuration Studio is displayed.

There are several reasons why the name resolution can fail in this special context.

- The host name has been entered incorrectly in the **MACHINE_HOSTNAME** variable in the .env file. For example, if you used capital letters in this variable, this can lead to authorization problems when using the IIoT API.
  Solution: Change the hostname in the *.env* file so that only lower case letters are used for the entire hostname. Then restart the IIoT Services and reload the website of the IIoT API. Authorize again.
- The firewall blocks the correct resolution of computer names. To check this, copy the URL specified in the error message and try to open the URL in the browser.
  Solution: Check the configuration of your firewall.

You can find further information on error codes and their reasons in the Swagger API documentation.

15.6.6 Windows hibernation: Different time stamps (Docker on Windows)

Using the **Hibernate** energy-saving option on your computer can lead to problems. The authentication to the Identity Service can fail if the computer is awoken from hibernation mode. This problem occurs when the internal time of Docker after hibernation mode is no longer synchronized automatically with the time of the computer. This results in different time stamps.

Solution: A complete restart of Docker fixes the problem. It is not enough to just restart the services.