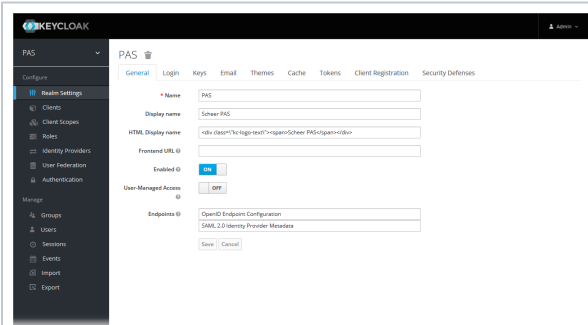


Keycloak

Keycloak: Managing Authentication



Scheer PAS uses [Keycloak](#) to save the platform users and their passwords. For [API Management](#) and [Kibana](#) we also manage roles in Keycloak.

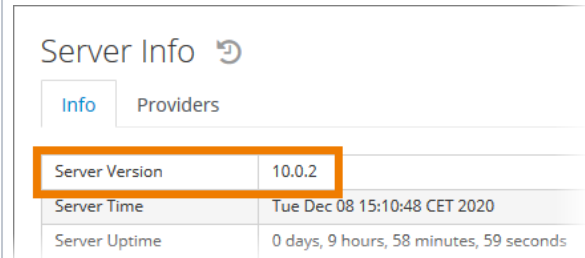
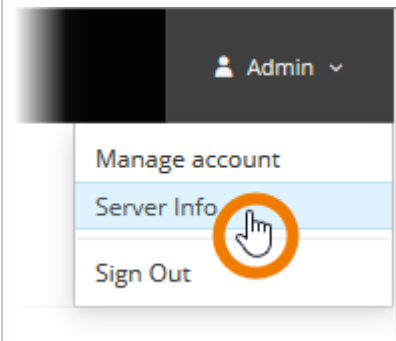
Keycloak is an identity and access management software that can manage users for multiple applications and enables them for single sign on. Data for each application is stored in so called **realms**.

On this Page:

- [Keycloak: Managing Authentication](#)
 - [Keycloak Roles for Usage with Scheer PAS](#)
 - [Realm Roles](#)
 - [Client Roles](#)
 - [How to Create Additional Keycloak Admins](#)
 - [Keycloak and API Management](#)
 - [Keycloak and Kibana](#)

Related Pages:

- [3rd Party Tools](#)
 - [Kibana](#)



✓ In the [Official Keycloak Documentation](#) you will find detailed descriptions of all Keycloak functionalities. Please note that the official documentation may vary according to different versions of the tool.

Related Documentation:

- [API Management](#)
- [Official Keycloak Documentation](#)

Before consulting the documentation check your Keycloak version:

- [Open the user menu and choose option **Server Info**](#).

- On page **S**erver Info, check the entry in field **S**erver Version.

Keycloak Roles for Usage with Scheer PAS



In Keycloak, you can assign two different types of roles:

- [Realm Roles](#)
- [Client Roles](#)

Realm Roles

Realm roles are a global namespace to define your roles. The following realm roles are used with **Scheer PAS**:

Role Name	Description
apiadmin	Grants administrator privileges for API Management.
apipublisher	Required to publish to the API gateway of API Management (applies to a technical user only).
apiuser	Grants user privileges for API Management.
devportaluser	Required to access the developer portal of API Management.
elasticadmin	Grants access to Elasticsearch and Kibana.

offline_access	Default Keycloak realm role.  See chapter Roles in the official Keycloak documentation.
uma_authorization	Default Keycloak realm role.  See chapter Roles in the official Keycloak documentation.

Client Roles

Client roles are namespaces dedicated to a client. To add a client role, you must first select the client from the drop-down list.

In case of **Scheer PAS**, the client **realm-management** is needed. This client defines client-level roles that specify permissions that can be granted to manage the realm.

Client	Role Name	Description
realm-management	view-users	Users with this role will only be able to use that specific part of the administration console.


How to Create Additional Keycloak Admins

You can create additional admins for **Keycloak** as follows:

1. Go to realm **Master**.
2. Select **Users**.
3. Create a new user or edit an existing user.
4. Switch to tab **Role Mappings**.
5. Assign role **admin** in section **Realm Roles**.

Keycloak and API Management

Scheer PAS API Management uses the Keycloak application to manage application access. Additionally, you can manage user roles in API Management to grant permissions within API Management.

 Visit the [API Management Guide](#) for further informations. You will find a detailed description how Keycloak is used to manage API Management users on page [Managing Users and Permissions](#).

Keycloak and Kibana

[Kibana](#) is delivered with every installation of **Scheer PAS 20.1**, including a Kibana admin user. Use this admin user to access Kibana.

Preview: Starting with PAS 21.1 you can assign the role **elasticadmin** to each PAS user individually. This role grants access to Kibana.