

Managing Permissions

Permissions are used to control the access to URLs, for example the URL of the portal, of the BPaaS cockpit or the Designer, or the URLs of specific apps. Permissions are assigned to [profiles](#). Depending on his allocated profiles, a user is authorized to open the portal, to work with process apps etc.

 The authorization check of the user management is based on [Role-based Access Control \(RBAC\)](#).

On this Page:

- [Menu "Permissions"](#)
- [The Permission Data Sheet](#)
 - [Master Data](#)
 - [Permission](#)
 - [Meta](#)
- [Standard Permissions](#)

Menu "Permissions"

 Users

 Profiles

 Roles

 Permissions

 Add Permission

 Activities

Menu **Permissions** allows you to create new permissions and to edit existing ones.

It contains the following menu options:

- **Permissions**
- **Add Permission**

Related Pages:

- [Managing Profiles](#)
 - [Adding a Profile](#)
- [Working With the User Management](#)

#	Name	Resource	Activated
1	service.tpaas.backend	/{path}/service/tpaas-backend/*	Yes
2	service.tpaas.ccsdpt	/{path}/service/tpaas-ccsdpt/*	Yes
3	SEP-Hostfile_001	/{path}/app/sep-hostfile-001/*	Yes
4	service.tpaas.robot.connector	/{path}/service/tpaas-robot-connector/*	Yes
5	service.tpaas.robot.connector.get_robot_logs	/{path}/service/tpaas-robot-connector/getRobotLogs	Yes
6	service.tpaas.robot.connector.get_robot_history	/{path}/service/tpaas-robot-connector/getRobotHistory	Yes
7	service.tpaas.robot.connector.get_robotids	/{path}/service/tpaas-robot-connector/getRobotids	Yes
8	service.tpaas.robot.connector.get_environments	/{path}/service/tpaas-robot-connector/getEnvironments	Yes
9	service.tpaas.robot.connector.set_error_status	/{path}/service/tpaas-robot-connector/setErrorStatus	Yes
10	service.tpaas.robot.connector.set_status	/{path}/service/tpaas-robot-connector/setStatus	Yes

In the **Permissions** menu, a list of all permissions is displayed in the content area.

Use the filter to limit the content of the list.

This list contains the following information:

- **Name**
- **Resource**
- **Activated** (Yes/No)

Master Data
☒ Permission active
Name

Permission
Possible Resource Information

```

1 {
2   "path": "/permission/path"
3 }

```

SAVE CANCEL

Select **Add Permission** to create a new permission.

An empty permission data sheet will open.

The Permission Data Sheet

The data of each permission is saved in a separate permission data sheet with the following sections:

1. [Master Data](#)
2. [Permission](#)
3. [Meta](#)

Master Data

Master Data ☒ Permission active Name	You can manage the basic data of a permission in the Master Data section. Name is mandatory.
☒ Permission active	Permission active The checkbox indicates whether the permission is activated or not. If you create a new permission, default is active.
Name	Name It is mandatory to insert the identifier for a new permission in the Name field.

Permission

Permission Possible Resource Information 1 { 2 "path": "/permission/path/*" 3 }	A permission grants access to a defined sector of the platform. In the Permission section, you will have to insert the resource of that sector.  Do not change or create permissions without any knowledge of JSON.
---	--

Permission

Path

Path with Variable

Information

ssion/path/*"

You can choose additional resources by using the drop down list **Possible Resource Information**.

Available options are:

- **Path**
- **Path with Variable**

Meta

Meta

UUID

Created at

Modified at

Modified by

The **Meta** section contains a list of the permission's meta data such as:

- **UUID:** The UUID allows to identify the permission.
- **Created at:** Contains the creation date of the permission.
- **Modified at:** Contains the date of the permissions' last modification.
- **Modified by** Contains another UUID which allows to identify the last editor of the permission.

i The content of section **Meta** is for documentation purposes only: meta data cannot be changed.

Standard Permissions

Each **Scheer PAS** platform is delivered with the standard permissions required for the use of the **Scheer PAS** components. The following table summarizes the standard permissions.

The table can be searched. To do this, enter your search phrase in the search field:

Permission Name	Description
activity-service	Access to the activities
app_admin	Component user management (UI): Access using URL admin
app_file	File storage without profile check
app_storage	File storage with profile check
app_user_admin	Component user management (UI): Access using URL user-admin For backward compatibility only - default is <i>app_admin</i>
app-userself	User self service (UI)
bpaas_editor	Component BPaaS: Access to modeling (UI)
bpaas_execution	Execution engine for BPaaS apps

bpaas-cockpit	Component BPaaS cockpit (UI): Access using URL bpaas-cockpit For backward compatibility only - default is <i>cockpit</i>
bpaas-forms-connector-service	Component BPaaS: Integration of Scheer PAS Mobile (Backend)
cockpit	Component BPaaS cockpit (UI): Access using URL cockpit
customer_service	Access to integration (xUML) services and UIs
designer-service	Component BPaaS: Editor for mobile forms (Backend)
form-designer-app	Component BPaaS: Editor for mobile forms (UI)
ldap-auth-service	Access to LDAP service
management_console	Access to component Administration (UI)
model_editor	Access to component Business Modeler (UI)
model_editor_service	Access to component Business Modeler (Backend)
pas_designer	Access to component Designer (UI)
pas_designer_service	Access to component Designer (Backend)
portal_administration	Access to the tile "Administration" in the Portal
portal_api_management	Access to the tile "API Management" in the Portal
portal_base_permission	Access to the Portal
portal_cockpit	Access to the tile "BPaaS" in the Portal
portal_cron_manager	Access to the tile "Cron Manager" in the Portal
portal_designer	Access to the tile "Designer" in the Portal
portal_dev_portal	Access to the tile "API Developer Portal" in the Portal
portal_file_manager	Access to the tile "File Manager" in the Portal and to the UI of component File Manager
portal_home_permission	Access to the homepage of the Portal
portal_integration	Access to the tile "Integration" in the Portal
portal_keycloak	Access to the tile "Keycloak" in the Portal
portal_kibana	Access to the tile "Kibana" in the Portal
portal_model_editor	Access to the tile "Business Modeler" in the Portal
portal_process_analytics	Access to the tile "Process Mining" in the Portal
portal_process_monitor	Access to the tile "Process Monitor" in the Portal
portal_user_management	Access to the tile "User Management" in the Portal
portal_xuml_monitor	Access to the tile "xUML Service Monitor" in the Portal
process_monitor	Access to component Process Monitor (UI)
sap-restifier	Access to SAP Restifier
service_analytics_api	Component Process Mining (Backend): Reading the analytics database
service_analytics_collector	Component Process Mining (Backend): Collecting data from services
service_analytics_etl	Component Process Mining (Backend): Importing data to the analytics database
service_bpaas_backend	Component BPaaS: Modeling (Backend)
service_bpaas_cockpit	Component BPaaS: Cockpit (Backend)

service_bpaas_robot_connector	Deprecated as of PAS 21.2 Component BPaaS: Integration of RPA (Backend)
service_bpaas_robot_connector_continue_instance	Deprecated as of PAS 21.2 Component BPaaS/RPA (Backend): Continuing an instance
service_bpaas_robot_connector_get_current_user_job_queue_items	Deprecated as of PAS 21.2 Component BPaaS/RPA (Backend): Showing robot processes by user
service_bpaas_robot_connector_get_environments	Deprecated as of PAS 21.2 Component BPaaS/RPA (Backend): Reading the robot environments
service_bpaas_robot_connector_get_instance	Deprecated as of PAS 21.2 Component BPaaS/RPA (Backend): Loading an instance
service_bpaas_robot_connector_get_instance_from_queue	Deprecated as of PAS 21.2 Component BPaaS/RPA (Backend): Listing all queued instances
service_bpaas_robot_connector_get_jobs_stats	Deprecated as of PAS 21.2 Component BPaaS/RPA (Backend): Getting list of robot jobs
service_bpaas_robot_connector_get_process_information	Deprecated as of PAS 21.2 Component BPaaS/RPA (Backend): Listing robot process information
service_bpaas_robot_connector_get_process_parameter	Deprecated as of PAS 21.2 Component BPaaS/RPA (Backend): Analyzing the UIPath process
service_bpaas_robot_connector_get_processes_by_environment	Deprecated as of PAS 21.2 Component BPaaS/RPA (Backend): Loading robot processes by environment
service_bpaas_robot_connector_get_robot_history	Deprecated as of PAS 21.2 Component BPaaS/RPA (Backend): Loading robot history
service_bpaas_robot_connector_get_robot_logs	Deprecated as of PAS 21.2 Component BPaaS/RPA (Backend): Loading robot logs
service_bpaas_robot_connector_get_robot_processes	Deprecated as of PAS 21.2 Component BPaaS/RPA (Backend): Loading robot processes
service_bpaas_robot_connector_get_robots	Deprecated as of PAS 21.2 Component BPaaS/RPA (Backend): Loading robots
service_bpaas_robot_connector_get_sessions	Deprecated as of PAS 21.2 Component BPaaS/RPA (Backend): Listing robot session information from the orchestrator
service_bpaas_robot_connector_get_sessions_stats	Deprecated as of PAS 21.2 Component BPaaS/RPA (Backend): Getting the total number of robots aggregated by robot state
service_bpaas_robot_connector_has_access_on_profile	Deprecated as of PAS 21.2 Component BPaaS/RPA (Backend): Checking robot authorization for profiles
service_bpaas_robot_connector_save_instance	Deprecated as of PAS 21.2 Component BPaaS/RPA (Backend): Saving an instance
service_bpaas_robot_connector_set_error_status	Deprecated as of PAS 21.2 Component BPaaS/RPA (Backend): Setting the error status of a robot
service_bpaas_robot_connector_set_status	Deprecated as of PAS 21.2 Component BPaaS/RPA (Backend): Setting the status of a robot
service_bpaas_robot_connector_start_instance	Deprecated as of PAS 21.2 Component BPaaS/RPA (Backend): Starting an instance
service_bpaas_robot_connector_start_robot	Deprecated as of PAS 21.2 Component BPaaS/RPA (Backend): Starting a robot
service_bpaas_robot_connector_start_robot_without_instance	Deprecated as of PAS 21.2 Component BPaaS/RPA (Backend): Starting a robot without requiring a PAS instance
service_cron	Component Cron Manager: Time-controlled event execution (Backend)
service_cron-service_availableServices	Component Cron Manager: Allows access to availableServices event of the service repository
service_cron-service_controlJob	Component Cron Manager: Allows access to controlJob event of cron-service

service_cron-service_getJobs	Component Cron Manager: Allows access to getJobs event of cron-service
service_cron-service_registerJob	Component Cron Manager: Allows access to registerJob event of cron-service
service_cron-service_unregisterJob	Component Cron Manager: Allows access to unregisterJob event of cron-service
service_mail	Email dispatch (Backend)
service_messaging-service	Access to messaging service
service_persistence	Database access (Backend)
service_service-repository__docker_build_image	Access for building a docker image (like xuml images)
service_service-repository__docker_deploy_container	Access for deploying arbitrary containers previously build and stored in local docker registry
service_service-repository__docker_get_images	Access for fetching all stored images in your private docker registry
service_service-repository__xuml_deploy_service	Access for deploying an xUML service running as container
service_service-repository__xuml_endpoints	Access for fetching registered endpoints from an xUML service running as container
service_service-repository__xuml_get_config	Access for fetching configuration for a deployed xUML service
service_service-repository__xuml_remove_service	Access for removing an xUML service running as container
service_service-repository__xuml_restart_service	Access for restarting an xUML service running as container
service_service-repository__xuml_start_service	Access for starting an xUML service running as container
service_service-repository__xuml_status_service	Access for fetching running state for an xUML service running as container
service_service-repository__xuml_stop_service	Access for stopping an xUML service running as container
service_service-repository_available_services	Access to availableServices API from service repository
service_service-repository_change_service_config	Access to changeServiceConfig API from service repository
service_service-repository_control_service	Access to controlService API from service repository
service_service-repository_get_config	Access to getConfig API from service repository
service_service-repository_get_service	Access to getService API from service repository
service_service-repository_get_services	Access to getServices API from service repository
service_service-repository_get_validation_patterns	Access to getValidationPatterns API from service repository
service_service-repository_get_version	Access to getVersion API from service repository
service_service-repository_update_log_level	Access to updateLogLevel API from service repository

service_user_*	The access to the user administration is separated in multiple single permissions. For further information how to use these, please contact the Sc heer PAS Support team.
service-reporting	MS Office data export (Backend)
service-userself	User self service (Backend)
transaction_logger	Component Process Mining: Saving the transaction log
xuml_access_forms	Access granting UI access for all xUML services running as container
xuml_library_repository	Access to xUML library repository
xuml_library_repository_service	Access to xUML library repository service
xuml_monitor	Access to xUML Service Monitor