FLOWX.AI

# PLATFORM SETUP GUIDES / License engine setup guide

# Contents

# PLATFORM SETUP GUIDES / License engine setup guide / Configuring access rights for License

Granular access rights can be configured for restricting access to the License component.

The following access authorizations are provided, with the specified access scopes:

1. **Manage-licenses** - for configuring access for managing license related details

Available scopes:

- read - users are able to view the license report
- edit - users are able to update the license model and sync license data
- admin - users are able to download the license data

The License component is preconfigured with the following default users roles for each of the access scopes mentioned above:

- manage-licenses
  - read:
    - ROLE_LICENSE_MANAGE_READ
    - ROLE_LICENSE_MANAGE_EDIT
    - ROLE_LICENSE_MANAGE_ADMIN
  - edit:
    - ROLE_LICENSE_MANAGE_EDIT
    - ROLE_LICENSE_MANAGE_ADMIN
  - admin:
    - ROLE_LICENSE_MANAGE_ADMIN

> **DANGER**
>
> These roles need to be defined in the chosen identity provider solution.

In case other custom roles are needed, you can configure them using environment variables. More than one role can be set for each access scope.

To configure access for each of the roles above, adapt the following input:

`SECURITY_ACCESSAUTHORIZATIONS_AUTHORIZATIONNAME_SCOPES_SCOPENAME_ROLESALLOWED: NEEDED_ROLE_NAMES`

Possible values for `AUTHORIZATIONNAME: MANAGELICENSES`.

Possible values for `SCOPENAME`: read, edit, admin.

For example, if you need to configure role access for read, insert this:

```
SECURITY_ACCESSAUTHORIZATIONS_MANAGELICENSES_SCOPES_READ_ROLES
ROLE_NAME_TEST
```

**Was this page helpful?**

# PLATFORM SETUP GUIDES / License engine setup guide / Configuring access roles

> ⚠ **CAUTION**
>
> Deprecated since platform version 1.16.0

The License engine is able to offer different levels of accessing license related information.

In order to restrict API calls by user role you will need to add the user roles in the application config. You can configure separate roles for the provided API base routes:

```
    - path: "/api/report"
      rolesAllowed: ${LICENSE_VIEW}
    - path: "/api/license-model"
      rolesAllowed: ${LICENSE_MANAGER}
    - path: "/api/sync/**"
      rolesAllowed: ${LICENSE_SUPER_MANAGER}
    - path: "/api/data/**"
      rolesAllowed: ${LICENSE_SUPER_USER}
```

- `LICENSE_VIEW` - users with this role will be able to view the status of the license (just the usage info, no extra details)
- `LICENSE_MANAGER` - users with this role will be able to configure the license
- `LICENSE_SUPER_MANAGER` - users with this role will be able to trigger sync for the existing license
- `LICENSE_SUPER_USER` - users with this role will be able to request a detailed report with details of custom identifiers and dates when they appear (this can contain personal data)

**Was this page helpful?**