Structure Permissions

Every structure has a list of permission rules, which define who is allowed to see, edit or configure the structure.

Access Levels

Each user has one of the following access levels to a structure:

None	The user does not see the structure at all and does not know that it exists.
View	The user can view the structure but cannot make changes.
Edit	The user can view the structure and can rearrange, add and remove issues from the structure. The user cannot, however, create or modify generators.
Automate	The user has full edit access to the structure, including modifying generators and effectors.
Control	The user can view, edit and configure the structure - including changing structure permission rules.

Default Access

By default, all users have **None** access level.

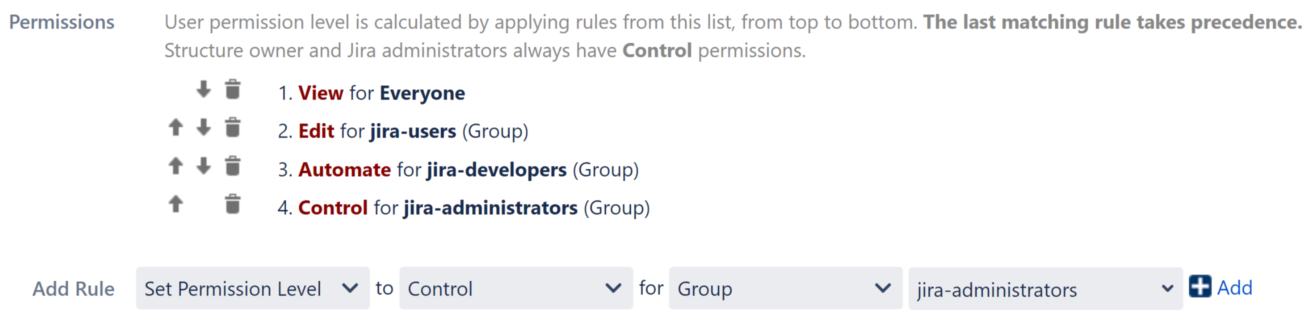
The structure's owner and Jira administrators always have **Control** access level.

Therefore, if you create a new structure and do not specify any permission rules, it will be a private structure that only you and Jira administrators will be able to see and modify.

Permission Rules

Users who have **Control** permission for a structure can define permission rules by Editing Structure Details.

The Permission Rules list is an ordered list that's used to calculate the access level for a given user. Each rule assigns an **access level** to a specific **condition** (category of users).



Permissions User permission level is calculated by applying rules from this list, from top to bottom. **The last matching rule takes precedence.** Structure owner and Jira administrators always have **Control** permissions.

1. **View** for **Everyone**
2. **Edit** for **jira-users** (Group)
3. **Automate** for **jira-developers** (Group)
4. **Control** for **jira-administrators** (Group)

Add Rule: Set Permission Level to Control for Group jira-administrators Add

The conditions are applied from top to bottom, and the **last matching rule has precedence** - so if a user fits multiple conditions, their access level will match the lowest-listed matching condition. For this reason, it is advisable to order permissions from least access (None) to most access (Control), as we've done above.

Setting Permissions

To set permissions, in the **Add Rule** section, select an access level and choose one of the following conditions for that access level:

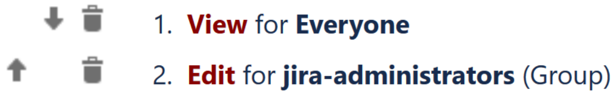
Everyone	Matches any user, including anonymous (not logged in). This condition can be used to set a default permission for everyone.
Group(G)	Matches users that belong to the group G.
Project Role(R,P)	Matches users that have role R in project P.
User	Adds the permission for a specific Jira user.

✓ To copy the permissions from another structure, click the **Set Permission Level** box and choose **Apply Permissions From**. This will let you apply the same permissions rules from any structure for which you have Control access level.

Permission Examples

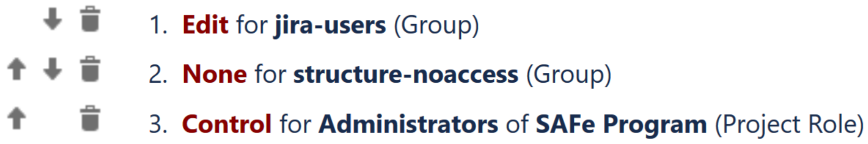
The following are examples of how your permissions list might look:

- Everyone can view the structure, Jira administrators can edit, only the owner and admins can control:



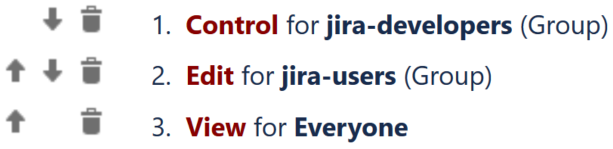
1. **View** for **Everyone**
2. **Edit** for **jira-administrators** (Group)

- Any logged-in user can edit the structure, except for the users from the structure-noaccess group, who can't even view the structure. Project administrators are allowed to control the structure:



1. **Edit** for **jira-users** (Group)
2. **None** for **structure-noaccess** (Group)
3. **Control** for **Administrators** of **SAFe Program** (Project Role)

- Incorrect configuration - in this example, everyone is given View access level:



1. **Control** for **jira-developers** (Group)
2. **Edit** for **jira-users** (Group)
3. **View** for **Everyone**

Although the configuration looks fine at first glance, remember that **the last matching rule has precedence**. So even if a user is part of the jira-developers or jira-users group, their access level will be set to View by the last rule.

Require Edit Issue permission on parent issue

When this option is selected on the Structure Details page, the user must have Edit Issue permission for a parent issue in order to adjust its sub-issues. In other words, direct sub-issues (or children issues) are treated as if they are part of the parent issue; therefore, adding sub-issues, removing sub-issues and rearranging sub-issues is actually changing the parent issue - for which the Edit Issue permission is required.

> The user must also have **Edit** access level to the structure to be able to make changes at all.

Note the following:

- Top-level issues do not have parent issues, and therefore are not affected by this flag: the user can add/rearrange issues at the top level of the structure if they have Edit access level.
- The Edit Issue permission applies only to the direct parent issue. If issue A has sub-issue B, and B has sub-issue C, then to be able to move or remove C from the structure, the user needs Edit Issue permission on B - not on A.

> Structure maintains a cache of users permissions with regards to each structure. In most cases, the cache is recalculated automatically, but in some cases Structure may miss a change in a user's groups or roles. This could mean that the changed permissions for the user do not take effect until several minutes later (but only with regards to Structure Permissions).
>
> A user can force the cache to be recalculated by doing a **hard refresh** from the browser. Typically, it's done by holding **Ctrl** or **Shift** or both and clicking the **Refresh** button.