

Jira Configuring projects

Managing project permissions

Project permissions are created within [permission schemes](#), which are then assigned to specific projects by Jira Administrators. Project permissions are able to be granted based on:

- Individual users
- Groups
- [Project roles](#)
- Issue roles such as 'Reporter', 'Project Lead', and 'Current Assignee'
- 'Anyone' (e.g. to allow anonymous access)
- A (multi-)user picker [custom field](#).
- A (multi-)group picker [custom field](#). This can either be an actual group picker custom field, or a (multi-)select-list whose values are group names.

Note that some permissions are dependent upon others to ensure that users can perform the actions needed. For example, in order for a user to be able to resolve an issue, that user must be granted both the Transition Issue permission and the Resolve Issue permission.

The following table lists the different types of project permissions and the functions they secure. Note that project permissions can also be used in [workflow conditions](#).

Project permissions overview

Project permissions	Explanation
Administer projects	Permission to administer a project in Jira. This includes the ability to edit project role membership , project components , project versions , and some project details ('Project Name', 'URL', 'Project Lead', 'Project Description').
Extended project administration	<p>Gives the project administrator the ability to edit workflows and screens under certain conditions.</p> <p>Restrictions for editing the project's workflows...</p> <ul style="list-style-type: none">• Restrictions for editing the project's screens... <p>*</p>
Browse projects	Permission to browse projects, use the Issue Navigator and view individual issues (except issues that have been restricted via issue-level security). Many other permissions are dependent on this permission , e.g. the 'Work On Issues' permission is only effective for users who also have the 'Browse Projects' permission.
Manage sprints (only available to Jira Software users)	<p>Permission to perform the following sprint-related actions for all projects in a board:</p> <ul style="list-style-type: none">• Creating sprints• Starting sprints• Completing sprints• Reopening sprints• Reordering future sprints• Adding sprint goals• Deleting sprints• Editing sprint information (sprint name, goal, dates)• Moving the sprint footer <p>Depending on the complexity of your board's filter query, you may need further consideration when configuring the 'Manage Sprints' permission for users. For more information on the impact of complex filters, and ways to simplify your filter query, see Using Manage Sprints permission for advanced cases.</p> <p>Notes on working with sprints</p> <ul style="list-style-type: none">•
View development tools (only available to Jira Software users)	Permission to view the Development panel , which provides you with just enough information to evaluate the status of an issue's development, at a glance.
View (read-only) workflow	Permission to view the project's 'read-only' workflow when viewing an issue. This permission provides the 'View Workflow' link against the 'Status' field of the 'View Issue' page.

Issue permissions	Explanation
Assign issues	Permission to assign issues to users. Also allows autocompletion of users in the Assign Issue dropdown. (See also Assignable User permission below)
Assignable user	Permission to be assigned issues. (Note that this does not include the ability to assign issues; see Assign Issue permission above).
Close issues	Permission to close issues based on the workflow conditions. (This permission is useful where, for example, developers resolve issues and testers close them). Requires the Transition issue and Resolve issue transitions. Also see the Resolve Issues permission.
Create issues	Permission to create issues in the project. (Note that the Create Attachments permission is required in order to create attachments.) Includes the ability to create sub-tasks (if sub-tasks are enabled).
Delete issues	Permission to delete issues. Think carefully about which groups or project roles you assign this permission to; usually it will only be given to administrators. Note that deleting an issue will delete all of its comments and attachments, even if the user does not have the Delete Comments or Delete Attachments permissions. However, the Delete Issues permission does not include the ability to delete individual comments or attachments.
Edit issues	Permission to edit issues (excluding the 'Due Date' field — see the Schedule Issues permission). Includes the ability to convert issues to sub-tasks and vice versa (if sub-tasks are enabled). Note that the Delete Issue permission is required in order to delete issues. The Edit Issue permission is usually given to any groups or project roles who have the Create Issue permission (perhaps the only exception to this is if you give everyone the ability to create issues — it may not be appropriate to give everyone the ability to edit too).
Link issues	Permission to link issues together. (Only relevant if Issue Linking is enabled).
Modify reporter	Permission to modify the 'Reporter' of an issue. This allows a user to create issues 'on behalf of' someone else. This permission should generally only be granted to administrators.
Move issues	Permission to move issues from one project to another, or from one workflow to another workflow within the same project. Note that a user can only move issues to a project for which they have Create Issue permission.
Resolve issues	Permission to resolve and reopen issues based on the workflow condition. This also includes the ability to set the 'Fix For version' field for issues. Requires the Transition issues permission. Also see the Close Issues permission.
Schedule issues	Permission to schedule an issue — that is, to edit the 'Due Date' of an issue. In older versions of Jira this also controlled the permission to view the 'Due Date' of an issue.
Set issues security	Permission to set the security level on an issue to control who can access the issue. Only relevant if issue security has been enabled .
Transition issues	Permission to transition (change) the status of an issue.
Voters & watchers permissions	Explanation
Manage watcher list	Permission to manage (i.e. view/add/remove users to/from) the watcher list of an issue.
View voters and watchers	Permission to view the voter list and watcher list of an issue. Also see the Manage Watcher List permission.
Comments permissions	Explanation
Add comments	Permission to add comments to issues. Note that this does not include the ability to edit or delete comments.
Delete all comments	Permission to delete any comments, regardless of who added them.
Delete own comments	Permission to delete comments that were added by the user.
Edit all comments	Permission to edit any comments, regardless of who added them.

Edit own comments	Permission to edit comments that were added by the user.
Attachments permissions	Explanation
Create attachments	Permission to attach files to an issue. (Only relevant if attachments are enabled). Note that this does not include the ability to delete attachments.
Delete all attachments	Permission to delete any attachments, regardless of who added them.
Delete own attachments	Permission to delete attachments that were added by the user.
Time-tracking Permissions	Explanation
Work on issues	Permission to log work against an issue, i.e. create a worklog entry. (Only relevant if time tracking is enabled).
Delete all worklogs	Permission to delete any worklog entries, regardless of who added them. (Only relevant if time tracking is enabled). Also see the Work On Issues permission.
Delete own worklogs	Permission to delete worklog entries that were added by the user. (Only relevant if time tracking is enabled). Also see the Work On Issues permission.
Edit all worklogs	Permission to edit any worklog entries, regardless of who added them. (Only relevant if time tracking is enabled). Also see the Work On Issues permission.
Edit own worklogs	Permission to edit worklog entries that were added by the user. (Only relevant if time tracking is enabled). Also see the Work On Issues permission.

Permission schemes

What is a permission scheme?


A permission scheme is a set of user/group/role assignments for the project permissions listed above. Every project has a permission scheme. One permission scheme can be associated with multiple projects.

Why permission schemes?


In many organizations, multiple projects have the same needs regarding access rights. (For example, only the specified project team may be authorized to assign and work on issues).

Permission schemes prevent having to set up permissions individually for every project. Once a permission scheme is set up it can be applied to all projects that have the same type of access requirements.

Creating a permission scheme


1. Choose  > **Issues**.
2. Select **Permission Schemes** to open the Permission Schemes page, which displays a list of all permission schemes in your Jira system and the projects that use each scheme.
3. Click the '**Add Permission Scheme**' link.
4. In the 'Add Permission Scheme' form, enter a name for the scheme, and a short description of the scheme. Select **Add**.
5. You will return to the 'Permission Schemes' page which now contains the newly added scheme.

Adding users, groups, or roles to a permission scheme


1. Choose  > **Issues**.
2. Select **Permission Schemes** to open the Permission Schemes page, which displays a list of all permission schemes in your Jira system and the projects that use each scheme.
3. Locate the permission scheme you would like to update, and select **Permissions** in the Operations column to view the scheme.
4. Select the '**Edit**' link for the permission you wish to add to, this displays the 'Grant permission' dialog.
5. Select who to add the selected permission to, and click the '**Grant**' button. The users/groups/roles will now be added to the selected permission. Note that [project roles](#) are useful for defining specific team members for each project. Referencing project roles (rather than users or groups) in your permissions can help you minimize the number of permission schemes in your system.

6. Repeat the last 2 steps until all required users/groups/roles have been added to the permissions.


Deleting users, groups, or roles from a permission scheme

1. Choose  > **Issues**.
2. Select **Permission Schemes** to open the Permission Schemes page, which displays a list of all permission schemes in your Jira system and the projects that use each scheme.
3. Locate the permission scheme of interest and click its name to show the list of 'Project Permissions' ([above](#)).
4. Click the **Remove** link for the permission you wish to remove the users, groups, or roles from.
5. Select the users, groups, or roles you wish to remove, and click the **Remove** button.


Associating a permission scheme with a project

1. Choose  > **Projects**, and select the relevant project.
2. Select the project of interest to open the **Project Summary** administration page for that project. See [Defining a project](#) for more information.
3. On the lower right, in the **Permissions** section, click the name of the current scheme (e.g. 'Default Permission Scheme') to display the details of the project's current permission scheme.
4. Click the **'Actions'** dropdown menu and choose **'Use a different scheme'**.
5. On the 'Associate Permission Scheme to Project' page, which lists all available permission schemes, select the permission scheme you want to associate with the project.
6. Click the **'Associate'** button to associate the project with the permission scheme.

Deleting a permission scheme

1. Choose  > **Issues**.
2. Select **Permission Schemes** to open the Permission Schemes page, which displays a list of all permission schemes in your Jira system and the projects that use each scheme.
3. Click the **Delete** link (in the **Operations** column) for the scheme that you want to delete.
4. A confirmation screen will appear. To delete click **Delete** otherwise click **Cancel**.
5. The scheme will be deleted and all associated projects will be automatically associated with the Default Permission Scheme. (Note that you cannot delete the Default Permission Scheme.)

Copying a permission scheme

1. Choose  > **Issues**.
2. Select **Permission Schemes** to open the Permission Schemes page, which displays a list of all permission schemes in your Jira system and the projects that use each scheme.
3. Click the **Copy** link (in the **Operations** column) for the scheme that you want to copy.
4. A new scheme will be created with the same permissions and the same users/groups/roles assigned to them.

Restricting Projects in Jira

As your Jira Software instance grows, restricting project access becomes an increasingly pressing need. Whether you're working with clients or just trying to control project visibility, strategic restrictions are vital.

There are numerous ways you can restrict access in Jira Software, but not all methods are created equal. The process covered below is the most scalable way to approach locking down your work.

3 Fundamental Concepts

There are three important Jira administration concepts to understand before we dive in.

1. **Users/Groups** - Users in Jira can be organized into [groups](#), i.e. Client X, iOS developers, et al., which allows for a more convenient treatment of a collection of like users.
2. **Permission schemes** - Every project has an associated permission scheme, [which is a list of potential actions one can take in any given project](#). A single permission scheme can be applied to any number of projects.
3. **Project Roles** - [Project roles](#) are a flexible way to associate users/groups with a Jira permission scheme. Project roles are somewhat similar to groups, in that they are groupings of like users. The main difference being that group membership is global whereas project role membership is project-specific.

Below, you'll find a handy chart on how all these concepts relate together. Don't worry if this strikes you as complicated right now; it'll make more sense in context.

Steps to Locking Down Projects

Say that you have two development projects, Project A and Project B. These projects should only be visible to their respective teams. Below, we'll cover how you can restrict said projects in a scalable fashion. We'll start with Project A. [*Note: We'll assume that you will stick to out-of-box project roles (administrators, developers).]*

User Management

1. In your site administration, group your existing user base. For developers in Project A, create a "Developers-A" group, for administrators, "Administrators-A," and so on.
2. Again in site administration, add your users to their determined group.
3. Now, go to your project settings for Project A. In the "People" section, assign your user groups to their respective project roles, i.e. Developers-A to the developers role.

At this point, you've mapped out your users to project roles.

Permission Scheme Management

You'll now need to create a new permission scheme. Assuming you're starting from scratch, all of your projects map to a default permission scheme (i.e. default software scheme), which permits access to all Jira users.

1. Toggle to the "Permission Schemes" section of your Jira administration. (A handy shortcut is to hit '.' and type in "Permission Schemes.")
2. Rather than build a permission scheme from scratch, clone the existing default permission scheme.
3. Once you have created the cloned scheme, click "Edit."
4. Of the many possible permissions, the one you'll want to really focus on is the "Browse Project" permission.
 - a. To this permission, add the project roles of administrator and developer
 - b. Remove the "any logged in user" access right.
5. Save this permission scheme with a unique, descriptive name.

Project Settings Management

Now that you've mapped your user groups to project roles and created a new permission scheme, your final step is to simply apply your new permission scheme to your project.

1. Go to Project A and in the lefthand toolbar toggle to "Settings" > "Permissions."
2. At the top right, select "Use a Different Scheme."
3. Choose the newly created permission scheme and you should be all set!

Project A should now be hidden from all users not in the user groups Developers-A and Administrators-A. To lock down Project B, you'll just repeat the steps above except you can skip creating a new permission scheme.

Final Considerations

As a whole, when you consider restricting projects, you'll want to keep 3 things in mind.

1. How you group your users together.
2. How user groups apply to project roles for a given project.

3. What permissions you grant those project roles according to the project's permission scheme.

As you get more comfortable with this process, you can probably start thinking of ways you can introduce additional project roles (client groups, executives, etc.) and what permissions you'd like to grant them through your permission schemes.

Hopefully, this helps clarify an admittedly complex process in Jira and gives you the groundwork to move forward and experiment further with locking down work.