


Contents

Permission manager


The permission manager can be accessed from the *Global actions* menu *under Administration > Permissions*. This link loads the page *Special:PermissionManager*.

There are four different permission settings. The default setting is "Private wiki". If you want to grant different permissions in different [namespaces](#), the setting "Custom setup" is required.


Permission manager

**Public wiki**


- Everyone can **view** and **edit** the content - including anonymous users
- Only users in the group "sysop" can perform **administrative actions**

**Protected wiki**

- Everyone can **view** the content - including anonymous users
- Only logged-in users can **edit** the content
- Only users in the group "sysop" can perform **administrative actions**

**Private wiki**

- Only logged-in users can **view** and **edit** the content
- Only users in the group "sysop" can perform **administrative actions**

**Custom setup**

- Assign roles and groups manually

Permissions manager

Contents

1	Settings	4
2	Custom setup	5
2.1	About role-based permissions	6
2.2	The roles matrix	7
2.3	Role inheritance	8
2.4	Default roles	8
3	Restricting read permissions	10
3.1	Extensions that are permissions-aware	10
3.2	Extensions and special pages that are not permissions-aware	10
3.3	Limited transclusion	11
4	Technical info	11
4.1	Logging	11
5	Configuration	11

6	Related info	11
---	--------------	----

Settings

An administrator can choose between three types of settings. The setting **Private Wiki** is activated by default.

Permission type	Description	Special permissions
Public wiki	<p>The wiki is visible to and editable by anyone, including anonymous users (that means users that are not logged in).</p> <ul style="list-style-type: none">• All users have edit rights by default.• To approve a page (if the approval mechanism is activated in a namespace), the groups "editor" and "reviewer" must be assigned to a user.• To manage the wiki (administrative tasks), the group "sysop" must be assigned to a user.	<p>Sonderverrehtung: [1]</p> <pre>// Anonymous and logged-in users can read and edit \$this->groupRoles ['*']['reader'] = true; \$this->groupRoles ['*']['editor'] = true;</pre>
Protected wiki	<p>The wiki is visible to anyone. Only logged-in users can edit the wiki.</p> <ul style="list-style-type: none">• All users have edit rights by default.• To approve a page (if the approval mechanism is activated in a namespace), the groups "editor" and "reviewer" must be assigned to a user.• To manage the wiki (administrative tasks), the group "sysop" must be assigned to a user.	<p>Sonderverrehtung: [1]:</p> <pre>// Anonymous users can read, logged-in users can edit \$this->groupRoles ['*']['reader'] = true; \$this->groupRoles ['*']['editor'] = false; \$this->groupRoles ['user']['editor'] = true;</pre>
Private wiki	<p>Only logged-in users can view and edit the wiki.</p> <ul style="list-style-type: none">• Logged-in users only have veiw permissions.• Important! To edit a page, users must be assigned to the "editor" group manually.• To approve a page (if the approval mechanism is activated in a namespace), the groups "editor" and "reviewer" must be assigned to a user.	<p>Sonderverrehtung: [1]</p> <pre>// Only logged-in users can read. The group "editor" has to be manually assigned to users. \$this->groupRoles ['*']['reader'] = false; \$this->groupRoles ['*']['editor'] = false; \$this->groupRoles ['user']['reader'] = true; \$this->groupRoles ['user']['editor'] = false; \$this->groupRoles</pre>

Permission type	Description	Special permissions
	<ul style="list-style-type: none">To manage the wiki (administrative tasks), the group "sysop" must be assigned to a user.	<pre>['editor']['editor'] = true; \$this->groupRoles ['sysop']['editor'] = true;</pre>
Custom setup (BlueSpice pro)	Roles and groups are assigned by an administrator. This is necessary if different namespaces need to have different user rights. See the next section for more info.	

1. ↑ [1.0](#) [1.1](#) [1.2](#) Global permissions (modified by the special permissions shown in the table above):

```
'bureaucrat' => [  
'accountmanager' => true  
],  
'sysop' => [  
'reader' => true,  
'editor' => true,  
'reviewer' => true,  
'admin' => true  
],  
'user' => [ 'editor' => true ],  
'editor' => [  
'reader' => true,  
'editor' => true  
],  
'reviewer' => [  
'reader' => true,  
'editor' => true,  
'reviewer' => true
```

Note: In BlueSpice pro Cloud, it is not possible to assign *edit*, *comment*, or *upload* rights to anonymous users.

Custom setup

Die Tabelle zeigt typische Standardeinstellungen für eine einfache Benutzerverwaltung:

Group	Default roles	Purpose of the group	Suggested roles
anonymous user (*)	-	Determines if anonymous users can read wiki pages.	(no role assignment) or <i>reader</i>
		Determines the rights authenticated users have if they don't	<i>reader</i> or

Group	Default roles	Purpose of the group	Suggested rolees
user	reader, editor	belong to any other groups. All groups except for anonymous users initially inherit permissions from this group.	<i>reader, editor</i>
editor	(von user geerbt), editor	Group members can edit the wiki.	<i>(editor)*</i>
reviewer	(von user geerbt), reviewer	Group members can approve page revisions if the approval feature is activated.	<i>reviewer</i>
sysop	(von user geerbt), editor, admin	Grants administrator rights to the wiki. Included in roles: admin, maintenanceadmin	<i>(editor)*, admin</i>
* can be inherited through the group "user"			

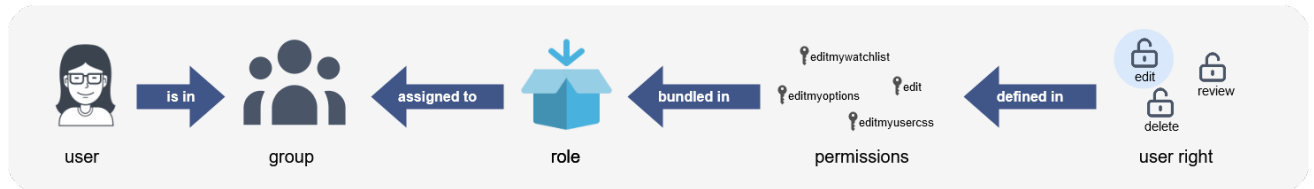
Note: If you have saved the personalized settings at least once and then switch back to a "protected" or "private" wiki, you do not lose your latest personalized settings. You can simply resave the latest saved settings to switch back to a wiki with personalized permissions.

About role-based permissions

Roles represent a **collection of individual permissions** that are necessary to perform certain functions in the wiki. For example, for a user who is supposed to only read the wiki, many permissions in addition to the "read" permission are needed: The ability to change their own settings, to search the wiki, to view page ratings, and so on.

All permissions that make up a logical group are encapsulated in a role, in this example the role "reader". If wiki administrators want to grant read-only rights to a user group, they only need to assign that group the "reader" role, instead of assigning many individual permissions that are needed to create a "read"-user.

By assigning roles to a group, all users belonging to that group receive the rights of these roles. Roles are never assigned directly to users, but always to groups instead. Users are then assigned to one or more groups.



As a result, the following administration pages play a role in rights management:

- **Namespace manager:** In the wiki, user groups can be granted different permissions via roles in individual namespaces.
- **Group manager:** Namespace permissions are assigned to user groups, not to individual users.
- **User manager:** Individual users are assigned to groups to obtain the permissions associated with the group.
- **Permission manager:** In the Permission manager, the user groups are assigned to their roles in the namespaces.

The roles matrix

The permission manager consists of the group tree (1) and the role matrix (2):

The screenshot shows the Permission manager interface. On the left, there is a **group tree** (labeled 1) showing a hierarchy of groups: **user**, **editor**, **groupfeedback**, **reviewer**, and **sysop**. The **user** group is selected. On the right, there is a **role matrix** (labeled 2) showing a table of roles and their permissions across various namespaces.

Role	Wiki	Namespaces					
		(Pages)	Geo/son	Minutes	QM	OM	OM
accountselfcreate	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
autocreateaccount	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
reader	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
commenter	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
author	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
editor	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
reviewer	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
structuremanager	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
accountmanager	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
admin	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
bot	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
maintenanceadmin	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

At the bottom of the interface, there are **Save** and **Reset** buttons. Below the table, it says "Associating groups with roles in namespaces". On the right side, there is a list of namespaces with checkboxes: **User**, **User talk**, **Project**, **Project talk**, **File**, **File talk**, **MediaWiki**, **MediaWiki talk**, **Template**, **Template talk**, **Help**, **Help talk**, **Category**, **Category talk**, **Property**, **Property talk**, **Form**, **Form talk**, **Concept**, **Concept talk**, **smw/schema**, **smw/schema talk**, **Rule**, **Rule talk**, **Widget**, and **Widget talk**.

The **group tree** shows all existing groups:

- **Group "*":** all non-logged-in (anonymous) users

- **Group "user"**: all logged-in users, the default group for all users
- **Subgroups of group "user"**: all groups that are defined on the wiki, either by default, by MediaWiki, or custom groups created in the [Group manager](#) by an administrator. System groups, created by MediaWiki, can be hidden by unchecking the "Show system groups" checkbox above the tree.

The columns in the **role matrix** are:

- **Role information** (info icon): Clicking the icon shows all the permissions in a role. This list is exportable.
- **Role name**
- **Wiki**: Assignment of a role to the entire wiki. By assigning the role in this column, a user group gets permissions in this role on the wiki (all namespaces).
- **Individual namespaces**: The following columns list every (applicable) namespace on the wiki.
 - Roles can be assigned to individual namespaces. For example, the group *user* can get the *editor* role only in the namespace *Public*. *Users in this group cannot edit content in any other* . By granting a role to a particular group in a particular namespace, means that all other groups will lose permissions from this role, eg. granting role "reader" in namespace "Private" to group "sysop" means that all users in any other groups won't be able to read pages in "Private" namespace, even if they have "reader" role granted on the wiki level ("Wiki" column).
 - The same role can be granted to multiple groups for the same namespace.
 - Additional namespaces can be added in the matrix by clicking on the arrow in table header, then "Columns". Then the namespaces can be selected.

Role inheritance

By default, all roles granted to the (*) group will be granted to the *user* group, and all roles granted to the *user* group are granted to its subgroups. If a group inherits the role from an upper-level group field, this is indicated in the role matrix with a green background, but the checkbox is empty.

Default roles

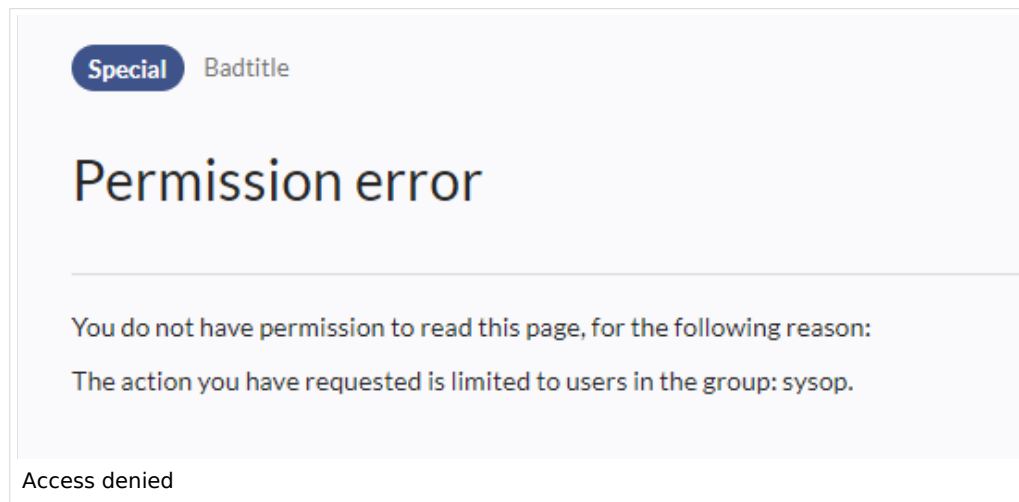
By default, the Permission manager includes a number of predefined roles that serve most user needs. The individual permissions contained in a role can be seen by clicking the info icon in front of the role name. It opens a dialog with a permissions list for the role.

Permission	Description
apihighlimits	Use higher limits in API queries
autoconfirmed	Not be affected by IP-based rate limits
autopatrol	Have one's own edits automatically marked as patrolled
autoreview	Have one's own edits automatically marked as "checked"
autoreviewrestore	Auto-review on rollback
bluespiceabout-viewspecia...	Access to the special page [[Special:BlueSpiceAbout]]
bot	Be treated as an automated process
checklistmodify	Modify the values of checklist items without editing the wiki ...

- **bot:** exists to achieve recurring system actions. This role is assigned to the user BSMaintenance in Bluespice via the group bot. The group bot should not be changed.
- **admin:** Grants access to all administrative special pages and to all typical administrative features.
- **maintenanceadmin:** Similar to the *admin* role, but with extended admin rights for maintaining wiki integrity.
- **author:** all permissions necessary for creating content on the wiki. Editing, moving, or deleting pages is not possible.
- **editor:** create content, edit and delete content.
- **reviewer:** If you have activated the review function and, therefore, work draft pages in a namespace, there must be at least one group with the role of reviewer. By default, the group "reviewer" is available for this purpose. Only users in the reviewer role can approve draft pages. Reviewers generally need read, write and review rights via the corresponding three roles of reader, editor and reviewer. However, if you have not activated the review function in any namespace, you do not need this role in your wiki.
- **accountmanager:** enables the administration of user accounts. Since user accounts are managed independently of namespaces in the wiki, this role cannot be restricted to individual namespaces. Grayed-out namespaces have no meaning here as long as the role in the wiki itself is highlighted in green.
- **structuremanager:** allows some actions for wiki maintenance such as moving pages, mass deleting pages or searching and replacing text, as well as renaming namespaces.
- **accountselfcreate:** enables the automatic creation of new user accounts and is required for single-sign-on. You can assign this role, for example, to anonymous users who can create their own account.
- **commenter:** allows the creation of discussion contributions and page ratings, but not of the pages themselves. The editor role includes all the rights of the commenter role. If a group has editor rights, it does not need special commenter rights.
- **reader:** Basic read access. Users can also edit their personal settings.

Restricting read permissions

It is possible to limit read permissions in a namespace by explicitly assigning the role *reader* to one or more particular groups. When users in other groups try to access a page in such a namespace, they will get a message that the permissions are denied.



While a user cannot access the content of the page, the wiki still shows links to these pages to all users in some contexts, even if a user does not have permissions to access the page content itself.

The following lists show which extensions or functionalities do not show links to restricted pages — because they are permissions-aware — and where the links are shown regardless of permissions.

Extensions that are permissions-aware

Query results and page lists provided by the following extensions do not show links to pages to which the current user has no access on the namespace level:

- [ExtendedSearch](#) (and functionality based on Extended Search in general, e.g. [TagSearch](#), [ExtendedFilelist](#)).
- [Semantic MediaWiki](#)
- [TopList](#)

Extensions and special pages that are not permissions-aware

Extensions that provide page lists and that do not hide links to read-restricted pages to the affected users. Examples:

- [DynamicPageList3](#)
- [SmartList](#)
- [WatchList](#) (both the tag and the special page)

In general, all MediaWiki special pages do not check permissions and therefore list these pages for the affected users. Most common examples:

- Special:All pages
- Special:RecentChanges
- Special:Bookshelf (**Note:** If this is an issue, you can limit access to the namespace *Book* to selected groups. The page *Special:Bookshelf* then won't show any links to books to users who do not have access to the *Book* namespace. Links to individual books can then be provided on various portal pages as needed).
- Category pages: All pages in the namespace *Category*

Limited transclusion

If you explicitly assign the *reader* role (or any other role that contains the *read* permission) in a namespace to a group or groups, that namespace is automatically configured so that its content cannot be transcluded. This is for security reasons, since MediaWiki does not check permissions when transcluding content.

Technical info

Logging

Every change to the roles is logged in `Special:Log`, in the `Permission Manager log`. These logs are available only to wiki administrators (users in groups with the role *admin*).

Configuration

All changes to the role matrix are backed up. By default, the last 5 backups are kept. This limit can be changed in [Config manager](#), under extension `BlueSpicePermissionManager`.

- **Backup limit:** Sets the number of backups for the permissions manager. Each time the page *Special:PermissionManager* is saved, a backup is created. If the backup limit is set to 5, the last five versions of the permissions configuration are saved as backups.

Related info

- [Reference:BlueSpicePermissionManager](#)
- [Group manager](#)

➔ [Technical Reference: BlueSpicePermissionManager](#)