+++ title = "Permissions" description = "Understand fine-grained access control permissions" keywords = ["grafana", "fine-grained access-control", "roles", "permissions", "enterprise"] weight = 110 +++

Permissions

A permission is an action and a scope. When creating a fine-grained access control, consider what specific action a user should be allowed to perform, and on what resources (its scope).

To grant permissions to a user, you create a built-in role assignment to map a role to a built-in role. A built-in role assignment *modifies* to one of the existing built-in roles in Grafana (Viewer, Editor, Admin). For more information, refer to [Built-in role assignments]({{< relref "./roles.md#built-in-role-assignments" >}}).

To learn more about which permissions are used for which resources, refer to [Resources with fine-grained permissions]({{< relref "./_index.md#resources-with-fine-grained-permissions" >}}).

action : The specific action on a resource defines what a user is allowed to perform if they have permission with the relevant action assigned to it.

scope : The scope describes where an action can be performed, such as reading a specific user profile. In such case, a permission is associated with the scope `users:<userId>` to the relevant role.

Action definitions

The following list contains fine-grained access control actions.

Action	Applicable scope	Description
`roles:list`	`roles:*`	List available roles without permissions.
`roles:read`	`roles:*` `roles:uid:*`	Read a specific role with its permissions.
`roles:write`	`permissions:delegate`	Create or update a custom role.
`roles:delete`	`permissions:delegate`	Delete a custom role.
`roles.builtin:list`	`roles:*`	List built-in role assignments.
`roles.builtin:add`	`permissions:delegate`	Create a built-in role assignment.
`roles.builtin:remove`	`permissions:delegate`	Delete a built-in role assignment.
`reports.admin:create`	n/a	Create reports.
`reports.admin:write`	`reports:*` `reports:id:*`	Update reports.
`reports:delete`	`reports:*` `reports:id:*`	Delete reports.
`reports:read`	`reports:*`	List all available reports or get a specific report.
`reports:send`	`reports:*`	Send a report email.
	n/a	Update report settings.

`reports.settings:write`		
`reports.settings:read`	n/a	Read report settings.
`provisioning:reload`	`provisioners:*`	Reload provisioning files. To find the exact scope for specific provisioner, see [Scope definitions]({{< relref "./permissions.md#scope-definitions" >}}).
`teams.roles:list`	`teams:*`	List roles assigned directly to a team.
`teams.roles:add`	`permissions:delegate`	Assign a role to a team.
`teams.roles:remove`	`permissions:delegate`	Unassign a role from a team.
`users:read`	`global.users:*`	Read or search user profiles.
`users:write`	`global.users:*` `global.users:id:*`	Update a user's profile.
`users.teams:read`	`global.users:*` `global.users:id:*`	Read a user's teams.
`users.authtoken:list`	`global.users:*` `global.users:id:*`	List authentication tokens that are assigned to a user.
`users.authtoken:update`	`global.users:*` `global.users:id:*`	Update authentication tokens that are assigned to a user.
`users.password:update`	`global.users:*` `global.users:id:*`	Update a user's password.
`users:delete`	`global.users:*` `global.users:id:*`	Delete a user.
`users:create`	n/a	Create a user.
`users:enable`	`globa.users:*` `global.users:id:*`	Enable a user.
`users:disable`	`global.users:*` `global.users:id:*`	Disable a user.
`users.permissions:update`	`global.users:*` `global.users:id:*`	Update a user's organization-level permissions.
`users:logout`	`global.users:*` `global.users:id:*`	Sign out a user.
`users.quotas:list`	`global.users:*` `global.users:id:*`	List a user's quotas.
`users.quotas:update`	`global.users:*` `global.users:id:*`	Update a user's quotas.
`users.roles:list`	`users:*`	List roles assigned directly to a user.

`users.roles:add`	`permissions:delegate`	Assign a role to a user.
`users.roles:remove`	`permissions:delegate`	Unassign a role from a user.
`users.permissions:list`	`users:*`	List permissions of a user.
`org.users:read`	`users:*` `users:id:*`	Get user profiles within an organization.
`org.users:add`	`users:*`	Add a user to an organization.
`org.users:remove`	`users:*` `users:id:*`	Remove a user from an organization.
`org.users.role:update`	`users:*` `users:id:*`	Update the organization role (`Viewer`, `Editor` or `Admin`) of an organization.
`orgs:read`	`orgs:*` `orgs:id:*`	Read one or more organizations.
`orgs:write`	`orgs:*` `orgs:id:*`	Update one or more organizations.
`org:create`	n/a	Create an organization.
`orgs:delete`	`orgs:*` `orgs:id:*`	Delete one or more organizations.
`orgs.quotas:read`	`orgs:*` `orgs:id:*`	Read organization quotas.
`orgs.quotas:write`	`orgs:*` `orgs:id:*`	Update organization quotas.
`orgs.preferences:read`	`orgs:*` `orgs:id:*`	Read organization preferences.
`orgs.preferences:write`	`orgs:*` `orgs:id:*`	Update organization preferences.
`ldap.user:read`	n/a	Read users via LDAP.
`ldap.user:sync`	n/a	Sync users via LDAP.
`ldap.status:read`	n/a	Verify the availability of the LDAP server or servers.
`ldap.config:reload`	n/a	Reload the LDAP configuration.
`status:accesscontrol`	`services:accesscontrol`	Get access-control enabled status.

`settings:read`	`settings:*`
	`settings:auth.saml:*`
	`settings:auth.saml:enabled` (property level)
		Read the [Grafana configuration settings]({{< relref "../../administration/configuration/_index.md >}})

`settings:write`	`settings:*`	Update any Grafana configuration settings

	`settings:auth.saml:*` `settings:auth.saml:enabled` (property level)	that can be [updated at runtime]({{< relref "../../enterprise/settings-updates/_index.md" >}}).
`server.stats:read`	n/a	Read Grafana instance statistics.
`datasources:explore`	n/a	Enable access to the **Explore** tab.

`datasources:read`	n/a
	`datasources:*`
	`datasources:id:*`
	`datasources:uid:*`
	`datasources:name:*`
		List data sources.

`datasources:query`	n/a
	`datasources:*`
	`datasources:id:*`
		Query data sources.

`datasources.id:read`	`datasources:*` `datasources:name:*`	Read data source IDs.
`datasources:create`	n/a	Create data sources.
`datasources:write`	`datasources:*` `datasources:id:*`	Update data sources.

`datasources:delete`	`datasources:id:*`
	`datasources:uid:*`
	`datasources:name:*`
		Delete data sources.

`datasources.permissions:read`	`datasources:*` `datasources:id:*`	List data source permissions.
`datasources.permissions:write`	`datasources:*` `datasources:id:*`	Update data source permissions.
`licensing:read`	n/a	Read licensing information.
`licensing:update`	n/a	Update the license token.
`licensing:delete`	n/a	Delete the license token.
`licensing.reports:read`	n/a	Get custom permission reports.
`teams:create`	n/a	Create teams.
`teams:read`	`teams:*` `teams:id:*`	Read one or more teams and team preferences.
`teams:write`	`teams:*` `teams:id:*`	Update one or more teams and team preferences.
`teams:delete`	`teams:*` `teams:id:*`	Delete one or more teams.
`teams.permissions:read`	`teams:*`	Read members and External Group

	`teams:id:*`	Synchronization setup for teams.
`teams.permissions:write`	`teams:*` `teams:id:*`	Add, remove and update members and manage External Group Synchronization setup for teams.

`dashboards:read`	`dashboards:*`
	`dashboards:id:*`
	`folders:*`
	`folders:id:*`
		Read one or more dashboards.

`dashboards:create`	`folders:*` `folders:id:*`	Create dashboards in one or more folders.

`dashboards:write`	`dashboards:*`
	`dashboards:id:*`
	`folders:*`
	`folders:id:*`
		Update one or more dashboards.

`dashboards:edit`	`dashboards:*`
	`dashboards:id:*`
	`folders:*`
	`folders:id:*`
		Edit one or more dashboards (only in ui).

`dashboards:delete`	`dashboards:*`
	`dashboards:id:*`
	`folders:*`
	`folders:id:*`
		Delete one or more dashboards.

`dashboards.permissions:read`	`dashboards:*`
	`dashboards:id:*`
	`folders:*`
	`folders:id:*`
		Read permissions for one or more dashboards.

`dashboards.permissions:write`	`dashboards:*`
	`dashboards:id:*`
	`folders:*`
	`folders:id:*`
		Update permissions for one or more dashboards.

`folders:read`	`folders:*` `folders:id:*`	Read one or more folders.
`folders:create`	n/a	Create folders.
`folders:write`	`folders:*` `folders:id:*`	Update one or more folders.
`folders:delete`	`folders:*` `folders:id:*`	Delete one or more folders.
`folers.permissions:read`	`folders:*` `folders:id:*`	Read permissions for one or more folders.
`folders.permissions:write`	`folders:*`	Update permissions for one or more folders.

	`folders:id:*`	
`annotations.read`	`annotations:*` `annotations:type:*`	Read annotations and annotation tags.
`annotations.create`	`annotations:*` `annotations:type:*`	Create annotations.
`annotations.write`	`annotations:*` `annotations:type:*`	Update annotations.
`annotations.delete`	`annotations:*` `annotations:type:*`	Delete annotations.

Scope definitions

The following list contains fine-grained access control scopes.

Scopes	Descriptions
`permissions:delegate`	The scope is only applicable for roles associated with the Access Control itself and indicates that you can delegate your permissions only, or a subset of it, by creating a new role or making an assignment.
`roles:*` `roles:uid:*`	Restrict an action to a set of roles. For example, `roles:*` matches any role and `roles:uid:randomuid` matches only the role whose UID is `randomuid`.
`reports:*` `reports:id:*`	Restrict an action to a set of reports. For example, `reports:*` matches any report and `reports:id:1` matches the report whose ID is `1`.
`services:accesscontrol`	Restrict an action to target only the fine-grained access control service. You can use this in conjunction with the `status:accesscontrol` actions.
`global.users:*` `global.users:id:*`	Restrict an action to a set of global users. For example, `global.users:*` matches any user and `global.users:id:1` matches the user whose ID is `1`.
`teams:*` `teams:id:*`	Restrict an action to a set of teams from an organization. For example, `teams:*` matches any team and `teams:id:1` matches the team whose ID is `1`.
`users:*` `users:id:*`	Restrict an action to a set of users from an organization. For example, `users:*` matches any user and `users:id:1` matches the user whose ID is `1`.
`orgs:*` `orgs:id:*`	Restrict an action to a set of organizations. For example, `orgs:*` matches any organization and `orgs:id:1` matches the organization whose ID is `1`.
`settings:*`	Restrict an action to a subset of settings. For example, `settings:*` matches all settings, `settings:auth.saml:*` matches all SAML settings, and `settings:auth.saml:enabled` matches the enable property on the SAML settings.
`provisioners:*`	Restrict an action to a set of provisioners. For example, `provisioners:*` matches any provisioner, and `provisioners:accesscontrol` matches the fine-grained access control [provisioner]({{< relref "./provisioning.md" >}}).
`datasources:*`	Restrict an action to a set of data sources. For example, `datasources:*` matches any

`datasources:id:*`
`datasources:uid:*`
`datasources:name:*`
	data source, and `datasources:name:postgres` matches the data source named `postgres`.

`folders:*` `folders:id:*`	Restrict an action to a set of folders. For example, `folders:*` matches any folder, and `folders:id:1` matches the folder whose ID is `1`.
`dashboards:*` `dashboards:id:*`	Restrict an action to a set of dashboards. For example, `dashboards:*` matches any dashboard, and `dashboards:id:1` matches the dashboard whose ID is `1`.
`annotations:*` `annotations:type:*`	Restrict an action to a set of annotations. For example, `annotations:*` matches any annotation, `annotations:type:dashboard` matches annotations associated with dashboards and `annotations:type:organization` matches organization annotations.