+++ title = "Fine-grained access control usage scenarios" description = "Fine-grained access control usage scenarios" keywords = ["grafana", "fine-grained-access-control", "roles", "permissions", "fine-grained-access-control-usage", "enterprise"] weight = 125 +++

Fine-grained access control usage scenarios

This guide contains several examples and usage scenarios of using fine-grained roles and permissions for controlling access to Grafana resources.

Before you get started, make sure to [enable fine-grained access control]({{< relref "./_index.md#enable-fine-grained-access-control" >}}).

Check all built-in role assignments

You can use the [Fine-grained access control HTTP API]({{< relref "../../http_api/access_control.md#get-all-built-in-role-assignments" >}}) to see all available built-in role assignments. The response contains a mapping between one of the organization roles (`Viewer` , `Editor` , `Admin`) or `Grafana Admin` to the custom or fixed roles.

Example request:

```
curl --location --request GET '<grafana_url>/api/access-control/builtin-roles' --
header 'Authorization: Basic YWRtaW46cGFzc3dvcmQ='
```

You must use the base64 username:password Basic Authorization here. Auth tokens are not applicable here.

Example response:

```
{
    "Admin": [
        ...
        {
            "version": 2,
            "uid": "qQui_LCMk",
            "name": "fixed:users:org:writer",
            "displayName": "Users Organization writer",
            "description": "Within a single organization, add a user, invite a user,
read information about a user and their role, remove a user from that organization, or
change the role of a user.",
            "global": true,
            "updated": "2021-05-17T20:49:18+02:00",
            "created": "2021-05-13T16:24:26+02:00"
        },
        {
            "version": 1,
            "uid": "Kz9m_YjGz",
            "name": "fixed:reports:writer",
            "displayName": "Report writer",
            "description": "Create, read, update, or delete all reports and shared
report settings.",
            "global": true,
            "updated": "2021-05-13T16:24:26+02:00",
            "created": "2021-05-13T16:24:26+02:00"
```

```
        }
        ...
    ],
    "Grafana Admin": [
        ...
        {
            "version": 2,
            "uid": "qQui_LCMk",
            "name": "fixed:users:writer",
            "displayName": "User writer",
            "description": "Read and update all attributes and settings for all users
in Grafana: update user information, read user information, create or enable or
disable a user, make a user a Grafana administrator, sign out a user, update a user's
authentication token, or update quotas for all users.",
            "global": true,
            "updated": "2021-05-17T20:49:18+02:00",
            "created": "2021-05-13T16:24:26+02:00"
        },
        {
            "version": 2,
            "uid": "ajum_YjGk",
            "name": "fixed:users:reader",
            "displayName": "User reader",
            "description": "Allows every read action for user organizations and in
addition allows to administer user organizations.",
            "global": true,
            "updated": "2021-05-17T20:49:17+02:00",
            "created": "2021-05-13T16:24:26+02:00"
        },
        ...
    ]
}
```

To see what permissions each of the assigned roles have, you can a [Get a role]({{< relref "../../http_api/access_control.md#get-a-role" >}}) by using an HTTP API.

Example request:

```
curl --location --request GET '<grafana_url>/api/access-control/roles/qQui_LCMk' --
header 'Authorization: Basic YWRtaW46cGFzc3dvcmQ='
```

Example response:

```
{
    "version": 2,
    "uid": "qQui_LCMk",
    "name": "fixed:users:writer",
    "displayName": "User writer",
    "description": "Read and update all attributes and settings for all users in
Grafana: update user information, read user information, create or enable or disable a
user, make a user a Grafana administrator, sign out a user, update a user's
authentication token, or update quotas for all users.",
```

```
    "global": true,
    "permissions": [
        {
            "action": "org.users:add",
            "scope": "users:*",
            "updated": "2021-05-17T20:49:18+02:00",
            "created": "2021-05-17T20:49:18+02:00"
        },
        {
            "action": "org.users:read",
            "scope": "users:*",
            "updated": "2021-05-17T20:49:18+02:00",
            "created": "2021-05-17T20:49:18+02:00"
        },
        {
            "action": "org.users:remove",
            "scope": "users:*",
            "updated": "2021-05-17T20:49:18+02:00",
            "created": "2021-05-17T20:49:18+02:00"
        },
        {
            "action": "org.users.role:update",
            "scope": "users:*",
            "updated": "2021-05-17T20:49:18+02:00",
            "created": "2021-05-17T20:49:18+02:00"
        }
    ],
    "updated": "2021-05-17T20:49:18+02:00",
    "created": "2021-05-13T16:24:26+02:00"
}
```

Manage roles granted directly to users

To learn about granting roles to users, refer to [Manage user role assignments]({{< relref "manage-role-assignments/manage-user-role-assignments.md" >}}) page.

Create your first custom role

You can create your custom role by either using an [HTTP API]({{< relref "../../http_api/access_control.md#create-a-new-custom-role" >}}) or by using [Grafana provisioning]({{< relref "./provisioning.md" >}}). You can take a look at [actions and scopes]({{< relref "./provisioning.md#action-definitions" >}}) to decide what permissions would you like to map to your role.

Example HTTP request:

```
curl --location --request POST '<grafana_url>/api/access-control/roles/' \
--header 'Authorization: Basic YWRtaW46cGFzc3dvcmQ=' \
--header 'Content-Type: application/json' \
--data-raw '{
    "version": 1,
    "uid": "jZrmlLCkGksdka",
    "name": "custom:users:admin",
```

```
    "displayName": "custom users admin",
    "description": "My custom role which gives users permissions to create users",
    "global": true,
    "permissions": [
        {
            "action": "users:create"
        }
    ]
}'
```

Example response:

```
{
    "version": 1,
    "uid": "jZrmlLCkGksdka",
    "name": "custom:users:admin",
    "displayName": "custom users admin",
    "description": "My custom role which gives users permissions to create users",
    "global": true,
    "permissions": [
        {
            "action": "users:create"
            "updated": "2021-05-17T22:07:31.569936+02:00",
            "created": "2021-05-17T22:07:31.569935+02:00"
        }
    ],
    "updated": "2021-05-17T22:07:31.564403+02:00",
    "created": "2021-05-17T22:07:31.564403+02:00"
}
```

Once the custom role is created, you can create a built-in role assignment by using an [HTTP API]({{< relref "../../http_api/access_control.md#create-a-built-in-role-assignment" >}}). If you created your role using [Grafana provisioning]({{< relref "./provisioning.md" >}}), you can also create the assignment with it.

Example HTTP request:

```
curl --location --request POST '<grafana_url>/api/access-control/builtin-roles' \
--header 'Authorization: Basic YWRtaW46cGFzc3dvcmQ=' \
--header 'Content-Type: application/json' \
--data-raw '{
    "roleUid": "jZrmlLCkGksdka",
    "builtinRole": "Viewer",
    "global": true
}'
```

Example response:

```
{
    "message": "Built-in role grant added"
}
```

Allow Viewers to create reports

In order to create reports, you need to have `reports.admin:write` permission. By default, a Grafana Admin or organization Admin can create reports as there is a [built-in role assignment]({{< relref "./roles#built-in-role-assignments" >}}) which comes with `reports.admin:write` permission.

If you want your users who have the `Viewer` organization role to create reports, you have two options:

1. Create a built-in role assignment and map the `fixed:reporting:admin:edit` fixed role to the `Viewer` built-in role. Note that the `fixed:reporting:admin:edit` fixed role allows doing more than creating reports. Refer to [fixed roles]({{< relref "./roles.md#fixed-roles" >}}) for full list of permission assignments.
2. [Create a custom role]({{< ref "#create-your-custom-role" >}}) with `reports.admin:write` permission, and create a built-in role assignment for `Viewer` organization role.

Prevent Grafana Admin from creating and inviting users

In order to create users, you need to have `users:create` permission. By default, a user with the Grafana Admin role can create users as there is a [built-in role assignment]({{< relref "./roles#built-in-role-assignments" >}}) which comes with `users:create` permission.

If you want to prevent Grafana Admin from creating users, you can do the following:

1. [Check all built-in role assignments]({{< ref "#check-all-built-in-role-assignments" >}}) to see what built-in role assignments are available.
2. From built-in role assignments, find the role which gives `users:create` permission. Refer to [fixed roles]({{< relref "./roles.md#fixed-roles" >}}) for full list of permission assignments.
3. Remove the built-in role assignment by using an [Fine-grained access control HTTP API]({{< relref "../../http_api/access_control.md" >}}) or by using [Grafana provisioning]({{< relref "./provisioning" >}}).

Allow Editors to create new custom roles

By default, the Grafana Server Admin is the only user who can create and manage custom roles. If you want your users to do the same, you have two options:

1. Create a built-in role assignment and map `fixed:permissions:admin:edit` and `fixed:permissions:admin:read` fixed roles to the `Editor` built-in role.
2. [Create a custom role]({{< ref "#create-your-custom-role" >}}) with `roles.builtin:add` and `roles:write` permissions, then create a built-in role assignment for `Editor` organization role.

Note that any user with the ability to modify roles can only create, update or delete roles with permissions they themselves have been granted. For example, a user with the `Editor` role would be able to create and manage roles only with the permissions they have, or with a subset of them.