

Roles and permissions

Another Check-In · A06

Before the demonstration

Bertland Hope

- Your assigned responsibility
- The correct training record
- Evidence of the result

Today's outcome

Learner can justify and verify a least-privilege role without treating menu hiding as the only control.

Existing role grants

Bertrand Hope

- Open Admin > Roles and inspect an existing training role and permissions

In this step

Start with the responsibilities the person has been approved to perform. Read the grants on the role instead of judging access by its name. A familiar role label can hide deliberate local changes.

A training role

Bertland Hope

- Create a named training role using the approved preset or selected grants

In this step

A preset provides a starting set of permissions for a named role. It does not mean a role with that name already exists. Use a disposable training role and document the intended responsibilities before assigning it.

Permission review

Bertrand Hope

- Review each permission against the job responsibility and save

In this step

Match each grant to a real duty. Organization-wide visibility should not be added simply to make missing data appear. A new permission label also cannot create an application feature that the code does not implement.

Effective access

Berland Hope

- Assign the role to the fictional user through Users, then verify allowed and denied surfaces in a separate session

In this step

Verify the fictional user in a separate session. Check both an allowed report and a prohibited change. Hidden menus help navigation, while server authorization determines whether a request is allowed.

Inactive roles

Bertland Hope

- Explain deactivate and restore-default behavior using a disposable role

In this step

Inactive roles stop contributing authority, but another active role may still grant the same permission. Verify the effective result on the next request. The built-in admin role is protected, and Restore Defaults can replace deliberate local choices.

Common problems

Berland Hope

- Assuming preset names already exist as roles
- granting organization.view_all to fix missing data
- thinking a custom permission name implements a feature

A useful support report

The task and record
The expected result
The actual message
The authorized next step

Guided practice

Berland Hope

- Create a restricted training role from report_viewer, assign it to a fictional user and show one allowed report plus a denied mutation

Evidence to show

The correct training case

The result or permitted decision

Your explanation of the next step

- Correct context and permitted action
- A result you can trace
- A clear next step if blocked

Expected result

Learner can justify and verify a least-privilege role without treating menu hiding as the only control.