Fabric Credentials Commands

The following tables discuss how user access control is managed using Fabric commands.

CREATE Command

CREATE commands are used to create a user, role or token in the k2auth keyspace in Cassandra:

CREATE USER

Description: Create a Fabric user. The user is saved in the system_auth.roles Cassandra table.

Usage:

CREATE USER <'username'> [WITH PASSWORD <'password'>] [SUPERUSER | NOSUPERUSER | (empty)]

Parameters:

- <'username'> – mandatory, user name.
- [WITH PASSWORD <'password'>] – optional, password.
- [SUPERUSER | NOSUPERUSER | (empty)] – optional, indicates whether the user is a superuser (admin) or not. If empty, the user is not a superuser.

Examples:

- Create a regular user named **test_user** without a password:
  - create user test_user;
- Create a superuser named **sup_user** without a password:
  - create a **sup_user** user without a password and as a superuser.
- Create a regular user named **psw_user** with password '12345':
  - create user psw_user with password ‘12345’;

CREATE ROLE

Description: Create a new role with a description.

Usage:

CREATE ROLE <'role_name'> [description <'role description'>]

Parameters:

- <'role_name'> – mandatory, role name.
- [description <'role description'>] – optional, role description.

Examples:

- Create a role named **test_role** without a description:
  - create role 'test_role';
- Create a role named **test_role** role with a description:
  - create role 'test_role' description 'test the desc';

CREATE TOKEN

Description: Create a new API Key.

Usage:

CREATE TOKEN <'token_name'> [SECURED]

Parameters:

- <'token_name'> – mandatory, API Key name.
- SECURED – optional in case of a secured API Key.

Examples:

- Create a **test_token** token. Do not assign it to any user:
  - create token 'test_token';

ASSIGN Command

ASSIGN commands are used to assign a role either for a user or for a token:

ASSIGN ROLE <ROLE> to user <USER>

Description: Assign a role to a user.

Usage:

ASSIGN ROLE <role> to USER <user>

Example:

assign role role1 to user test_user;

ASSIGN ROLE <ROLE> to token <API Key>	**Description**: Assign a role to a token. **Usage**: ASSIGN ROLE <role> to TOKEN <token> **Example:** assign role role1 to token test_token;

GRANT Command

GRANT commands are used to enable specific roles to access Fabric. These commands insert records into the **Permissions table** in the k2auth keyspaces in Cassandra. Several **GRANT** commands can be run for the same LU. The permissions granted by the **GRANT** command will be appended.

The following **GRANT** commands are supported:

GRANT <OPERATION> ON <RESOURCE> TO <ROLE>

Below is a list of GRANT OPERATION command parameters:

Parameter Name	Mandatory	Description
Operation	Yes	Fabric operations included in the permission. **Examples:** • ALL - give permission on all Fabric activities. • ALL_WS - give permission on all Fabric Web Services. • Web Service name • Role activities: REVOKE_ROLE, ASSIGN_ROLE, EDIT_ROLE • Other Fabric activities: READ, DROP_LUTYPE, DEPLOY, MIGRATE… Run a help grant; command to view the full list of operations.
Resource	Yes	Fabric resources included in the permission: • * - grant permissions on all Fabric resources. • <LUT> - LU name. Grant permissions on a given LU. • <LUT>.<List of LUIs> - grant permissions for a list of instance IDs in a given LU.
Role	Yes	The permissions are granted to a given role.

GRANT OPERATION Command - Examples

Description	Example
Allow the role to access all instances of all LUs.	grant all on * to role1;
Allow the role to access all instances of a specific LU.	Grant all CRM LU permissions to the role1 role: grant all on CRM to role1;
Allow the role to access specific instances of a specific LU.	Grant all permissions for Instance IDs 41 and 42 of CRM LU to the role1 role: grant all on CRM.41, CRM.42 to role1;
Allow the role to deploy CRM LU on Fabric.	grant deploy on CRM to role1;
Allow the role of migrating Instance IDs on Customer LU.	grant migrate on Customer to role1;
Allow the role to access Instance IDs 1, 2, 4, and 6 of CRM LU.	grant all on CRM.1, CRM.2, CRM.4,CRM.6 to role1;
Allow the role to access Instance IDs 1 and 2 of CRM LU, and Instance ID 57 of Customer LU.	grant all on CRM.1, CRM.2, Customer.57 to role1;
Allow the role to invoke wsGetCustomerDetails Fabric Web Service to access CRM LU	grant wsGetCustomerDetails on CRM to role1;

GRANT <WS_NAME> TO <ROLE>

Enables users to give generic access to a given Web Service to access the Fabric database.

Notes:

- Use the **GRANT OPERATION** command to limit the access of the Web Service to a given LU or LUI.
- Use the **GRANT OPERATION** command to grant an access to all Web Services: populate the **Operation** parameter by **ALL_WS**.

Below is a list of GRANT WS_NAME command parameters:

Parameter Name	Mandatory	Description
WS_Name	Yes	Fabric Web Service name.
Role	Yes	The permissions are granted to a given role.

GRANT WS_NAME Command - Examples

Description	Example
Allow the role to invoke wsGetCustomerDetails	grant wsGetCustomerDetails to role1;

Web Services Authorization

Web Service authorization is performed using the **API Key**:

- Project Web Services: give permission to the **role** in the Web Service or all Web Services and assign the API Key to the role.
- Product Web Services: assign the API key to the user. The permissions for Product Web Services are defined by combining the API Key assigned to the user and the permissions of the roles assigned to the user.

Example

```
create user 'test_read';
create role 'readonly';
grant READ on * to 'readonly';
assign 'readonly' to 'test_read';
assign role 'readonly' to user 'test_read';
create token 'test_read_token' user 'test_read';
```

When invoking the DELETE WS: /lu/{luName}/{iid} using the 'test_token' token, Fabric throws the following error: "Com.k2view.cdbms.exceptions.UnauthorizedException: test_read is not allowed to perform [DELETE_INSTANCE]"

Additional Commands

Description: Drop the user from Fabric.

DROP USER Usage: DROP USER <user>

Example: drop user test_user;

Description: Drop a role from Fabric.

DROP ROLE Usage: DROP ROLE <role name>

Example: drop role role1;

Description: Drop a token from Fabric.

DROP TOKEN Usage: DROP TOKEN <API Key>

Example: drop token test_token;

Description: Remove the granted permissions from a role.

Usage: Revoke syntax is the same as Grant commands with the exception that "Revoke" replaces "Grant" and "From" replaces "To".

REVOKE Examples:

- revoke all on CRM from role1;
- revoke migrate on Customer from role1;
- revoke wsGetCustomerDetails on Customer from role1;

Description: Unassign a role from a user or a token.

Usage:

- REVOKE ROLE <ROLE> FROM USER <user>

REVOKE ROLE

- REVOKE ROLE <ROLE> FROM TOKEN <token>

Examples:

- revoke test_role from test_user;

Description: Check the permission for the user on the given method (operation).

CHECK PERMISSION

Usage: CHECK_PERMISSION FOR <user> ON <operation>

Example: check_permission for test_user on deploy;