

Authorization

Authorization is the function of specifying access rights to resources and operations. In Jethro, once a user is properly [authenticated](#) and granted an access to the system, defined ROLES and OWNERSHIP will control which content and objects will the user be able to access, and which actions a user, or group of users, would be able to perform on that content.

*While authorization and authentication are closely connected, it is important not to confuse between them, as they are distinct from one another. Authentication verifies the identity of a client, while Authorization determines its operations and access policies. To learn more about Authentication, visit its [documentation page](#).

The following subjects will be covered under this section:

- [Ownership Permissions Model](#)
- [Roles Permissions Model](#)
 - [Objects Hierarchy](#)
 - [Permissions for objects](#)
 - [Permissions for operations](#)
- [Commands Syntax](#)
 - [Roles](#)
 - [CREATE ROLE](#)
 - [DROP ROLE](#)
 - [GRANT ROLE](#)
 - [REVOKE ROLE](#)
 - [SET ROLE](#)
 - [SHOW ROLES](#)
 - [SHOW GRANT ROLES](#)
 - [PERMISSIONS](#)
 - [GRANT <PERMISSION>](#)
 - [REVOKE <PERMISSION>](#)
 - [SHOW GRANT](#)
 - [ALTER](#)
 - [ALTER SCHEMA ... OWNER TO](#)
 - [ALTER TABLE ... OWNER TO](#)
 - [SHOW](#)
 - [SHOW LDAP GROUPS](#)
 - [SHOW SCHEMAS](#)
 - [SHOW TABLES](#)
- [Command Line Operations](#)

Ownership Permissions Model

- The user that creates a TABLE, VIEW or SCHEMA, becomes its OWNER.
- The object OWNER gets all the PERMISSIONS for the object.
- Object ownership can be altered via [ALTER TABLE](#) command.

As a result:

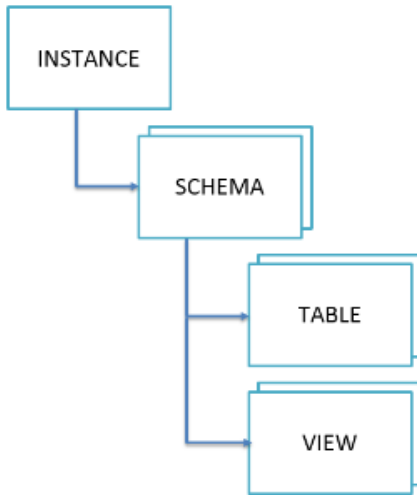
- A User that creates a table/view will have full access to it.
- Any other user can gain access to this object, if he/she are granted a permission, via **ROLES** that are granted to his/her group.

Roles Permissions Model

Objects Hierarchy

Permissions are managed according to a parent/child hierarchy.

Any permission granted to an object, applies also to its descendant objects.



Permissions for objects

| Permission | Objects | Description |
|------------|-------------------------|--|
| ALL | INSTANCE, SCHEMA, TABLE | Lets you create or modify an object. Required to run DDL statements. |
| INSERT | INSTANCE, SCHEMA, TABLE | Lets you write data to a table. |
| SELECT | INSTANCE, SCHEMA, TABLE | Lets you read data from a table or view. |

Permissions for operations

| Operation | Scope | Required Permission |
|----------------------------------|----------|---|
| Schemas | | |
| ALTER SCHEMA ... OWNER TO | INSTANCE | ALL |
| SHOW SCHEMAS | | The output includes only the accessible schemas: <ul style="list-style-type: none"> - Of which the user holds SELECT/INSERT/ALL permissions to - Which contains objects (table/view) he is authorized for (SELECT/INSERT/ALL) |
| Tables | | |
| CREATE TABLE | SCHEMA | ALL |
| DROP TABLE | TABLE | ALL |
| TRUNCATE TABLE | TABLE | ALL |
| DESCRIBE TABLE | TABLE | SELECT/INSERT |
| ALTER TABLE ... ADD COLUMNS | SCHEMA | ALL |
| ALTER TABLE ... DROP COLUMN | SCHEMA | ALL |
| ALTER TABLE ... DROP PARTITION | TABLE | ALL |
| ALTER TABLE ... ADD PRIMARY KEY | SCHEMA | ALL |
| ALTER TABLE ... DROP PRIMARY KEY | SCHEMA | ALL |
| ALTER TABLE ... OWNER TO | SCHEMA | ALL |
| CREATE EXTERNAL TABLE | SCHEMA | ALL |
| SHOW TABLE COLUMNS | TABLE | SELECT/INSERT |
| SHOW TABLE PARTITIONS | TABLE | SELECT/INSERT |
| SHOW TABLES MAINT | | The output includes only accessible tables (has SELECT/INSERT/ALL) |

| | | |
|---------------------------|------------|---|
| SHOW TABLES EXTENDED | | The output includes only accessible tables (has SELECT /INSERT/ALL) |
| SHOW TABLES | | The output includes only accessible tables (has SELECT /INSERT/ALL) |
| SHOW ALL TABLES | | The output includes only accessible tables (has SELECT /INSERT/ALL) |
| SHOW FULL TABLES | | The output includes only accessible tables (has SELECT /INSERT/ALL) |
| SHOW FULL COLUMNS | | The output includes only accessible tables (has SELECT /INSERT/ALL) |
| Views | | |
| DROP VIEW | VIEW | ALL |
| SHOW VIEWS | | The output includes only accessible views (has SELECT /INSERT/ALL) |
| Special Indexes | | |
| CREATE JOIN INDEX | TABLES | ALL |
| DROP JOIN INDEX | TABLES | ALL |
| SHOW JOIN INDEXES | | Join indexes on tables that are accessible (has SELECT /INSERT) |
| CREATE RANGE INDEX | TABLE | ALL |
| DROP RANGE INDEX | TABLE | ALL |
| SHOW RANGE INDEXES | | Range indexes on tables that are accessible (has SELECT /INSERT) |
| Cubes | | |
| GENERATE CUBES | TABLES | ALL |
| SHOW CUBES | | Cubes that the tables in their select statement are accessible (has SELECT access to) |
| DROP CUBES | INSTANCE | ALL |
| Select | | |
| SELECT | TABLE/VIEW | SELECT |
| Parameters | | |
| SET | Any | |
| SET/UNSET GLOBAL | INSTANCE | ALL |
| SHOW PARAM | Any | |
| SHOW PARAM EXTENDED | INSTANCE | ALL |
| Admin | | |
| SHOW ADAPTIVE CACHE | INSTANCE | ALL |
| DROP ADAPTIVE CACHE | INSTANCE | ALL |
| DROP ADAPTIVE STORAGE | INSTANCE | ALL |
| SHOW LOCAL CACHE EXTENDED | INSTANCE | ALL |
| SHOW ACTIVE QUERIES | INSTANCE | ALL |
| CREATE SCHEDULED LOAD | TABLE | INSERT |
| SHOW SCHEDULED LOADS | INSTANCE | ALL |
| DROP SCHEDULED LOADS | TABLE | INSERT |
| DROP ALL SCHEDULED LOADS | INSTANCE | ALL |

| External Data Source | | |
|---|--|-----|
| CREATE EXTERNAL DATA SOURCE | INSTANCE | ALL |
| CREATE EXTERNAL TABLE | SCHEMA | ALL |
| Authorizations | | |
| CREATE ROLE | INSTANCE | ALL |
| DROP ROLE | INSTANCE | ALL |
| GRANT ROLE | INSTANCE | ALL |
| REVOKE ROLE | INSTANCE | ALL |
| GRANT PERMISSION .. TO ROLE | INSTANCE | ALL |
| REVOKE PERMISSION .. TO ROLE | INSTANCE | ALL |
| SHOW GRANT ON [ALL TABLE ...] VIEW ... [SCHEMA...] | INSTANCE | ALL |
| SHOW ROLES | INSTANCE | ALL |
| SHOW CURRENT ROLE | Any | |
| SHOW ROLES GRANT GROUP | INSTANCE or group members | ALL |
| SHOW ROLES GRANT GROUPS | INSTANCE | ALL |
| SHOW GRANT ROLE | INSTANCE + users that had been granted this role | ALL |
| SHOW LDAP GROUPS | INSTANCE | ALL |

Commands Syntax

Roles

CREATE ROLE

Create a role to which permissions are granted

```
CREATE ROLE [role-name]
```

DROP ROLE

Remove an existing role. Once role is drop the role will be revoked from all users to whom it was previously assigned

```
DROP ROLE [role-name]
```

GRANT ROLE

Grant role to group(s).

```
GRANT ROLE role-name [, role-name] TO GROUP group-name [, GROUP group-name]
```

REVOKE ROLE

Revoke role from group

```
REVOKE ROLE role-name [, role-name] FROM GROUP group-name [, GROUP group-name]
```

SET ROLE

Enable specific role for the current session. Only granted roles can be enabled. When setting specific roles, any other roles that not set at session level are disabled. By default all roles for user are enabled (SET ROLE ALL).

Enable a specific role:

```
SET ROLE <role-name>
```

Enable all roles:

SET ROLE ALL

Disable all roles:

SET ROLE NONE

SHOW ROLES

List all the roles in the system:

SHOW ROLES

List all roles in effect for the current user session:

SHOW CURRENT ROLES

List all roles assigned to a given group:

SHOW ROLES GRANT GROUP <group-name>

List all roles granted to all groups.

SHOW ROLES GRANT GROUPS

SHOW GRANT ROLES

List all permissions granted to a role:

SHOW GRANT ROLE <role-name>

List all permissions granted to a role on a give object

SHOW GRANT ROLE <role-name> on object <object-name>

PERMISSIONS

GRANT <PERMISSION>

Grant a permission on an object to a role.

GRANT <PERMISSION> [, <PERMISSION>] ON <OBJECT_TYPE> <object-name> TO ROLE <role-name> [, ROLE <role-name>]

<OBJECT_TYPE> - type of object INSTANCE, SCHEMA, TABLE or VIEW

<object-name> - instance name, schema name, table name or view name.

REVOKE <PERMISSION>

Revoke a permission on an object from a role

REVOKE <PERMISSION> [, <PERMISSION>] ON <OBJECT_TYPE> <object-name> FROM ROLE <role-name> [, ROLE <role-name>]

SHOW GRANT

Show granted permissions, roles and ownership on objects

SHOW GRANT ON (ALL | (SCHEMA schema-name | TABLE table-name | VIEW view-name)

ALTER

ALTER SCHEMA ... OWNER TO

Change the owner of a schema. The new owner must be an existing user (if the user is an LDAP user, it must be found in the LDAP directory)

ALTER SCHEMA <schema-name> OWNER TO <new_owner>

ALTER TABLE ... OWNER TO

Change the owner of a table. The new owner must be an existing user (if the user is an LDAP user, it must be found in the LDAP directory)

ALTER TABLE <table-name> OWNER TO <new_owner>

SHOW

SHOW LDAP GROUPS

List all LDAP groups relevant for the instance found using ldap config/filter rules

```
SHOW LDAP GROUP
```

SHOW SCHEMAS

List all schemas for which the current user has schema or table level access.

```
SHOW SCHEMAS
```

SHOW TABLES

List all tables for which the current user has table level access:

```
SHOW TABLES [EXTENDED]
```

List all columns for all tables for which the current user has table level access:

```
SHOW ALL TABLES [...]
```

Command Line Operations

The following Jethro operations are executed via the command line:

- Create/Attach/Detach/Remove/list Jethro Instances via JethroAdmin
- Start/stop services
- Load data via JethroLoader

The security for those operations is based on a Jethro Linux user (by default it is user 'jethro').

See also

[Authentication](#)

[Auditing](#)

[Show](#)