Collection-Level Access Control

By creating a role with *privileges* (page 325) that are scoped to a specific collection in a particular database, administrators can implement collection-level access control.

See *Collection-Level Access Control* (page 326) for more information.

Users

MongoDB stores user credentials in the protected `admin.system.users` (page 291). Use the *user management methods* to view and edit user credentials.

Role Assignment to Users

User administrators create the users that access the system's databases. MongoDB's *user management commands* let administrators create users and assign them roles.

MongoDB scopes a user to the database in which the user is created. MongoDB stores all user definitions in the `admin` database, no matter which database the user is scoped to. MongoDB stores users in the `admin` database's *system.users collection* (page 416). Do not access this collection directly but instead use the *user management commands*.

The first role assigned in a database should be either `userAdmin` (page 407) or `userAdminAnyDatabase` (page 412). This user can then create all other users in the system. See *Create a User Administrator* (page 384).

Protect the User and Role Collections

MongoDB stores role and user data in the protected `admin.system.roles` (page 291) and `admin.system.users` (page 291) collections, which are only accessible using the *user management methods*.

If you disable access control, **do not** modify the `admin.system.roles` (page 291) and `admin.system.users` (page 291) collections using normal `insert()` and `update()` operations.

Additional Information

See the reference section for documentation of all *built-in-roles* (page 405) and all available *privilege actions* (page 419). Also consider the reference for the form of the *resource documents* (page 417).

To create users see the *Create a User Administrator* (page 384) and *Add a User to a Database* (page 386) tutorials.

6.2.3 Collection-Level Access Control

Collection-level access control allows administrators to grant users privileges that are scoped to specific collections.

Administrators can implement collection-level access control through *user-defined roles* (page 325). By creating a role with *privileges* (page 325) that are scoped to a specific collection in a particular database, administrators can provision users with roles that grant privileges on a collection level.