Permission Syntax

<u>AUTHORIZATION PERMISSION ON SECURABLE::NAME TO PRINCIPAL</u>
• AUTHORIZATION must be GRANT, REVOKE or DENY.

PERMISSION is listed in the charts below.

Most permission statements have the format :

• ON SECURABLE::NAME is the server, server object, database, or database object and its name. (ON SECURABLE::NAME is omitted

for server-wide and database-wide permissions.)

• PRINCIPAL is the login, user, or role which receives or loses the permission. Grant permissions to roles whenever possible.

Sample grant statement: GRANT UPDATE ON OBJECT::Production.Parts TO PartsTeam

Denying a permission at any level, overrides a related grant.

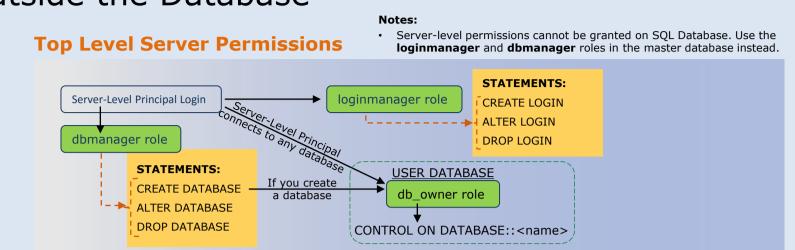
To remove a previously granted permission, use REVOKE, not DENY.

How to Read this Chart • Most of the more granular permissions are included in more than one higher level scope permission. So permissions can be inherited

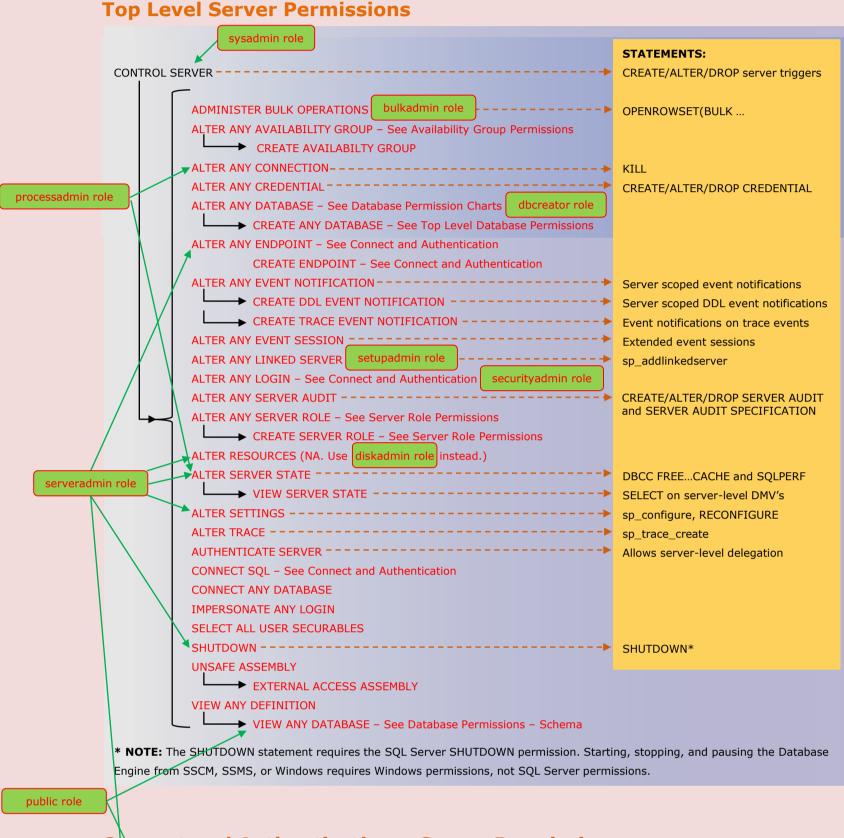
Most of the more granular permissions are included in more than one higher level scope permission. So permissions can be inherited from more than one type of higher scope.
Black, green, and purple arrows and boxes point to subordinate permissions that are included in the scope of higher a level permission.

- Brown arrows and boxes indicate some of the statements that can use the permission.
 Permissions in black apply to both SQL Server 2016 and Azure SQL Database
- Permissions in red apply only to SQL Server 2016
- Permissions in blue apply only to Azure SQL DatabaseThe newest permissions are underlined

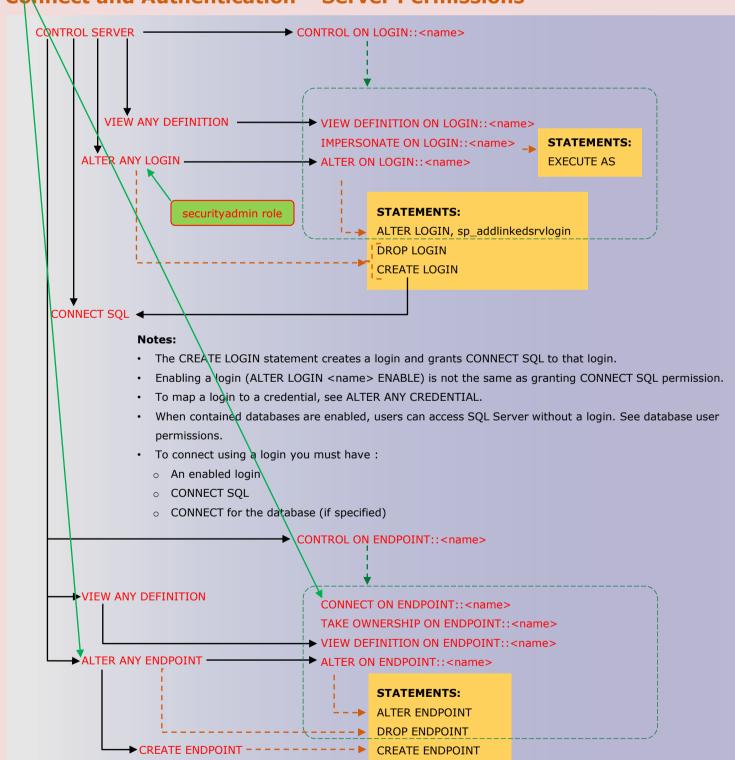
Azure SQL Database Permissions Outside the Database



Server Level Permissions for SQL Server



Connect and Authentication – Server Permissions



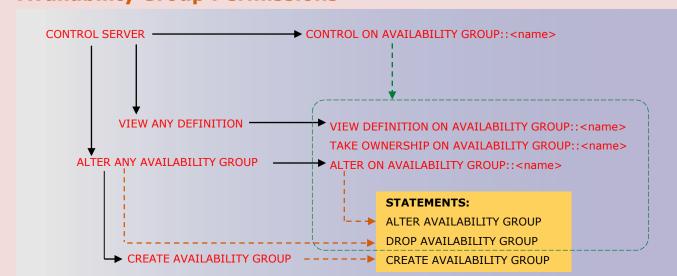
CONTROL SERVER OLE::<name> VIEW ANY DEFINITION VIEW DEFINITION ON SERVER ROLE::<name> TAKE OWNERSHIP ON SERVER ROLE::<name> ALTER ANY SERVER ROLE ALTER ON SERVER ROLE::<name> STATEMENTS: ALTER SERVER ROLE ORONGO DROP SERVER ROLE CREATE SERVER ROLE

Availability Group Permissions

NOTES: To add a member to a fixed server role, you must be a member of

that fixed server role, or be a member of the sysadmin fixed server role.

Server Role Permissions





• The CONTROL SERVER permission has all permissions on the instance of SQL Server or SQL Database.

However, it is sometimes possible to impersonate between roles and equivalent permissions.

• Permissions do not imply role memberships and role memberships do not grant permissions. (E.g. CONTROL SERVER does not imply

membership in the sysadmin fixed server role. Membership in the db_owner role does not grant the CONTROL DATABASE permission.)

• The CONTROL DATABASE permission has all permissions on the database.

