

**PostgreSQL Access Control Policy**

**Purpose:**

This document outlines the PostgreSQL access control policy for service accounts and developers across all environments (Development, Test, Staging, and Production). The goal is to enforce the Principle of Least Privilege (PoLP), ensuring each role is granted only the permissions necessary for its function.

To maintain strong access controls:

1. We avoid using PostgreSQL global attributes such as rolname, rolsuper, rolcanlogin, rolinherit, rolcreaterole, rolcreatedb, and rolreplication except where absolutely necessary, as they confer instance-level privileges.

2. Instead, specific access requirements are implemented by creating roles (groups) with scoped permissions, then assigning users and service accounts to those roles to enforce consistency and security.[[1]](#footnote-1)

**Role-Based Access by Environment**

**1. Development**

Developers

- Assigned as owners of their respective databases. This means their role has full privileges to manage the database and its objects.  
 (Note: In PostgreSQL, a database 'owner' is a role explicitly assigned as the owner of the database. Ownership provides full DDL and DCL rights, including the ability to DROP the database.)

- Enables full control over schema changes, function development, and data setup for application testing.

Service Accounts

- Granted: CONNECT to the database; USAGE, CREATE on relevant schemas; SELECT, INSERT, UPDATE, DELETE; EXECUTE on functions.  
- DEFAULT PRIVILEGES are applied to automatically grant these permissions on newly created tables, sequences, and functions.

**2. Test & Staging**

Developers

- Not database owners.  
- Granted: CONNECT to the database; USAGE, CREATE on designated schemas; SELECT, INSERT, UPDATE, DELETE; EXECUTE on all relevant functions.  
- DEFAULT PRIVILEGES applied to ensure consistent access to future objects.  
- Past configurations may have granted ownership; this will be phased out in favor of controlled privileges.

Service Accounts

- Same access as in Development.  
- Includes DEFAULT PRIVILEGES for consistency with deployments.

**3. Production**

Developers

- Granted read-only access: CONNECT; USAGE on specific schemas; SELECT and EXECUTE on necessary views and functions.  
- No ownership or object-level modification privileges.

Service Accounts

- Granted: CONNECT, USAGE, CREATE, SELECT, INSERT, UPDATE, DELETE, EXECUTE.  
- Access is limited to application-owned schemas and objects.  
- DEFAULT PRIVILEGES are configured for deployment readiness.

**Governance & Review**

The DBA team collaborates with engineering and security to govern access control.  
All permissions are provisioned based on operational requirements and are subject to routine review to maintain compliance and minimize risk.





1. GRANT parent\_role TO child\_role; [↑](#footnote-ref-1)