Row-Level Security (RLS) in Azure Cosmos DB for PostgreSQL

# What is Row-Level Security (RLS)?

Row-Level Security (RLS) is a PostgreSQL feature that restricts which rows users can view or modify. It allows fine-grained access control directly at the row level in the database.

# Why is RLS Important?

• It enforces data isolation in multi-tenant databases.  
• It prevents users from seeing or modifying data that doesn’t belong to them.  
• It reduces the risk of security breaches due to application bugs.  
• Security policies live inside the database, reducing complexity in application code.

# Demo Script: Implementing RLS

Step 1: Create a sample table with tenant\_id

CREATE TABLE orders (  
 order\_id BIGINT,  
 tenant\_id INT,  
 item TEXT,  
 amount DECIMAL,  
 PRIMARY KEY (order\_id, tenant\_id)  
);

Step 2: Enable Row-Level Security

ALTER TABLE orders ENABLE ROW LEVEL SECURITY;

Step 3: Create a policy to restrict access based on tenant\_id

CREATE POLICY customer\_policy  
 ON orders  
 USING (tenant\_id = current\_setting('app.tenant\_id')::int);

Step 4: Enforce the policy

ALTER TABLE orders FORCE ROW LEVEL SECURITY;

Step 5: Simulate tenant access by setting the tenant ID

SET app.tenant\_id = '101';

# Analogy

Imagine a spreadsheet shared by different teams. Each team sees only the rows with their name. RLS acts like a filter that shows only relevant rows based on the user's context.

# Summary

Row-Level Security allows defining security at the data level. With tenant-specific policies, you ensure each user or tenant sees only their own data, making your application secure and scalable.