Resolution of Permission Denied Error on PostgreSQL View

The following steps outline how the permission denied error on the view "vw_pcpl_digipro_sud_loan_status" was resolved for the user "magicxpiuser". These steps involve checking and granting the necessary privileges and ensuring schema-level access.

1. Checking Table Privileges

The following query was used to check if the user 'magicxpiuser' had the necessary SELECT privilege on the view 'vw_pcpl_digipro_sud_loan_status':

SELECT has_table_privilege('magicxpiuser', 'vw_pcpl_digipro_sud_loan_status', 'SELECT');

This query checks if the user has the specific 'SELECT' privilege on the view. If it returned 'false', it confirmed that the user did not have the necessary privileges.

2. Granting the SELECT Privilege

To grant the necessary SELECT privilege on the view, the following command was executed:

GRANT SELECT ON vw_pcpl_digipro_sud_loan_status TO magicxpiuser;

This grants the user 'magicxpiuser' permission to execute SELECT queries on the view.

3. Changing the View Owner

Altering the ownership of the view was also considered to make 'magicxpiuser' the owner. The command for this is:

ALTER VIEW vw_pcpl_digipro_sud_loan_status OWNER TO magicxpiuser;

This would grant 'magicxpiuser' full privileges on the view, but this step required confirmation with the infrastructure team.

4. Checking Schema-Level Permissions

To ensure that 'magicxpiuser' had access to the schema where the view resides, the following command was used:

GRANT USAGE ON SCHEMA your_schema TO magicxpiuser;

This command grants the 'USAGE' privilege on the schema, which is necessary for accessing any objects (like tables or views) within that schema.

Summary

The following steps were taken to resolve the permission denied error for 'magicxpiuser':

- Verified whether the user had SELECT privileges using 'has_table_privilege'.
- Granted SELECT access to the user on the view using the GRANT command.
- Considered changing the view's owner, but this step required further confirmation.
- Ensured the user had schema-level access by granting USAGE on the schema.