



Universidad Nacional
Autónoma de México
Facultad de Ingeniería



Bases de Datos

Semestre: 2026-2

Tarea 3

Profesor: Arreola

Alumno: Parra Bello Carlos Enrique

Número de cuenta: 321135344

Grupo: 01

Creación de usuario con límite de conexiones y vigencia en PostgreSQL

En PostgreSQL, la gestión de usuarios se realiza mediante **roles**, ya que en este sistema no existe una separación estricta entre usuario y rol. Un rol puede autenticarse (LOGIN) y tener restricciones como límite de conexiones y fecha de expiración.

Para crear un usuario con contraseña, límite de conexiones simultáneas y vigencia de un mes, se utilizan los siguientes parámetros:

- LOGIN → permite autenticación.
- PASSWORD → define contraseña.
- CONNECTION LIMIT → limita conexiones concurrentes.
- VALID UNTIL → establece fecha de expiración.

Ejemplo:

```
CREATE ROLE usuario_est
WITH LOGIN
PASSWORD 'contraseña123'
CONNECTION LIMIT 2
VALID UNTIL CURRENT_DATE + INTERVAL '30 days';
```

En este caso:

- El usuario podrá tener máximo 2 conexiones simultáneas.
- La cuenta expirará automáticamente en 30 días.
- La contraseña es obligatoria para autenticación.

Creación de un rol y asignación de permisos

En PostgreSQL, los roles también pueden utilizarse para agrupar privilegios y facilitar la administración.

Para crear un rol que tenga permisos de lectura, actualización y borrado sobre la tabla estudiante, se utilizan los siguientes comandos:

```
CREATE ROLE rol_estudiante;
```

```
GRANT SELECT, UPDATE, DELETE
ON TABLE estudiante
TO rol_estudiante;
```

Posteriormente, se asigna el rol al usuario creado anteriormente:

```
GRANT rol_estudiante TO usuario_est;
```

De esta manera, el usuario hereda automáticamente los privilegios definidos en el rol.

Importancia en la administración de seguridad

El uso de roles y restricciones en PostgreSQL permite:

- Aplicar el principio de **mínimo privilegio**.
- Controlar accesos simultáneos.
- Establecer políticas de expiración de cuentas.
- Facilitar la administración de permisos en entornos multiusuario.

Este mecanismo mejora la seguridad y el control de acceso dentro del sistema de base de datos.

Bibliografía

[1] PostgreSQL Global Development Group, “PostgreSQL 16 Documentation,” PostgreSQL Documentation, 2023. [Online]. Available: <https://www.postgresql.org/docs/>

[2] A. Silberschatz, H. F. Korth y S. Sudarshan, Database System Concepts, 7th ed. New York, NY, USA: McGraw-Hill, 2019.

[3] T. Connolly y C. Begg, Database Systems: A Practical Approach to Design, Implementation, and Management, 6th ed. Harlow, England: Pearson, 2015.