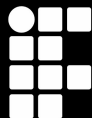


Tecnologias de Banco de Dados

Msc. Eliezio Soares
eliezio.soares@ifrn.edu.br



INSTITUTO FEDERAL
Rio Grande do Norte

Campus
Carrais Novos

Roles

Roles são usuários ou grupos para os quais o PostgreSQL controla as permissões de acesso.

- Roles assumem esse caráter híbrido de grupo ou individual.
- Não possuem relação com usuário do sistema hospedeiro, é vinculado ao cluster de banco de dados.
 - As roles não pertencem à bases de dados;
 - Uma role é de escopo do servidor e pode ou não ter permissão de acesso a 0:N bases de dados;
- Ao inicializar uma instância (initdb), uma role é criada com o mesmo nome do usuário do SO hospedeiro (postgres);

Criação de Roles

- Uma ROLE deve ser criada sempre que se desejar:
 - Criar um login para acesso de um usuário do SGBD;
 - Criar um login para identificar uma aplicação cliente específica;
- Comando:
 - ***CREATE ROLE***
- Opções:
 - LOGIN
 - PASSWORD
 - VALID UNTIL
 - CREATEROLE;
 - SUPERUSER;
 - CREATEDB;
 - REPLICATION;

Criação de Roles

- Criação de ROLE para acesso (usuário):

```
postgres=# CREATE ROLE fulano LOGIN PASSWORD 'abcd';
```

- LOGIN
 - Informa que a role pode conectar-se ao servidor;
- PASSWORD
 - Especifica um parâmetro de senha;

Criação de Roles

- Criação de ROLE para acesso (usuário):

```
postgres=# CREATE ROLE fulano LOGIN PASSWORD 'abcd' VALID UNTIL '2019-01-01';
```

- VALID UNTIL
 - Define uma data de expiração para a senha do usuário;

```
postgres=# CREATE ROLE fulano LOGIN PASSWORD 'abcd' VALID UNTIL '2019-01-01'  
CREATE ROLE;
```

- CREATE ROLE
 - Define que esse role tem o privilégio de criar outras roles;

Criação de Roles

- Criação de ROLE com privilégios de super usuário:

```
postgres=# CREATE ROLE fulano LOGIN PASSWORD 'abcd' SUPERUSER;
```

- Criação de ROLE com privilégios para criação de novas bases de dados;

```
postgres=# CREATE ROLE fulano LOGIN PASSWORD 'abcd' CREATEDB;
```

- Criação de ROLE com privilégios para operações de replicação;

```
postgres=# CREATE ROLE fulano LOGIN PASSWORD 'abcd' REPLICATION;
```

Criação de Roles

- Criação de ROLE com comportamento de grupo:

```
postgres=# CREATE ROLE grupo1;
```

- Para vincular usuários a essa role se atribui uma role à outra como um privilégio:

```
postgres=# GRANT grupo1 TO fulano;
```

```
postgres=# GRANT grupo1 TO siclano;
```

Exclusão de Roles

- Exclusão de uma ROLE:

```
postgres=# DROP ROLE grupo1;
```

- Caso uma role seja proprietária ou possua privilégios sobre objetos ou bases de dados:

```
postgres=# DROP ROLE roleexemplo3;  
ERROR:  role "roleexemplo3" cannot be dropped because some objects depend on it  
DETALHE:  owner of database dbrole2
```

- Para remover todos os objetos de uma role, é possível utilizar o comando:

```
postgres=# DROP OWNED BY roleexemplo2;
```


Alteração de Roles

- Para alterar a propriedade dos objetos de uma role atribuindo-os a uma outra role:

```
postgres=# REASSIGN OWNED BY roleexemplo1 TO roleexemplo2;
```

Privilégios

Os comandos **GRANT** e **REVOKE** são aplicados para garantir e revogar permissões em objetos e bases de dados.

- Ao criar um novo banco de dados:
 - Se atribui um proprietário com privilégios para fazer qualquer alteração nesse objeto ou base de dados;
 - Caso não se explicita nenhum proprietário, o objeto será atribuído ao usuário da sessão.
 - Para conceder privilégios a outros usuários é necessário utilizar o comando GRANT.

Privilégios

- Comando GRANT:

- GRANT <permissão> ON <tipo objeto> <objeto> TO <role>;

```
postgres=# GRANT CREATE ON DATABASE exemploaula TO roleexemplo2;
```

- Criação de objetos no schema;

```
postgres=# GRANT CREATE ON SCHEMA vendas TO vendedor;
```

- Uso de objetos no schema;

```
postgres=# GRANT USAGE ON SCHEMA vendas TO vendedor;
```

- Conexão na base de dados;

```
postgres=# GRANT CONNECT ON DATABASE exemploaula TO aluno;
```

Privilégios

O que está sendo concedido nesses exemplos???

1

```
sis_contabil=> GRANT CONNECT ON DATABASE sis_contabil TO contabil;  
sis_contabil=> GRANT USAGE ON SCHEMA geral TO contabil;  
sis_contabil=> GRANT SELECT, INSERT, UPDATE, DELETE  
                ON ALL TABLES IN SCHEMA geral  
                TO contabil;
```

2

```
sis_contabil=> GRANT SELECT ON geral.balanco TO contabilidade;  
sis_contabil=> GRANT EXECUTE ON FUNCTION geral.lancamento()  
                TO contabilidade;
```

Repasse de Privilégios

Se uma role possui um privilégio, ela pode repassar a outras roles.

- Para garantir esse poder a uma role é necessário especificar com a declaração:
 - **WITH GRANT OPTION;**

```
postgres=# GRANT SELECT ON DATABASE exemploaula TO roleexemplo2 WITH GRANT  
OPTION;
```

Privilégios x Objetos

Base de dados:

- CONNECT: permite a role conectar-se à base;
- CREATE: permite a role criar schemas na base;
- TEMP or TEMPORARY: permite a role criar tabelas temporárias na base.

Schemas:

- CREATE: permite criar objetos no schema;
- USAGE: permite acessar objetos do schema, mas ainda depende de permissão no objeto.

Privilégios x Objetos

Tabelas e Views:

- SELECT
- INSERT
- UPDATE
- DELETE
- TRUNCATE
- TRIGGER
- REFERENCES

* Para utilizar WHERE em updates ou deletes, é preciso ter permissão também para select.

Privilégios x Objetos

Tabelas:

- É possível definir permissão apenas para coluna(s) específica(s).

```
postgres=# GRANT SELECT (matricula) ON alunos TO roleexemplo2;
```

```
postgres=# GRANT INSERT (matricula) ON alunos TO roleexemplo2;
```

```
postgres=# GRANT UPDATE (matricula), SELECT (matricula) , INSERT (matricula) ON alunos TO  
roleexemplo2;
```


Privilégios x Objetos

Funções:

- EXECUTE

Cláusula ALL

Para atribuir permissão a todos os objetos de um tipo específico em um schema, utiliza-se a cláusula **ALL <...> IN SCHEMA**.

```
postgres=# GRANT USAGE ON ALL TABLES IN SCHEMA public TO roleexemplo2;
```

Revoke Privilégios

O **REVOKE ... FROM** é utilizado para remover um privilégio específico em um objeto.

```
postgres=# REVOKE USAGE ON SCHEMA public FROM roleexemplo2;
```

```
postgres=# REVOKE GRANT OPTION FOR SELECT, INSERT ON public.aluno FROM roleexemplo2;
```

Revoke “Grupos”

Para remover de um usuário as permissões de uma role, ou “remover um usuário de um grupo” também se utiliza o REVOKE.

```
postgres=# GRANT contabilidade TO jsilva;
```

```
postgres=# GRANT contabilidade TO moliveira;
```

```
postgres=# REVOKE contabilidade FROM moliveira;
```

Consulta de Privilégios

Para consultar os privilégios em cada objeto no psql se utiliza o comando `\dp`.

- `\d+p`: Lista os objetos (tabelas, visões e sequências) + permissões.

```
exemploaula=# \dp
```

Esquema	Nome	Tipo	Privilégios de acesso	Privilégios de coluna	Políticas
public	exemplo	tabela	postgres=arwdDxt/postgres+ roleexemplo3=rDx/postgres		
public	exemplo_id_seq	sequência			

(2 registros)

→ Quem concedeu o privilégio

→ Privilégios - Um por letra.

→ Quem tem o privilégio

Consulta de Privilégios

- O * representa a opção WITH GRANT OPTION;
- Letras e permissões:

a	INSERT (append)
r	SELECT (read)
w	UPDATE (write)
d	DELETE
D	TRUNCATE
x	REFERENCES
t	TRIGGER
X	EXECUTE
U	USAGE
C	CREATE
c	CONNECT
T	TEMPORARY

exemploaula=# \dp

Esquema	Nome	Tipo	Privilégios de acesso	Privilégios de acesso	Privilégios de coluna	Políticas
public	exemplo	tabela	postgres=arwdDxt/postgres	+	id:	+
			roleexemplo3=rDx/postgres	+	roleexemplo2=a/postgres	
public	exemplo_id_seq	sequência	roleexemplo2=a*r*w*d*/postgres			
(2 registros)						

→ Campo com permissão^T

Consulta de Privilégios

- Para as bases de dados \l
 - **T** - Tempt (tabelas temporárias)
 - **c** - Connect (conectar-se)
 - **C** - Create (criar)

```
exemploaula=# \l
```

Lista dos bancos de dados					
Nome	Dono	Codificação	Collate	Ctype	Privilégios de acesso
bdexemplotemplate0	postgres	UTF8	pt_BR.UTF-8	pt_BR.UTF-8	
bdexemplotemplate1	postgres	UTF8	pt_BR.UTF-8	pt_BR.UTF-8	
bdexemplotemplatedefault	postgres	UTF8	pt_BR.UTF-8	pt_BR.UTF-8	
dbrole2	roleexemplo2	UTF8	pt_BR.UTF-8	pt_BR.UTF-8	
exemploaula	postgres	UTF8	pt_BR.UTF-8	pt_BR.UTF-8	=Tc/postgres + postgres=Ctc/postgres + roleexemplo=C/postgres + roleexemplo3=c/postgres
postgres	postgres	UTF8	pt_BR.UTF-8	pt_BR.UTF-8	
template0	postgres	UTF8	pt_BR.UTF-8	pt_BR.UTF-8	=c/postgres + postgres=Ctc/postgres
template1	postgres	UTF8	pt_BR.UTF-8	pt_BR.UTF-8	=c/postgres + postgres=Ctc/postgres

(8 registros)

Consulta de Privilégios

- Para os schemas `\dn+`
 - **U** - Usage
 - **C** - Create (criar)

```
curso=# \dn+
```

List of schemas			
Name	Owner	Access privileges	Description
avaliacao	postgres		
extra	postgres	aluno=U/postgres	+
		postgres=UC/postgres	
public	postgres	postgres=UC/postgres+	standard public schema
		=UC/postgres	

Bibliografia Utilizada

CAIUT, Fábio. Administração de banco de dados. 1ª Edição. Rio de Janeiro. RNP/ESR, 2015.

PostgreSQL 10.5

Documentation. Disponível em:
<<https://www.postgresql.org/docs/10/static/index.html>>. Acesso em 09 Ago. 2018.

