睿创神脑
RUICHUANGSHENNAO

PolarDB
PostgreSQL

PostgreSQL
用户与角色管理

睿创神脑
RUICHUANGSHENNAO

PolarDB PostgreSQL

Objectives

- 创建用户
- 创建角色
- 权限介绍
- 给用户授权
- 给角色授权

睿创神脑
RUICHUANGSHENNAO

用户角色（概述）

PolarDB PostgreSQL

- 用户与角色
  - 数据库用户用来访问、管理数据库中的对象（表、索引...）
  - 数据库角色用来管理数据库访问权限，简化权限的管理
  - 用户和角色在整个数据库集簇中是全局性的，不是针对某个单一数据库，只要有足够的权限，用户可以访问所有数据库的对象。
  - 数据库用户可以分为两类：
    超级用户 -- postgres
    普通用户 -- 根据需要创建

睿创神脑
RUICHUANGSHENNAO

用户与角色概述

PolarDB PostgreSQL

- 用户与角色的区别
  - user：拥有login登陆数据库权限的role
  - role: 可以拥有数据库对象，如表、索引，也可以把这些对象上的权限赋予其它角色，以控制哪些用户对哪些对象拥有哪些权限
  - group：不拥有replication/noreplication、connection limit属性的role
  - 在PG 8.1之前，user与group是不同类型的实体，现在可以被看作是role，任意一个role均可自由的在user与group间转换

睿创神脑
RUICHUANGSHENNAO

创建用户

PolarDB PostgreSQL

- 创建用户
  - 方式1：在系统命令行使用createuser命令

    createuser username

    createuser -U postgres -p 7788 user1 （--for windows）
  - 方式2：在psql命令行使用create user(role)指令

    CREATE user[ROLE] rolename;

睿创神脑
RUICHUANGSHENNAO

创建用户

PolarDB PostgreSQL

- CREATE USER name [[WITH] option [...]]

这里的option可以是：

```
    SUPERUSER | NOSUPERUSER
  | CREATEDB | NOCREATEDB
  | CREATEROLE | NOCREATEROLE
  | CREATEUSER | NOCREATEUSER
  | INHERIT | NOINHERIT    #角色是其他角色的成员，这些子句决定新角色是否从那些角色中“继承”特权
  | LOGIN | NOLOGIN
  | REPLICATION | NOREPLICATION
  | BYPASSRLS | NOBYPASSRLS  #决定是否一个角色可以绕过每一条行级安全性（RLS）策略。
  | CONNECTION LIMIT connlimit
  | [ ENCRYPTED | UNENCRYPTED ] PASSWORD 'password'
  | VALID UNTIL 'timestamp'
  | IN ROLE role_name [, ...]
  | IN GROUP role_name [, ...]
  | ROLE role_name [, ...]
  | ADMIN role_name [, ...]  #ADMIN子句与ROLE相似，让它们能够把这个角色中的成员关系授予给其他人。
  | USER role_name [, ...]
  | SYSID uid
```

睿创神脑
RUICHUANGSHENNAO

创建用户

PolarDB PostgreSQL

- 创建用户示例：

CREATE USER u1 SUPERUSER PASSWORD 'u1'; --创建一个具有超级用户权限的用户

CREATE USER u2 CREATEDB PASSWORD 'u2'; --创建一个具有建库权限的用户

CREATE USER u3 LOGIN; --创建一个具有登录权限的用户，默认值

CREATE ROLE u4 encrypted PASSWORD '123456' VALID UNTIL '2018-08-16'; --创建一个带有加密密码且具有有效时间的用户

注： 属性LOGIN、SUPERUSER和CREATEROLE被视为特殊权限，它们不会像其它数据库对象的普通权限那样被继承。

睿创神脑
RUICHUANGSHENNAO

创建角色

PolarDB PostgreSQL

- CREATE ROLE name [[WITH] option [...]]

这里的option可以是：

```
    SUPERUSER | NOSUPERUSER
  | CREATEDB | NOCREATEDB
  | CREATEROLE | NOCREATEROLE
  | CREATEUSER | NOCREATEUSER
  | INHERIT | NOINHERIT
  | LOGIN | NOLOGIN
  | REPLICATION | NOREPLICATION
  | CONNECTION LIMIT connlimit
  | [ ENCRYPTED | UNENCRYPTED ] PASSWORD 'password'
  | VALID UNTIL 'timestamp'
  | IN ROLE role_name [, ...]
  | IN GROUP role_name [, ...]
  | ROLE role_name [, ...]
  | ADMIN role_name [, ...]
  | USER role_name [, ...]
  | SYSID uid
```

睿创神脑
RUICHUANGSHENNAO

创建角色

PolarDB PostgreSQL

- 创建角色示例

```
CREATE ROLE manager;  --创建一个角色

CREATE ROLE  dev createdb;     --创建一个具有建库权限的角色

CREATE ROLE  r1  LOGIN;   --创建一个具有登录权限的角色，类似于用户

CREATE ROLE  u4 encrypted  PASSWORD   '123456'   VALID UNTIL
  '2018-08-16';   --创建一个带有加密密码且具有有效时间的角色
```

睿创神脑
RUICHUANGSHENNAO

查看用户与角色

PolarDB PostgreSQL

- 查看用户与角色信息

◆ \du 指令显示用户和角色属性

```
postgres=# \du
                                   角色列表
 角色名称 |                         属性                         | 成员属于
----------+------------------------------------------------------+----------
 dev      | 建立 DB, 无法登录                                    | {}
 manager  | 无法登录                                             | {dev}
 postgres | 超级用户, 建立角色, 建立 DB, 复制, 绕过RLS           | {}
 u2       | 建立 DB                                              | {}
 u3       |                                                      | {u2}
 u4       | 无法登录                                            +| {}
          | 密码有效直至2018-08-16 00:00:00+08                   |
```

睿创神脑
RUICHUANGSHENNAO

查看用户与角色

PolarDB PostgreSQL

- 通过数据库字典表来查看用户信息

```
postgres=# \d pg_user
                    视图 "pg_catalog.pg_user"
     栏位      |           类型           | 校对规则 | 可空的 | 预设
---------------+--------------------------+----------+--------+------
 usename       | name                     |          |        |
 usesysid      | oid                      |          |        |
 usecreatedb   | boolean                  |          |        |
 usesuper      | boolean                  |          |        |
 userepl       | boolean                  |          |        |
 usebypassrls  | boolean                  |          |        |
 passwd        | text                     |          |        |
 valuntil      | timestamp with time zone |          |        |
 useconfig     | text[]                   | C        |        ||
```

睿创神脑
RUICHUANGSHENNAO

查看用户与角色

PolarDB PostgreSQL

- 通过数据库字典表来查看

```
postgres=# select * from pg_user where usename='u2';
 usename | usesysid | usecreatedb | usesuper | userepl | usebypassrls |  passwd  | valuntil | useconfig
---------+----------+-------------+----------+---------+--------------+----------+----------+-----------
 u2      |    65559 | t           | f        | f       | f            | ******** |          |
```

睿创神脑
RUICHUANGSHENNAO

查看用户与角色

PolarDB PostgreSQL

- 通过数据库字典表来查看角色信息

```
postgres=# \d pg_roles

                    视图 "pg_catalog.pg_roles"
     栏位      |           类型           | 校对规则 | 可空的 | 预设
---------------+--------------------------+----------+--------+------
 rolname       | name                     |          |        |
 rolsuper      | boolean                  |          |        |
 rolinherit    | boolean                  |          |        |
 rolcreaterole | boolean                  |          |        |
 rolcreatedb   | boolean                  |          |        |
 rolcanlogin   | boolean                  |          |        |
 rolreplication| boolean                  |          |        |
 rolconnlimit  | integer                  |          |        |
 rolpassword   | text                     |          |        |
 rolvaliduntil | timestamp with time zone |          |        |
 rolbypassrls  | boolean                  |          |        |
 rolconfig     | text[]                   | C        |        |
 oid           | oid                      |          |        |
```

睿创神脑
RUICHUANGSHENNAO

查看用户与角色

PolarDB PostgreSQL

- 通过数据库字典表来查看角色信息

```
postgres=# select
rolname ,rolsuper ,rolinherit ,rolcreatedb ,rolcanlogin  from pg_roles
where rolname='dev';

 rolname | rolsuper | rolinherit | rolcreatedb | rolcanlogin
---------+----------+------------+-------------+-------------
 dev     | f        | t          | t           | f
(1 行记录)
```

睿创神脑
RUICHUANGSHENNAO

修改用户

PolarDB PostgreSQL

- 修改用户属性

postgres=# \h alter user
命令：　　ALTER USER
描述：　　更改数据库角色
语法：
ALTER USER role_specification [WITH] 选项 [...]

选项可以是

SUPERUSER | NOSUPERUSER
| CREATEDB | NOCREATEDB
| CREATEROLE | NOCREATEROLE
| INHERIT | NOINHERIT
| LOGIN | NOLOGIN
| REPLICATION | NOREPLICATION
| BYPASSRLS | NOBYPASSRLS
| CONNECTION LIMIT 连接限制
| [ENCRYPTED] PASSWORD '口令' | PASSWORD NULL
| VALID UNTIL '时间戳'

ALTER USER 名称 RENAME TO 新的名称

……

睿创神脑
RUICHUANGSHENNAO

修改用户

PolarDB PostgreSQL

- 修改用户示例

ALTER USER u2 RENAME TO u22; --修改用户的名字

ALTER USER u22 PASSWORD ‘u22’; --修改用户的密码

ALTER USER u22 CREATEROLE; --修改用户的权限

ALTER USER u22 IN DATABASE testdb RESET ALL; --修改数据库testdb中的参数重设为默认值

只有createrole权限的用户可以修改其它用户的名字。

睿创神脑
RUICHUANGSHENNAO

修改角色

PolarDB PostgreSQL

- 修改角色示例

```
ALTER ROLE dev RENAME TO dev1;  --修改角色的名字

ALTER ROLE dev1 SUPERUSER;   --修改角色的权限

ALTER ROLE dev1 LOGIN;    --修改角色的权限

……
```

睿创神脑
RUICHUANGSHENNAO

删除用户

PolarDB PostgreSQL

- 删除用户方法：

  ➢ 方式1:在系统命令行使用dropuser命令删除用户
  dropuser –U postgres –p 7788 username;

  ➢ 方式2:在psql命令行使用drop删除
  drop role rolename; 或 drop user username;
  DROP ROLE IF EXISTS role_name;

  注意事项：
  1、只用超级用户能够删除超级用户
  2、只有具有createrole权限的用户能删除非超级用户
  3、删除用户前，需要先删除依赖该用户的对象、权限等信息
  4、任何属于该组角色的对象都必须先被删除或者将对象的所有者赋予其它角色，任何赋予该组角色的权限也都必须被撤消。
  5、删除组role只会删除组的role本身，组的成员并不会被删除

删除用户与角色

- 删除用户与角色示例：

```
DROP USER u22;

DROP USER IF EXISTS u3;

DROP ROLE IF EXISTS u4;
```

注意：删除用户和角色所用命令可以通用

启用角色赋予的权限

- 启用某个角色权限：

分配给用户的角色在用户登录时其权限不会自动生效，需要启用该角色。

命令：

Set role manager;

注意：此时登陆的用户名变成manager。在此期间创建的对象也是属于manager。

睿创神脑
RUICHUANGSHENNAO

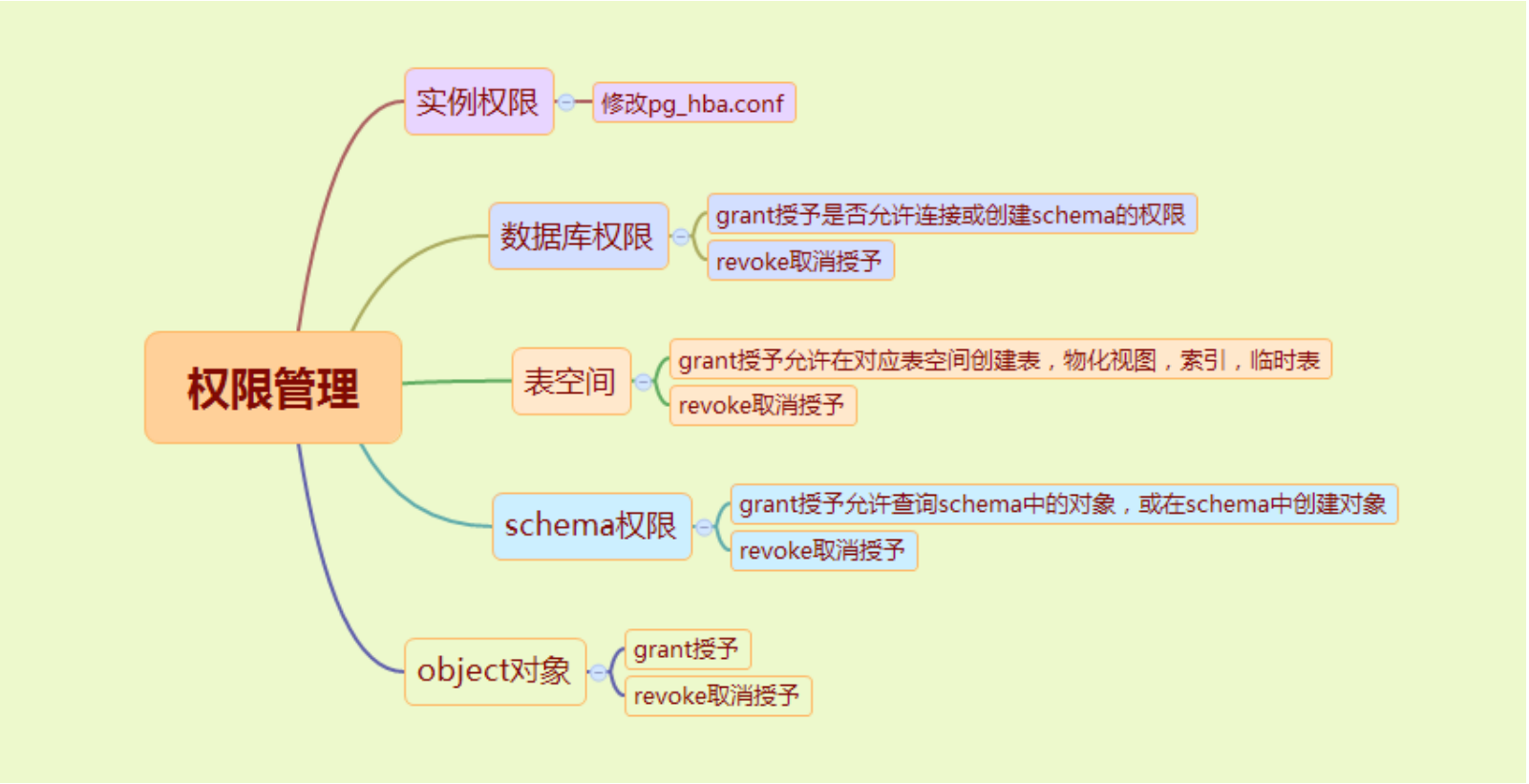
权限管理

PolarDB PostgreSQL

- 每个数据库对象都有一个所有者，默认情况下，所有者拥有该对象的所有权限
- 在数据库中所有的权限都和角色挂钩
- 对超级用户postgres不做权限检查，其它用户走ACL(Access ControL List)
- 对于数据库对象，开始只有所有者和超级用户可以做任何操作，其它走ACL

权限管理

- 权限管理结构层次图



睿创神脑
RUICHUANGSHENNAO

PolarDB PostgreSQL

总结

- 创建用户
- 创建角色
- 权限介绍
- 给用户授权
- 给角色授权

睿创神脑
RUICHUANGSHENNAO

PolarDB
PostgreSQL

练习

- 1、创建3个角色a、b、c, 每个角色分别创建1张表. 将角色b赋予给角色a, 观察a和c是否可以查询b创建的表的数据? 观察b和c是否可以查询a创建的表的数据?

睿创神脑
RUICHUANGSHENNAO

PolarDB PostgreSQL