**Setting up a User Account in Postgres**

The superuser in a Postgres database can create new users and assign them passwords on the server.

They can also grant each user very specific privileges to carry out actions on specific databases, or even specific tables within each database. Note we have to grant privileges to each database as we go, this I a high level of granularity.

A user could be allowed on SELECT operations (and not DELETE, INSERT, CREATE or UPDATE operations) on specific databases and tables. This allows the superuser (ie the database manager) highly granular control of the access to the database, which helps avoid damage or misuse.

For some of the work we will want to do with our databases (bank, dvdrental and chinook), we will want to create a user account that is not the superuser to log in with and work with data.

**Instructions**

We will create a new user, and give them SELECT access only to each of the three databases.

1. Login in as superuser (ie username postgres) using the command line interface, the postgres command window
2. Run the command:

create user bob with encrypted password 'pwd1';

*create your own new user and password here, it doesn’t have to be bob, but write down that password*

1. Connect to the bank database

\connect bank

1. Once connected to bank, grant bob the ability to run SELECT on tables in bank

GRANT SELECT ON ALL TABLES IN SCHEMA public TO bob;

*Note that SCHEMA public means all publicly visible tables in bank*

1. Now connect to the dvdrental database and grant select privileges to bob

\connect dvdrental

GRANT SELECT ON ALL TABLES IN SCHEMA public TO bob;

1. Lastly connect to chinook and do the same thing

\connect dvdrenta

GRANT SELECT ON ALL TABLES IN SCHEMA public TO bob;

Note: the GRANT SELECT ON ALL TABLES command grants access to all currently existing tables, if

we create new TABLEs or VIEWs (a summary or report that acts like a table, more or less), and

then we will need to run GRANT SELECT again for the user. There is a way to alter the default

behavior of postgress to change this.

**Closing comments**

In pgAdmin4, thcreere is a way to create groups of users called User Roles. Each User Role has a set of privileges for each database. It is then possible to create a user with specific User roles, instead of having to assign privileges one by one to each user.

For example, you might have a “Teller” role in a bank, which has specific rights to access and update specific tables only. All the employees in the “Teller” position would be placed in the “Teller” User Role in the database and assigned the same set of privileges. There might be other roles, like “Head Teller” or “Loan Officer” that had a different set of privileges, perhaps to different databases on the server.

Setting up User Roles in pgAdmin4 probably isn’t that difficult, but we don’t need to learn it right now. It’s a matter of reading the manual\s and experimenting with it really.