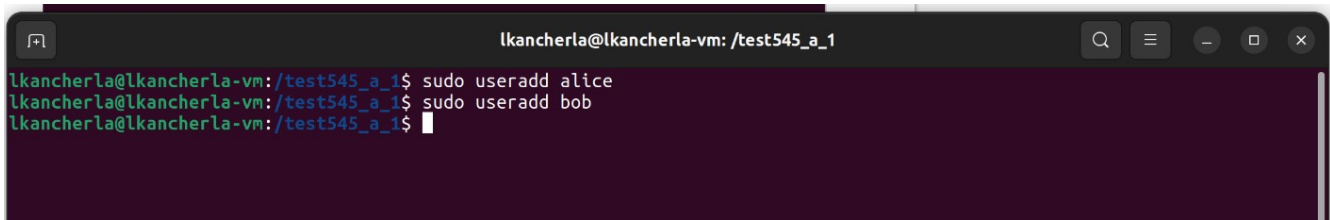# Software Security Assignment 01

Lushaank Kancherla
1224601664

Question 1: Please provide the command lines of Linux terminal and the screenshots of command line responses to show you're your commands executed successfully

a. Create two users alice and bob.

sudo useradd alice
sudo useradd bob

b. Switch to user alice and create a file named "test1" under alice's home folder. Revoke all the privileges except the owners'. Write some text into the file.

sudo touch test1.txt
sudo chown alice test1.txt
su alice
chmod 600 test1.txt
echo "………." > test1.txt

c. Grant bob read access to the file "test1".

setfacl -m u:bob:r test1.txt

```
$ setfacl -m u:bob:r test1.txt
$ 
```

d. Switch to bob. Use vim or emacs or cat to read the file "test1".

su bob
cat test1.txt

```
lkancherla@lkancherla-vm:/test545_a_1$ su bob
Password:
$ cat test1.txt
This is alice's text. Others shouldn't read this
$ 
```

e. Modify the content of "test1" and save. Show the results.

su bob
nano test1.txt

```
$ nano test1.txt
```

```
  GNU nano 6.2                         test1.txt *

This is alice's text. Others shouldn't read this




                       [ File 'test1.txt' is unwritable ] ...
^G Help        ^O Write Out    ^W Where Is    ^K Cut        ^T Execute    ^C Location    M-U Undo      M-A Set Mark
^X Exit        ^R Read File    ^\ Replace     ^U Paste      ^J Justify    ^/ Go To Line  M-E Redo      M-6 Copy
```

```
 GNU nano 6.2                          test1.txt *                              R

This is alice's text. Others shouldn't read this
I, Bob have modified the file.



                          [ Error writing test1.txt: Permission denied ]
^G Help        ^O Write Out   ^W Where Is    ^K Cut        ^T Execute    ^C Location    M-U Undo    M-A Set Mark
^X Exit        ^R Read File   ^\ Replace     ^U Paste      ^J Justify    ^/ Go To Line  M-E Redo    M-6 Copy
```

f. Go to bob's home folder. Create a file named "test2". Grant alice read and write access to the file "test2".

sudo touch test2.txt
sudo chown bob test2.txt
su bob
setfacl -m u:alice:rw test2.txt


```
lkancherla@lkancherla-vm:/test545_a_1$ sudo chown bob test2.txt
lkancherla@lkancherla-vm:/test545_a_1$ ls -la
total 12
drwxr-xr-x   2 root   root 4096 Sep 15 21:21 .
drwxr-xr-x  21 root   root 4096 Sep 15 20:45 ..
-rw-r-----+  1 alice  root   49 Sep 15 20:55 test1.txt
-rw-r--r--   1 bob    root    0 Sep 15 21:21 test2.txt
lkancherla@lkancherla-vm:/test545_a_1$ su bob
Password:
$ setfacl -m u:alice:rw test2.txt
$
```

g. Switch to user alice, modify the file "test2", and save. Show the results.

su alice
nano test2.txt

GNU nano 6.2   test2.txt *

I am Alice, writing in the file created by Bob

Save modified buffer?
Y Yes
N No   ^C Cancel

---

lkancherla@lkancherla-vm:/test545_a_1$ su alice
Password:
$ nano test2.txt
Unable to create directory /home/alice/.local/share/nano/: No such file or directory
It is required for saving/loading search history or cursor positions.

$ cat test2.txt

I am Alice, writing in the file created by Bob

$

Question 2: If a user alice does not have read and execute access to a file, can you provide the commands for allowing alice read and execute access using both user privileges and group privileges (create a new group and assign the users and the file to the group)?

sudo useradd alice
sudo useradd bob
sudo groupadd lab
sudo useradd -a -G lab alice
sudo useradd -a -G lab bob
grep lab /etc/group
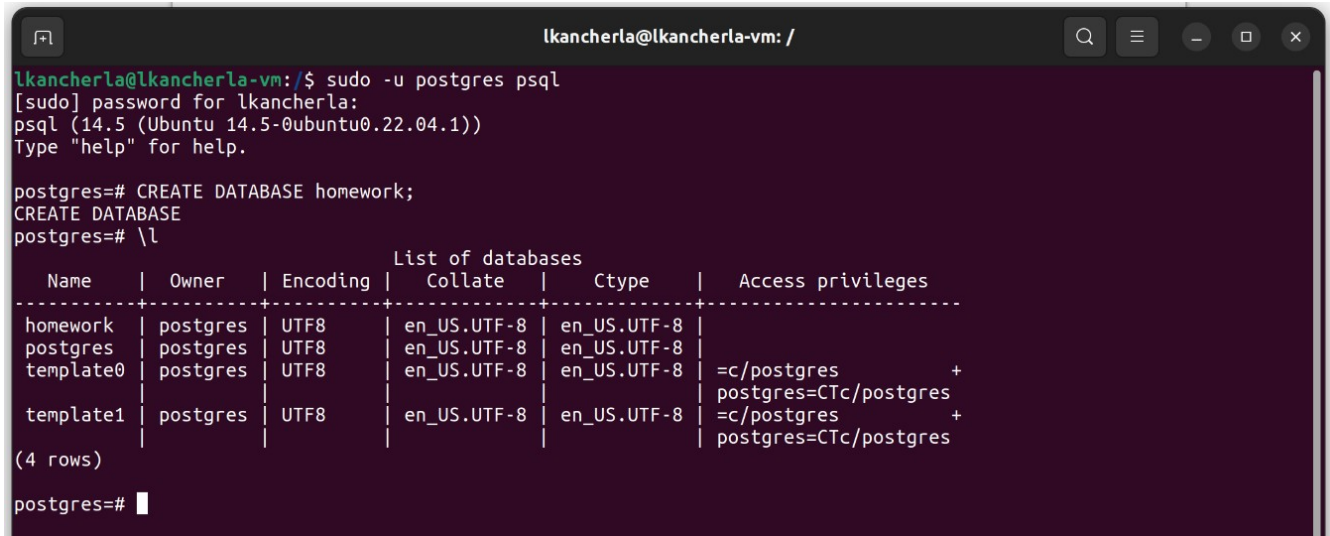

sudo touch test3.txt
sudo chown :lab test3.txt
sudo chmod 654 test3.txt
sudo chown bob test3.txt
su bob
setfacl -m u:alice:rx test3.txt

Question 3: Please provide the commands at psql and the screenshots of command line responses to show your commands executed successfully

a. First login using postgres (admin user) and create a database "homework".

sudo -u postgres psql
CREATE DATABASE homework;

b. Create a user with the name being your asuid, and assign login and createrole permissions. Grant all privileges to the created user.

sudo -u postgres psql
CREATE ROLE lkancher WITH LOGIN PASSWORD 'lkancher' CREATEROLE;

c. Login using your created user. Create a table "friends" with columns "id", "name", and "room". Insert 5 rows into the table.
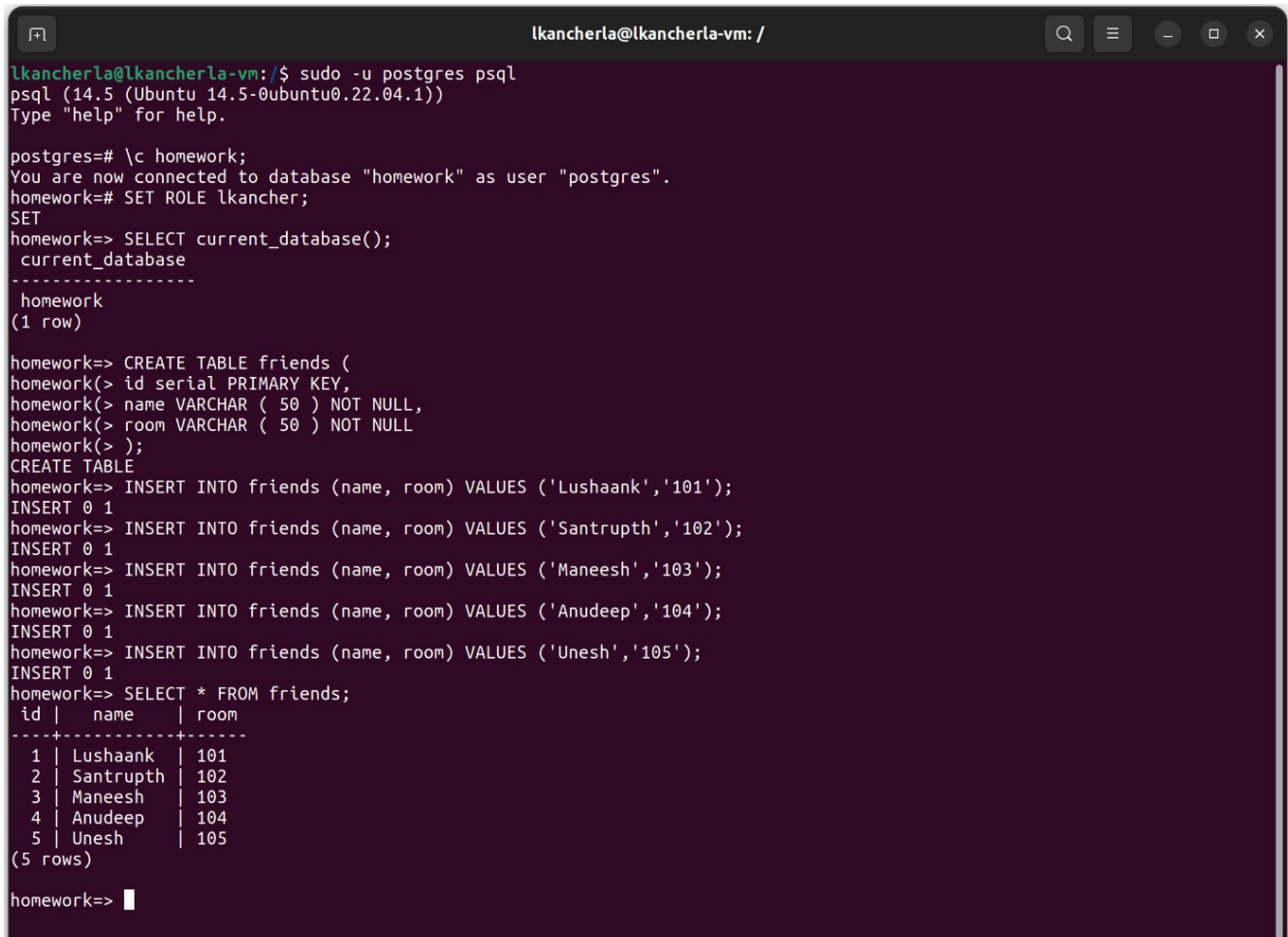
sudo -u postgres psql
\c homework;

SET ROLE lkancher;
CREATE TABLE friends (
        id serial PRIMARY KEY,
        name VARCHAR (50) NOT NULL,
        room VARCHAR (50) NOT NULL);
INSERT INTO friends (name, room) VALUES ('…', '…');
…
...

```
lkancherla@lkancherla-vm: /

lkancherla@lkancherla-vm:/$ sudo -u postgres psql
psql (14.5 (Ubuntu 14.5-0ubuntu0.22.04.1))
Type "help" for help.

postgres=# \c homework;
You are now connected to database "homework" as user "postgres".
homework=# SET ROLE lkancher;
SET
homework=> SELECT current_database();
 current_database
------------------
 homework
(1 row)

homework=> CREATE TABLE friends (
homework(> id serial PRIMARY KEY,
homework(> name VARCHAR ( 50 ) NOT NULL,
homework(> room VARCHAR ( 50 ) NOT NULL
homework(> );
CREATE TABLE
homework=> INSERT INTO friends (name, room) VALUES ('Lushaank','101');
INSERT 0 1
homework=> INSERT INTO friends (name, room) VALUES ('Santrupth','102');
INSERT 0 1
homework=> INSERT INTO friends (name, room) VALUES ('Maneesh','103');
INSERT 0 1
homework=> INSERT INTO friends (name, room) VALUES ('Anudeep','104');
INSERT 0 1
homework=> INSERT INTO friends (name, room) VALUES ('Unesh','105');
INSERT 0 1
homework=> SELECT * FROM friends;
 id |   name    | room
----+-----------+------
  1 | Lushaank  | 101
  2 | Santrupth | 102
  3 | Maneesh   | 103
  4 | Anudeep   | 104
  5 | Unesh     | 105
(5 rows)

homework=>
```
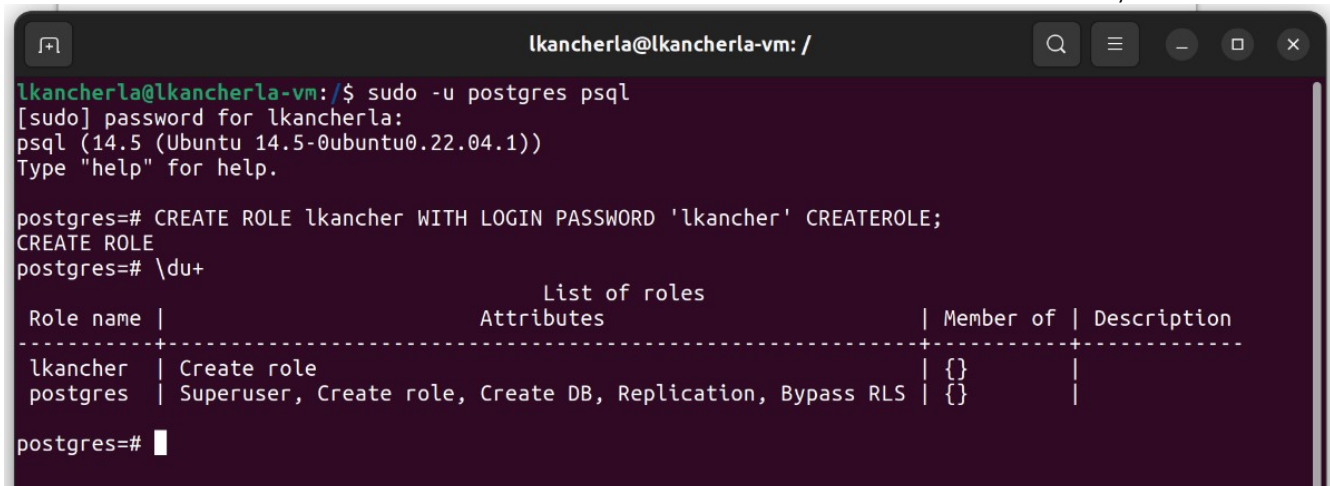
d. Create another user with the name "sub". Make sure the created user can login and can query the table "friends".

sudo -u postgres psql
\c homework;
CREATE ROLE sub WITH LOGIN PASSWORD 'sub' CREATEROLE;
SET ROLE lkancher;
GRANT SELECT ON friends TO sub;
SET ROLE sub;
SELECT * FROM friends;

```
lkancherla@lkancherla-vm:/$ sudo -u postgres psql
psql (14.5 (Ubuntu 14.5-0ubuntu0.22.04.1))
Type "help" for help.

postgres=# \c homework;
You are now connected to database "homework" as user "postgres".
homework=# CREATE ROLE sub WITH LOGIN PASSWORD 'sub' CREATEROLE;
CREATE ROLE
homework=# SET ROLE sub;
SET
homework=> SELECT * FROM friends;
ERROR:  permission denied for table friends
homework=> SET ROLE lkancher;
SET
homework=> SET ROLE postgres;
SET
homework=# GRANT SELECT ON friends TO sub;
GRANT
homework=# SET ROLE sub;
SET
homework=> SELECT * FROM friends;
 id |   name    | room
----+-----------+------
  1 | Lushaank  | 101
  2 | Santrupth | 102
  3 | Maneesh   | 103
  4 | Anudeep   | 104
  5 | Unesh     | 105
(5 rows)

homework=>
```

e. Login to the new user and create another table "notes" with columns "id" and "note". Insert 5 rows. Make sure the user using your asuid can query and insert in this new table.

sudo -u postgres psql
\c homework;
SET ROLE sub;
CREATE TABLE notes (
        id serial PRIMARY KEY,
        note VARCHAR (100) NOT NULL);
INSERT INTO notes (note) VALUES (' … ');
...
…
GRANT SELECT, INSERT ON notes TO lkancher;

```
lkancherla@lkancherla-vm:/$ sudo -u postgres psql
psql (14.5 (Ubuntu 14.5-0ubuntu0.22.04.1))
Type "help" for help.

postgres=# \c homework;
You are now connected to database "homework" as user "postgres".
homework=# SET ROLE sub;
SET
homework=> CREATE TABLE notes (
homework(> id serial PRIMARY KEY,
homework(> note VARCHAR ( 100 ) NOT NULL
homework(> );
CREATE TABLE
homework=> INSERT INTO notes (note) VALUES ('Course Bootstrap');
INSERT 0 1
homework=> INSERT INTO notes (note) VALUES ('History of Hacking');
INSERT 0 1
homework=> INSERT INTO notes (note) VALUES ('Ethical Hacking');
INSERT 0 1
homework=> INSERT INTO notes (note) VALUES ('Access Control');
INSERT 0 1
homework=> INSERT INTO notes (note) VALUES ('Confidentiality Policies');
INSERT 0 1
homework=> GRANT SELECT, INSERT ON notes TO lkancher;
GRANT
homework=>
```

f. Login as the user using your asuid and make changes to the table "notes".

sudo -u postgres psql
\c homework;
SET ROLE lkancher;
INSERT INTO notes (id, note) VALUES (6, …);
SELECT * FROM notes;


```
lkancherla@lkancherla-vm:/$ sudo -u postgres psql
psql (14.5 (Ubuntu 14.5-0ubuntu0.22.04.1))
Type "help" for help.

postgres=# \c homework;
You are now connected to database "homework" as user "postgres".
homework=# SET ROLE lkancher;
SET
homework=> SELECT * FROM notes;
 id |          note
----+--------------------------
  1 | Course Bootstrap
  2 | History of Hacking
  3 | Ethical Hacking
  4 | Access Control
  5 | Confidentiality Policies
(5 rows)

homework=> INSERT INTO notes (id, note) VALUES (6, 'Lab 1 Access Control');
INSERT 0 1
homework=> SELECT * FROM notes;
 id |          note
----+--------------------------
  1 | Course Bootstrap
  2 | History of Hacking
  3 | Ethical Hacking
  4 | Access Control
  5 | Confidentiality Policies
  6 | Lab 1 Access Control
(6 rows)

homework=>
```

g. Create a third user with the name "thirduser". Make sure he can query friends but not notes.

sudo -u postgres psql
\c homework;
CREATE ROLE thirduser WITH LOGIN PASSWORD 'thirduser' CREATEROLE;
GRANT SELECT ON friends TO thirduser;

```
lkancherla@lkancherla-vm: /                              Q  ≡  —  □  ×

lkancherla@lkancherla-vm:/$ sudo -u postgres psql
psql (14.5 (Ubuntu 14.5-0ubuntu0.22.04.1))
Type "help" for help.

postgres=# \c homework;
You are now connected to database "homework" as user "postgres".
homework=# CREATE ROLE thirduser WITH LOGIN PASSWORD 'thirduser' CREATEROLE;
CREATE ROLE
homework=# GRANT SELECT ON friends TO thirduser;
GRANT
homework=#
```

h. Login as the third user and try to query the tables notes and friends. Show the responses.

sudo -u postgres psql
\c homework;
SET ROLE thirduser;
SELECT * FROM notes;
SELECT * FROM friends;

```
lkancherla@lkancherla-vm: /                              Q  ≡  —  □  ×

lkancherla@lkancherla-vm:/$ sudo -u postgres psql
psql (14.5 (Ubuntu 14.5-0ubuntu0.22.04.1))
Type "help" for help.

postgres=# \c homework;
You are now connected to database "homework" as user "postgres".
homework=# SET ROLE thirduser;
SET
homework=> SELECT * FROM notes;
ERROR:  permission denied for table notes
homework=> SELECT * FROM friends;
 id |   name    | room
----+-----------+------
  1 | Lushaank  | 101
  2 | Santrupth | 102
  3 | Maneesh   | 103
  4 | Anudeep   | 104
  5 | Unesh     | 105
(5 rows)

homework=>
```
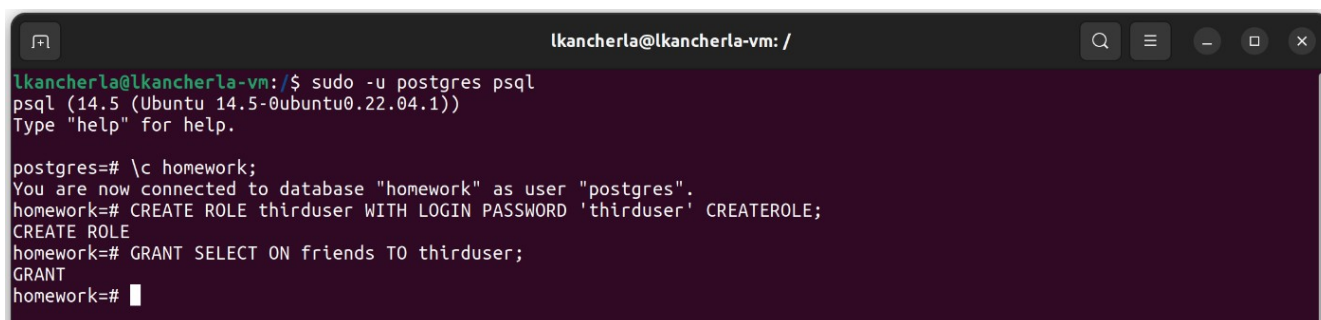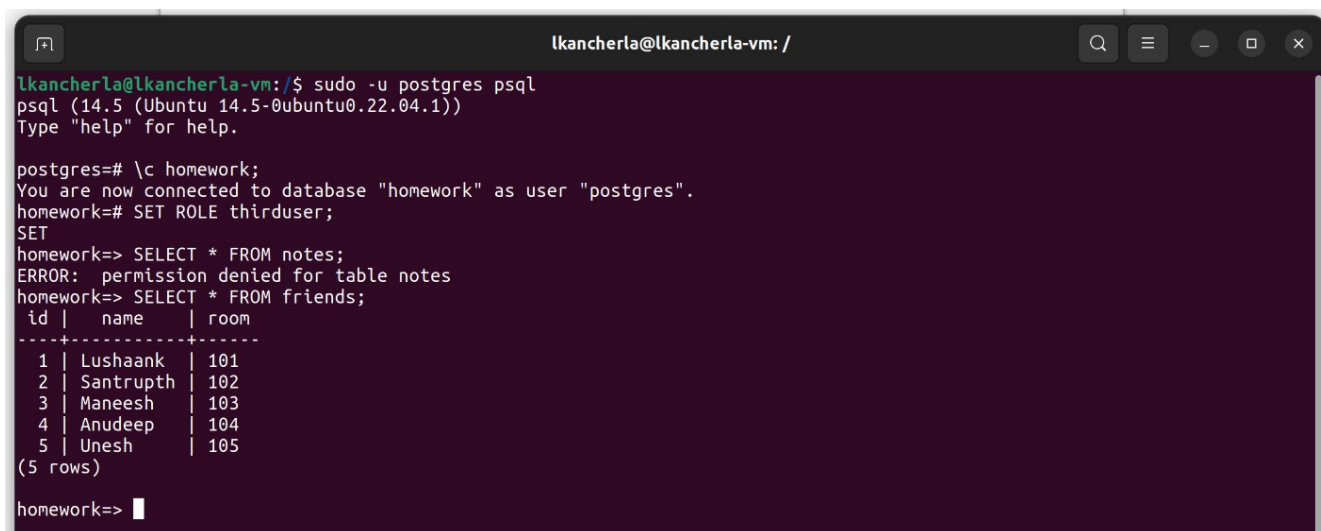
i. Create a group "noteusers" and make sure the group can query the table "notes".

sudo -u postgres psql
\c homework;
CREATE GROUP noteusers;
GRANT SELECT ON notes TO noteusers;

```
lkancherla@lkancherla-vm:/$ sudo -u postgres psql
psql (14.5 (Ubuntu 14.5-0ubuntu0.22.04.1))
Type "help" for help.

postgres=# \c homework;
You are now connected to database "homework" as user "postgres".
homework=# CREATE GROUP noteusers;
CREATE ROLE
homework=# GRANT SELECT ON notes TO noteusers;
GRANT
homework=#
```

j. Let the third user join the group "noteusers" and then query the "notes" table again to see the responses.

sudo -u postgres psql
\c homework;
ALTER GROUP noteusers ADD USER thirduser;
SET ROLE thirduser;
SELECT * FROM notes;


```
lkancherla@lkancherla-vm:/$ sudo -u postgres psql
psql (14.5 (Ubuntu 14.5-0ubuntu0.22.04.1))
Type "help" for help.

postgres=# \c homework;
You are now connected to database "homework" as user "postgres".
homework=# SET ROLE thirduser;
SET
homework=> SELECT * FROM notes;
ERROR:  permission denied for table notes
homework=> SET ROLE postgres;
SET
homework=# ALTER GROUP noteusers ADD USER thirduser;
ALTER ROLE
homework=# SET ROLE thirduser;
SET
homework=> SELECT * FROM notes;
 id |          note
----+-------------------------
  1 | Course Bootstrap
  2 | History of Hacking
  3 | Ethical Hacking
  4 | Access Control
  5 | Confidentiality Policies
  6 | Lab 1 Access Control
(6 rows)

homework=>
```