МИНИСТЕРСТВО НАУКИ
И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное
образовательное учреждение высшего образования
«НОВОСИБИРСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

НГТУ
НЭТИ | **Факультет прикладной математики и информатики**

Кафедра теоретической и прикладной информатики

Лабораторная работа № 6

по дисциплине «Администрирование информационных систем»

РЕПЛИКАЦИЯ

Бригада 2 ХАЙДАЕВ К.Е.

Группа ПМИ-82 ЗЯБЛИЦЕВА У.П.

Вариант 2

Преподаватели АВРУНЕВ О.Е.

Новосибирск, 2022

1 Убедиться, что значение параметра wal_level равно replica, если нет, то установить. Установить значение параметра max_wal_senders в 2

show wal_level;

alter system set max_wal_senders=2;

```
[dba@centos-7 ~]$  psql demo
psql (14.1)
Type "help" for help.

demo=# show wal_level;
 wal_level
-----------
 replica
(1 row)

demo=#  alter system set max_wal_senders=2;
ALTER SYSTEM
demo=#
```

Проверить наличие в pg_hba.conf записи для подключения по протоколу репликации, и установить запись вместо существующих
host replication all 127.0.0.1/32 trust.

sudo vi /var/lib/pgpro/std-14/data/pg_hba.conf

```
# Allow replication connections from localhost, by a user with the
# replication privilege.
local   replication     all                                     peer
host    replication     all             127.0.0.1/32            trust
host    replication     all             ::1/128                 md5
```

systemctl restart postgrespro-std-14

select * from pg_hba_file_rules;

```
demo=# select * from pg_hba_file_rules;
 line_number | type  |   database    | user_name |  address   |                 netmask                 | auth_method | options | error
-------------+-------+---------------+-----------+------------+-----------------------------------------+-------------+---------+-------
          85 | local | {all}         | {all}     |            |                                         | trust       |         |
          87 | host  | {all}         | {all}     | 127.0.0.1  | 255.255.255.255                         | md5         |         |
          89 | host  | {all}         | {all}     | ::1        | ffff:ffff:ffff:ffff:ffff:ffff:ffff:ffff | md5         |         |
          92 | local | {replication} | {all}     |            |                                         | peer        |         |
          93 | host  | {replication} | {all}     | 127.0.0.1  | 255.255.255.255                         | trust       |         |
          94 | host  | {replication} | {all}     | ::1        | ffff:ffff:ffff:ffff:ffff:ffff:ffff:ffff | md5         |         |
          95 | host  | {demo}        | {dba}     | 172.17.6.1 | 255.255.0.0                             | trust       |         |
(7 rows)
```

2 Создать автономную резервную копию основного кластера кластер

pg_basebackup --pgdata=/usr/local/pgsql2/data -R

cd /usr/local/

mkdir pgsql2

chown postgres:postgres pgsql2

sudo su – postgres

pg_basebackup --pgdata=/usr/local/pgsql2/data -R

cd /usr/local/pgsql2/data

```
-bash-4.2$ cd /usr/local/pgsql2/data
-bash-4.2$ ls
backup_label     current_logfiles  pg_commit_ts  pg_ident.conf  pg_notify     pg_snapshots  pg_subtrans  PG_VERSION  postgresql.auto.conf
backup_manifest  global            pg_dynshmem   pg_logical     pg_replslot   pg_stat       pg_tblspc    pg_wal      postgresql.conf
base             log               pg_hba.conf   pg_multixact   pg_serial     pg_stat_tmp   pg_twophase  pg_xact     standby.signal
```

Перенести ее в каталог данных второго, предварительно остановив его, и удалив из каталога данных /var/lib/pgpro/std-14-sec/data содержимое

pg_ctl stop -D /var/lib/pgpro/std-14-sec/data

rm -rf /var/lib/pgpro/std-14-sec/data/*

mv /usr/local/pgsql2/data/* /var/lib/pgpro/std-14-sec/data

После этого привести содержание файла recovery.conf из/var/lib/pgpro/std-14-sec/data

Файла recovery.conf начиная с Postgres 12 не существует.

3 Поменять порт резервного кластера на 5433, и установить параметру hot_standby значение on

sudo vi /var/lib/pgpro/std-14-sec/data/postgresql.conf

```
#------------------------------------------------------------------------------
# CONNECTIONS AND AUTHENTICATION
#------------------------------------------------------------------------------

# - Connection Settings -

listen_addresses = '*'          # what IP address(es) to listen on;
                                # comma-separated list of addresses;
                                # defaults to 'localhost'; use '*' for all
                                # (change requires restart)
port = 5433                     # (change requires restart)
max_connections = 100           # (change requires restart)
```

```
# These settings are ignored on a primary server.

#primary_conninfo = ''                  # connection string to sending server
#primary_slot_name = ''                 # replication slot on sending server
#promote_trigger_file = ''              # file name whose presence ends recovery
hot_standby = on                        # "off" disallows queries during recovery
```

Запустить второй сервер

sudo su - postgres

pg_ctl start -D /var/lib/pgpro/std-14-sec/data

```
-bash-4.2$ pg_ctl start -D /var/lib/pgpro/std-14-sec/data
waiting for server to start....2022-03-30 07:10:25.842 +07 [4066] LOG:  redirecting log output to logging collector process
2022-03-30 07:10:25.842 +07 [4066] HINT:  Future log output will appear in directory "log".
 done
server started
```

Проверить наличие процесса wal receiver для второго сервера

ps -o pid,command --ppid `head -n 1 /var/lib/pgpro/std-14-sec/data/postmaster.pid` (тут косые `` - как в sql, ps -eaf | egrep "receiver|startup" – тоже показывает).

```
-bash-4.2$ ps -o pid,command --ppid  `head -n 1 /var/lib/pgpro/std-14-sec/data/postmaster.pid`
  PID COMMAND
 4067 postgres: logger
 4068 postgres: startup recovering 000000010000000000000002C
 4069 postgres: checkpointer
 4070 postgres: background writer
 4071 postgres: stats collector
 4072 postgres: walreceiver streaming 0/2C000758
```

Walsender:

ps -o pid,command --ppid `head -n 1 /var/lib/pgpro/std-14/data/postmaster.pid`

(тут косые `` - как в sql, ps -eaf | grep sender – тоже показывает).

```
-bash-4.2$ ps -o pid,command --ppid  `head -n 1 /var/lib/pgpro/std-14/data/postmaster.pid`
  PID COMMAND
 3278 postgres: logger
 3280 postgres: checkpointer
 3281 postgres: background writer
 3282 postgres: walwriter
 3283 postgres: autovacuum launcher
 3284 postgres: archiver last was 000000010000000000000002B.00000028.backup
 3285 postgres: stats collector
 3286 postgres: logical replication launcher
 4073 postgres: walsender postgres [local] streaming 0/2C000758
```

4 Проверка репликации

Получить информацию о процессе репликации на основном сервере

SELECT * FROM pg_stat_replication \gx

```
-bash-4.2$ psql demo
psql (14.1)
Type "help" for help.

demo=# SELECT * FROM pg_stat_replication \gx
-[ RECORD 1 ]----+------------------------------
pid              | 4073
usesysid         | 10
usename          | postgres
application_name | walreceiver
client_addr      |
client_hostname  |
client_port      | -1
backend_start    | 2022-03-30 07:10:26.005276+07
backend_xmin     |
state            | streaming
sent_lsn         | 0/2C000758
write_lsn        | 0/2C000758
flush_lsn        | 0/2C000758
replay_lsn       | 0/2C000758
write_lag        |
flush_lag        |
replay_lag       |
sync_priority    | 0
sync_state       | async
reply_time       | 2022-03-30 07:23:26.255922+07
```

Внести изменения в одну из таблиц б.д. demo

Проверить, что они применились на втором сервере.

Убедиться, что попытка модификации данных на резервном сервере не допускается

select * from bookings.bookings limit 1;

```
demo=# select * from bookings.bookings limit 1;
 book_ref |       book_date        | total_amount
----------+------------------------+--------------
 00000F   | 2016-09-02 06:12:00+07 |    265700.00
(1 row)
```

update bookings.bookings

set total_amount=total_amount-5700

where book_ref='00000F';

select * from bookings.bookings where book_ref='00000F';

```
demo=# update bookings.bookings
set total_amount=total_amount-5700
where book_ref='00000F';
UPDATE 1
demo=# select * from bookings.bookings where book_ref='00000F';
 book_ref |       book_date        | total_amount
----------+------------------------+--------------
 00000F   | 2016-09-02 06:12:00+07 |    260000.00
(1 row)
```

psql demo -p 5433

select * from bookings.bookings where book_ref='00000F';

```
demo=# \q
-bash-4.2$ psql demo -p 5433
psql (14.1)
Type "help" for help.

demo=# select * from bookings.bookings where book_ref='00000F';
 book_ref |       book_date        | total_amount
----------+------------------------+--------------
 00000F   | 2016-09-02 06:12:00+07 |    260000.00
(1 row)

demo=# update bookings.bookings
set total_amount=total_amount-5700
where book_ref='00000F';
ERROR:  cannot execute UPDATE in a read-only transaction
```

5 Логическая репликация

Перевести второй сервер из режима восстановления в обычный режим

/opt/pgpro/std-14/bin/pg_ctl -w -D /var/lib/pgpro/std-14-sec/data -l logfile promote

```
-bash-4.2$ /opt/pgpro/std-14/bin/pg_ctl -w -D /var/lib/pgpro/std-14-sec/data -l logfile promote
waiting for server to promote.... done
server promoted
```

У основного сервера изменить значение параметра wal_level на logical и перезапустить его.

alter system set wal_level='logical';

systemctl restart postgrespro-std-14

show wal_level;

```
demo=# alter system set wal_level='logical';
ALTER SYSTEM
demo=# \q
-bash-4.2$ systemctl restart postgrespro-std-14
==== AUTHENTICATING FOR org.freedesktop.systemd1.manage-units ===
Authentication is required to manage system services or units.
Authenticating as: dba
Password:
==== AUTHENTICATION COMPLETE ===
-bash-4.2$ psql demo
psql (14.1)
Type "help" for help.

demo=# show wal_level
demo-# ;
 wal_level
-----------
 logical
(1 row)
```

6 На первом сервере создать публикацию для одной из таблиц б.д. demo

Вариант	Реплицируемые операции
1,2, 9	Все

create publication pub_lab6 for table bookings.bookings;

(create publication pub_lab6 for table bookings.bookings with (publish= 'insert, update, delete'); - то же самое)

7 Получить данные о публикации командой psql - \dRp+

psql - \dRp+

```
demo=# create publication pub_lab6 for table bookings.bookings;
CREATE PUBLICATION
demo=# psql - \dRp+
                            Publication pub_lab6
  Owner   | All tables | Inserts | Updates | Deletes | Truncates | Via root
----------+------------+---------+---------+---------+-----------+----------
 postgres | f          | t       | t       | t       | t         | f
Tables:
    "bookings.bookings"
```

8 На втором сервере создать подписку на эту публикацию с опцией (copy_data = false).

psql -d demo -p 5433

create subscription sub_lab6

connection 'host=127.0.0.1 port=5432 user=dba

password=sladkiyKot dbname=demo'

publication pub_lab6

with (copy_data = false);

```
demo=# create subscription sub_lab6
connection 'host=127.0.0.1 port=5432 user=dba
password=sladkiyKot dbname=demo'
publication insert_publication
with (copy_data = false);
NOTICE:  created replication slot "sub_lab6" on publisher
CREATE SUBSCRIPTION
```

9 Получить данные о состоянии подписки

SELECT * FROM pg_stat_subscription \gx

```
demo=# SELECT * FROM pg_stat_subscription \gx
-[ RECORD 1 ]---------+------------------------------
subid                 | 16566
subname               | sub_lab6
pid                   | 5328
relid                 |
received_lsn          | 0/2D01A718
last_msg_send_time    | 2022-03-30 08:29:50.262648+07
last_msg_receipt_time | 2022-03-30 08:29:50.262948+07
latest_end_lsn        | 0/2D01A718
latest_end_time       | 2022-03-30 08:29:50.262648+07
```

Проверить работу подписки выполнив операции над данными таблицы на основном сервере и посмотрев наличие этих данных на втором.

На основном сервере выполним операции вставки и модификации.

INSERT INTO bookings.bookings (book_ref, book_date, total_amount)

VALUES ('FFFFFF', ' 2022-03-30 06:12:00+07', 1);

select * from bookings.bookings where book_ref='FFFFFF';

update bookings.bookings

set total_amount=total_amount+1

where book_ref='FFFFFF';

select * from bookings.bookings where book_ref='FFFFFF';

delete from bookings.bookings where book_ref='FFFFFF';

```
INSERT 0 1
demo=#
demo=# select * from bookings.bookings where book_ref='FFFFFF';
 book_ref |       book_date        | total_amount
----------+------------------------+--------------
 FFFFFF   | 2022-03-30 06:12:00+07 |         1.00
(1 row)

demo=#
demo=# update bookings.bookings
demo-# set total_amount=total_amount+1
demo-# where book_ref='FFFFFF';
UPDATE 1
demo=#
demo=# select * from bookings.bookings where book_ref='FFFFFF';
 book_ref |       book_date        | total_amount
----------+------------------------+--------------
 FFFFFF   | 2022-03-30 06:12:00+07 |         2.00
(1 row)

demo=# \q
-bash-4.2$ psql -d demo -p 5433
psql (14.1)
Type "help" for help.

demo=# select * from bookings.bookings where book_ref='FFFFFF';
 book_ref |       book_date        | total_amount
----------+------------------------+--------------
 FFFFFF   | 2022-03-30 06:12:00+07 |         2.00
(1 row)

demo=# \q
-bash-4.2$ psql demo
psql (14.1)
Type "help" for help.

demo=# delete from bookings.bookings where book_ref='FFFFFF';
DELETE 1
demo=# select * from bookings.bookings where book_ref='FFFFFF';
 book_ref | book_date | total_amount
----------+-----------+--------------
(0 rows)

demo=# \q
-bash-4.2$ psql -d demo -p 5433
psql (14.1)
Type "help" for help.

demo=# select * from bookings.bookings where book_ref='FFFFFF';
 book_ref | book_date | total_amount
----------+-----------+--------------
(0 rows)
```

Все запросы реплицируются.