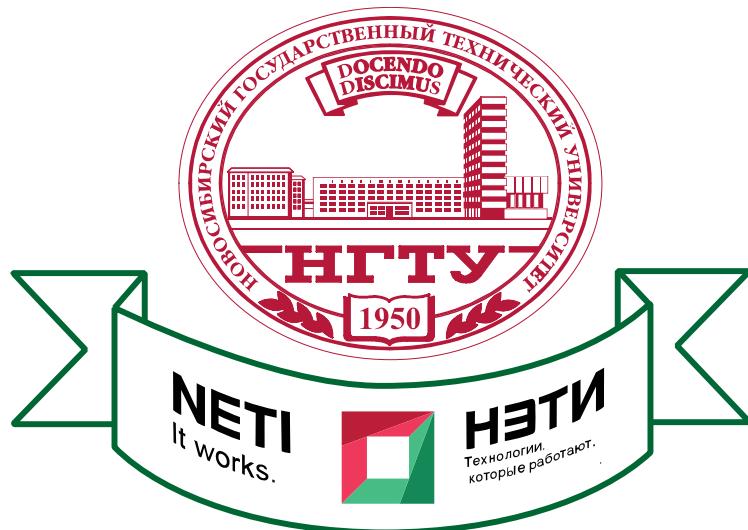


Министерство науки и высшего образования
Российской Федерации

Федеральное государственное бюджетное
образовательное учреждение высшего образования

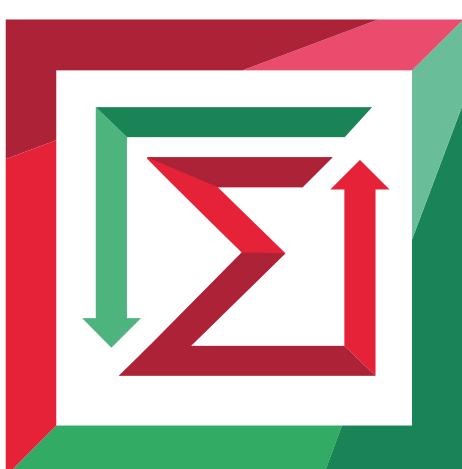
«НОВОСИБИРСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»



Кафедра теоретической и прикладной информатики

Лабораторная работа № 6
по дисциплине «Администрирование информационных систем»

РЕПЛИКАЦИЯ



Факультет: ПМИ
Группа: ПМИ-02
Бригада: 8
Студенты: Сидоров Даниил,
Дюков Богдан
Преподаватель: Аврунев О.Е.

Новосибирск

2026

Ход работы

1. Физическая репликация

- Убедились, что значение параметра wal_level равно replica:

```
#-----  
# WRITE-AHEAD LOG  
#-----  
  
# - Settings -  
  
#wal_level = replica          # minimal, replica, or logical  
                                # (change requires restart)
```

Установили значение параметра max_wal_senders в 2 (было 10).

```
#-----  
# REPLICATION  
#-----  
  
# - Sending Servers -  
  
# Set these on the primary and on any standby that will send replication data.  
  
#max_wal_senders = 2          # max number of walsender processes  
                                # (change requires restart)
```

Проверили наличие в pg_hba.conf записи для подключения по протоколу репликации и установили запись вместо существующих:

#	TYPE	DATABASE	USER	ADDRESS	METHOD
	local	all	all		peer
	host	all	all	127.0.0.1/32	md5
	host	all	all	::1/128	md5
	local	replication	all		peer
	host	replication	all	127.0.0.1/32	trust
	host	demo	user1	172.17.7.71/32	trust

Перезапустили postgres:

```
[root@centos-stream-8 ~]# systemctl restart postgresql-15.service  
[root@centos-stream-8 ~]#
```

- Создали автономную резервную копию основного кластера кластер (перед этим создали соответствующие директории):

```
[postgres@centos-stream-8 ~]$ pg_basebackup --pgdata=/usr/local/pgsql12/data -R  
[postgres@centos-stream-8 ~]$
```

Остановили каталог данных второго кластера:

```
[postgres@centos-stream-8 ~]$ /opt/pgpro/std-15/bin/pg_ctl -D /var/lib/pgpro/std-11-sec/data -l log file stop
waiting for server to shut down.... done
server stopped
[postgres@centos-stream-8 ~]$
```

Удалили содержимое из каталога данных /var/lib/pgpro/std-11-sec/data и добавили нужные права:

```
[postgres@centos-stream-8 std-11-sec]$ rm -rf data
[postgres@centos-stream-8 std-11-sec]$ mkdir data
[postgres@centos-stream-8 std-11-sec]$ su root
Password:
[root@centos-stream-8 std-11-sec]# chmod 750 data
[root@centos-stream-8 std-11-sec]#
```

Перенесли созданную резервную копию в каталог данных второго кластера:

```
[postgres@centos-stream-8 ~]$ cp -r /usr/local/pgsql12/data/* /var/lib/pgpro/std-11-sec/
[postgres@centos-stream-8 ~]$
```

Файл `standby.signal`, который появился во втором кластере, означает настройку `standby_mode=on` в `recovery.conf` до PostgreSQL 12.

1.3. Поменяли порт резервного кластера на 5433:

```
#  
# CONNECTIONS AND AUTHENTICATION  
#-----  
  
# - Connection Settings -  
  
listen_addresses = 'localhost' # 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)  
superuser_reserved_connections = 3 # (change requires restart)  
unix_socket_directories = '/tmp' # comma-separated list of directories  
                                 # (change requires restart)  
unix_socket_group = '' # (change requires restart)  
unix_socket_permissions = 0777 # begin with 0 to use octal notation  
                             # (change requires restart)  
bonjour = off # advertise server via Bonjour  
               # (change requires restart)  
bonjour_name = '' # defaults to the computer name  
                  # (change requires restart)
```

Установили параметру hot_standby значение оп:

Запустили второй сервер:

```
[postgres@centos-stream-8 std-11-sec]$ /opt/pgpro/std-15/bin/pg_ctl -w -D /var/lib/pgpro/std-11-sec/data -l logfile start
waiting for server to start.... done
server started
[postgres@centos-stream-8 std-11-sec]$
```

Проверили наличие процесса wal receiver для второго сервера:

```
[postgres@centos-stream-8 std-ll-sec]$ ps -o pid,command --ppid $(head -n 1 /var/lib/pgpro/std-ll-sec/data/postmaster.pid)
 PID COMMAND
 21002 postgres: logger
 21003 postgres: checkpointer
 21004 postgres: background writer
 21005 postgres: startup recovering 0000000100000000000000000000002B
 21006 postgres: walreceiver streaming 0/2B000018
[postgres@centos-stream-8 std-ll-sec]$
```

Проверили наличие процесса wal sender для основного сервера:

```
[postgres@centos-stream-8 std-11-sec]$ ps -o pid,command --ppid $(head -n 1 /var/lib/pgpro/std-15/data/postmaster.pid)
    PID COMMAND
 20856 postgres: logger
 20857 postgres: checkpointer
 20858 postgres: background writer
 20860 postgres: walwriter
 20861 postgres: autovacuum launcher
 20862 postgres: logical replication launcher
 20865 postgres: dba postgres 127.0.0.1(21525) idle
21007 postgres: walsender postgres [local] streaming 0/2B000018
```

1.4. Проверка репликации

Получили информацию о процессе репликации на основном сервере:

```
demo=# SELECT * FROM pg_stat_replication \gx
-[ RECORD 1 ]-----+
pid          | 21007
usesysid     | 10
username      | postgres
application_name | walreceiver
client_addr   |
client_hostname |
client_port    | -1
backend_start  | 2024-03-13 00:00:11.511021+07
backend_xmin   |
state         | streaming
sent_lsn      | 0/2B000018
write_lsn     | 0/2B000018
flush_lsn     | 0/2B000018
replay_lsn    | 0/2B000018
write_lag     |
flush_lag     |
replay_lag    |
sync_priority | 0
sync_state    | async
reply_time    | 2024-03-13 00:27:06.012492+07

demo=#
```

Внесли изменения в одну из таблиц бд. demo:

Data Output Messages Notifications

	id [PK] integer	name character varying (50)
1	123	hello

✓ Data saved successfully.

Total rows: 0 of 0 | Query complete 00:00:00.253 ✓ Successfully run. Total query runtime: 253 msec. 0 rows affected.

Проверили, что они применились на втором сервере:

new_server

Databases (2)

- demo
 - Casts
 - Catalogs
 - Event Triggers
 - Extensions
 - Foreign Data Wrappers
 - Languages
 - Publications
 - Schemas (4)
 - bookings
 - myschema
 - Aggregates
 - Collations
 - Domains
 - FTS Configurations
 - FTS Dictionaries
 - FTS Parsers
 - FTS Templates
 - Foreign Tables
 - Functions
 - Materialized Views
 - Operators
 - Procedures
 - Sequences
 - Tables (2)
 - table1
 - table2

Убедились, что попытка модификации данных на резервном сервере не допускается:

ERROR: cannot execute DELETE in a read-only transaction

SQL state: 25006

2. Логическая репликация

- 2.1. Перевели второй сервер из режима восстановления в обычный режим:

```
[postgres@centos-stream-8 ~]$ /opt/pgpro/std-15/bin/pg_ctl -w -D /var/lib/pgpro/std-11-sec/data -l logfile  
waiting for server to promote.... done  
server promoted  
[postgres@centos-stream-8 ~]$
```

У основного сервера изменили значение параметра wal_level на logical и перезапустили его:

```
#-----  
# WRITE-AHEAD LOG  
#-----  
  
# - Settings -  
  
wal_level = logical                                # minimal, replica, or logical  
# (change requires restart)  
#fsync = on                                         # flush data to disk for crash safety  
# (turning this off can cause  
# unrecoverable data corruption)
```

- 2.2. На первом сервере создали публикацию для одной из таблиц б.д. demo.

Реплицируемая операция согласно бригадному варианту: **Вставка**:

```
demo=# CREATE PUBLICATION my_publication  
FOR TABLE myschema.table2  
WITH (publish = 'insert');  
CREATE PUBLICATION  
demo=#
```

- 2.3. Получили данные о публикации командой psql - \dRp+:

```
demo=# \dRp+  
                               Publication my_publication  
   Owner | All tables | Inserts | Updates | Deletes | Truncates | Via root  
-----+-----+-----+-----+-----+-----+-----+-----+-----+  
    dba   | f          | t       | f       | f       | f       | f  
Tables:  
      "myschema.table2"  
demo=#
```

- 2.4. На втором сервере создали подписку на эту публикацию с опцией copy_data = false (в строке соединения - пользователь dba):

```
demo=# CREATE SUBSCRIPTION my_subscription  
CONNECTION 'host=127.0.0.1 port=5432 dbname=demo user=dba password=disidorow'  
  PUBLICATION my_publication  
    WITH (copy_data = false);  
NOTICE: created replication slot "my_subscription" on publisher  
CREATE SUBSCRIPTION  
demo=#
```

- 2.5. Получили данные о состоянии подписки:

```
demo=# SELECT * FROM pg_stat_subscription \gx
-[ RECORD 1 ]-----+
subid          | 16562
subname        | my_subscription
pid            | 6500
relid          |
received_lsn   | 0/220052D0
last_msg_send_time | 2024-03-13 01:10:26.914169+07
last_msg_receipt_time | 2024-03-13 01:10:26.914546+07
latest_end_lsn | 0/220052D0
latest_end_time  | 2024-03-13 01:10:26.914169+07

demo=#
```

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

```
[dba@centos-stream-8 ~]$ psql -d demo
psql (15.5)
Type "help" for help.

demo=# INSERT INTO myschema.table2 VALUES (1, 'Привет');
INSERT 0 1
demo=# \q
[dba@centos-stream-8 ~]$ psql -d demo -p 5433
psql (15.5)
Type "help" for help.

demo=# SELECT * FROM myschema.table2;
 id | name
----+-----
  1 | Привет
(1 row)

demo=#
```