

**Федеральное государственное автономное образовательное  
учреждение высшего образования**

**НИУ ИТМО**

**Факультет программной инженерии и компьютерной техники**

**Дисциплина:**

«Распределенные системы хранения данных»

**ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №4**

**Выполнили:**

студенты группы Р33302

Сабитов Д.Т.

Алексеев Д.И.

**Преподаватель:**

Шешуков Дмитрий Михайлович

Санкт-Петербург

2023г.

## Задание (вариант 17)

В качестве хостов использовать одинаковые виртуальные машины.

В первую очередь настроить сеть виртуальных машин:

Если ВМ запускаются на одном хосте, рекомендуется использовать NAT сеть.

Если ВМ запускаются на различных хостах, рекомендуется использовать сетевые интерфейсы в режиме "Bridge"; для связи рекомендуется использовать проводное соединение.

Проверить сетевую связность между всеми узлами (ping, ssh).

Для подключения к СУБД (например, через psql), использовать отдельную виртуальную или физическую машину.

Перед тем как "сломать" узел на этапе 2, рекомендуется выполнить снимок виртуальной машины.

Для демонстрации наполнения базы, а также доступа на запись (см. задание ниже) использовать не меньше двух таблиц, трёх столбцов, пяти строк, двух транзакций, двух клиентских сессий. Данные не обязаны быть осмысленными, но должны быть легко отличимы - повторяющиеся строки запрещены.

### Этап 1 Настройка:

Развернуть postgres на двух узлах в режиме потоковой репликации. Не использовать дополнительные пакеты. Продемонстрировать доступ в режиме чтение/запись на основном сервере, а также что новые данные синхронизируются на резервный.

### Этап 2.1 Подготовка:

- a. Установить несколько клиентских подключений к СУБД.
- b. Продемонстрировать состояние данных и работу клиентов в режиме чтение/запись.

### Этап 2.2 Сбой:

Симулировать недоступность основного узла - отключить сетевой интерфейс виртуальной машины, переключить его в изолированную подсеть и т.п.

### Этап 2.3 Отработка:

- a. Найти продемонстрировать в логах релевантные сообщения об ошибках.
- b. Выполнить фейловер на резервный сервер.
- c. Продемонстрировать состояние данных и работу клиентов в режиме чтение/запись.

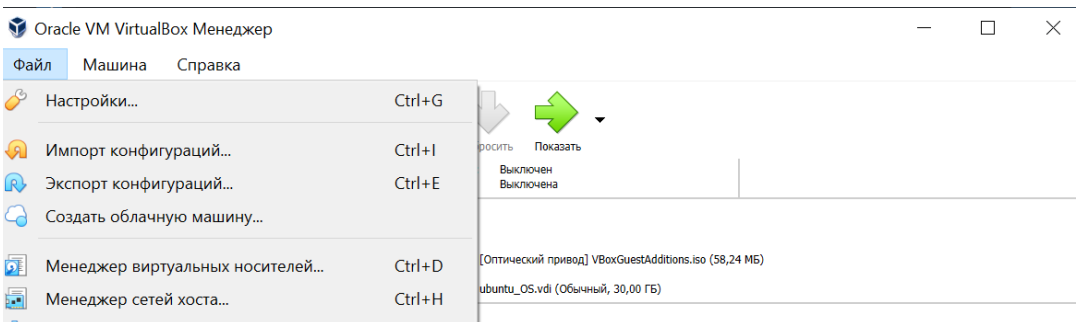
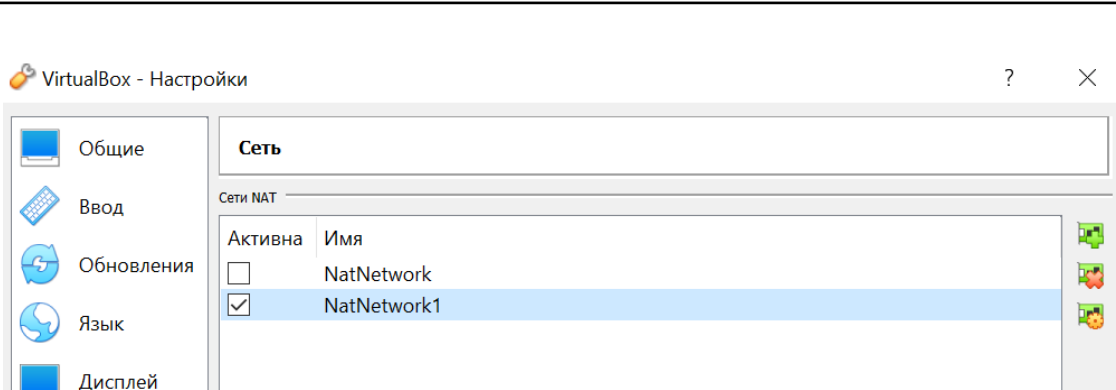
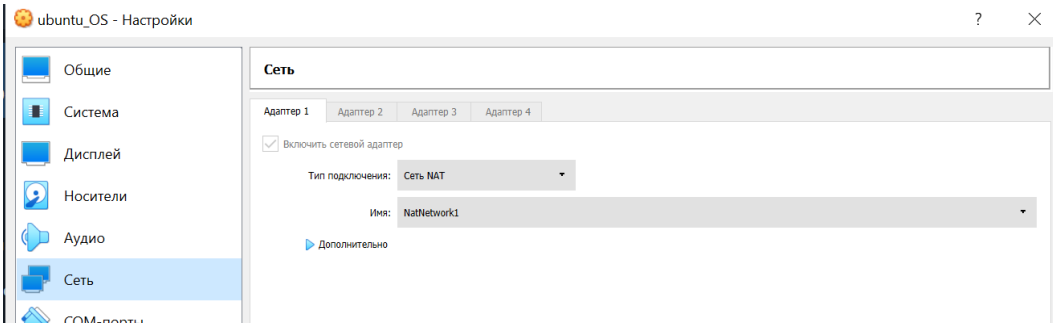
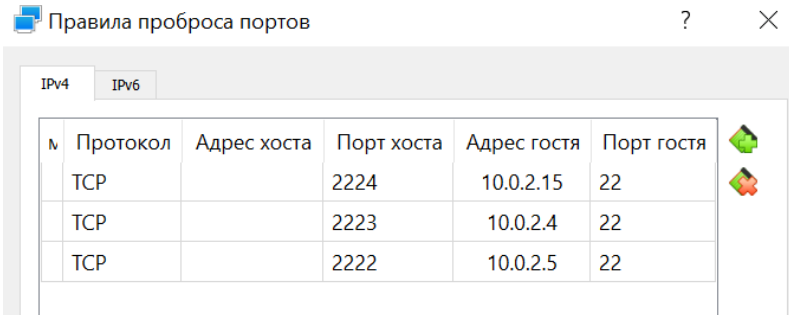
### Этап 3 Восстановление:

- a. Восстановить работу основного узла - откатить действие, выполненное с виртуальной машиной на этапе 2.2.
- b. Актуализировать состояние базы на основном узле - накатить все изменения данных, выполненные на этапе 2.3.
- c. Восстановить работу узлов в исходной конфигурации (в соответствии с этапом 1).
- d. Продемонстрировать состояние данных и работу клиентов в режиме чтение/запись.

# Этап 1 Настройка

## Создание изолированной сети и виртуальных машин

Работу было решено выполнять в VirtualBox. Создаем NAT сеть:

заходим в настройки																									
создаем NAT сеть																									
пробрасываем порты, чтобы можно было подключиться к виртуалкам с основного компьютера	  <table><tr><th>И</th><th>Протокол</th><th>Адрес хоста</th><th>Порт хоста</th><th>Адрес гостя</th><th>Порт гостя</th></tr><tr><td></td><td>TCP</td><td></td><td>2224</td><td>10.0.2.15</td><td>22</td></tr><tr><td></td><td>TCP</td><td></td><td>2223</td><td>10.0.2.4</td><td>22</td></tr><tr><td></td><td>TCP</td><td></td><td>2222</td><td>10.0.2.5</td><td>22</td></tr></table>	И	Протокол	Адрес хоста	Порт хоста	Адрес гостя	Порт гостя		TCP		2224	10.0.2.15	22		TCP		2223	10.0.2.4	22		TCP		2222	10.0.2.5	22
И	Протокол	Адрес хоста	Порт хоста	Адрес гостя	Порт гостя																				
	TCP		2224	10.0.2.15	22																				
	TCP		2223	10.0.2.4	22																				
	TCP		2222	10.0.2.5	22																				

На момент выполнения работы у нас уже была сконфигурированная виртуальная машина на Ubuntu, поэтому мы просто сделаем полное клонирование.

Запустим машины и удостоверимся в том, что они подключены к созданной сети.

master	<pre>sabitov@sabitov-VirtualBox:~\$ hostname -I 10.0.2.15 172.18.0.1 172.17.0.1 172.19.0.1</pre>
slave	<pre>sabitov@sabitov-VirtualBox:~\$ hostname -I 10.0.2.5 172.19.0.1 172.18.0.1 172.17.0.1 sabitov@sabitov-VirtualBox:~\$</pre>

Также проверим работу ssh, попробуем подключиться:

master -ssh-> slave	<pre>sabitov@sabitov-VirtualBox:~\$ ssh sabitov@10.0.2.5 The authenticity of host '10.0.2.5 (10.0.2.5)' can't be established. ECDSA key fingerprint is SHA256:bm4oHs56rNpL2LS3X0UPQHVOEAQdVZwkQusBJ3Nme/k. Are you sure you want to continue connecting (yes/no)? yes Warning: Permanently added '10.0.2.5' (ECDSA) to the list of known hosts. sabitov@10.0.2.5's password: Welcome to Ubuntu 18.04.6 LTS (GNU/Linux 5.4.0-148-generic x86_64)   * Documentation:  https://help.ubuntu.com  * Management:    https://landscape.canonical.com  * Support:        https://ubuntu.com/advantage  63 updates can be applied immediately. 40 of these updates are standard security updates. To see these additional updates run: apt list --upgradable  New release '20.04.6 LTS' available. Run 'do-release-upgrade' to upgrade to it.  Your Hardware Enablement Stack (HWE) is supported until April 2023. Last login: Thu Jun 15 08:14:40 2023 from 10.0.2.2 sabitov@sabitov-VirtualBox:~\$</pre>
windows -ssh-> master	<pre>C:\Users\spide&gt;ssh sabitov@localhost -p 2224 The authenticity of host '[localhost]:2224 ([127.0.0.1]:2224)' can't be established. ECDSA key fingerprint is SHA256:bm4oHs56rNpL2LS3X0UPQHVOEAQdVZwkQusBJ3Nme/k. Are you sure you want to continue connecting (yes/no/[fingerprint])? yes Warning: Permanently added '[localhost]:2224' (ECDSA) to the list of known hosts. sabitov@localhost's password: Welcome to Ubuntu 18.04.6 LTS (GNU/Linux 5.4.0-148-generic x86_64)   * Documentation:  https://help.ubuntu.com  * Management:    https://landscape.canonical.com  * Support:        https://ubuntu.com/advantage  63 updates can be applied immediately. 40 of these updates are standard security updates. To see these additional updates run: apt list --upgradable  Your Hardware Enablement Stack (HWE) is supported until April 2023.  The programs included with the Ubuntu system are free software; the exact distribution terms for each program are described in the individual files in /usr/share/doc/*/copyright.  Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by applicable law.  sabitov@sabitov-VirtualBox:~\$ who sabitov  :0          2023-06-15 08:00 (:0) sabitov  pts/1      2023-06-15 08:14 (10.0.2.2) sabitov@sabitov-VirtualBox:~\$ hostname -I 10.0.2.15 172.18.0.1 172.17.0.1 172.19.0.1 sabitov@sabitov-VirtualBox:~\$</pre>

## Конфигурация postgresql

Сначала добавим правила аутентификации в pg\_hba.conf. В **master'е** добавим подключения с **slave'а** и наоборот в **slave**:

```
# IPv4 local connections:
host    all             all             127.0.0.1/32          md5
# IPv6 local connections:
host    all             all             ::1/128               md5
# Allow replication connections from localhost, by a user with the
# replication privilege.
local   replication     all                               peer
host    replication     all             127.0.0.1/32          md5
host    replication     all             ::1/128               md5
# for slave
host    replication     postgres       10.0.2.5/32           md5
```

```
# IPv4 local connections:
host    all             all             127.0.0.1/32          md5
# IPv6 local connections:
host    all             all             ::1/128               md5
# Allow replication connections from localhost, by a user with the
# replication privilege.
local   replication     all                               peer
host    replication     all             127.0.0.1/32          md5
host    replication     all             ::1/128               md5
# for master
host    replication     postgres       10.0.2.15/32          md5
```

## Теперь конфигурируем узлы: делаем изменения в postgresql.conf.

Сначала для **master'a**:

which IPs the server will answer on	<code>listen_addresses = 'localhost, 10.0.2.15'</code>
включаем архивацию WAL файлов	<code>archive_mode = on</code> <code>archive_command = 'cd .'</code>
включаем горячий резерв (можно будет отправлять запросы на чтение пока узел в процессе восстановления или архивации)	<code>wal_level = hot_standby</code> <code>hot_standby = on</code>
число одновременных подключений резервных серверов или клиентов потокового копирования	<code>max_wal_senders = 2</code>

Теперь для **slave'a**:

```
listen_addresses = 'localhost, 10.0.2.5'
archive_mode = on
archive_command = 'cd .'
wal_level = hot_standby
hot_standby = on
max_wal_senders = 2
```

## Резервное копирование кластера на резервный узел:

удаляем файлы базы на slave	<pre>postgres@sabitov-VirtualBox:~/10\$ rm -rf main; mkdir main; chmod go-rwx main postgres@sabitov-VirtualBox:~/10\$</pre>
с помощью ps_basebackup выполняем копию основного узла	<pre>postgres@sabitov-VirtualBox:~/10\$ pg_basebackup -P -R -X stream -c fast -h 10.0.2.15 -U postgres -D ./main Password: 23759/23759 kB (100%), 1/1 tablespace postgres@sabitov-VirtualBox:~/10\$</pre>
проверяем наличие файлов (всё в порядке)	<pre>postgres@sabitov-VirtualBox:~/10\$ ls -l main total 84 -rw----- 1 postgres postgres 206 июн 15 09:14 backup_label drwx----- 5 postgres postgres 4096 июн 15 09:14 base drwx----- 2 postgres postgres 4096 июн 15 09:14 global drwx----- 2 postgres postgres 4096 июн 15 09:14 pg_commit_ts drwx----- 2 postgres postgres 4096 июн 15 09:14 pg_dynshmem drwx----- 4 postgres postgres 4096 июн 15 09:14 pg_logical drwx----- 4 postgres postgres 4096 июн 15 09:14 pg_multixact drwx----- 2 postgres postgres 4096 июн 15 09:14 pg_notify drwx----- 2 postgres postgres 4096 июн 15 09:14 pg_replslot drwx----- 2 postgres postgres 4096 июн 15 09:14 pg_serial drwx----- 2 postgres postgres 4096 июн 15 09:14 pg_snapshots drwx----- 2 postgres postgres 4096 июн 15 09:14 pg_stat drwx----- 2 postgres postgres 4096 июн 15 09:14 pg_stat_tmp drwx----- 2 postgres postgres 4096 июн 15 09:14 pg_subtrans drwx----- 2 postgres postgres 4096 июн 15 09:14 pg_tblspc drwx----- 2 postgres postgres 4096 июн 15 09:14 pg_twophase -rw----- 1 postgres postgres 3 июн 15 09:14 PG_VERSION drwx----- 3 postgres postgres 4096 июн 15 09:14 pg_wal drwx----- 2 postgres postgres 4096 июн 15 09:14 pg_xact -rw----- 1 postgres postgres 88 июн 15 09:14 postgresql.auto.conf -rw-rw-r-- 1 postgres postgres 170 июн 15 09:14 recovery.conf postgres@sabitov-VirtualBox:~/10\$</pre>

## Этап 2.1 Подготовка

После копирования запустим узлы, на мастере увидим информацию о резервном узле находящимся по адресу **slave**, который работает в режиме репликации

```
postgres=# select client_addr, state from pg_stat_replication;
 client_addr | state 
-----+-----
 10.0.2.5    | streaming
(1 row)
```

### Наполнение базы

лишний раз убеждаемся в том, что с резервного это сделать нельзя	<pre>postgres=# create table test(id int, name text); ERROR:  cannot execute CREATE TABLE in a read-only transaction postgres=#</pre>
а с основного МОЖНО	<pre>postgres=# INSERT INTO test_table2 (name, age) VALUES ('Biba', 21), ('Boba', 19) , ('aaaaaa', 80), ('Dima', 15); INSERT 0 4</pre>

С резервного узла получается считать данные:

```
postgres=# select * from test_table1;
 id | name  | age 
----+-----+----
  1 | Danil |  21 
  2 | Abobus | 19 
  3 | John  | 80 
  4 | Misha | 15 
(4 rows)

postgres=# select * from test_table2;
 id | name  | age 
----+-----+----
  1 | Biba  |  21 
  2 | Boba  |  19 
  3 | aaaaaa | 80 
  4 | Dima  | 15 
(4 rows)
```



## Этап 2.2 Сбой

Симулируем сбой: отключаем сетевой интерфейс на **master'e**.

На резервном узле в логах видим сообщения о недоступности сервера

```
sabitov@sabitov-VirtualBox:/var/log/postgresql$ tail -n 20 postgresql-10-main.log
Is the server running on host "10.0.2.15" and accepting
TCP/IP connections on port 5432?
2023-06-15 09:47:02.190 MSK [6770] FATAL:  could not connect to the primary server: could not connect to server: Network is unreachable
Is the server running on host "10.0.2.15" and accepting
TCP/IP connections on port 5432?
2023-06-15 09:47:07.195 MSK [6771] FATAL:  could not connect to the primary server: could not connect to server: Network is unreachable
Is the server running on host "10.0.2.15" and accepting
TCP/IP connections on port 5432?
2023-06-15 09:47:12.198 MSK [6772] FATAL:  could not connect to the primary server: could not connect to server: Network is unreachable
Is the server running on host "10.0.2.15" and accepting
TCP/IP connections on port 5432?
2023-06-15 09:47:17.205 MSK [6773] FATAL:  could not connect to the primary server: could not connect to server: Network is unreachable
Is the server running on host "10.0.2.15" and accepting
TCP/IP connections on port 5432?
2023-06-15 09:47:22.209 MSK [6774] FATAL:  could not connect to the primary server: could not connect to server: Network is unreachable
Is the server running on host "10.0.2.15" and accepting
TCP/IP connections on port 5432?
2023-06-15 09:47:27.215 MSK [6776] FATAL:  could not connect to the primary server: could not connect to server: Network is unreachable
Is the server running on host "10.0.2.15" and accepting
TCP/IP connections on port 5432?
sabitov@sabitov-VirtualBox:/var/log/postgresql$
```

На резервном узле с помощью команды `pg_ctl promote` указываем узлу выйти из режима резерва и начать операции чтения/записи.

```
sabitov@sabitov-VirtualBox:/var/log/postgresql$ sudo su - postgres
postgres@sabitov-VirtualBox:~$ /usr/lib/postgresql/12/bin/pg_ctl promote -D /var/lib/postgresql/12/main
-su: /usr/lib/postgresql/12/bin/pg_ctl: No such file or directory
postgres@sabitov-VirtualBox:~$ /usr/lib/postgresql/10/bin/pg_ctl promote -D /var/lib/postgresql/10/main
waiting for server to promote.... done
server promoted
postgres@sabitov-VirtualBox:~$
```

В логах сообщения об успешном выполнении предыдущей команды

```
sabitov@sabitov-VirtualBox:/var/log/postgresql$ tail -n 20 postgresql-10-main.log
Is the server running on host "10.0.2.15" and accepting
TCP/IP connections on port 5432?
2023-06-15 09:49:52.336 MSK [6833] FATAL:  could not connect to the primary server: could not connect to server: Network is unreachable
Is the server running on host "10.0.2.15" and accepting
TCP/IP connections on port 5432?
2023-06-15 09:49:57.339 MSK [6835] FATAL:  could not connect to the primary server: could not connect to server: Network is unreachable
Is the server running on host "10.0.2.15" and accepting
TCP/IP connections on port 5432?
2023-06-15 09:50:02.344 MSK [6836] FATAL:  could not connect to the primary server: could not connect to server: Network is unreachable
Is the server running on host "10.0.2.15" and accepting
TCP/IP connections on port 5432?
2023-06-15 09:50:07.350 MSK [6837] FATAL:  could not connect to the primary server: could not connect to server: Network is unreachable
Is the server running on host "10.0.2.15" and accepting
TCP/IP connections on port 5432?
2023-06-15 09:50:07.756 MSK [6228] LOG:  received promote request
2023-06-15 09:50:07.756 MSK [6228] LOG:  redo done at 0/3020658
2023-06-15 09:50:07.756 MSK [6228] LOG:  last completed transaction was at log time 2023-06-15 09:39:20.833487+03
2023-06-15 09:50:07.763 MSK [6228] LOG:  selected new timeline ID: 2
2023-06-15 09:50:07.819 MSK [6228] LOG:  archive recovery complete
2023-06-15 09:50:07.866 MSK [6227] LOG:  database system is ready to accept connections
sabitov@sabitov-VirtualBox:/var/log/postgresql$
```

Теперь резервный узел работает как основной и можно попробовать добавить новые данные:

```
postgres@sabitov-VirtualBox:~$ psql -c "select * from test_table2;"
 id | name | age
-----+-----+-----
  1 | Biba |  21
  2 | Boba |  19
  3 | aaaaaa |  80
  4 | Dima |  15
(4 rows)

postgres@sabitov-VirtualBox:~$ psql -c "INSERT INTO test_table2 (name, age) VALUES ('kkk', 3), ('aaa', 76), ('bbb', 23),
> ('dd', 12), ('Egor', 2);"
INSERT 0 5
postgres@sabitov-VirtualBox:~$ psql -c "select * from test_table2;"
 id | name | age
-----+-----+-----
  1 | Biba |  21
  2 | Boba |  19
  3 | aaaaaa |  80
  4 | Dima |  15
 34 | kkk |   3
 35 | aaa |  76
 36 | bbb |  23
 37 | dd |  12
 38 | Egor |   2
(9 rows)

postgres@sabitov-VirtualBox:~$
```

## Этап 3 Восстановление

На основном узле удаляем файлы данных, предварительно сохранив файлы конфигурации. При помощи утилиты rsync выполним синхронизацию файлов с резервным узлом, добавим туда файлы конфигурации.

синхронизация	<pre>sabitov@sabitov-VirtualBox:/etc/postgresql/10\$ rsync -avv postgres@10.0.2.5:~/10 /* ~/10/ opening connection using: ssh -l postgres 10.0.2.5 rsync --server --sender -vvlo gDtpre.iLsfXC . "~/10/*" (10 args) postgres@10.0.2.5's password: receiving incremental file list created directory /home/sabitov/10 delta-transmission enabled main/ main/PG_VERSION main/backup_label.old main/postgresql.auto.conf main/postmaster.opts main/postmaster.pid main/recovery.done main/base/ main/base/1/ main/base/1/112 main/base/1/113 main/base/1/1247 main/base/1/1247_fsm main/base/1/1247_vm</pre>
закончилась успешно	<pre>sabitov@sabitov-VirtualBox: /etc/postgresql/10 File Edit View Search Terminal Help main/pg_replslot/ main/pg_serial/ main/pg_snapshots/ main/pg_stat/ main/pg_stat_tmp/ main/pg_subtrans/ main/pg_subtrans/0000 main/pg_tblspc/ main/pg_twophase/ main/pg_wal/ main/pg_wal/00000000100000000000000000000003.partial main/pg_wal/000000002.history main/pg_wal/00000000200000000000000000000003 main/pg_wal/00000000200000000000000000000004 main/pg_wal/archive_status/ main/pg_wal/archive_status/00000000100000000000000000000003.partial.done main/pg_wal/archive_status/000000002.history.done main/pg_xact/ main/pg_xact/0000 total: matches=0 hash_hits=0 false_alarms=0 data=74332683  sent 18,537 bytes received 74,407,474 bytes 11,450,155.54 bytes/sec total size is 74,332,683 speedup is 1.00 sabitov@sabitov-VirtualBox:/etc/postgresql/10\$</pre>

На резервном узле вновь добавим файл standby.signal, сигнализирующий о том, что узел должен работать в резервном режиме.

```
postgres@sabitov-VirtualBox:~$ touch ~/10/main/standby.signal
postgres@sabitov-VirtualBox:~$ service postgresql restart
==== AUTHENTICATING FOR org.freedesktop.systemd1.manage-units ====
Authentication is required to restart 'postgresql.service'.
Authenticating as: sabitov,,, (sabitov)
Password:
==== AUTHENTICATION COMPLETE ====
postgres@sabitov-VirtualBox:~$ service postgresql status
● postgresql.service - PostgreSQL RDBMS
   Loaded: loaded (/lib/systemd/system/postgresql.service; enabled; vendor preset: enabled)
   Active: active (exited) since Thu 2023-06-15 10:10:51 MSK; 19s ago
     Process: 7300 ExecStart=/bin/true (code=exited, status=0/SUCCESS)
    Main PID: 7300 (code=exited, status=0/SUCCESS)
postgres@sabitov-VirtualBox:~$
```

Запустим основной узел и проверим данные:

запустили	<pre>sabitov@sabitov-VirtualBox:/etc/postgresql/10\$ service postgresql start sabitov@sabitov-VirtualBox:/etc/postgresql/10\$ service postgresql status ● postgresql.service - PostgreSQL RDBMS    Loaded: loaded (/lib/systemd/system/postgresql.service; enabled; vendor preset: enabled)    Active: active (exited) since Thu 2023-06-15 08:56:44 MSK; 1h 15min ago      Process: 5857 ExecStart=/bin/true (code=exited, status=0/SUCCESS)     Main PID: 5857 (code=exited, status=0/SUCCESS)  июн 15 08:56:44 sabitov-VirtualBox systemd[1]: Starting PostgreSQL RDBMS... июн 15 08:56:44 sabitov-VirtualBox systemd[1]: Started PostgreSQL RDBMS.</pre>
данные появились (причем новые - добавленные на <b>slave</b> ), всё в порядке	<pre>postgres@sabitov-VirtualBox:/usr/lib/postgresql/10/bin\$ psql Password: psql (10.23 (Ubuntu 10.23-0ubuntu0.18.04.2)) Type "help" for help.  postgres=# \d            List of relations Schema        Name        Type    Owner -----+-----+-----+----- public   test_table1      table   postgres public   test_table1_id_seq   sequence   postgres public   test_table2      table   postgres public   test_table2_id_seq   sequence   postgres (4 rows)  postgres=# select * from test_table2;  id   name    age ----+-----+----   1   Biba    21   2   Boba    19   3   aaaaaa   80   4   Dima    15  34   kkk      3  35   aaa     76  36   bbb     23  37   dd      12  38   Egor     2 (9 rows)</pre>

Опять проверим, что работает синхронизация при добавлении данных на основной узел:

добавили данные на  
основном узле

```
postgres=# insert into test_table2 (name, age) values ('Gigachad', 100000);
INSERT 0 1
postgres=# select * from test_table2;
 id |  name  | age 
----+-----+----
  1 | Biba   |  21
  2 | Boba   |  19
  3 | aaaaaa |  80
  4 | Dima   |  15
 34 | kkk    |   3
 35 | aaa    |  76
 36 | bbb    |  23
 37 | dd     |  12
 38 | Egor   |   2
 39 | Gigachad | 100000
(10 rows)
```

прочитали данные на **slave**,  
а новые теперь добавить  
нельзя, потому что узел  
помечен как standby

```
postgres@sabitov-VirtualBox:~/10$ psql
psql (10.23 (Ubuntu 10.23-0ubuntu0.18.04.2))
Type "help" for help.

postgres=# select * from test_table2;
 id |  name  | age 
----+-----+----
  1 | Biba   |  21
  2 | Boba   |  19
  3 | aaaaaa |  80
  4 | Dima   |  15
 34 | kkk    |   3
 35 | aaa    |  76
 36 | bbb    |  23
 37 | dd     |  12
 38 | Egor   |   2
 39 | Gigachad | 100000
(10 rows)

postgres=# insert into test_table2 (name, age) values ('CATTTT', 1);
ERROR:  cannot execute INSERT in a read-only transaction
postgres=#
```

## Вывод

В результате выполнения данной лабораторной работы изучены основные принципы функционирования распределенной системы хранения данных. Удалось на практике изучить, как достигается отказоустойчивость и балансировка нагрузки при помощи механизма репликации.