

3.2

Создаем стенд

Текущий стенд состоит из 1-ого узла: `vs01.sXXXXXX`, (XXXXXX - ваш номер студента). Работа стенда 6 часов. 2 попытки запуска стенда.

1. Над текстом нажмите кнопку "Создать стенд". Каждый стенд создан под определенную тему курса. Сейчас вы находитесь в теме - **3. Физическая резервная копия**, этот стенд не подойдет под практические занятия из других пунктов курса.

Запуск обычно идёт до 10 минут, в редких случаях - до 30 минут.

2. После создания стенда вам нужно зайти по SSH на **sbox.slurm.io** с логином и паролем из личного кабинета (<https://edu.slurm.io/>):

```
ssh s000001@sbox.slurm.io  
s000001 нужно заменить на ваш номер студента
```

Далее вам нужно перейти на сам стенд (sbox является просто jump-хостом и не понадобится кроме как для входа):

```
ssh vs01.s000001  
s000001 снова нужно заменить на ваш личный номер студента
```

На хосте vs01 у вас есть полные права (включая беспарольный sudo):

```
sudo -i
```

Будьте осторожны и не сломайте стенд до прохождения всех заданий :)

Практика: Физическая резервная копия

Можете запустить скрипт на стенде `~/practice/lecture1/2.physical_backups.sh`, который запускал спикер в видео.

Или пройтись по шагам текущей практики

- http://gitlab.slurm.io/postgres/slurm_course/

[/blob/main/practice/lecture1/2.physical_backups.md](http://gitlab.slurm.io/postgres/slurm_course/blob/main/practice/lecture1/2.physical_backups.md), также представленной ниже по тексту.

КУПЛЕНО НА

SKLADCHIK.COM

Холодное резервирование

На прошлой практике мы создали с вами три БД: dbcoru1 на первом сервере, который находится на порту 5432, и dbcoru1 и dbcoru2 на втором сервере на порту 5433. Давайте теперь создадим резервную копию сервера, находящегося на порту 5432. Восстанавливать сервер будем на порт 5433. На примере холодного копирования.

1. Останавливаем PostgreSQL на порту 5432 и 5433, предварительно выполним checkpoint на сервере 5432

```
postgres$ psql
5432=> checkpoint;
5432=> \q

postgres$ pg_ctl -D /var/lib/pgsql/12/main5432 stop
postgres$ pg_ctl -D /var/lib/pgsql/12/main5433 stop
```

2. Удаляем каталог 5433, что мы создали на первой практике

```
postgres$ rm -rf /var/lib/pgsql/12/main5433/*
```

3. Копируем данные с

каталога /var/lib/pgsql/12/main5432/ на /var/lib/pgsql/12/main5433

```
postgres$ cp -R /var/lib/pgsql/12/main5432/* /var/lib/pgsql/12/main5433
```

4. Проверяем, что данные скопировались

```
postgres$ ls ./12/main5433/
total 136
drwx----- 20 postgres postgres 4096 Mar 30 13:02 ./
drwxr-xr-x   3 postgres postgres 4096 Mar 30 13:02 ../
drwx-----  6 postgres postgres 4096 Mar 30 13:02 base/
-rw-----  1 postgres postgres   30 Mar 30 13:02 current_logfiles
drwx-----  2 postgres postgres 4096 Mar 30 13:02 global/
drwx-----  2 postgres postgres 4096 Mar 30 13:02 log/
drwx-----  2 postgres postgres 4096 Mar 30 13:02 pg_commit_ts/
drwx-----  2 postgres postgres 4096 Mar 30 13:02 pg_dynshmem/
-rw-----  1 postgres postgres 4760 Mar 30 13:02 pg_hba.conf
-rw-----  1 postgres postgres 1636 Mar 30 13:02 pg_ident.conf
drwx-----  4 postgres postgres 4096 Mar 30 13:02 pg_logical/
drwx-----  4 postgres postgres 4096 Mar 30 13:02 pg_multixact/
drwx-----  2 postgres postgres 4096 Mar 30 13:02 pg_notify/
drwx-----  2 postgres postgres 4096 Mar 30 13:02 pg_replslot/
drwx-----  2 postgres postgres 4096 Mar 30 13:02 pg_serial/
drwx-----  2 postgres postgres 4096 Mar 30 13:02 pg_snapshots/
drwx-----  2 postgres postgres 4096 Mar 30 13:02 pg_stat/
drwx-----  2 postgres postgres 4096 Mar 30 13:02 pg_stat_tmp/
drwx-----  2 postgres postgres 4096 Mar 30 13:02 pg_subtrans/
drwx-----  2 postgres postgres 4096 Mar 30 13:02 pg_tblspc/
drwx-----  2 postgres postgres 4096 Mar 30 13:02 pg_twophase/
-rw-----  1 postgres postgres   3 Mar 30 13:02 PG_VERSION
drwx-----  3 postgres postgres 4096 Mar 30 13:02 pg_wal/
drwx-----  2 postgres postgres 4096 Mar 30 13:02 pg_xact/
-rw-----  1 postgres postgres  88 Mar 30 13:02 postgresql.auto.conf
-rw-----  1 postgres postgres 26642 Mar 30 13:02 postgresql.conf
-rw-----  1 postgres postgres  61 Mar 30 13:02 postmaster.opts
```

5. Меняем порт с 5432 на 5433 в папке /var/lib/pgsql/12/main5433

```
postgres$ sed -i 's/#port = 5432/port = 5433/'  
/var/lib/pgsql/12/main5433/postgresql.conf
```

6. Запускаем кластера

```
postgres$ pg_ctl -D /var/lib/pgsql/12/main5432 start  
waiting for server to start....2021-03-30 13:11:29.637 UTC [20448] LOG: starting  
PostgreSQL 12.6 on x86_64-pc-linux-gnu, compiled by gcc (GCC) 4.8.5 20150623 (Red Hat  
4.8.5-44), 64-bit  
2021-03-30 13:11:29.637 UTC [20448] LOG: listening on IPv4 address "127.0.0.1", port  
5432  
2021-03-30 13:11:29.641 UTC [20448] LOG: listening on Unix socket  
"/var/run/postgresql/.s.PGSQL.5432"  
2021-03-30 13:11:29.647 UTC [20448] LOG: listening on Unix socket  
"/tmp/.s.PGSQL.5432"  
2021-03-30 13:11:29.660 UTC [20448] LOG: redirecting log output to logging collector  
process  
2021-03-30 13:11:29.660 UTC [20448] HINT: Future log output will appear in directory  
"log".  
done  
server started
```

```
postgres$ pg_ctl -D /var/lib/pgsql/12/main5433 start  
waiting for server to start....2021-03-30 13:12:01.327 UTC [20464] LOG: starting  
PostgreSQL 12.6 on x86_64-pc-linux-gnu, compiled by gcc (GCC) 4.8.5 20150623 (Red Hat  
4.8.5-44), 64-bit  
2021-03-30 13:12:01.327 UTC [20464] LOG: listening on IPv4 address "127.0.0.1", port  
5433  
2021-03-30 13:12:01.361 UTC [20464] LOG: listening on Unix socket  
"/var/run/postgresql/.s.PGSQL.5433"  
2021-03-30 13:12:01.442 UTC [20464] LOG: listening on Unix socket  
"/tmp/.s.PGSQL.5433"  
2021-03-30 13:12:01.454 UTC [20464] LOG: redirecting log output to logging collector  
process  
2021-03-30 13:12:01.454 UTC [20464] HINT: Future log output will appear in directory  
"log".  
done  
server started
```

7. Проверяем работу серверов

```
postgres$ pg_ctl -D /var/lib/pgsql/12/main5432 status  
pg_ctl: server is running (PID: 20448)  
/usr/pgsql-12/bin/postgres "-D" "/var/lib/pgsql/12/main5432"  
postgres$ pg_ctl -D /var/lib/pgsql/12/main5433 status  
pg_ctl: server is running (PID: 20464)  
/usr/pgsql-12/bin/postgres "-D" "/var/lib/pgsql/12/main5433"
```

Физическая резервная копия. Горячее резервирование

8. Горячее резервирование можно выполнить только с помощью специальных утилит. Сейчас рассмотрим `pg_basebackup`.

В нашем случае и сервер-источник, и резервная копия будут располагаться на одном сервере. Поскольку мы собираемся тут же развернуть новый сервер, выбираем формат `plain`, а в качестве каталога для сохранения используем `PGDATA` целевого сервера.

Остановим PostgreSQL на порту 5433

```
postgres$ pg_ctl -D /var/lib/pgsql/12/main5433 stop  
waiting for server to shut down.... done
```

```
server stopped
```

9. Удаляем каталог `/var/lib/pgsql/12/main5433`

```
postgres$ rm -rf /var/lib/pgsql/12/main5433/*
```

10. Если не передавать дополнительные параметры, то `pg_basebackup` будет ждать очередную контрольную точку. Параметры следующей контрольной точки определяются двумя

параметрами `checkpoint_completion_target` и `checkpoint_timeout`.

```
postgres$ psql
5432=> \x
5432=> select name, setting, unit from pg_settings where name in
('checkpoint_completion_target','checkpoint_timeout');
-[ RECORD 1 ]-----
name      | checkpoint_completion_target
setting   | 0.5
unit      |
-[ RECORD 2 ]-----
name      | checkpoint_timeout
setting   | 300
unit      | s
```

Это может занять существенное время: если контрольные точки выполняются по расписанию.

11. Создадим резервную копию

```
postgres$ pg_basebackup -U postgres --pgdata=/var/lib/pgsql/12/main5433
```

Если требуется выполнить контрольную точку как можно быстрее, надо указать ключ `--checkpoint=fast`.

Сам резервный сервер уже предварительно собран и установлен.

12. Проверим содержимое каталога с данными, которое было записано `pg_basebackup`:

```
postgres$ ls -la /var/lib/pgsql/12/main5433
total 136
drwx----- 20 postgres postgres 4096 Mar 30 13:38 ./
drwx-----  6 postgres postgres 4096 Mar 30 12:59 ../
-rw-----  1 postgres postgres  224 Mar 30 13:38 backup_label
drwx-----  6 postgres postgres 4096 Mar 30 13:38 base/
-rw-----  1 postgres postgres   30 Mar 30 13:38 current_logfiles
drwx-----  2 postgres postgres 4096 Mar 30 13:38 global/
drwx-----  2 postgres postgres 4096 Mar 30 13:38 log/
drwx-----  2 postgres postgres 4096 Mar 30 13:38 pg_commit_ts/
drwx-----  2 postgres postgres 4096 Mar 30 13:38 pg_dynshmem/
-rw-----  1 postgres postgres 4760 Mar 30 13:38 pg_hba.conf
-rw-----  1 postgres postgres 1636 Mar 30 13:38 pg_ident.conf
drwx-----  4 postgres postgres 4096 Mar 30 13:38 pg_logical/
drwx-----  4 postgres postgres 4096 Mar 30 13:38 pg_multixact/
drwx-----  2 postgres postgres 4096 Mar 30 13:38 pg_notify/
drwx-----  2 postgres postgres 4096 Mar 30 13:38 pg_replslot/
drwx-----  2 postgres postgres 4096 Mar 30 13:38 pg_serial/
drwx-----  2 postgres postgres 4096 Mar 30 13:38 pg_snapshots/
drwx-----  2 postgres postgres 4096 Mar 30 13:38 pg_stat/
drwx-----  2 postgres postgres 4096 Mar 30 13:38 pg_stat_tmp/
```

```

drwx----- 2 postgres postgres 4096 Mar 30 13:38 pg_subtrans/
drwx----- 2 postgres postgres 4096 Mar 30 13:38 pg_tblspc/
drwx----- 2 postgres postgres 4096 Mar 30 13:38 pg_twophase/
-rw----- 1 postgres postgres   3 Mar 30 13:38 PG_VERSION
drwx----- 3 postgres postgres 4096 Mar 30 13:38 pg_wal/
drwx----- 2 postgres postgres 4096 Mar 30 13:38 pg_xact/
-rw----- 1 postgres postgres  88 Mar 30 13:38 postgresql.auto.conf
-rw----- 1 postgres postgres 26642 Mar 30 13:38 postgresql.conf

```

13. Все необходимые файлы журнала находятся в каталоге pg_wal:

```

postgres$ ls -l /var/lib/pgsql/12/main5433/pg_wal
total 16396
drwx----- 3 postgres postgres 4096 Mar 30 13:38 ./
drwx----- 20 postgres postgres 4096 Mar 30 13:38 ../
-rw----- 1 postgres postgres 16777216 Mar 30 13:38 00000001000000000000000002
drwx----- 2 postgres postgres 4096 Mar 30 13:38 archive_status/

```

Восстановление из базовой резервной копии

14. Так как на порту 5432 у нас уже есть инстанс PostgreSQL, то меняем порт с 5432 на порт 5433

```

postgres$ sed -i 's/#port = 5432/port = 5433/'
/var/lib/pgsql/12/main5433/postgresql.conf

```

15. Запускаем кластер

```

postgres$ pg_ctl -D /var/lib/pgsql/12/main5433 start
waiting for server to start....2021-03-30 13:41:40.252 UTC [20827] LOG:  starting
PostgreSQL 12.6 on x86_64-pc-linux-gnu, compiled by gcc (GCC) 4.8.5 20150623 (Red Hat
4.8.5-44), 64-bit
2021-03-30 13:41:40.252 UTC [20827] LOG:  listening on IPv4 address "127.0.0.1", port
5433
2021-03-30 13:41:40.256 UTC [20827] LOG:  listening on Unix socket
"/var/run/postgresql/.s.PGSQL.5433"
2021-03-30 13:41:40.263 UTC [20827] LOG:  listening on Unix socket
"/tmp/.s.PGSQL.5433"
2021-03-30 13:41:40.276 UTC [20827] LOG:  redirecting log output to logging collector
process
2021-03-30 13:41:40.276 UTC [20827] HINT:  Future log output will appear in directory
"log".
done
server started

```

16. Проверяем статус кластера

```

postgres$ pg_ctl -D /var/lib/pgsql/12/main5433 status
pg_ctl: server is running (PID: 20827)
/usr/pgsql-12/bin/postgres "-D" "/var/lib/pgsql/12/main5433"

```

17. Теперь оба сервера работают одновременно и независимо. Проверим:

```

postgres$ psql -p 5432 -l

```

List of databases					
Name	Owner	Encoding	Collate	Ctype	Access privileges
dbcopy1	postgres	UTF8	en_US.utf8	en_US.utf8	
postgres	postgres	UTF8	en_US.utf8	en_US.utf8	
student	postgres	UTF8	en_US.utf8	en_US.utf8	
template0	postgres	UTF8	en_US.utf8	en_US.utf8	=c/postgres + postgres=CTc/postgres
template1	postgres	UTF8	en_US.utf8	en_US.utf8	=c/postgres +

```

(5 rows) | | | | | postgres=CtC/postgres

postgres$ psql -p 5433 -l
                                List of databases
  Name      | Owner   | Encoding | Collate  | Ctype    | Access privileges
-----+-----+-----+-----+-----+-----
 dbcopy1    | postgres | UTF8     | en_US.utf8 | en_US.utf8 |
 postgres   | postgres | UTF8     | en_US.utf8 | en_US.utf8 |
 student    | postgres | UTF8     | en_US.utf8 | en_US.utf8 |
 template0  | postgres | UTF8     | en_US.utf8 | en_US.utf8 | =c/postgres      +
            |          |          |          |          | postgres=CtC/postgres
 template1  | postgres | UTF8     | en_US.utf8 | en_US.utf8 | =c/postgres      +
            |          |          |          |          | postgres=CtC/postgres
(5 rows)

```

3.3

Домашнее задание.

1. Остановите кластер на порту 5433. Удалите каталог данных по пути `/var/lib/pgsql/12/main5433`.
2. Выполните резервную копию сервера 5432 с помощью утилиты `rsync`. Предварительно выполнив команду `pg_start_backup`
3. После завершения копирования выполните `pg_stop_backup`
4. Поменяйте порт в конфигурационном файле `/var/lib/pgsql/12/main5433/postgresql.conf` с 5432 на 5433.
5. Запустите кластер на порту 5433

Чек-лист

1. Кластер на порту 5433 был остановлен и удален каталог данных `/var/lib/pgsql/12/main5433`.
2. В одном сеансе была выполнена команда `pg_start_backup`. В другом выполнена команда копирования каталога `/var/lib/pgsql/12/main5432` в `/var/lib/pgsql/12/main5433`
3. После завершения копирования была выполнена команда `pg_stop_backup`.
4. Порт в конфигурационном файле `/var/lib/pgsql/12/main5433/postgresql.conf` был изменен с 5432 на 5433.
5. Кластер на порту 5433 запустился

Решение:

1. Остановите кластер на порту 5433. Удалите каталог данных по пути `/var/lib/pgsql/12/main5433`.

```
postgres$ pg_ctl -D /var/lib/pgsql/12/main5433 stop
waiting for server to shut down.... done
server stopped
```

```
postgres$ rm -R ./12/main5433
```

2. Выполните резервную копию сервера 5432 с помощью утилиты rsync.
Предварительно выполнив команду pg_start_backup

```
5432=>$ SELECT pg_start_backup('rsync_backup', true, false);
pg_start_backup
-----
 0/E000028
(1 row)
```

ОТКРЫВАЕМ НОВУЮ СЕССИЮ!!!

Выполняем следующие команды:

```
postgres$ rsync -r "/var/lib/pgsql/12/main5432/"
"/var/lib/pgsql/12/main5433/"
postgres$ chmod 700 /var/lib/pgsql/12/main5433
postgres$ rm ./12/main5433/postmaster.pid
```

3. После завершения копирования выполните pg_stop_backup **СТАРАЯ СЕССИЯ!!!**

```
5432=>$ select pg_stop_backup(false, true);
NOTICE: WAL archiving is not enabled; you must ensure that all required WAL
segments are copied through other means to complete the backup
          pg_stop_backup
-----
(0/E000138,"START WAL LOCATION: 0/E000028 (file 000000010000000000000000E)+
CHECKPOINT LOCATION: 0/E000060
BACKUP METHOD: streamed
BACKUP FROM: master
START TIME: 2021-04-05 13:20:31 UTC
LABEL: rsync_backup
START TIMELINE: 1
", "")
(1 row)
5432=> \q
```

4. Поменяйте порт в конфигурационном файле
/var/lib/pgsql/12/main5433/postgresql.conf с 5432 на 5433.

```
postgres$ echo port=5433 > ./12/main5433/postgresql.auto.conf
```

5. Запустите кластер на порту 5433

```
postgres$ pg_ctl -D /var/lib/pgsql/12/main5433 start
waiting for server to start....2021-04-05 13:31:48.409 UTC [5701] LOG:
starting PostgreSQL 12.6 on x86_64-pc-linux-gnu, compiled by gcc (GCC) 4.8.5
20150623 (Red Hat 4.8.5-44), 64-bit
2021-04-05 13:31:48.409 UTC [5701] LOG:  listening on IPv4 address
"127.0.0.1", port 5433
2021-04-05 13:31:48.413 UTC [5701] LOG:  listening on Unix socket
"/var/run/postgresql/.s.PGSQL.5433"
2021-04-05 13:31:48.421 UTC [5701] LOG:  listening on Unix socket
"/tmp/.s.PGSQL.5433"
```

2021-04-05 13:31:48.439 UTC [5701] LOG: redirecting log output to logging collector process
2021-04-05 13:31:48.439 UTC [5701] HINT: Future log output will appear in directory "log".
done
server started