

НИУ ИТМО

Факультет программной инженерии и компьютерных технологий

Отчет по лабораторной работе №3

по дисциплине Распределенные системы хранения данных

Студент группы № Р33151

Шипулин Павел Андреевич

Преподаватель

Перцев Тимофей Сергеевич

Санкт-Петербург

2024

Задание

Цель работы – настроить процедуру периодического резервного копирования базы данных, сконфигурированной в ходе выполнения лабораторной работы №2, а также разработать и отладить сценарии восстановления в случае сбоев.

Узел из предыдущей лабораторной работы используется в качестве основного. Новый узел используется в качестве резервного. Учётные данные для подключения к новому узлу выдаёт преподаватель. В сценариях восстановления необходимо использовать копию данных, полученную на первом этапе данной лабораторной работы.

Требования к отчёту

Отчет должен быть самостоятельным документом (без ссылок на внешние ресурсы), содержать всю последовательность команд и исходный код скриптов по каждому пункту задания. Для демонстрации результатов приводить команду вместе с выводом (самой наглядной частью вывода, при необходимости).

Этап 1. Резервное копирование

- Настроить резервное копирование с основного узла на резервный следующим образом:
 - Периодические холодные полные копии.
 - Полная копия (rsync) по расписанию (cron) раз в сутки. СУБД на время копирования должна останавливаться. На резервном узле хранить 14 копий, после успешного создания пятнадцатой копии, самую старую автоматически уничтожать.

- Подсчитать, каков будет объем резервных копий спустя месяц работы системы, исходя из следующих условий:
 - Средний объем новых данных в БД за сутки: 750МБ.
 - Средний объем измененных данных за сутки: 300МБ.
- Проанализировать результаты.

Этап 2. Потеря основного узла

Этот сценарий подразумевает полную недоступность основного узла. Необходимо восстановить работу СУБД на РЕЗЕРВНОМ узле, продемонстрировать успешный запуск СУБД и доступность данных.

Этап 3. Повреждение файлов БД

Этот сценарий подразумевает потерю данных (например, в результате сбоя диска или файловой системы) при сохранении доступности основного узла. Необходимо выполнить полное восстановление данных из резервной копии и перезапустить СУБД на ОСНОВНОМ узле.

Ход работы:

- Симулировать сбой:
 - удалить с диска директорию любой таблицы со всем содержимым.
- Проверить работу СУБД, доступность данных, перезапустить СУБД, проанализировать результаты.
- Выполнить восстановление данных из резервной копии, учитывая следующее условие:
 - исходное расположение директории PGDATA недоступно - разместить данные в другой директории и скорректировать конфигурацию.

- Запустить СУБД, проверить работу и доступность данных, проанализировать результаты.

Этап 4. Логическое повреждение данных

Этот сценарий подразумевает частичную потерю данных (в результате нежелательной или ошибочной операции) при сохранении доступности основного узла. Необходимо выполнить восстановление данных на ОСНОВНОМ узле следующим способом:

- Восстановление с использованием архивных WAL файлов. (СУБД должна работать в режиме архивирования WAL, потребуется задать параметры восстановления).

Ход работы:

- В каждую таблицу базы добавить 2-3 новые строки, зафиксировать результат.
- Зафиксировать время и симулировать ошибку:
 - удалить любые две таблицы (DROP TABLE)
- Продемонстрировать результат.
- Выполнить восстановление данных указанным способом.
- Продемонстрировать и проанализировать результат.

Выполнение

Этап 1. Резервное копирование

Настройка резервного копирования

Файл cold_copy.sh на основном узле

```
1  reserve_node='postgres2@pg191';
2  now_date=$(date "+%Y-%m-%d_%H-%M-%S");
3  new_backup="backup_at_${now_date}";
4
5  pg_ctl stop -D ~/yqi56;
6
7  rsync -avv ~/yqi56 ~/uzb16 ~/mwd84 ~/orw97 $reserve_node:~/backups/$new_backup;
8
9  pg_ctl start -D ~/yqi56;
10
11 ssh $reserve_node "bash ~/remove_old_backups.sh";
12
13 echo "Скрипт холодного копирования завершил работу";
```

Файл remove_old_backups.sh на дополнительном узле

```
1  copies_count=`ls backups/ | grep backup_at_ | wc -l`;
2  max_copies_count=14;
3  remove_count=$(( $copies_count - $max_copies_count ));
4
5  echo "Количество холодных копий: $(( $copies_count ))";
6
7  if (( $remove_count <= 0 )); then
8      echo "Еще можно создать копий: $(( -$remove_count ))"
9  else
10     echo "Количество копий для удаления: ${remove_count}";
11
12     while (( $remove_count > 0 )); do
13         oldest_copy=`ls backups/ | grep backup_at_ | head -1`;
14
15         rm -rf backups/$oldest_copy &&
16         echo "Копия ${oldest_copy} успешно удалена" ||
17         echo "Не удалось удалить копию ${oldest_copy}";
18
19         remove_count=$(( $remove_count-- ))
20     done;
21 fi;
22
23 echo "Скрипт удаления старых копий завершил работу";
```

Подсчет объема

Так как при создании резервной копии происходит копирование файлов БД целиком, то объем измененных данных не будет влиять на объем таких копий.

По условию:

- прирост объема данных в сутки $V = 750$ МБ;
- максимальное количество копий равно 14;
- копии создаются раз в сутки;
- требуется подсчитать объем за месяц.

Объем информации на хранилище (дни от n до m) можно найти как сумму арифметической прогрессии:

$$a_1 = V$$

$$a_2 = a_1 + V$$

$$a_k = a_{k-1} + V = a_1 + (k - 1)V$$

$$S_{1,k} = \frac{a_1 + a_k}{2} \cdot k$$

$$S_{n,m} = S_{1,m} - S_{1,n-1}, m > n$$

Так как максимальное количество копий равно 14, то

$$m - n + 1 = 14$$

$$n = m - 13$$

Номер последней копии будет равен номеру дня за месяц, пусть будет 30. Значит

$$m = 30$$

$$n = 30 - 13 = 17$$

Объем данных резервных копий с номерами от 17 до 30 включительно:

$$S_{n,m} = S_{17,30} = S_{1,30} - S_{1,16} = \frac{V + V + 29V}{2} \cdot 30 - \frac{V + V + 15V}{2} \cdot 16 =$$

$$= \frac{31}{2}V \cdot 30 + \frac{17V}{2} \cdot 16 = (13 \cdot 15 + 17 \cdot 8) \cdot 750\text{МБ} = 248250\text{МБ}$$

$$\approx 242,432 \text{ ГБ}$$

Анализ

При физическом копировании, обновление данных не влияет на размер копии, поэтому этот вид резервного копирования полезен, когда значение объема обновленных данных мало. Хотя даже при небольшом приросте данных в сутки, объем всех резервных копий может достигать слишком больших значений.

Этап 2. Потеря основного узла

Восстановление базы данных на резервном узле

Файл run_last_backup.sh на резервном узле

```

1 last_backup=`ls backups/ | grep backup_at_ | tail -1`;
2
3 cp -r backups/$last_backup/* ~/;
4
5 cd yqi56/pg_tblspc;
6
7 ln -s ../../uzb16 16388;
8 ln -s ../../mwd84 16389;
9 ln -s ../../orw97 16390;
10
11 cd ../../;
12
13 pg_ctl start -D yqi56/ || pg_ctl restart -D yqi56/;
```

```

[postgres2@pg191 ~]$ ./run_last_backup.sh
cp: backups/backup_at_2024-05-10_12-27-12/uzb16/uzb16: No such file or directory
cp: backups/backup_at_2024-05-10_12-27-12/yqi56/pg_tblspc/16388: Permission denied
cp: backups/backup_at_2024-05-10_12-27-12/yqi56/pg_tblspc/16389: Permission denied
cp: backups/backup_at_2024-05-10_12-27-12/yqi56/pg_tblspc/16390: Permission denied
ожидание запуска сервера...2024-05-10 12:30:05.114 MSK [23683] СООБЩЕНИЕ: передача вывода в протокол процессу сбора протоколов
2024-05-10 12:30:05.114 MSK [23683] ПОДСКАЗКА: В дальнейшем протоколы будут выводиться в каталог "log".
готово
сервер запущен
```

Так как в директории pg_tblspc кластера БД хранятся символические ссылки на табличные пространства, а операция копирования cp не смогла

скопировать эти файлы (нет доступа) – их необходимо создать заново для каждого табличного пространства.

Демонстрация доступности данных

```
[postgres2@pg191 ~]$ psql -p 9142 -h localhost -d sickorangecity -U s311817
Пароль пользователя s311817:
psql (14.2)
Введите "help", чтобы получить справку.
```

```
sickorangecity=> \dt+
```

Список отношений							
Схема	Имя	Тип	Владелец	Хранение	Метод доступа	Размер	Описание
public	building	таблица	postgres6	постоянное	heap	16 kB	
public	sickorange	таблица	postgres6	постоянное	heap	16 kB	
public	street	таблица	postgres6	постоянное	heap	16 kB	

(3 строки)


```
sickorangepcity=> select * from building;
```

id	street_id	name
1	1	Building132763172
2	1	Building18237
3	1	Building9812376
4	1	Building1236712
5	2	Building99999
6	2	Building182620
7	2	Building87123671
8	3	Building11111111
9	3	Building331823782
10	3	Building323232323
11	3	Building8718273
12	3	Building9919237
13	3	Building3267666
14	4	Building8129362
15	4	Building993827
16	4	Building38382173

(16 строк)

```
sickorangepcity=> select * from sickorange;
```

id	name	birthdate
1	SickOrange3812536	2000-01-01
2	SickOrange812738	2001-02-01
3	SickOrange1234512	2000-01-02
4	SickOrange94721	2003-03-03
5	SickOrange712637	2000-11-11
6	SickOrange761273	2006-11-09
7	SickOrange9812378	2000-08-28
8	SickOrange761237	2004-07-17

(8 строк)

```
sickorangepcity=> select * from street;
```

id	name
1	Main
2	Branch13231
3	Branch3211
4	Branch98392

(4 строки)

Этап 3. Повреждение файлов БД

Симуляция сбоя

Удаление с диска директории таблицы “street” со всем содержимым

```
[postgres6@pg100 ~]$ rm -rf uzb16/
[postgres6@pg100 ~]$ ls -l
total 15
-rwxr--r--  1 postgres6 postgres6 386 10 мая 12:02 cold_copy.sh
drwxr-xr-x  2 postgres6 postgres6  5 26 апр. 17:03 lab2
drwx-----  3 postgres6 postgres6  3 26 апр. 15:57 mwd84
drwx-----  3 postgres6 postgres6  3 26 апр. 15:57 orw97
drwx----- 20 postgres6 postgres6 30 10 мая 13:21 yqi56
```

Проверка работы СУБД

```
[postgres6@pg100 ~]$ psql -p 9142 -h localhost -d sickorangecity -U s311817
Пароль пользователя s311817:
psql (14.2)
Введите "help", чтобы получить справку.

sickorangecity=> \dt
          Список отношений
Схема | Имя      | Тип  | Владелец
-----+-----+-----+-----
public | building | таблица | postgres6
public | sickorange | таблица | postgres6
public | street   | таблица | postgres6
(3 строки)

sickorangecity=> select * from street;
ОШИБКА: не удалось открыть файл "pg_tblspc/16388/PG_14_202107181/16384/16445": No such file or directory
```

Восстановление

Файл load_and_run_last_backup.sh на основном узле

```
1  reserve_node='postgres2@pg191';
2  last_backup=`ssh ${reserve_node} "ls backups/ | grep backup_at_ | tail -1"`;
3  db_copy_dir='yqi56'
4  db_put_dir='yqi99'
5
6  echo "Загрузка резервной копии: ${last_backup}"
7
8  mkdir $db_put_dir;
9  rsync -avv $reserve_node:~/backups/$last_backup/$db_copy_dir/ $db_put_dir &&
10 echo "Директория основных файлов БД загружена" ||
11 echo "Не удалось загрузить директорию основных файлов БД";
12
13 mkdir uzb16;
14 rsync -avv $reserve_node:~/backups/$last_backup/uzb16/ uzb16 &&
15 echo "Директория файлов таблицы 'street' загружена" ||
16 echo "Не удалось загрузить директорию файлов таблицы 'street'";
17
18 pg_ctl start -D $db_put_dir/ || pg_ctl restart -D $db_put_dir/;
```

```
[postgres6@pg100 ~]$ psql -p 9142 -d sickorangecity
psql (14.2)
Введите "help", чтобы получить справку.

sickorangecity=# \dt
                Список отношений
 Схема |      Имя      | Тип   | Владелец 
-----+-----+-----+-----
 public | building      | таблица | postgres6
 public | sickorange    | таблица | postgres6
 public | street        | таблица | postgres6
(3 строки)

sickorangecity=# select * from street;
 id |      name 
----+-----
  1 | Main
  2 | Branch13231
  3 | Branch3211
  4 | Branch98392
(4 строки)
```

Этап 4. Логическое повреждение данных

Добавление новых данных

```
1 insert into sickorange values (default, 'Step4_SickOrange91238', '04/05/2006');
2 insert into sickorange values (default, 'Step4_SickOrange1131', '09/08/2007');
3
4 insert into street values (default, 'Step4_Branch1321');
5 insert into street values (default, 'Step4_Branch666');
6
7 insert into building values (default, 5, 'Step4_Building3212');
8 insert into building values (default, 6, 'Step4_Building1232');
```

Фиксация времени и симуляция ошибки

```
sickorangepcity=# \i lab3/insert.sql
INSERT 0 1
INSERT 0 1
INSERT 0 1
INSERT 0 1
INSERT 0 1
INSERT 0 1
sickorangepcity=# select * from sickorange;
 id |          name          | birthdate
-----+-----+-----
  1 | SickOrange3812536     | 2000-01-01
  2 | SickOrange812738      | 2001-02-01
  3 | SickOrange1234512     | 2000-01-02
  4 | SickOrange94721       | 2003-03-03
  5 | SickOrange712637      | 2000-11-11
  6 | SickOrange761273      | 2006-11-09
  7 | SickOrange9812378     | 2000-08-28
  8 | SickOrange761237      | 2004-07-17
  9 | Step4_SickOrange91238 | 2006-05-04
 10 | Step4_SickOrange1131  | 2007-08-09
(10 строк)
```

```
sickorangepcity=# select now();
          now
-----
2024-05-24 15:44:35.260532+03
(1 строка)
```

```
sickorangepcity=# drop table building, sickorange;
DROP TABLE
```

Результат выполнения ошибки

```
sickorangecity=# \dt
                Список отношений
 Схема  |  Имя   |  Тип   | Владелец
-----+-----+-----+-----
 public | street | таблица | postgres6
(1 строка)
```

Восстановление

```
[postgres6@pg100 ~]$ ./load_and_run_last_backup.sh
Загрузка резервной копии: backup_at_2024-05-10_16-54-08
mkdir: yqi56: File exists
opening connection using: ssh -l postgres2 pg191 rsync --server --sender -vvlogDtpre
rgs)
receiving incremental file list
[sender] expand file_list pointer array to 1024 bytes, did move
[sender] expand file_list pointer array to 4096 bytes, did move
[sender] expand file_list pointer array to 1024 bytes, did move
[sender] expand file_list pointer array to 4096 bytes, did move
[sender] expand file_list pointer array to 1024 bytes, did move
[sender] expand file_list pointer array to 4096 bytes, did move
[sender] expand file_list pointer array to 1024 bytes, did move
[sender] expand file_list pointer array to 4096 bytes, did move
delta-transmission enabled
PG_VERSION is uptodate
pg_hba.conf is uptodate
```

Загрузка последней копии с резервного узла

```
[postgres6@pg100 ~]$ ls -l yqi56/ | grep recover
-rw-r--r--  1 postgres6  postgres        0 24 мая   15:49 recovery.signal
```

Создание файла recovery.signal в PGDATA

```
# - Recovery Target -

# Set these only when performing a targeted recovery.

#recovery_target = ''          # 'immediate' to end recovery as soon as a
                                # consistent state is reached
                                # (change requires restart)
#recovery_target_name = ''     # the named restore point to which recovery will proceed
                                # (change requires restart)
recovery_target_time = '2024-05-24 15:44:35.260532+03' # the time stamp up to which recovery will proceed
                                # (change requires restart)
#recovery_target_xid = ''      # the transaction ID up to which recovery will proceed
                                # (change requires restart)
```

Изменение параметра `recovery_target_time` в `postgresql.conf`.

```
[postgres@pg100 ~]$ pg_ctl start -D yqi56
ожидание запуска сервера...2024-05-24 16:10:42.524 MSK [94078] СООБЩЕНИЕ:  передача вывода в протокол процессу сбора протоколов
2024-05-24 16:10:42.524 MSK [94078] ПОДСКАЗКА:  В дальнейшем протоколы будут выводиться в каталог "log".
    готово
сервер запущен

[postgres@pg100 ~]$ cat yqi56/log/postgresql-2024-05-24_161042.log
2024-05-24 16:10:42.524 MSK [94078] СООБЩЕНИЕ:  запускается PostgreSQL 14.2 on amd64-portbld-freebsd13.0, compiled by P
github.com:llvm/llvm-project.git llvmorg-11.0.1-0-g43ff75f2c3fe), 64-bit
2024-05-24 16:10:42.524 MSK [94078] СООБЩЕНИЕ:  для приёма подключений по адресу IPv6 ":::" открыт порт 9142
2024-05-24 16:10:42.524 MSK [94078] СООБЩЕНИЕ:  для приёма подключений по адресу IPv4 "0.0.0.0" открыт порт 9142
2024-05-24 16:10:42.536 MSK [94078] СООБЩЕНИЕ:  для приёма подключений открыт Unix-сокет "/tmp/.s.PGSQL.9142"
2024-05-24 16:10:42.562 MSK [94080] СООБЩЕНИЕ:  система БД была выключена: 2024-05-10 16:54:08 MSK
cp: /var/db/postgres6/lab3/pg_wal/00000002.history: No such file or directory
2024-05-24 16:10:42.564 MSK [94080] СООБЩЕНИЕ:  начинается восстановление точки во времени до 2024-05-24 15:44:35.26053
2024-05-24 16:10:42.565 MSK [94080] СООБЩЕНИЕ:  файл журнала "0000000100000000000000010" восстановлен из архива
2024-05-24 16:10:42.601 MSK [94080] СООБЩЕНИЕ:  согласованное состояние восстановления достигнуто по смещению 0/1000036
2024-05-24 16:10:42.601 MSK [94080] СООБЩЕНИЕ:  запись REDO начинается со смещения 0/10000360
2024-05-24 16:10:42.602 MSK [94080] СООБЩЕНИЕ:  восстановление останавливается перед фиксированием транзакции 792, вре
2024-05-24 16:10:42.602 MSK [94080] СООБЩЕНИЕ:  остановка в конце восстановления
2024-05-24 16:10:42.602 MSK [94080] ПОДСКАЗКА:  Выполните pg_wal_replay_resume() для повышения.
2024-05-24 16:10:42.603 MSK [94078] СООБЩЕНИЕ:  система БД готова принимать подключения в режиме "только чтение"
```

Результат восстановления

```
sickorangecity=# \dt
                Список отношений
 Schema |      Имя      | Тип   | Владелец
-----+-----+-----+-----
 public | building      | таблица | postgres
 public | sickorange    | таблица | postgres
 public | street        | таблица | postgres
(3 строки)
```

Таблицы восстановлены.

```
sickorangecity=# select * from sickorange;
 id |          name          | birthdate
----+-----+-----
  1 | SickOrange3812536     | 2000-01-01
  2 | SickOrange812738      | 2001-02-01
  3 | SickOrange1234512     | 2000-01-02
  4 | SickOrange94721       | 2003-03-03
  5 | SickOrange712637      | 2000-11-11
  6 | SickOrange761273      | 2006-11-09
  7 | SickOrange9812378     | 2000-08-28
  8 | SickOrange761237      | 2004-07-17
  9 | Step4_SickOrange91238 | 2006-05-04
 10 | Step4_SickOrange1131  | 2007-08-09
(10 строк)
```

Данные в таблицах соответствуют тому, что было перед их удалением.