**Санкт-Петербургский политехнический университет Петра Великого**

**Институт кибербезопасности и защиты информации**

**ЛАБОРАТОРНАЯ РАБОТА №4**

**Журнализация и восстановление**

по дисциплине «Основы построения защищенных баз данных»

Выполнил

студент гр. 4841001/00101 *<подпись>* А.С. Петушков

Проверил

преподаватель *<подпись>* М.А. Полтавцева

Санкт-Петербург  
2022

**Цель работы**

Изучить механизмы журнализации и восстановления в СУБД. Получить навыки обеспечения доступности баз данных и расследования инцидентов на основе логов.

**Задачи работы**

1. Зафиксируйте доступные характеристики журнала транзакций.
2. Оцените влияние расположения журнала и данных на производительность. Для этого:
   1. Определите местоположение файла журнала и файла данных СУБД. Разместите их на одном физическом диске.
   2. Выберите наиболее затратный, с точки зрения производительности (в том числе по памяти) запрос из предыдущих лабораторных работ (используйте запросы из лабораторной работы №1 для представлений) или составьте отдельный запрос для тестирования журнала. Запрос должен быть осмысленным с точки зрения предметной области. Сформируйте с ним транзакции:
3. Производительной выборки данных;
4. Выборки и обновления данных (объедините операции в одну транзакцию);
5. Вставки и удаления данных объедините операции в одну транзакцию).
   1. Оцените время выполнения транзакций (сделайте несколько тестов со случайными значениями нивелировав попадание в кэш и возьмите средние значения).
   2. Разместите файлы журнала и данных на разных физических дисках. Если транзакции используют разные отношения и СУБД позволяет разместить их на разных физических дисках, сделайте это (но не на диске с журналом).
   3. Повторите оценку времени. Сделайте выводы.
6. Оцените возможности по манипулированию журналом в СУБД.
   1. Определите после выполнения п.2 характеристики журнала. Сравните с предшествующими.
   2. (если это возможно) получите журнальные записи, соответствующие вашим транзакциям (1 набор на 1 транзакцию). Сделайте выводы о структуре индивидуального журнала и записи.
   3. Повторите п. 4.b для транзакции типа (2) пункт 3.b.ii, выполнив ее с откатом в конце. Сделайте выводы.
7. Оцените применение моделей восстановления. Для этого:
   1. Выполните (если это возможно) процедуру усечения журнала.
   2. Определите текущей режим резервирования.
   3. Выберите режим резервирования и настройте ведение контрольных точек. Аргументируйте.
   4. Создайте резервную копию БД.
   5. Выполните восстановление БД из резервной копии.

**Ход работы**

**Вариант №10.**

СУБД PostgreSQL ведет журнал предзаписи (WAL – Write Ahead Log). Изменения в файлах с данными (где находятся таблицы и индексы) должны записываться только после того, как эти изменения были занесены в журнал.

WAL хранится в директории

/var/lib/postgresql/11/main/pg\_wal

в виде набора файлов сегментов размером по 16Мб. На рисунке 1 приведено содержимое этой директории.

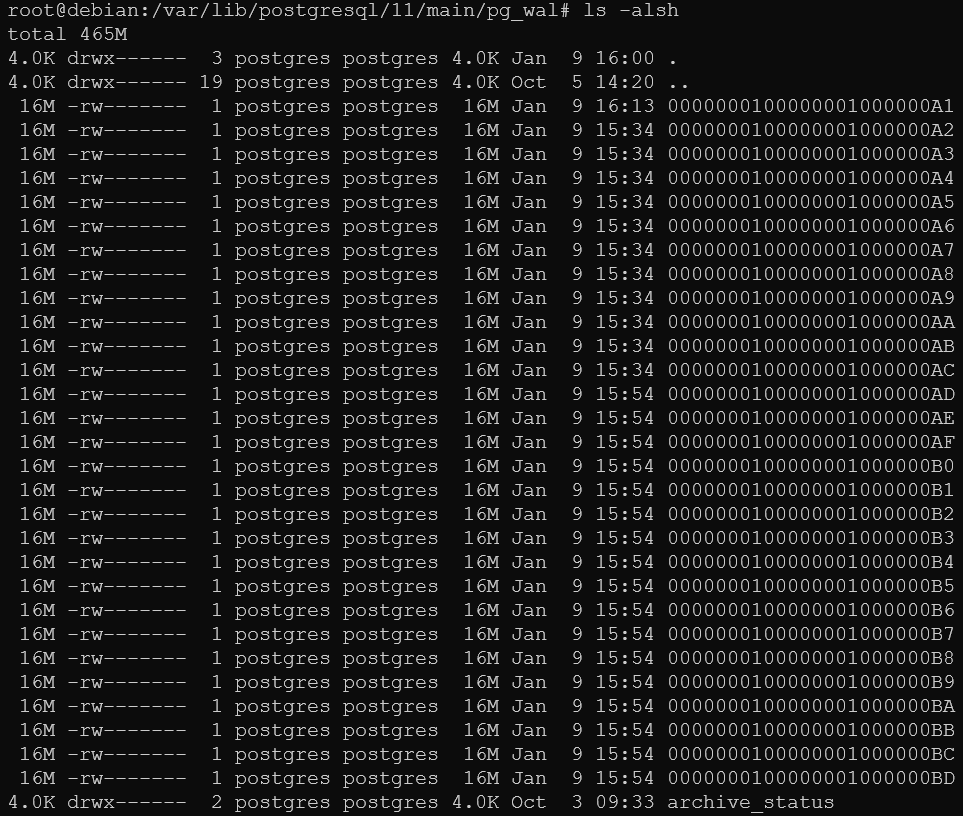


Рисунок 1 – Содержимое директории WAL

На рисунке 2 показан процесс получения информации об уровне ведения журнала.

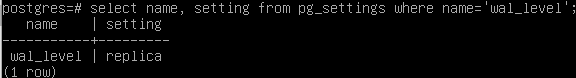


Рисунок 2 – Уровень ведения журнала

Видно, что текущий уровень ведения журнала – replica (по умолчанию).

Всего возможны три уровня:

1. minimal – гарантирует только восстановление после сбоя;
2. replica – гарантирует восстановление из резервной копии;
3. logical – обеспечивает возможность работы логического декодирования и логической репликации.

Для выполнения задач лабораторной работы текущий уровень ведения журнала подходит, поэтому изменен не будет.

Для удобного просмотра журнала используется утилита pg\_waldump, поставляемая вместе с СУБД. Для того чтобы узнать текущую позицию в журнале, используется запрос

SELECT pg\_current\_wal\_insert\_lsn();

Была выполнена следующая транзакция **выборки** данных:

BEGIN;

SELECT \* FROM "Gradebook"

WHERE "id\_students" = 2 AND "type" = 'Занятие';

COMMIT;

На его выполнение в среднем затрачивается 64 мс.

В журнал данная транзакция не записывается.

Была выполнена следующая транзакция **выборки и обновления** данных:

BEGIN;

SELECT \* FROM "Gradebook"

WHERE "id\_students" = 2 AND "type" = 'Занятие';

UPDATE "Courses" SET "price" = "price" + 5000;

COMMIT;

На его выполнение в среднем затрачивается 84 мс.

Записи журнала для данной транзакции представлены на рисунке 3.



Рисунок 3 – Записи в журнале для транзакции выборки и обновления

В журнал записана только операция обновления данных.

Была выполнена следующая транзакция **вставки и удаления** данных:

BEGIN;

INSERT INTO "Teachers" ("id", "username", "name", "speciality", "degree", "rating", "salary")

VALUES (7, 'kamenev', 'Каменев Семен Романович', 'Информационная безопасность', 'Доктор наук', 99, 220000);

DELETE FROM "Gradebook" WHERE "id\_students" = 1;

COMMIT;

На его выполнение в среднем затрачивается 84 мс.

Записи журнала для данной транзакции представлены на рисунке 4.

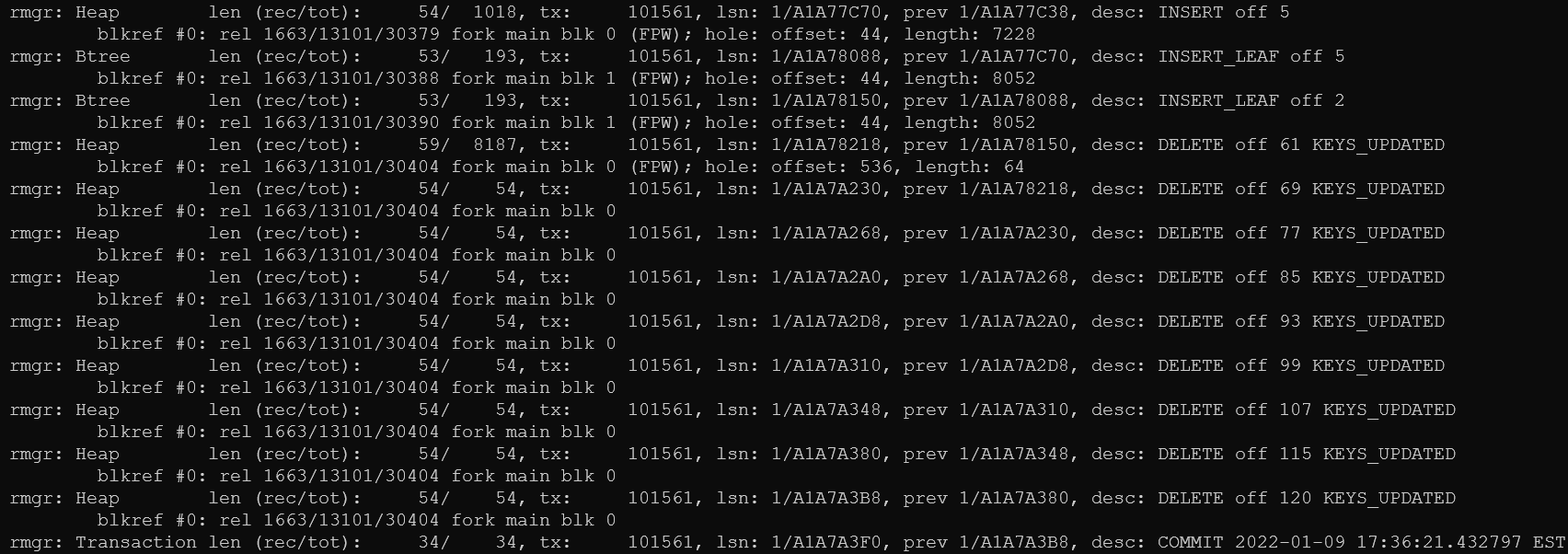


Рисунок 4 – Записи в журнале для транзакции добавления и удаления

В журнал записаны все операции.

Была выполнена транзакция **выборки и обновления** данных (такая же, как и ранее) с откатом в конце.

Записи журнала для данной транзакции представлены на рисунке 5.

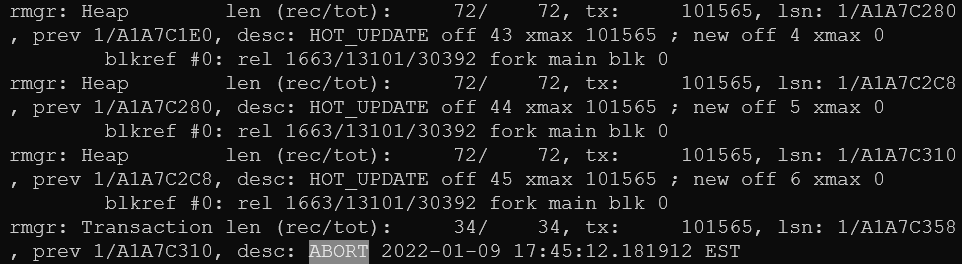


Рисунок 5 – Записи в журнале для отмененной транзакции выборки и обновления

На рисунке 5 выделено, что транзакция была отменена (если сравнить с обычной транзакцией, то у последней указано COMMIT – рисунок 3).

Аппаратное обеспечение машины, на которой выполнялась лабораторная работа, не позволяют поместить журнал отложенной записи на другом физическом диске. Однако в реальных системах это рекомендуется делать.

Для создания резервной копии БД используется утилита pg\_dump. Дамп может быть сохранен в файл для последующего восстановления.

Была создана резервная копия командой

$ pg\_dump>/tmp/backup.dump

После этого база данных была очищена.

Затем БД была успешно восстановлена из полученного дампа с помощью команды

$ psql < /tmp/backup.dump

**Вывод**

В результате выполнения данной лабораторной работы были изучены механизмы журнализации и восстановления в СУБД. Получены навыки обеспечения доступности баз данных и расследования инцидентов на основе логов на примере СУБД PostgreSQL.