|  |
| --- |
|  |
| МИНОБРНАУКИ РОССИИ  Федеральное государственное бюджетное образовательное учреждение  высшего образования  **«МИРЭА – Российский технологический университет»**  **РТУ МИРЭА** |

Институт комплексной безопасности и специального приборостроения

Кафедра КБ-14 «Интеллектуальные системы информационной безопасности»

### Политики безопасности баз данных

### Практическая работа № 3

# ОТЧЁТ

# Выполнил студент группы БСБО-07-20 Любовский С.В.

# Выполнение задания 1.

1. Создать базу данных с именем vacuum\_db.

**CREATE DATABASE vacuum\_db;**

1. Создать таблицу users, отключив параметр автоочистки (CREATE TABLE ... WITH (autovacuum\_enabled = off);),  со следующими полями:

id: уникальный идентификатор пользователя (integer, primary key, auto-increment).

username: имя пользователя (varchar(255)).

email: электронный адрес пользователя (varchar(255)).

category: категория (char(3))  
  
**CREATE TABLE users (**

**id serial primary key,**

**username varchar(255),**

**email varchar(255),**

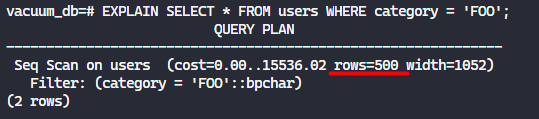
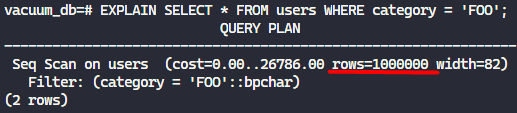
**category char(3)**

**) WITH (autovacuum\_enabled = off);**

1. Написать скрипт заполняющий таблицу users 1000000 рандомными записями, в поле category всегда должна находиться запись ‘FOO’.  
     
   **INSERT INTO users (username, email, category)**

**SELECT gen\_random\_uuid() as username, gen\_random\_uuid() as email, 'FOO'::char(3) AS category**

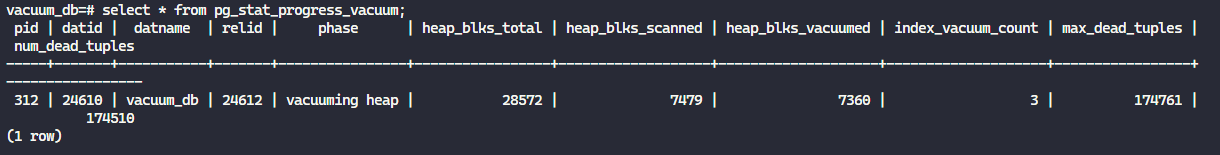
**FROM generate\_series(1,1000000);**

1. Используя оператор Explain выведите из таблицы users все записи, которые в поле category имеют значение ‘FOO’;  
     
   **EXPLAIN SELECT \* FROM users WHERE category = 'FOO';**Видно, что планировщик запросов сильно промахивается с оценкой количества строк.
2. Выполните команду ANALYZE;  
     
   **ANALYZE users;**
3. Используя оператор Explain выведите из таблицы users все записи, которые в поле category имеют значение ‘FOO’;  
     
   **EXPLAIN SELECT \* FROM users WHERE category = 'FOO';**После актуализации статистики, планировщик правильно оценивает стоимость запроса и количество строк.
4. Отличаются ли методы доступа к данным и почему?  
     
   Методы доступа к данным не отличаются, но планировщик грамотно оценивает стоимость запроса и количество строк в выборке.
5. Временно уменьшите значение maintenance\_work\_mem чтоб оно стало равно 1MB (не забудьте выполнить функцию pg\_reload\_conf())  
     
   **SET maintenance\_work\_mem to '1MB';**

**SELECT pg\_reload\_conf();**

1. Измените значение поля category на ‘BPP’  
     
   **UPDATE users SET category = 'BPP';**
2. Запустите очистку VACUUM VERBOSE. Заодно через небольшое время в другом сеансе обратитесь к pg\_stat\_progress\_vacuum.  
     
   **VACUUM VERBOSE;**

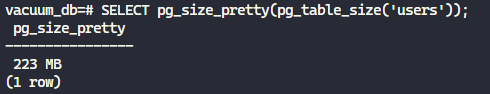
**-- в другом сеансе**

**SELECT \* FROM pg\_stat\_progress\_vacuum;** 

1. Верните значение maintenance\_work\_mem к исходному значению.  
     
   Стандартное значение – 64MB  
     
   **SET maintenance\_work\_mem to '64MB';**

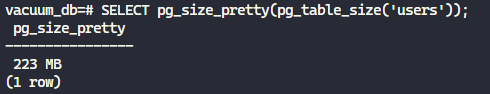
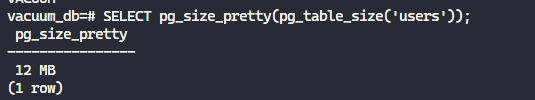
**SELECT pg\_reload\_conf();**

# Выполнение задания 2.

1. Узнать текущий размер файла данных таблицы users при помощи функции: pg\_size\_pretty(pg\_table\_size('название таблицы'))  
     
   **SELECT pg\_size\_pretty(pg\_table\_size('users'));**  
   
2. Удалите 90% случайных строк (Случайность важна, чтобы в каждой странице остались какие-нибудь не удаленные строки)  
     
   Воспользуемся TEMPSAMLE и методом формирования выборки BERNOULLI для получения случайной выборки 90% таблицы.  
     
   **DELETE FROM users WHERE id IN (**

**SELECT id FROM users TABLESAMPLE BERNOULLI (90)**

**);**

1. Выполните очистку  
     
   **VACUUM;**
2. Ещё раз узнайте текущий размер файла данных таблицы users и сравните его с первым пунктом. Объясните результат.  
     
   **SELECT pg\_size\_pretty(pg\_table\_size('users'));**  
   Vacuum очищает страницы от старых данных оставляя пустые места и не сжимает страницы. Размер таблицы остается такой же.
3. Заново заполните таблицу и повторите пункты 1 и 2.
4. Выполните полную очистку  
     
   **VACUUM FULL;**
5. Ещё раз узнайте текущий размер файла данных таблицы users и сравните его с результатом пункта 5. Объясните результат.  
     
   **SELECT pg\_size\_pretty(pg\_table\_size('users'));**Произошла очистка таблицы и сжатие страниц (удаление пустот). Размер страниц уменьшился.

# Выполнение задания 3.

1. Включите параметр автоочистки в таблице users  
     
   **ALTER TABLE users SET (**

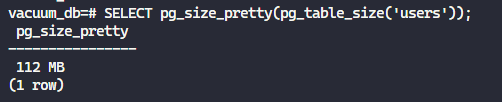
**autovacuum\_enabled = true**

**);**

1. Настройте автоочистку на запуск при изменении 10 % строк, время «сна» — одна секунда (autovacuum\_vacuum\_threshold = 0, autovacuum\_vacuum\_scale\_factor = 0.1, autovacuum\_naptime = '1s')  
     
   Все эти настройки прописываются в файле postgresql.conf, после чего postgres необходимо перезагрузить.
2. Заполните таблицу users до 1000000 записей.  
     
   **INSERT INTO users (username, email, category)**

**SELECT gen\_random\_uuid() as username, gen\_random\_uuid() as email, 'FOO'::char(3) AS category**

**FROM generate\_series(1,1000000);**

1. Узнать текущий размер файла данных таблицы users при помощи функции: pg\_size\_pretty(pg\_table\_size('название таблицы'))  
     
   
2. Напишите скрипт который двадцать раз с интервалом в несколько секунд изменяет по 5 % случайных строк. Каждое изменение выполняйте в отдельной транзакции.

DO

$BODY$

BEGIN

FOR i IN 1..20 LOOP

DELETE FROM users WHERE id IN (

SELECT id FROM users TABLESAMPLE BERNOULLI (5)

);

COMMIT;

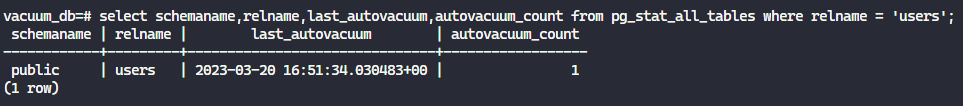
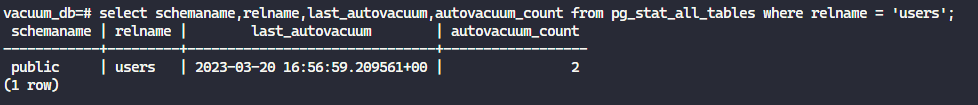
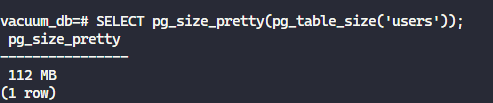
PERFORM pg\_sleep(2);

END LOOP;

END

$BODY$

LANGUAGE plpgsql;

1. При помощи **pg\_stat\_all\_tables** узнайте сколько раз выполнялась автоочистка (**autovacuum\_count**)  
     
   Результат до применения скрипта  
     
     
   Результат после применения скрипта  
   
2. Сравнить размеры таблицы до и после обновлений  
     
     
   Размер таблицы не изменился, так как не был выполнен глубокий вакуум.
3. Совпадают ли результаты с ожидаемыми и как их объяснить?

В связи с установлением параметра autovacuum\_vacuum\_threshold количество автоочисток уменьшилось, что сложилось с фактором autovacuum\_vacuum\_scale\_factor порог срабатывания очистки срабатывает в 10%, при том, что мы изменяли лишь 5%.