федеральное государственное автономное образовательное учреждение высшего образования

**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО»**

**ОТЧЕТ**

по лабораторной работе №4

по дисциплине «**Распределенные системы хранения данных**»

Вариант 15

Автор: Кулаков Н. В.

Буторин В.А.

Факультет: ПИиКТ

Группа: P33312

Преподаватель: Шешуков Д. М.



Санкт-Петербург 2023

# **Постановка задачи и исходные данные**

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 снапшот 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:

1. Настроить репликацию postgres на трёх узлах: A - основной, B и C - резервные. Для управления использовать pgpool-II. Репликация с A на B синхронная. Репликация с A на C асинхронная. Продемонстрировать, что новые данные реплицируются на B в синхронном режиме, а на C с задержкой.

Этап 2.1 Подготовка:

1. Установить несколько клиентских подключений к СУБД.
2. Продемонстрировать состояние данных и работу клиентов в режиме чтение/запись.

Этап 2.2 Сбой:

1. Симулировать отказ основного узла - выполнить жесткое выключение виртуальной машины.

Этап 2.3 Отработка:

1. Найти продемонстрировать в логах релевантные сообщения об ошибках.
2. Выполнить фейловер на резервный сервер.
3. Продемонстрировать состояние данных и работу клиентов в режиме чтение/запись.

Этап 3 Восстановление:

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

# Выполнение

Для выполнения сейчас и далее использовался Docker.

## Настройка рабочего окружения

Создаем Docker образ, на который будем накатывать узлы:

postgres-node.Dockefile

```

FROM ubuntu:22.04

ARG DEBIAN\_FRONTEND=noninteractive

RUN apt-get update

RUN apt-get -y install curl wget sudo

RUN apt-get -y install ca-certificates gnupg

RUN sudo sh -c 'echo "deb http://apt.postgresql.org/pub/repos/apt $(lsb\_release -cs)-pgdg main" > /etc/apt/sources.list.d/pgdg.list'

RUN wget --quiet -O - https://www.postgresql.org/media/keys/ACCC4CF8.asc | sudo apt-key add -

RUN apt-get update

RUN apt-get -y install postgresql-14

RUN apt-get -y install ssh iputils-ping

RUN mkdir -p /app

WORKDIR /app

```

# Вывод