# Лабораторная работа 6. Обеспечения безопасности и аварийного восстановления данных

**1. Обоснование выбранных механизмов репликации и резервного копирования.**

А) Резервное копирование базы данных проводится с целью поучения возможности отката изменений в БД или восстановления данных в случае инцидента ИБ.

Б) Репликация БД может проводится с целью повышения доступности данных в случае географической отдаленности серверов или отказа одного из них, также репликация может значительно повысить безопасность инфраструктуры данных.

**2. Описание необходимого оборудования и ПО.**

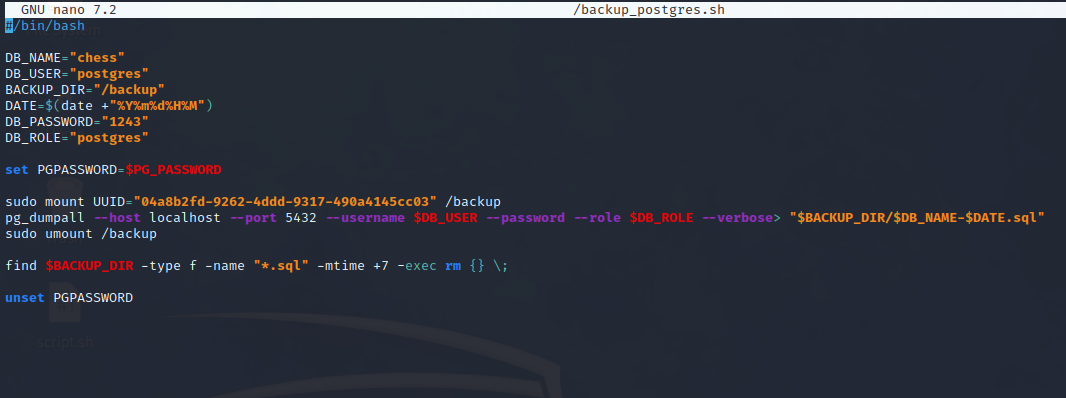
В нашем примере будут использованы:

1. Для резервного копирования: Виртуальная машина под управлением ОС Linux, с подсистемой контроля служб Systemd и установленной СУБД Postgresql
2. Для репликации: Две виртуальных машины с ос линукс и одинаковыми версиями postgresql, находящиеся в одной сети.

**3. Регламент резервного копирования и репликации базы данных с подробной инструкцией по реализации.**

**Резервное копирование:**

1. На сервере в корневой директории был создан скрипт:

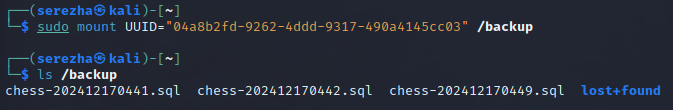


1. Запустить скрипт командой:



Бэкапы сохраняются на диск, монтируемый и демонтируемый в ходе выполнения сценария, это должно повысить шансы на сохранение данной копии в случае некоторых проишествий.

1. Просмотреть, отправить или воспользоваться копией можно примонтировав данный диск. Сделать это можно следующим образом:

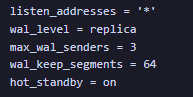


1. Также можно настроить резервное копирование в определённое время с помощью system, а также на включение/выключение сервера.
2. Резервные копии стоит на постоянной основе выгружать в хранилище независимое от основного сервера.

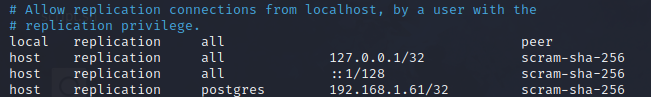
**Репликация**

1. Созданы две машины с одинаковыми версиями postgresql иОС(кали линукс)
2. На основной и резервной машине были отредактированы конфигурационные файлы:

postgresql.conf:



pg\_hba.conf:



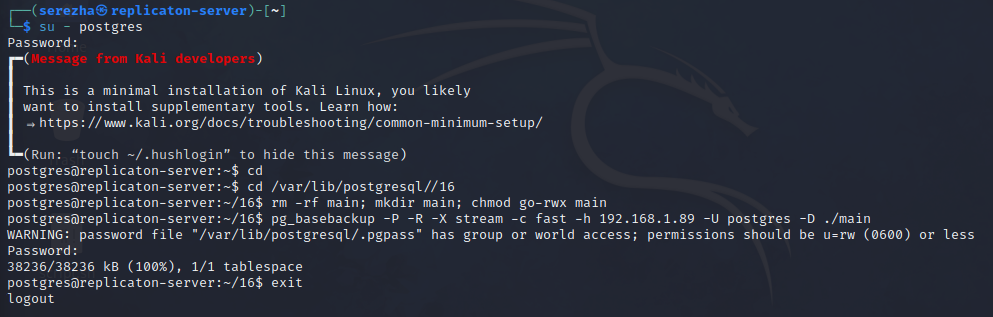
1. Был перезапущен сервис postgresql: 
2. На резервной машине был выполнен небольшой набор комманд:

$ su postgres

$ cd /var/lib/postgresql/16/

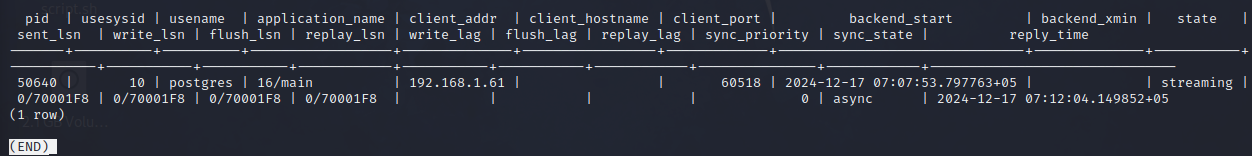
$ rm -rf main; mkdir main; chmod go-rwx main

$pg\_basebackup -P -R -X stream -c fast -h 192.168.1.89\_IP -U postgres -D ./main



1. На основноый машине была проверена репликация:

$ sudo -u postgres psql -c "SELECT \* FROM pg\_stat\_replication;"



**4. План аварийного восстановления данных с подробной инструкцией по реализации.**

Вариант 1. Восстановление с файла-бэкапа:

1. Достать самый актуальный бэкап с жесткого диска сервера если это возможно, в ином случае с места, куда бэкапы отправлялись
2. Прописать команду

$ pg\_restore -h <адрес сервера> -U postgres -d сhess <файл бэкапа>

Вариант 2. Перевод реплицирующего сервера в рабочий режим:

1. Подключится к реплицирующему серверу (например через SSH)
2. Выполнить команду:

$ pg\_restore -h localhost -U postgres -d ChessSchool D:\PGSQLBackup\backups\ChessSchool.backup