

# Министерство науки и высшего образования Российской Федерации

---

**Федеральное государственное автономное образовательное учреждение  
высшего образования  
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»**

*Институт перспективной инженерии*

*Департамент цифровых, робототехнических систем и электроники*

*Межинститутская базовая кафедра*

---

## Отчет по лабораторной работе №0

дисциплины "Администрирование баз данных"

**Выполнил:** Душин Александр Владимирович

4 курс, группа ПИЖ-6-о-22-1

09.03.04 "Программная инженерия"

направленность (профиль) "Разработка и сопровождение программного обеспечения"

очная форма обучения

**Руководитель практики:** Щеголев Алексей Алексеевич

старший преподаватель департамента цифровых, робототехнических систем и электроники института перспективной инженерии

---

## Тема работы

Установка PostgreSQL и базовое управление сервером

## Цель работы

Освоить основные способы установки PostgreSQL (из пакетов и исходных кодов) в предустановленном окружении. Получить практические навыки создания кластера баз данных, запуска и остановки сервера, а также проверки его состояния. Научиться использовать утилиты `initdb`, `pg_ctl`, `pg_checksums` и `psql` для базового взаимодействия с СУБД.

---

## Ход выполнения работы

Часть 1. Установка из пакетов (Ubuntu) и управление кластером

### 1. Установка PostgreSQL 16 из официальных пакетов

Установлен PostgreSQL 16 с помощью пакетного менеджера `apt`. (Рис.1)

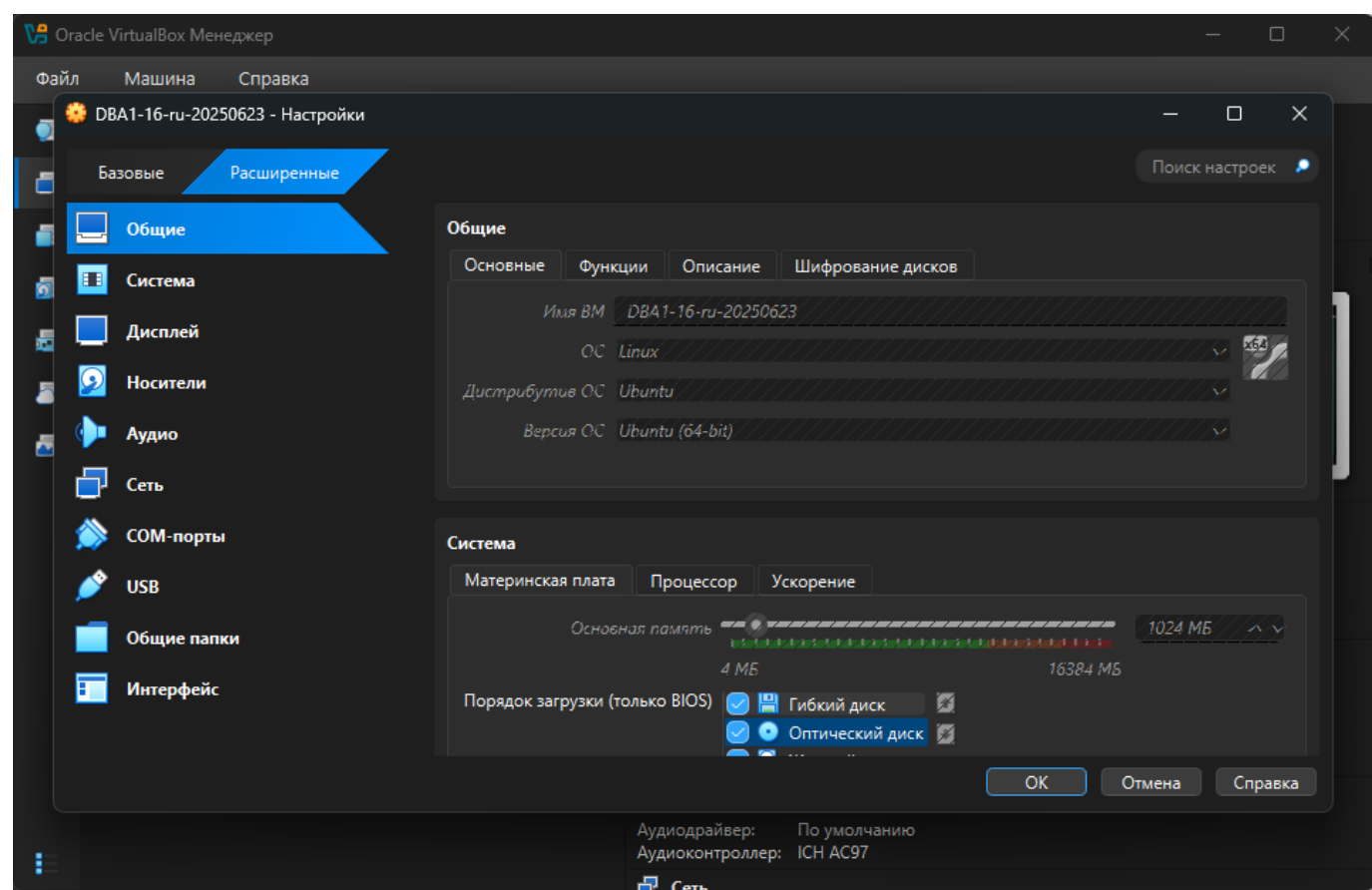


Рисунок 1 – Установка образа с PostgreSQL

## 2. Проверка кластера

Пакетный менеджер автоматически создал кластер **main**. Его состояние проверено с помощью команды **pg\_lsclusters**. (Рис.2)

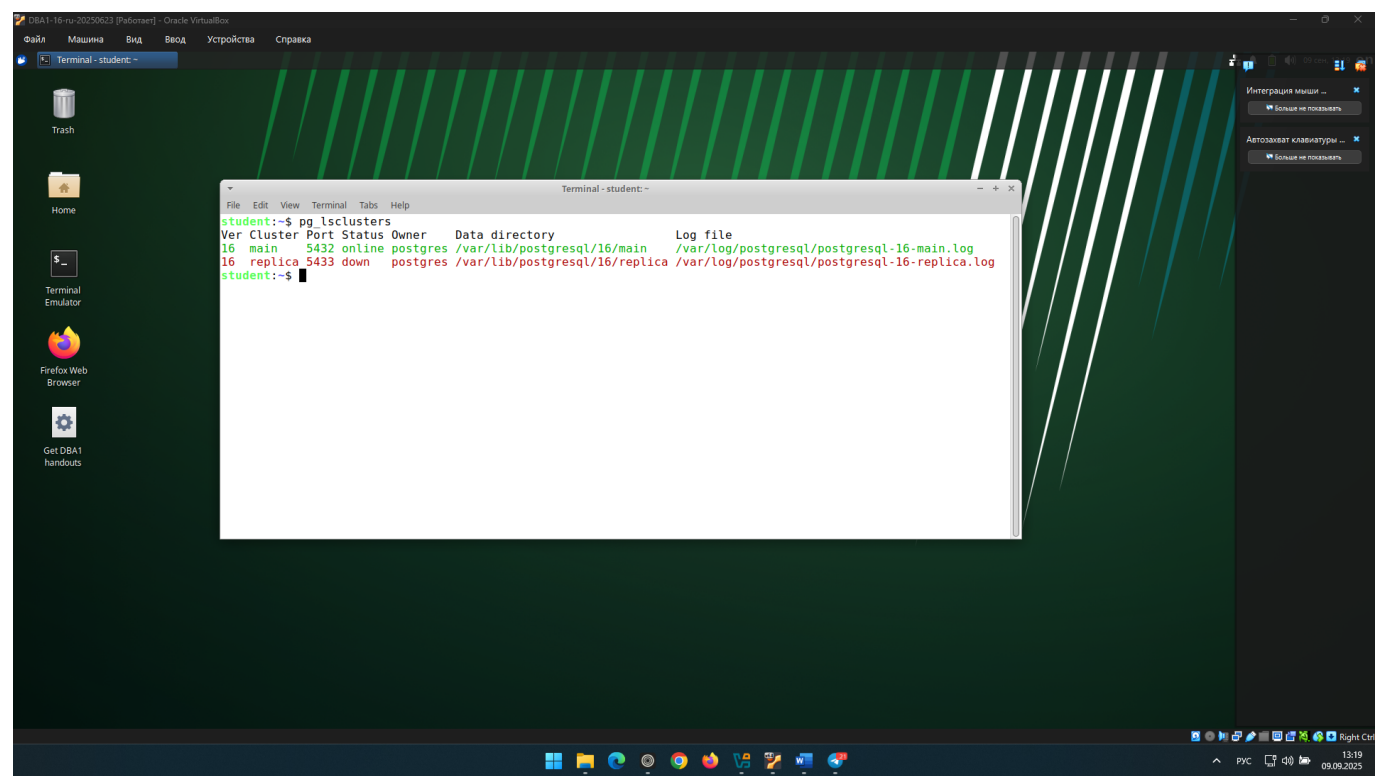
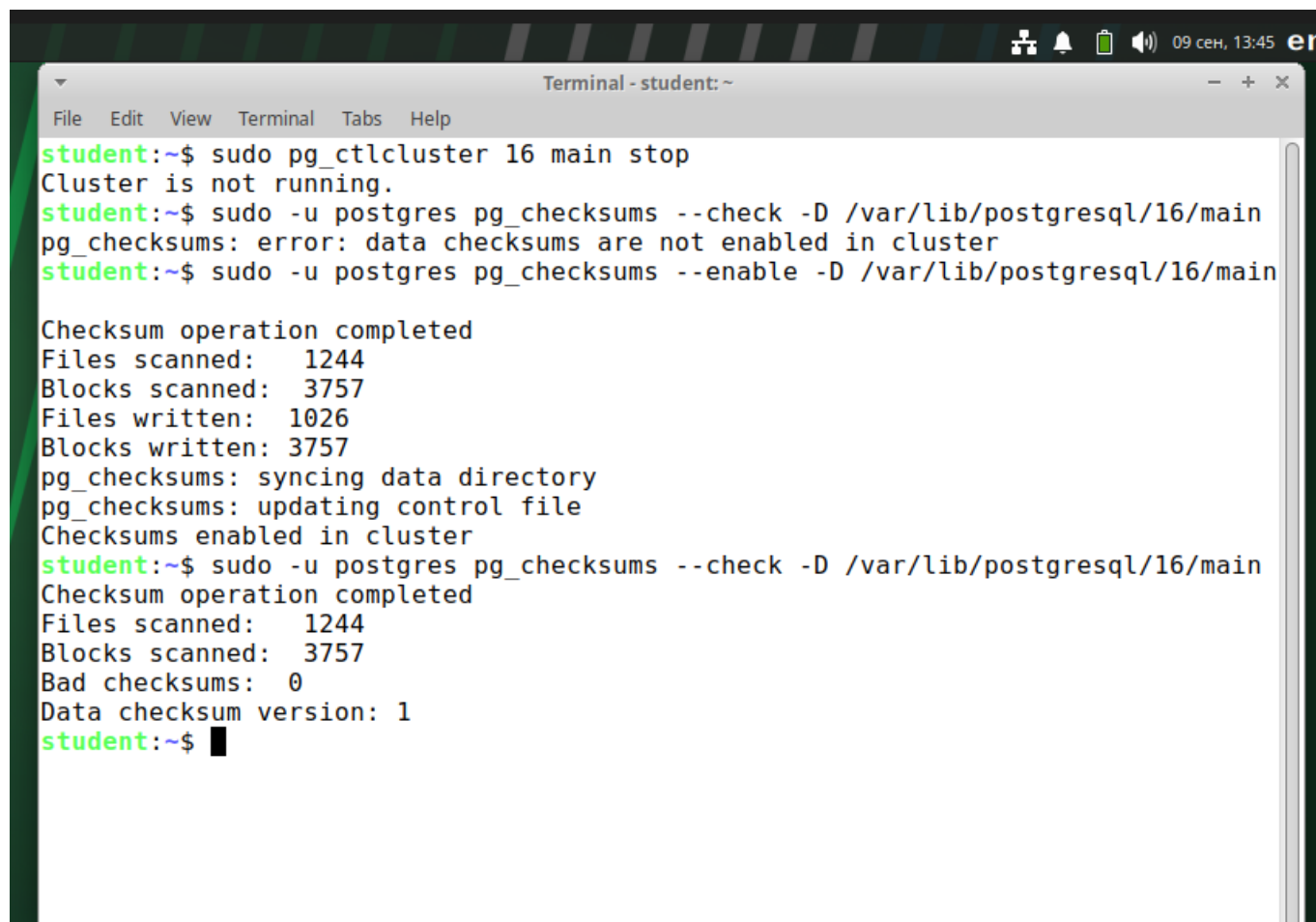


Рисунок 2 – Проверка статуса с помощью pg\_lsclusters

### 3. Включение контрольных сумм

Кластер остановлен, выполнена проверка состояния контрольных сумм (`pg_checksums --check`). Далее включен расчет контрольных сумм (`pg_checksums --enable`), после чего кластер снова запущен. (Рис.3)



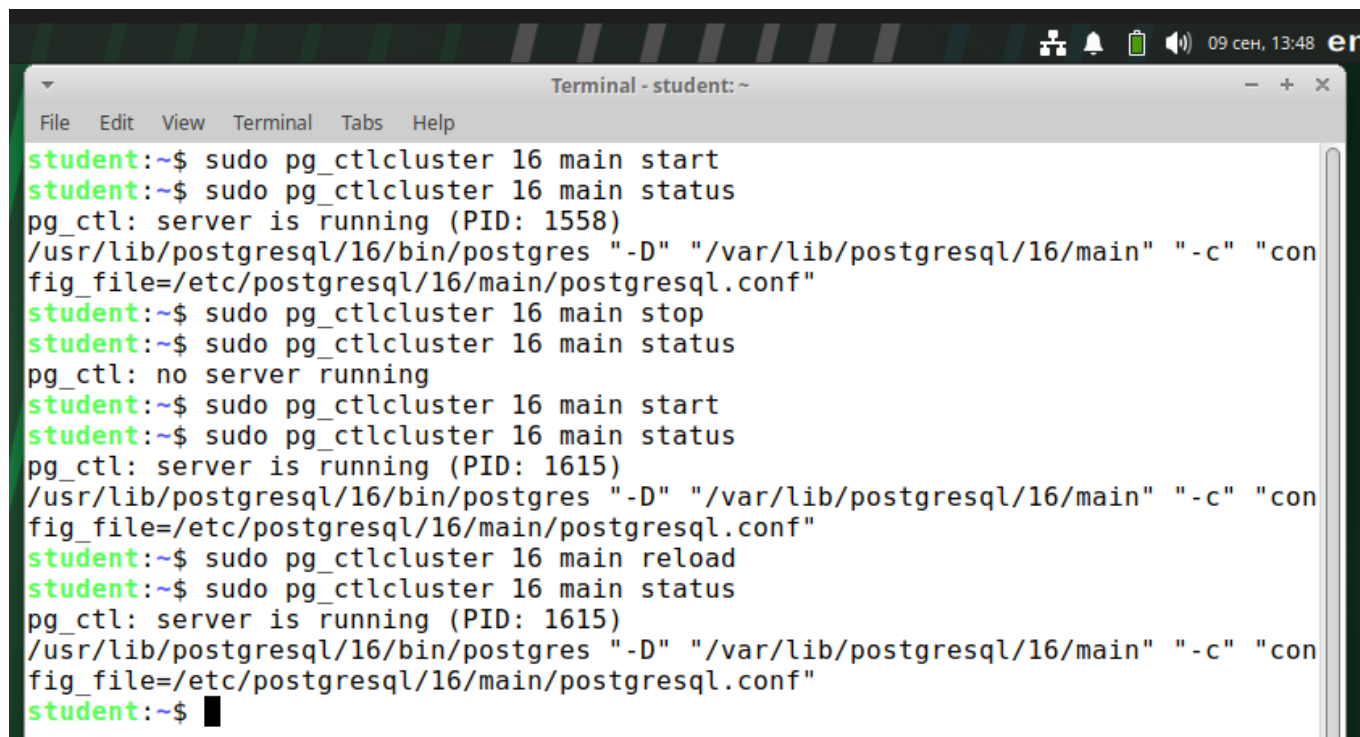
```
student:~$ sudo pg_ctlcluster 16 main stop
Cluster is not running.
student:~$ sudo -u postgres pg_checksums --check -D /var/lib/postgresql/16/main
pg_checksums: error: data checksums are not enabled in cluster
student:~$ sudo -u postgres pg_checksums --enable -D /var/lib/postgresql/16/main

Checksum operation completed
Files scanned: 1244
Blocks scanned: 3757
Files written: 1026
Blocks written: 3757
pg_checksums: syncing data directory
pg_checksums: updating control file
Checksums enabled in cluster
student:~$ sudo -u postgres pg_checksums --check -D /var/lib/postgresql/16/main
Checksum operation completed
Files scanned: 1244
Blocks scanned: 3757
Bad checksums: 0
Data checksum version: 1
student:~$
```

Рисунок 3 – Включение контрольных сумм

### 4. Управление сервером

С помощью утилиты `pg_ctlcluster` выполнены остановка, запуск и перезапуск кластера. После каждой операции проверялся статус. (Рис.4)


A terminal window titled "Terminal - student: ~" with a menu bar (File, Edit, View, Terminal, Tabs, Help). The terminal shows a series of commands to start, stop, and reload a PostgreSQL 16 cluster. The user is prompted for a password. The output shows the server is running with PID 1558, then 1615, and finally 1615 after a reload. The configuration file is /etc/postgresql/16/main/postgresql.conf.

```
student:~$ sudo pg_ctlcluster 16 main start
student:~$ sudo pg_ctlcluster 16 main status
pg_ctl: server is running (PID: 1558)
/usr/lib/postgresql/16/bin/postgres "-D" "/var/lib/postgresql/16/main" "-c" "con
fig_file=/etc/postgresql/16/main/postgresql.conf"
student:~$ sudo pg_ctlcluster 16 main stop
student:~$ sudo pg_ctlcluster 16 main status
pg_ctl: no server running
student:~$ sudo pg_ctlcluster 16 main start
student:~$ sudo pg_ctlcluster 16 main status
pg_ctl: server is running (PID: 1615)
/usr/lib/postgresql/16/bin/postgres "-D" "/var/lib/postgresql/16/main" "-c" "con
fig_file=/etc/postgresql/16/main/postgresql.conf"
student:~$ sudo pg_ctlcluster 16 main reload
student:~$ sudo pg_ctlcluster 16 main status
pg_ctl: server is running (PID: 1615)
/usr/lib/postgresql/16/bin/postgres "-D" "/var/lib/postgresql/16/main" "-c" "con
fig_file=/etc/postgresql/16/main/postgresql.conf"
student:~$
```

Рисунок 4 – Отработка управления сервером

## 5. Подключение и выполнение запроса

Выполнено подключение к кластеру через `psql` от имени пользователя `postgres`. Выполнен тестовый запрос `SELECT version();`. (Рис.5)

A terminal window titled "Terminal - student: ~" with a menu bar (File, Edit, View, Terminal, Tabs, Help). The user runs 'sudo -u postgres psql' and is prompted for a password. The prompt changes to 'postgres=#'. The user enters 'SELECT version();' and the output shows the PostgreSQL version and build information.

```
student:~$ sudo -u postgres psql
psql (16.9 (Ubuntu 16.9-1.pgdg24.04+1))
Type "help" for help.

postgres=# SELECT version();

              version
-----
PostgreSQL 16.9 (Ubuntu 16.9-1.pgdg24.04+1) on x86_64-pc-linux-gnu, compiled by gcc (Ubuntu 13.3.0-6ubuntu2~24.04) 13.3.0, 64-bit
(1 row)

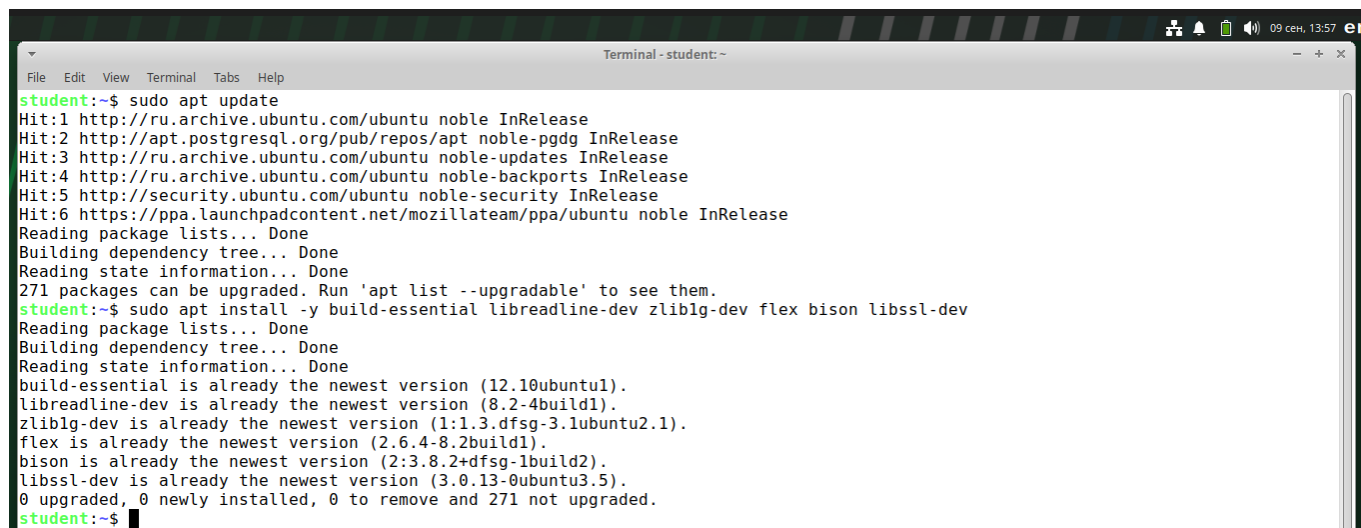
postgres=#
```

Рисунок 5 – Выполнение простого запроса

## Часть 2. Установка из исходных кодов

### 1. Подготовка инструментов для сборки

Установлены необходимые инструменты: `gcc`, `make`, `libreadline-dev` и другие. (Рис.6)



```
student:~$ sudo apt update
Hit:1 http://ru.archive.ubuntu.com/ubuntu noble InRelease
Hit:2 http://apt.postgresql.org/pub/repos/apt noble-pgdg InRelease
Hit:3 http://ru.archive.ubuntu.com/ubuntu noble-updates InRelease
Hit:4 http://ru.archive.ubuntu.com/ubuntu noble-backports InRelease
Hit:5 http://security.ubuntu.com/ubuntu noble-security InRelease
Hit:6 https://ppa.launchpadcontent.net/mozillateam/ppa/ubuntu noble InRelease
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
271 packages can be upgraded. Run 'apt list --upgradable' to see them.
student:~$ sudo apt install -y build-essential libreadline-dev zlib1g-dev flex bison libssl-dev
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
build-essential is already the newest version (12.10ubuntu1).
libreadline-dev is already the newest version (8.2-4build1).
zlib1g-dev is already the newest version (1:1.3.dfsg-3.1ubuntu2.1).
flex is already the newest version (2.6.4-8.2build1).
bison is already the newest version (2:3.8.2+dfsg-1build2).
libssl-dev is already the newest version (3.0.13-0ubuntu3.5).
0 upgraded, 0 newly installed, 0 to remove and 271 not upgraded.
student:~$
```

Рисунок 6 – Установка инструментов для сборки

## 2. Сборка и установка PostgreSQL

Скачан архив с исходными кодами PostgreSQL 16. Архив распакован, сборка сконфигурирована с указанием каталога установки (`--prefix`) и нестандартного порта (`--with-pgport=5433`). Далее выполнены команды `make` и `make install`. (Рис.7-8)

```
tar -xf postgresql-16.9.tar.gz
cd postgresql-16.9
./configure --prefix="$HOME/pg16custom" --with-pgport=5433
make
make install
```

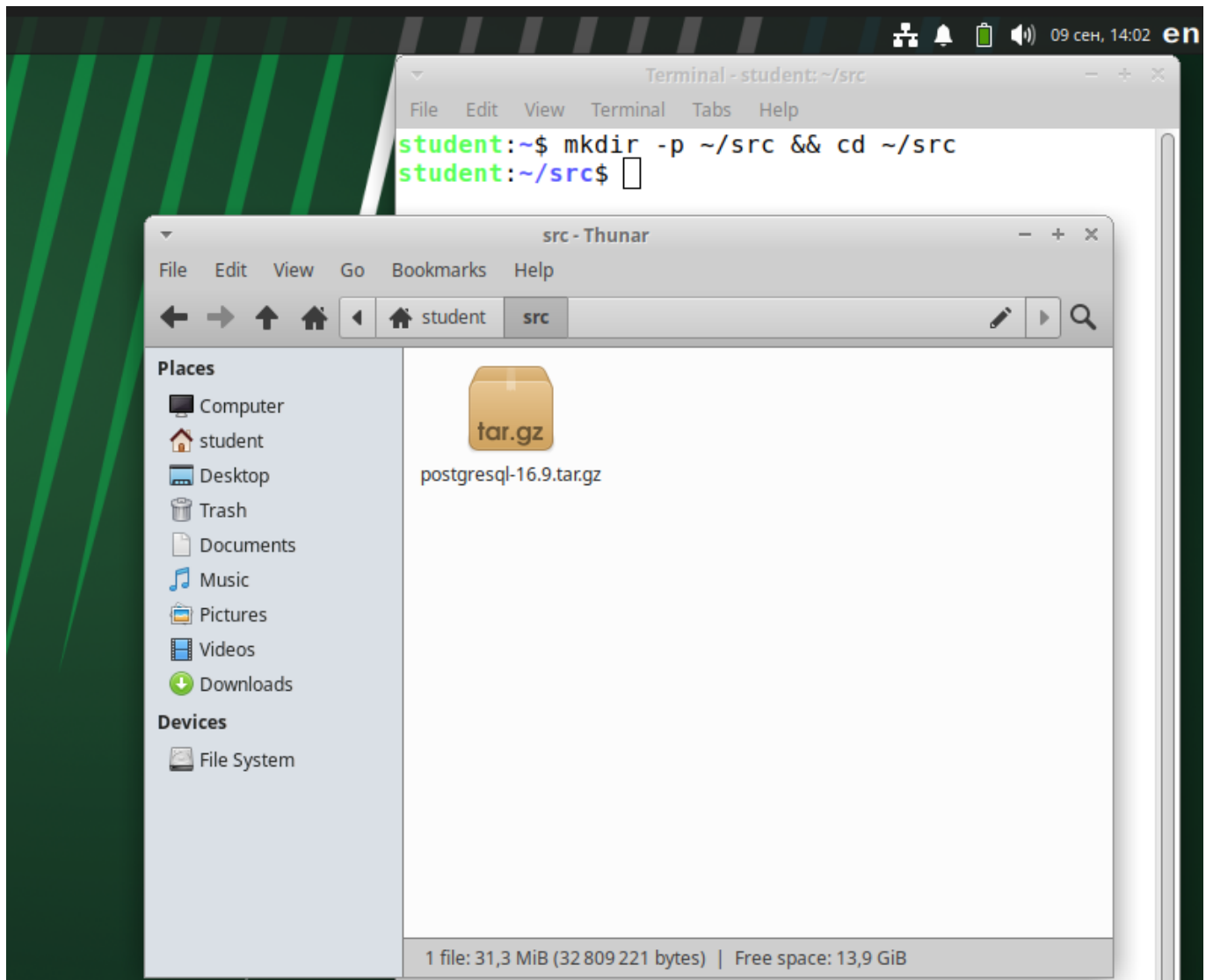


Рисунок 7 – Архив PostgreSQL

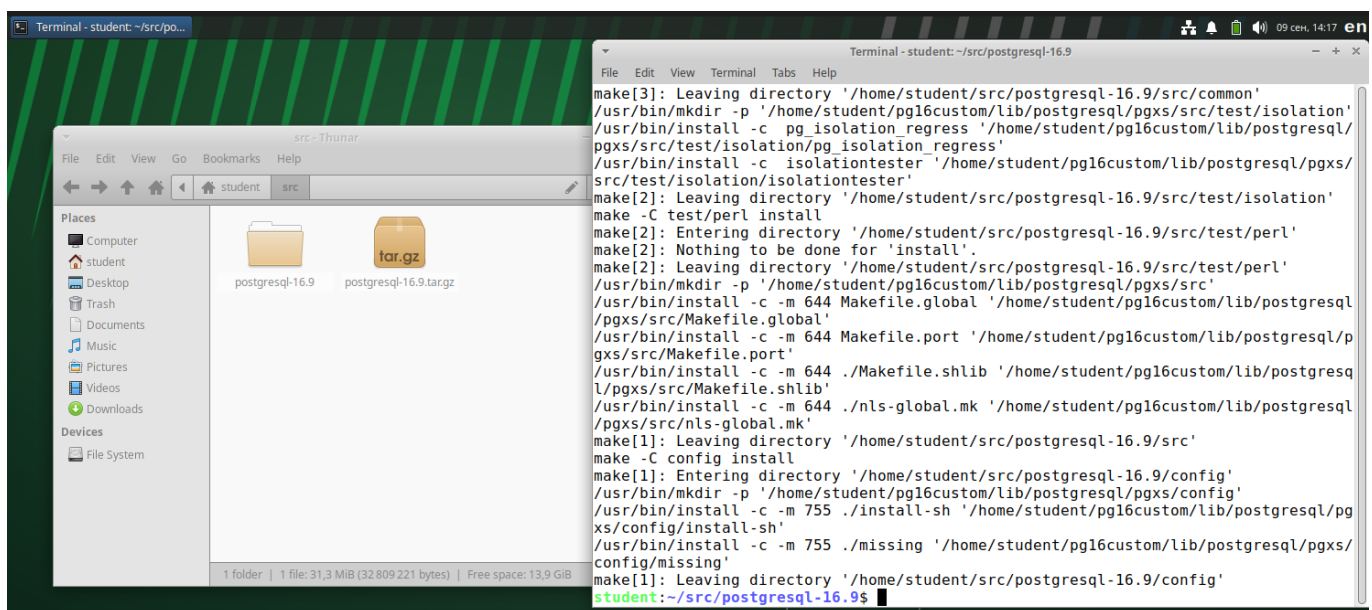
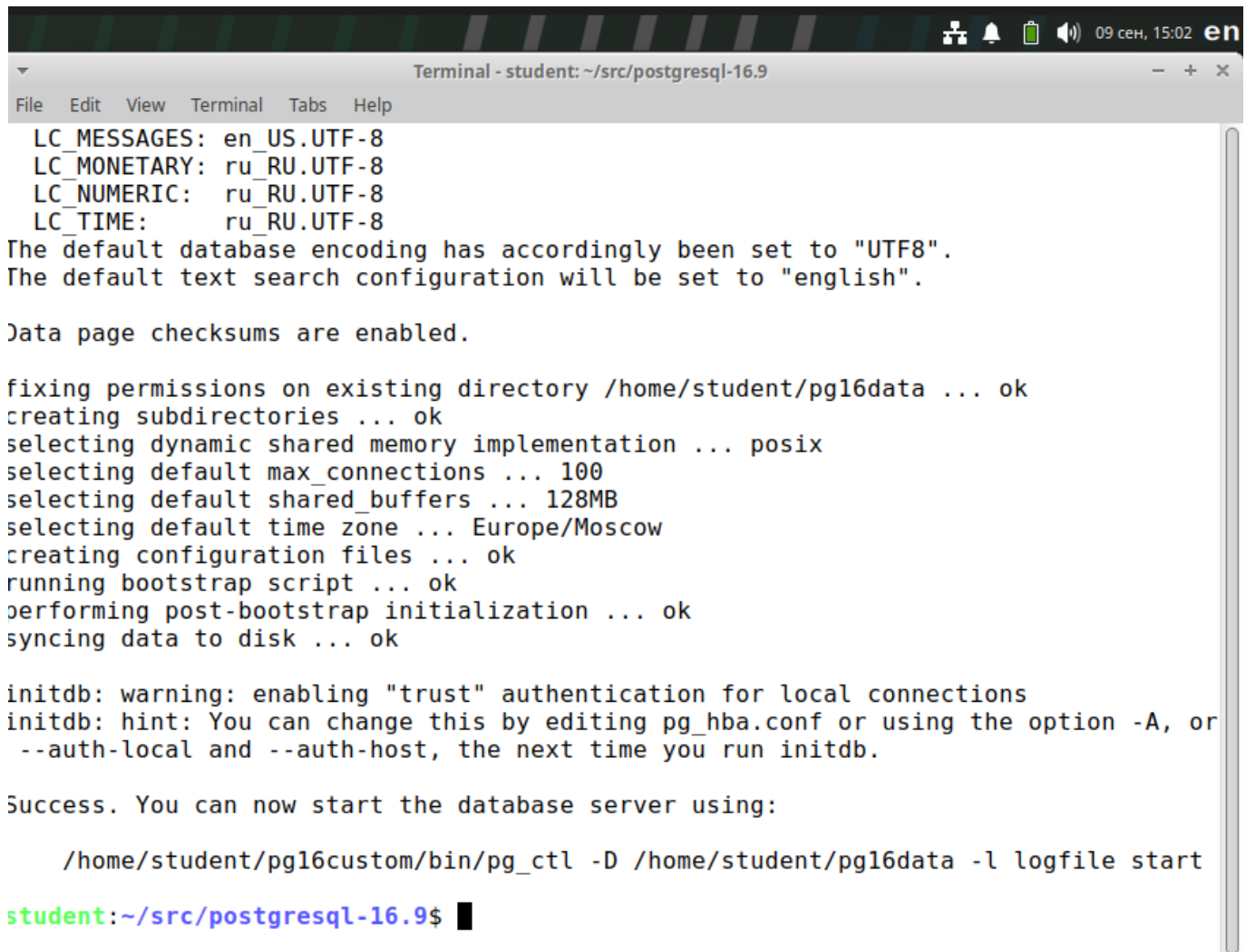


Рисунок 8 – Распаковка, конфигурация и установка

### 3. Создание нового кластера

Создан каталог для данных (**PGDATA**). Инициализирован новый кластер БД с указанием суперпользователя **postgres** и включением контрольных сумм. (Рис.9)



```
LC_MESSAGES: en_US.UTF-8
LC_MONETARY: ru_RU.UTF-8
LC_NUMERIC:  ru_RU.UTF-8
LC_TIME:     ru_RU.UTF-8
The default database encoding has accordingly been set to "UTF8".
The default text search configuration will be set to "english".

Data page checksums are enabled.

fixing permissions on existing directory /home/student/pg16data ... ok
creating subdirectories ... ok
selecting dynamic shared memory implementation ... posix
selecting default max_connections ... 100
selecting default shared_buffers ... 128MB
selecting default time zone ... Europe/Moscow
creating configuration files ... ok
running bootstrap script ... ok
performing post-bootstrap initialization ... ok
syncing data to disk ... ok

initdb: warning: enabling "trust" authentication for local connections
initdb: hint: You can change this by editing pg_hba.conf or using the option -A, or
--auth-local and --auth-host, the next time you run initdb.

Success. You can now start the database server using:

    /home/student/pg16custom/bin/pg_ctl -D /home/student/pg16data -l logfile start

student:~/src/postgresql-16.9$
```

Рисунок 9 – Создание кластера

#### 4. Запуск и проверка сервера

Сервер запущен с помощью **pg\_ctl** с указанием файла для логов. Выполнено подключение через **psql** на нестандартном порту и выполнен запрос. После тестирования сервер остановлен. (Рис.10-11)



```

Terminal - student: ~/src/pg...
student-Thunar
File Edit View Terminal Tabs Help
Terminal - student: ~/src/postgresql-16.9
LC_MONETARY: ru_RU.UTF-8
LC_NUMERIC: ru_RU.UTF-8
LC_TIME: ru_RU.UTF-8
The default database encoding has accordingly been set to "UTF8".
The default text search configuration will be set to "english".
Data page checksums are enabled.
fixing permissions on existing directory /home/student/pg16data2 ... ok
creating subdirectories ... ok
selecting dynamic shared memory implementation ... posix
selecting default max_connections ... 100
selecting default shared_buffers ... 128MB
selecting default time zone ... Europe/Moscow
creating configuration files ... ok
running bootstrap script ... ok
performing post-bootstrap initialization ... ok
syncing data to disk ... ok
initdb: warning: enabling "trust" authentication for local connections
initdb: hint: You can change this by editing pg_hba.conf or using the option -A, or --auth-local and --auth-host, the
next time you run initdb.
Success. You can now start the database server using:

        /home/student/pg16custom/bin/pg_ctl -D /home/student/pg16data2 -l logfile start

student:~/src/postgresql-16.9$ /home/student/pg16custom/bin/pg_ctl -D /home/student/pg16data2 -l logfile start
waiting for server to start.... done
server started
student:~/src/postgresql-16.9$ /home/student/pg16custom/bin/pg_ctl -D /home/student/pg16data2 status
pg_ctl: server is running (PID: 13526)
/home/student/pg16custom/bin/postgres "-D" "/home/student/pg16data2"
student:~/src/postgresql-16.9$

```

Рисунок 10 – Запуск сервера с помощью pg\_ctl

```

al - student: ~/src/pg...
Terminal - student: ~/src/postgresql-16.9
student:~/src/postgresql-16.9$ "$HOME/pg16custom/bin/psql" -p 5433 -U postgres -c "SELECT version();"
version
-----
PostgreSQL 16.9 on x86_64-pc-linux-gnu, compiled by gcc (Ubuntu 13.3.0-6ubuntu2-24.04) 13.3.0, 64-bit
(1 row)

student:~/src/postgresql-16.9$ "$HOME/pg16custom/bin/pg_ctl" -D "$HOME/pg16data2" stop
waiting for server to shut down.... done
server stopped
student:~/src/postgresql-16.9$ /home/student/pg16custom/bin/pg_ctl -D /home/student/pg16data2 status
pg_ctl: no server running
student:~/src/postgresql-16.9$

```

Рисунок 11 – Подключение к новому серверу на нестандартном порту и остановка сервера

## Теоретическая часть

### Зачем нужны контрольные суммы данных в PostgreSQL?

Контрольные суммы данных — это механизм защиты от «тихих» повреждений, когда информация на диске искажается без явных ошибок со стороны ОС.

PostgreSQL рассчитывает контрольную сумму для каждой 8KB страницы при записи и проверяет её при чтении. Несовпадение суммы означает повреждение данных — сервер сообщает об ошибке и предотвращает использование некорректной информации.



Этот механизм особенно важен для продакшн-систем, так как позволяет выявлять сбои оборудования (дисков, памяти, контроллеров) на раннем этапе.

Недостатки: небольшое снижение производительности (3–5%) и невозможность включения механизма на уже работающем кластере (только при инициализации).

## Сравнение способов установки PostgreSQL

### Установка из пакетов:

- Простота и скорость (1–3 минуты).
- Автоматическое создание кластера.
- Интеграция с ОС и системными сервисами.
- Удобные обновления через пакетный менеджер.
- Время до первого подключения: 5–10 минут.

### Сборка из исходников:

- Более трудоемкий процесс (45–90 минут).
- Требуется установка инструментов и ручной настройки.
- Дает гибкость в конфигурации (порт, каталог, модули).
- Оптимизация под конкретное оборудование.
- Полезно для глубокого изучения и специфических задач.

---

## Выводы

В ходе выполнения лабораторной работы были освоены два подхода к установке PostgreSQL и базовые приемы администрирования.

Пакетная установка показала простоту и надежность, что делает её оптимальной для большинства продакшн-сценариев. Сборка из исходных кодов оказалась более сложной и затратной по времени, но дала гибкость конфигурации и лучшее понимание архитектуры PostgreSQL.

Практика работы с контрольными суммами подтвердила их значимость для обеспечения целостности данных. Освоены утилиты `pg_ctl`, `pg_checksums`, `initdb` и `psql`, которые являются основой для дальнейшего изучения администрирования PostgreSQL.

Выбор метода установки зависит от задач:

- для стандартных сценариев рекомендуется установка из пакетов;
- для специализированных задач или учебных целей целесообразна сборка из исходников.