



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

РАБОТА ДОПУЩЕНА К ЗАЩИТЕ

Руководитель
программы _____ А.Т. Тарланов

«17» июня 2025 г.

ИТОГОВАЯ АТТЕСТАЦИОННАЯ РАБОТА
по дополнительной программе профессиональной переподготовки
«Технологии Девопс»

На тему: «Набор тематических заданий по проведению итоговой аттестации DevOps-инженера – вариант 1122»

Обучающийся _____
Подпись

Павлов Никита Сергеевич
Фамилия, имя, отчество

группа ИКБО-50-23

Руководитель работы _____
подпись

А.Т. Тарланов

Москва 2025 г.

СОДЕРЖАНИЕ

1. ВВЕДЕНИЕ	3
2. ОСНОВНАЯ ЧАСТЬ.....	5
2.1 Модуль 6. Администрирование баз данных.....	5
2.2 Модуль 7. Масштабируемость и отказоустойчивость	10
2.3 Модуль 8. Мониторинг, логирование и оповещение событий.....	14
2.4 Модуль 9. Виртуализация в DevOps	20
2.5 Модуль 12. Конфигурационное управление. Что такое IaC.....	22
2.6 Модуль 13. Системы контроля версий. Распределённая система управления версиями Git.....	25
2.7 Модуль 14. Жизненный цикл ПО	28
2.8 Модуль 15. Практические навыки работы с Docker.....	33
3 ЗАКЛЮЧЕНИЕ	35
СПИСОК ЛИТЕРАТУРЫ.....	37

1. ВВЕДЕНИЕ

Современная DevOps-экосистема требует комплексного подхода, объединяющего администрирование инфраструктуры, автоматизацию процессов, контроль версий и мониторинг систем. Рост спроса на специалистов, способных работать с распределёнными системами, конвейерами CI/CD и облачными технологиями, делает освоение ключевых инструментов DevOps критически важным для ИТ-отрасли.

Данная итоговая работа посвящена практическому освоению технологий, формирующих основу DevOps-практик. В рамках выполнения заданий модулей №6, 7, 8, 9, 12, 13, 14 и 15 дополнительной профессиональной программы «Технологии Девопс» были реализованы следующие аспекты:

- *Администрирование PostgreSQL:* от установки СУБД до проектирования структуры данных, создания пользователей, управления базами данных и выполнения SQL-запросов, включая операции SELECT, INSERT и другие.
- *Настройка отказоустойчивого кластера:* работа с инструментами HAProxy и Pacemaker для обеспечения масштабируемости и отказоустойчивости, включая настройку SSH-ключей, синхронизацию времени и управление узлами.
- *Мониторинг и логирование:* развёртывание стека мониторинга на базе Prometheus, Grafana и Node Exporter, настройка дашбордов и сбор метрик для анализа производительности системы.
- *Виртуализация:* установка и настройка инструментов, таких как vim, mc и net-tools, для работы в виртуальных средах.
- *Конфигурационное управление (IaC):* использование Ansible для автоматизации управления инфраструктурой, создание и выполнение playbook'ов.
- *Системы контроля версий:* развёртывание Git-сервера, управление репозиториями, создание веток, слияние изменений и организация совместной работы.

- *Жизненный цикл ПО*: настройка Jenkins для автоматизации CI/CD-процессов, включая установку плагинов, создание пользователей и управление задачами.
- *Работа с Docker*: создание Docker-образов, настройка контейнеров для развёртывания приложений, включая PostgreSQL, и управление их жизненным циклом.

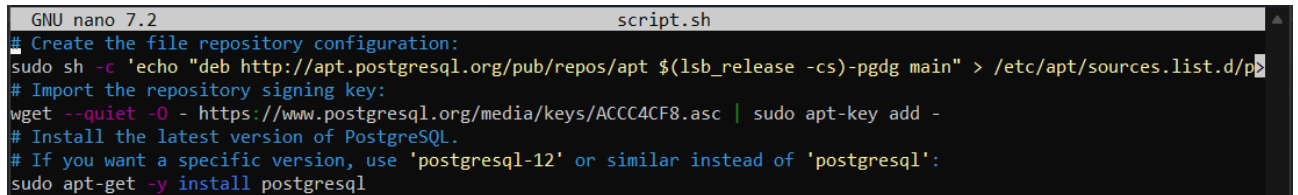
Особое внимание уделено интеграции компонентов в единую экосистему: от контроля версий исходного кода через Git до визуализации метрик работы системы в Grafana. Практические задания охватывают полный цикл DevOps-процессов — настройку окружения, развёртывание сервисов, конфигурацию инструментов мониторинга и автоматизацию сборки.

Цель работы — формирование профессиональных компетенций в области DevOps через практическое взаимодействие с ключевыми технологиями: управление базами данных, обеспечение отказоустойчивости, настройка систем мониторинга, автоматизация жизненного цикла ПО и работа с контейнеризацией. Результаты демонстрируют готовность к реализации реальных задач в области сопровождения и развития ИТ-инфраструктуры.

2. ОСНОВНАЯ ЧАСТЬ

2.1 Модуль 6. Администрирование баз данных

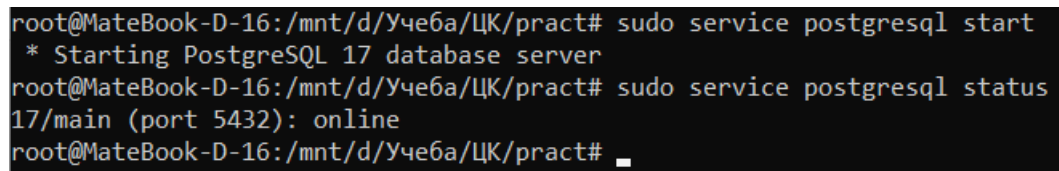
Начальный этап включал загрузку и установку PostgreSQL с использованием скрипта (Рисунок 1). Это обеспечило корректную инсталляцию СУБД и подготовку окружения для дальнейшей работы.

A screenshot of a terminal window with a dark background. The title bar at the top shows 'GNU nano 7.2' on the left and 'script.sh' on the right. The terminal content shows a series of commands for installing PostgreSQL: creating a repository configuration file, importing a signing key, and installing the latest version. The commands are: 'sudo sh -c "deb http://apt.postgresql.org/pub/repos/apt \$(lsb_release -cs)-pgdg main" > /etc/apt/sources.list.d/pgdg.list', 'wget --quiet -O - https://www.postgresql.org/media/keys/ACCC4CF8.asc | sudo apt-key add -', and 'sudo apt-get -y install postgresql'. There are also comments in the script explaining each step.

```
GNU nano 7.2 script.sh
# Create the file repository configuration:
sudo sh -c "deb http://apt.postgresql.org/pub/repos/apt $(lsb_release -cs)-pgdg main" > /etc/apt/sources.list.d/pgdg.list
# Import the repository signing key:
wget --quiet -O - https://www.postgresql.org/media/keys/ACCC4CF8.asc | sudo apt-key add -
# Install the latest version of PostgreSQL.
# If you want a specific version, use 'postgresql-12' or similar instead of 'postgresql':
sudo apt-get -y install postgresql
```

Рисунок 1 – Скрипт установки PostgreSQL

После установки была запущена служба PostgreSQL (Рисунок 2), что подтвердило её работоспособность. Проверка статуса службы показала, что СУБД готова к использованию.

A screenshot of a terminal window with a dark background. The terminal shows the commands to start and check the status of the PostgreSQL service. The commands are: 'sudo service postgresql start' and 'sudo service postgresql status'. The output shows the service starting successfully and being online. The prompt is 'root@MateBook-D-16:/mnt/d/Учеба/ЦК/pract#'.

```
root@MateBook-D-16:/mnt/d/Учеба/ЦК/pract# sudo service postgresql start
* Starting PostgreSQL 17 database server
root@MateBook-D-16:/mnt/d/Учеба/ЦК/pract# sudo service postgresql status
17/main (port 5432): online
root@MateBook-D-16:/mnt/d/Учеба/ЦК/pract#
```

Рисунок 2 – Запуск службы PostgreSQL и проверка работоспособности

Для выполнения административных задач был осуществлён вход в системный аккаунт postgres (Рисунок 3). Это позволило управлять базами данных и пользователями.

```

root@MateBook-D-16:/mnt/d/Учеба/ЦК/pract# sudo -i -u postgres
Welcome to Ubuntu 24.04 LTS (GNU/Linux 5.10.16.3-microsoft-standard-WSL2 x86_64)

* Documentation:  https://help.ubuntu.com
* Management:    https://landscape.canonical.com
* Support:       https://ubuntu.com/pro

System information as of Sat May 24 11:09:17 MSK 2025

System load:  0.01               Processes:            18
Usage of /:   0.8% of 250.98GB   Users logged in:     0
Memory usage: 2%                IPv4 address for eth0: 172.26.68.211
Swap usage:   0%

This message is shown once a day. To disable it please create the
/var/lib/postgresql/.hushlogin file.
postgres@MateBook-D-16:~$

```

Рисунок 3 – Вход в аккаунт postgres

Работоспособность консоли PostgreSQL была проверена (Рисунок 4), что подтвердило возможность выполнения SQL-запросов и взаимодействия с СУБД.

```

postgres@MateBook-D-16:~$ psql
psql (17.5 (Ubuntu 17.5-1.pgdg24.04+1))
Type "help" for help.

postgres=# \conninfo
You are connected to database "postgres" as user "postgres" via socket in "/var/run/postgresql" at port "5432".
postgres=# \q
postgres@MateBook-D-16:~$

```

Рисунок 4 – Проверка работоспособности консоли

Был создан пользователь nikita (Рисунок 5) с назначенными правами. Это обеспечило разграничение доступа и безопасность работы с базами данных.

```

postgres@MateBook-D-16:~$ createuser --interactive
Enter name of role to add: nikita
Shall the new role be a superuser? (y/n) y
postgres@MateBook-D-16:~$ man createuser
postgres@MateBook-D-16:~$

```

Рисунок 5 – Создание нового пользователя

Под учётной записью nikita была создана новая база данных (Рисунки 6 и 7). Это продемонстрировало навыки управления объектами СУБД.

```

root@MateBook-D-16:/mnt/d/Учеба/ЦК/pract# sudo -i -u nikita
Welcome to Ubuntu 24.04 LTS (GNU/Linux 5.10.16.3-microsoft-standard-WSL2 x86_64)

* Documentation:  https://help.ubuntu.com
* Management:    https://landscape.canonical.com
* Support:       https://ubuntu.com/pro

System information as of Sat May 24 11:25:42 MSK 2025

System load:  0.0          Processes:            18
Usage of /:   0.8% of 250.98GB Users logged in:          0
Memory usage: 2%          IPv4 address for eth0: 172.26.68.211
Swap usage:   0%

This message is shown once a day. To disable it please create the
/home/nikita/.hushlogin file.
nikita@MateBook-D-16:~$ psql
psql (17.5 (Ubuntu 17.5-1.pgdg24.04+1))
Type "help" for help.

nikita=# \conninfo
You are connected to database "nikita" as user "nikita" via socket in "/var/run/postgresql" at port "5432".
nikita=#

```

Рисунок 6 – Создание базы данных под учетной записью nikita

```

nikita=# CREATE TABLE bicycle (
nikita(# bicycle_id serial PRIMARY KEY,
nikita(# brand varchar(50) NOT NULL,
nikita(# model varchar(50) NOT NULL,
nikita(# year_of_release smallint NOT NULL,
nikita(# color varchar(30) NOT NULL
nikita(# );
CREATE TABLE
nikita=# \d

```

Schema	Name	Type	Owner
public	bicycle	table	nikita
public	bicycle_bicycle_id_seq	sequence	nikita

(2 rows)

```

nikita=# \dt

```

Schema	Name	Type	Owner
public	bicycle	table	nikita

(1 row)

Рисунок 7 – Создание и просмотр базы данных

В созданной базе данных была сформирована таблица, заполнена данными (Рисунок 8) и проверена с помощью оператора SELECT (Рисунки 9 и 11). Также были добавлены новые записи (Рисунок 10), что показало умение выполнять основные операции с данными.

```

nikita=# INSERT INTO bicycle (brand, model, year_of_release, color) VALUES
nikita-# ('Scott', 'Spark RC World Cup EVO', 2025, 'carbon black'),
nikita-# ('Colnago', 'Y1Rs', 2025, 'UAE Emirates'),
nikita-# ('Specialized', 'S-Works Epic 8', 2025, 'Gloss Carbon');
INSERT 0 3

```

Рисунок 8 – Операция заполнения таблицы данными

```

nikita=# SELECT * FROM bicycle;
bicycle_id | brand      | model                | year_of_release | color
-----+-----+-----+-----+-----
          1 | Scott     | Spark RC World Cup EVO |          2025 | carbon black
          2 | Colnago   | Y1Rs                 |          2025 | UAE Emirates
          3 | Specialized | S-Works Epic 8       |          2025 | Gloss Carbon
(3 rows)

```

Рисунок 9 – Использование оператора SELECT для вывода записей на экран

```

nikita=# INSERT INTO bicycle (brand, model, year_of_release, color) VALUES
nikita-# -- Шоссейные велосипеды
nikita-# ('Specialized', 'S-Works Tarmac SL8', 2024, 'Red/Black'),
nikita-# ('Trek', 'Emonda SLR 9', 2024, 'Blue/White'),
nikita-# ('Canyon', 'Aeroad CFR', 2025, 'Stealth Black'),
nikita-# ('Pinarello', 'Dogma F', 2024, 'Team Colors'),
nikita-# ('Cervelo', 'S5', 2025, 'Black/Red'),
nikita-#
nikita-# -- Горные велосипеды (MTB)
nikita-# ('Santa Cruz', 'Hightower 3', 2024, 'Gloss Green'),
nikita-# ('Yeti', 'SB160', 2025, 'Turquoise'),
nikita-# ('Trek', 'Fuel EX 9.9', 2024, 'Matte Black'),
nikita-# ('Specialized', 'S-Works Epic EVO', 2025, 'Carbon/Red'),
nikita-# ('Canyon', 'Lux CFR', 2024, 'Yellow/Black'),
nikita-#
nikita-# -- Гравел-велосипеды
nikita-# ('Cannondale', 'SuperSix EVO SE', 2024, 'Olive Green'),
nikita-# ('Specialized', 'S-Works Crux', 2025, 'White/Black'),
nikita-# ('Trek', 'Checkpoint SLR 9', 2024, 'Blue/Grey'),
nikita-# ('Cervelo', 'Aspero-5', 2025, 'Gloss Black'),
nikita-# ('Scott', 'Addict Gravel Tuned', 2024, 'Red/White'),
nikita-#
nikita-# -- Городские велосипеды
nikita-# ('VanMoof', 'S5', 2024, 'Light Grey'),
nikita-# ('Gazelle', 'Ultimate C8+ HMB', 2025, 'Dark Blue');
INSERT 0 17

```

Рисунок 10 – Добавление новых данных в таблицу БД


```
nikita=# SELECT * FROM bicycle;
```

bicycle_id	brand	model	year_of_release	color
1	Scott	Spark RC World Cup EVO	2025	carbon black
2	Colnago	Y1Rs	2025	UAE Emirates
3	Specialized	S-Works Epic 8	2025	Gloss Carbon
4	Specialized	S-Works Tarmac SL8	2024	Red/Black
5	Trek	Emonda SLR 9	2024	Blue/White
6	Canyon	Aeroad CFR	2025	Stealth Black
7	Pinarello	Dogma F	2024	Team Colors
8	Cervelo	S5	2025	Black/Red
9	Santa Cruz	Hightower 3	2024	Gloss Green
10	Yeti	SB160	2025	Turquoise
11	Trek	Fuel EX 9.9	2024	Matte Black
12	Specialized	S-Works Epic EVO	2025	Carbon/Red
13	Canyon	Lux CFR	2024	Yellow/Black
14	Cannondale	SuperSix EVO SE	2024	Olive Green
15	Specialized	S-Works Crux	2025	White/Black
16	Trek	Checkpoint SLR 9	2024	Blue/Grey
17	Cervelo	Aspero-5	2025	Gloss Black
18	Scott	Addict Gravel Tuned	2024	Red/White
19	VanMoof	S5	2024	Light Grey
20	Gazelle	Ultimate C8+ HMB	2025	Dark Blue

(20 rows)

Рисунок 11 – Использование оператора SELECT для вывода всех записей БД

2.2 Модуль 7. Масштабируемость и отказоустойчивость

Для корректной работы кластера была выполнена синхронизация времени между узлами с помощью `timedatectl` (Рисунок 12). Это важно для согласованного взаимодействия серверов в распределённой системе.

```
root@PAVLOV:/home/nikita# sudo apt install -y chrony
Чтение списков пакетов... Готово
Построение дерева зависимостей... Готово
Чтение информации о состоянии... Готово
Уже установлен пакет chrony самой новой версии (4.3-2+deb12u1).
Обновлено 0 пакетов, установлено 0 новых пакетов, для удаления отмечено 0 пакетов, и 0 пакетов не обновлено.
root@PAVLOV:/home/nikita# sudo systemctl enable --now chrony
Synchronizing state of chrony.service with SysV service script with /lib/systemd/systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install enable chrony
root@PAVLOV:/home/nikita# sudo timedatectl set-timezone Europe/Moscow
root@PAVLOV:/home/nikita#
```

Рисунок 12 – Настройка синхронизации времени

Проверены параметры IP-адресов узлов (Рисунок 13), созданы и распределены SSH-ключи (Рисунки 14–15). Это обеспечило безопасное управление узлами кластера без необходимости ввода пароля.

```
GNU nano 7.2 hosts
127.0.0.1 localhost
127.0.1.1 PAVLOV.nikita PAVLOV
192.168.1.125 node0
192.168.1.60 node1
192.168.1.187 node2

# The following lines are desirable for IPv6 capable hosts
::1 localhost ip6-localhost ip6-loopback
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
```

Рисунок 13 – Параметры ip узлов в сети

```
root@PAVLOV:/home/nikita# ssh-keygen -t rsa -b 4096
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /root/.ssh/id_rsa
Your public key has been saved in /root/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:+umlEbYElw7wf4Lxxg0hViRPQ5CXfyRuieSD/BcKs6Y root@PAVLOV
The key's randomart image is:
+----[RSA 4096]-----+
|      o=*o      |
|      o=B..    |
|    ..@.= +    |
|    B.X * .    |
|    . *S@ o    |
|    o++o0 .    |
|    o. oo.+    |
|    E . =      |
|    . =        |
+----[SHA256]-----+
```

Рисунок 14 – Создание ssh-ключа

```

root@PAVLOV:/home/nikita# ssh-copy-id node1
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/root/.ssh/id_rsa.pub"
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted now it is to install the new keys
root@node1's password:

Number of key(s) added: 1

Now try logging into the machine, with:  "ssh 'node1'"
and check to make sure that only the key(s) you wanted were added.

root@PAVLOV:/home/nikita#

```

Рисунок 15 – Настройка связи по ssh

Установлен пароль для пользователя hacluster (Рисунок 16) и выполнена аутентификация на узле node1 (Рисунок 17).

```

root@PAVLOV:/etc/ssh# sudo passwd hacluster
Новый пароль:
Повторите ввод нового пароля:
passwd: пароль успешно обновлён
root@PAVLOV:/etc/ssh#

```

Рисунок 16 – Смена пароля пользователя hacluster

```

root@PAVLOV:/home/nikita# sudo pcs host auth node1
Username: hacluster
Password:
node1: Authorized
root@PAVLOV:/home/nikita#

```

Рисунок 17 – Аутентификация на node1

Создан кластер pgcluster (Рисунок 18), запущен и активирован (Рисунки 19–20). Проверка статуса (Рисунок 21) подтвердила его работоспособность.

```

root@PAVLOV:/home/nikita# sudo pcs cluster setup --force pgcluster node0 addr=192.168.1.125 node1
addr=192.168.1.60 node2 addr=192.168.1.187
No addresses specified for host 'node1', using 'node1'
Warning: node1: The host seems to be in a cluster already as the following services are found to be running: 'corosync', 'pacemaker'. If the host is not part of a cluster, stop the services and retry
Warning: node1: The host seems to be in a cluster already as cluster configuration files have been found on the host
. If the host is not part of a cluster, run 'pcs cluster destroy' on host 'node1' to remove those configuration files
Destroying cluster on hosts: 'node0', 'node1'...
node0: Successfully destroyed cluster
node1: Successfully destroyed cluster
Requesting remove 'pcsd settings' from 'node0', 'node1'
node0: successful removal of the file 'pcsd settings'
node1: successful removal of the file 'pcsd settings'
Sending 'corosync authkey', 'pacemaker authkey' to 'node0', 'node1'
node0: successful distribution of the file 'corosync authkey'
node0: successful distribution of the file 'pacemaker authkey'
node1: successful distribution of the file 'corosync authkey'
node1: successful distribution of the file 'pacemaker authkey'
Sending 'corosync.conf' to 'node0', 'node1'
node0: successful distribution of the file 'corosync.conf'
node1: successful distribution of the file 'corosync.conf'
Cluster has been successfully set up.
bash: node2: команда не найдена
root@PAVLOV:/home/nikita#

```

Рисунок 18 – Создание кластера pgcluster

```
root@PAVLOV:/home/nikita# sudo pcs cluster start --all
node0: Starting Cluster...
node1: Starting Cluster...
```

Рисунок 19 – Запуск кластера

```
root@PAVLOV:/home/nikita# sudo pcs cluster enable --all
node0: Cluster Enabled
node1: Cluster Enabled
```

Рисунок 20 – Выполнение enable

```
root@PAVLOV:/home/nikita# sudo pcs status
Cluster name: pgcluster

WARNINGS:
No stonith devices and stonith-enabled is not false

Status of pacemakerd: 'Pacemaker is running' (last updated 2025-05-01 13:34:08 +03:00)
Cluster Summary:
* Stack: corosync
* Current DC: node0 (version 2.1.5-a3f44794f94) - partition with quorum
* Last updated: Thu May 1 13:34:08 2025
* Last change: Thu May 1 13:33:33 2025 by hacluster via crmd on node0
* 2 nodes configured
* 0 resource instances configured

Node List:
* Online: [ node0 node1 ]

Full List of Resources:
* No resources

Daemon Status:
corosync: active/enabled
pacemaker: active/enabled
pcsd: active/enabled
root@PAVLOV:/home/nikita#
```

Рисунок 21 – Проверка статуса кластера

Создана кластерная БД (Рисунок 22), запущена (Рисунок 23) и настроена (Рисунок 25). Смена пароля пользователя (Рисунок 24) обеспечила безопасность доступа.

```
root@PAVLOV:/home/nikita# sudo systemctl stop postgresql
root@PAVLOV:/home/nikita# sudo rm -rf /var/lib/postgresql/12/main/*
root@PAVLOV:/home/nikita# sudo -u postgres /usr/lib/postgresql/12/bin/initdb -D /var/lib/postgresql/12/main
could not change directory to "/home/nikita": Отказано в доступе
The files belonging to this database system will be owned by user "postgres".
This user must also own the server process.

The database cluster will be initialized with locale "ru_RU.UTF-8".
The default database encoding has accordingly been set to "UTF8".
The default text search configuration will be set to "russian".
```

Рисунок 22 – Создание кластера БД


```

root@PAVLOV:/home/nikita# sudo -u postgres /usr/lib/postgresql/12/bin/pg_ctl -D /var/lib/postgresql/12/main start
could not change directory to "/home/nikita": Отказано в доступе
waiting for server to start...2025-05-01 13:41:39.033 MSK [16170] СООБЩЕНИЕ:  запускается PostgreSQL 12.22 (Debian
12.22-2.pgdg120+1) on x86_64-pc-linux-gnu, compiled by gcc (Debian 12.2.0-14) 12.2.0, 64-bit
2025-05-01 13:41:39.033 MSK [16170] СООБЩЕНИЕ:  для приёма подключений по адресу IPv6 "::1" открыт порт 5432
2025-05-01 13:41:39.033 MSK [16170] СООБЩЕНИЕ:  для приёма подключений по адресу IPv4 "127.0.0.1" открыт порт 5432
2025-05-01 13:41:39.035 MSK [16170] СООБЩЕНИЕ:  для приёма подключений открыт Unix-сокет "/var/run/postgresql/.s.PGSQL.5432"
2025-05-01 13:41:39.044 MSK [16171] СООБЩЕНИЕ:  система БД была выключена: 2025-05-01 13:41:09 MSK
2025-05-01 13:41:39.047 MSK [16170] СООБЩЕНИЕ:  система БД готова принимать подключения
done
server started
root@PAVLOV:/home/nikita#

```

Рисунок 23 – Запуск БД

```

root@PAVLOV:/home/nikita# sudo -u postgres psql -c "ALTER USER postgres WITH PASSWORD '12345';"
ALTER ROLE
root@PAVLOV:/home/nikita# sudo -u postgres createuser --replication -P repl
Введите пароль для новой роли:
Повторите его:
root@PAVLOV:/home/nikita#

```

Рисунок 24 – Смена пароля пользователя

```

# replication privilege.
local    replication    all                                     trust
host     replication    all                                     127.0.0.1/32          trust
host     replication    all                                     ::1/128               trust
host     replication    all                                     192.168.1.60/24       trust
host     all            all                                     192.168.1.60/24       trust

```

Рисунок 25 – Содержимое файла конфигурации

2.3 Модуль 8. Мониторинг, логирование и оповещение событий

Загружен и распакован пакет `node_exporter` (Рисунки 26–27) для сбора системных метрик.

```
root@MateBook-D-16:/mnt/d/Учеба/ЦК/pract# wget https://github.com/prometheus/node_exporter/releases/download/v1.9.1/node_exporter-1.9.1.linux-amd64.tar.gz
--2025-05-25 15:39:00-- https://github.com/prometheus/node_exporter/releases/download/v1.9.1/node_exporter-1.9.1.linux-amd64.tar.gz
Resolving github.com (github.com)... 140.82.121.3
Connecting to github.com (github.com)|140.82.121.3|:443... connected.
HTTP request sent, awaiting response... 302 Found
Location: https://objects.githubusercontent.com/github-production-release-asset-2e65be/9524057/dc8ec09c-2975-42a2-9591-57dd1ffff7b7?X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Credential=releaseassetproduction%2F20250525%2Fus-east-1%2Fs3%2Faws4_request&X-Amz-Date=20250525T123900Z&X-Amz-Expires=300&X-Amz-Signature=052031c516984b2fa59e212cb63d48450c3ea28f2b8f24b6462dec6bfa8b1631&X-Amz-SignedHeaders=host&response-content-disposition=attachment%3B%20filename%3Dnode_exporter-1.9.1.linux-amd64.tar.gz&response-content-type=application%2Foctet-stream [following]
--2025-05-25 15:39:00-- https://objects.githubusercontent.com/github-production-release-asset-2e65be/9524057/dc8ec09c-2975-42a2-9591-57dd1ffff7b7?X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Credential=releaseassetproduction%2F20250525%2Fus-east-1%2Fs3%2Faws4_request&X-Amz-Date=20250525T123900Z&X-Amz-Expires=300&X-Amz-Signature=052031c516984b2fa59e212cb63d48450c3ea28f2b8f24b6462dec6bfa8b1631&X-Amz-SignedHeaders=host&response-content-disposition=attachment%3B%20filename%3Dnode_exporter-1.9.1.linux-amd64.tar.gz&response-content-type=application%2Foctet-stream
Resolving objects.githubusercontent.com (objects.githubusercontent.com)... 185.199.108.133, 185.199.109.133, 185.199.110.133, ...
Connecting to objects.githubusercontent.com (objects.githubusercontent.com)|185.199.108.133|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 11582410 (11M) [application/octet-stream]
Saving to: 'node_exporter-1.9.1.linux-amd64.tar.gz'

node_exporter-1.9.1.linux-amd 100%[=====>] 11.04M  3.54MB/s   in 3.1s

2025-05-25 15:39:04 (3.54 MB/s) - 'node_exporter-1.9.1.linux-amd64.tar.gz' saved [11582410/11582410]
```

Рисунок 26 – Загрузка пакета `node_exporter`

```
root@MateBook-D-16:/mnt/d/Учеба/ЦК/pract# tar xvfz node_exporter-1.9.1.linux-amd64.tar.gz
node_exporter-1.9.1.linux-amd64/
node_exporter-1.9.1.linux-amd64/LICENSE
node_exporter-1.9.1.linux-amd64/NOTICE
node_exporter-1.9.1.linux-amd64/node_exporter
```

Рисунок 27 – Извлечение файла из архива

Настроены права доступа (Рисунок 28) и запущен сервис (Рисунки 29–30).
Проверка через порт 9100 (Рисунок 31) подтвердила сбор данных.

```
root@MateBook-D-16:/mnt/d/Учеба/ЦК/pract# cd node_exporter-1.9.1.linux-amd64/
root@MateBook-D-16:/mnt/d/Учеба/ЦК/pract/node_exporter-1.9.1.linux-amd64# chmod +x node_exporter
```

Рисунок 28 – Смена разрешений файла (исполняемый)

```

root@MateBook-D-16:/mnt/d/Y4eba/UK/pract/node_exporter-1.9.1.linux-amd64# ./node_exporter
time=2025-05-25T12:43:15.847Z level=INFO source=node_exporter.go:216 msg="Starting node_exporter" version="(version=1.9.1, branch=HEAD, revision=f2ec547b49af53815038a50265aa2adcd1275959)"
time=2025-05-25T12:43:15.847Z level=INFO source=node_exporter.go:217 msg="Build context" build_context="(go=go1.23.7, platform=linux/amd64, user=root@7023beaa563a, date=20250401-15:19:01, tags=unknown)"
time=2025-05-25T12:43:15.849Z level=WARN source=node_exporter.go:219 msg="Node Exporter is running as root user. This exporter is designed to run as unprivileged user, root is not required."
time=2025-05-25T12:43:15.850Z level=INFO source=filesystem_common.go:265 msg="Parsed flag --collector.filesystem.mount-points-exclude" collector=filesystem flag=^(/dev|proc|run|credentials/.+|sys|var/lib/docker/.+|var/lib/containers/storage/.+)(/|$)
time=2025-05-25T12:43:15.850Z level=INFO source=filesystem_common.go:294 msg="Parsed flag --collector.filesystem.fs-types-exclude" collector=filesystem flag=^(autofs|binfmt_misc|bpf|cgroup2?|configfs|debugfs|devpts|devtmpfs|fusectl|hugetlbfs|iso9660|queue|nsfs|overlay|proc|procfs|pstore|rpc_pipefs|securityfs|selinuxfs|squashfs|sysfs|tracefs)$
time=2025-05-25T12:43:15.851Z level=INFO source=diskstats_common.go:110 msg="Parsed flag --collector.diskstats.device-exclude" collector=diskstats flag=^(zram|loop|fd|(h|s|v|xv)d[a-z]|nvme\d+n\d+p)\d+$
time=2025-05-25T12:43:15.851Z level=ERROR source=diskstats_linux.go:264 msg="Failed to open directory, disabling udev device properties" collector=diskstats path=/run/udev/data
time=2025-05-25T12:43:15.852Z level=INFO source=node_exporter.go:135 msg="Enabled collectors"
time=2025-05-25T12:43:15.852Z level=INFO source=node_exporter.go:141 msg=arp
time=2025-05-25T12:43:15.852Z level=INFO source=node_exporter.go:141 msg=bcache
time=2025-05-25T12:43:15.852Z level=INFO source=node_exporter.go:141 msg=bonding
time=2025-05-25T12:43:15.852Z level=INFO source=node_exporter.go:141 msg=btrfs
time=2025-05-25T12:43:15.852Z level=INFO source=node_exporter.go:141 msg=conntrack
time=2025-05-25T12:43:15.852Z level=INFO source=node_exporter.go:141 msg=cpu
time=2025-05-25T12:43:15.852Z level=INFO source=node_exporter.go:141 msg=cputime
time=2025-05-25T12:43:15.852Z level=INFO source=node_exporter.go:141 msg=diskstats
time=2025-05-25T12:43:15.852Z level=INFO source=node_exporter.go:141 msg=dmi
time=2025-05-25T12:43:15.852Z level=INFO source=node_exporter.go:141 msg=edac
time=2025-05-25T12:43:15.852Z level=INFO source=node_exporter.go:141 msg=entropy
time=2025-05-25T12:43:15.852Z level=INFO source=node_exporter.go:141 msg=fibrechannel
time=2025-05-25T12:43:15.852Z level=INFO source=node_exporter.go:141 msg=filefd
time=2025-05-25T12:43:15.852Z level=INFO source=node_exporter.go:141 msg=filesystem
time=2025-05-25T12:43:15.852Z level=INFO source=node_exporter.go:141 msg=hwmon
time=2025-05-25T12:43:15.852Z level=INFO source=node_exporter.go:141 msg=infiniband
time=2025-05-25T12:43:15.852Z level=INFO source=node_exporter.go:141 msg=ipvs
time=2025-05-25T12:43:15.852Z level=INFO source=node_exporter.go:141 msg=loadavg

```

Рисунок 29 – Запуск node_exporter (часть 1)

```

time=2025-05-25T12:43:15.852Z level=INFO source=node_exporter.go:141 msg=mdadm
time=2025-05-25T12:43:15.852Z level=INFO source=node_exporter.go:141 msg=meminfo
time=2025-05-25T12:43:15.852Z level=INFO source=node_exporter.go:141 msg=netclass
time=2025-05-25T12:43:15.852Z level=INFO source=node_exporter.go:141 msg=netdev
time=2025-05-25T12:43:15.852Z level=INFO source=node_exporter.go:141 msg=netstat
time=2025-05-25T12:43:15.852Z level=INFO source=node_exporter.go:141 msg=nfs
time=2025-05-25T12:43:15.852Z level=INFO source=node_exporter.go:141 msg=nfsd
time=2025-05-25T12:43:15.852Z level=INFO source=node_exporter.go:141 msg=nvme
time=2025-05-25T12:43:15.852Z level=INFO source=node_exporter.go:141 msg=os
time=2025-05-25T12:43:15.852Z level=INFO source=node_exporter.go:141 msg=powersupplyclass
time=2025-05-25T12:43:15.852Z level=INFO source=node_exporter.go:141 msg=pressure
time=2025-05-25T12:43:15.852Z level=INFO source=node_exporter.go:141 msg=rapl
time=2025-05-25T12:43:15.852Z level=INFO source=node_exporter.go:141 msg=schedstat
time=2025-05-25T12:43:15.852Z level=INFO source=node_exporter.go:141 msg=selinux
time=2025-05-25T12:43:15.852Z level=INFO source=node_exporter.go:141 msg=sockstat
time=2025-05-25T12:43:15.852Z level=INFO source=node_exporter.go:141 msg=softnet
time=2025-05-25T12:43:15.852Z level=INFO source=node_exporter.go:141 msg=stat
time=2025-05-25T12:43:15.852Z level=INFO source=node_exporter.go:141 msg=tapestats
time=2025-05-25T12:43:15.852Z level=INFO source=node_exporter.go:141 msg=textfile
time=2025-05-25T12:43:15.852Z level=INFO source=node_exporter.go:141 msg=thermal_zone
time=2025-05-25T12:43:15.852Z level=INFO source=node_exporter.go:141 msg=time
time=2025-05-25T12:43:15.852Z level=INFO source=node_exporter.go:141 msg=timex
time=2025-05-25T12:43:15.852Z level=INFO source=node_exporter.go:141 msg=udp_queues
time=2025-05-25T12:43:15.852Z level=INFO source=node_exporter.go:141 msg=uname
time=2025-05-25T12:43:15.852Z level=INFO source=node_exporter.go:141 msg=vmstat
time=2025-05-25T12:43:15.852Z level=INFO source=node_exporter.go:141 msg=watchdog
time=2025-05-25T12:43:15.852Z level=INFO source=node_exporter.go:141 msg=xfs
time=2025-05-25T12:43:15.852Z level=INFO source=node_exporter.go:141 msg=zfs
time=2025-05-25T12:43:15.861Z level=INFO source=tls_config.go:347 msg="Listening on" address=[::]:9100
time=2025-05-25T12:43:15.862Z level=INFO source=tls_config.go:350 msg="TLS is disabled." http2=false address=[::]:9100

```

Рисунок 30 – Запуск node_exporter (часть 2)

```

root@MateBook-D-16:/mnt/d/Y4eba/UK/pract/node_exporter-1.9.1.linux-amd64# curl http://localhost:9100/metrics
# HELP go_gc_duration_seconds A summary of the wall-time pause (stop-the-world) duration in garbage collection cycles.
# TYPE go_gc_duration_seconds summary
go_gc_duration_seconds{quantile="0"} 0
go_gc_duration_seconds{quantile="0.25"} 0
go_gc_duration_seconds{quantile="0.5"} 0
go_gc_duration_seconds{quantile="0.75"} 0
go_gc_duration_seconds{quantile="1"} 0
go_gc_duration_seconds_sum 0
go_gc_duration_seconds_count 0

```

Рисунок 31 – Проверка метрик через порт 9100

Загружен архив Prometheus (Рисунок 32), распакован (Рисунок 33) и настроен конфигурационный файл (Рисунок 34).

```
root@MateBook-D-16:/mnt/d/Учеба/ЦК/pract# wget https://github.com/prometheus/prometheus/releases/download/v3.3.1/prometheus-3.3.1.linux-amd64.tar.gz
--2025-05-25 15:51:17-- https://github.com/prometheus/prometheus/releases/download/v3.3.1/prometheus-3.3.1.linux-amd64.tar.gz
Resolving github.com (github.com)... 140.82.121.3
Connecting to github.com (github.com)|140.82.121.3|:443... connected.
HTTP request sent, awaiting response... 302 Found
Location: https://objects.githubusercontent.com/github-production-release-asset-2e65be/6838921/39e5bf8c-512d-4389-9add-1986b432ee5b?X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Credential=releaseassetproduction%2F20250525%2Fus-east-1%2Fs3%2Faws4_request&X-Amz-Date=20250525T125117Z&X-Amz-Expires=300&X-Amz-Signature=1d8d19f4d8962d5414529851567e70f0f1134124d2754ad9b5b12255fc070117&X-Amz-SignedHeaders=host&response-content-disposition=attachment%3B%20filename%3Dprometheus-3.3.1.linux-amd64.tar.gz&response-content-type=application%2Foctet-stream [following]
--2025-05-25 15:51:17-- https://objects.githubusercontent.com/github-production-release-asset-2e65be/6838921/39e5bf8c-512d-4389-9add-1986b432ee5b?X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Credential=releaseassetproduction%2F20250525%2Fus-east-1%2Fs3%2Faws4_request&X-Amz-Date=20250525T125117Z&X-Amz-Expires=300&X-Amz-Signature=1d8d19f4d8962d5414529851567e70f0f1134124d2754ad9b5b12255fc070117&X-Amz-SignedHeaders=host&response-content-disposition=attachment%3B%20filename%3Dprometheus-3.3.1.linux-amd64.tar.gz&response-content-type=application%2Foctet-stream
Resolving objects.githubusercontent.com (objects.githubusercontent.com)... 185.199.108.133, 185.199.109.133, 185.199.110.133, ...
Connecting to objects.githubusercontent.com (objects.githubusercontent.com)|185.199.108.133|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 116682535 (111M) [application/octet-stream]
Saving to: 'prometheus-3.3.1.linux-amd64.tar.gz'

prometheus-3.3.1.linux-amd64. 100%[=====>] 111.28M  4.77MB/s   in 21s

2025-05-25 15:51:38 (5.39 MB/s) - 'prometheus-3.3.1.linux-amd64.tar.gz' saved [116682535/116682535]
```

Рисунок 32 – Загрузка архива пакета Prometheus

```
root@MateBook-D-16:/mnt/d/Учеба/ЦК/pract# tar xvf prometheus-3.3.1.linux-amd64.tar.gz
prometheus-3.3.1.linux-amd64/
prometheus-3.3.1.linux-amd64/LICENSE
prometheus-3.3.1.linux-amd64/NOTICE
prometheus-3.3.1.linux-amd64/prometheus
prometheus-3.3.1.linux-amd64/prometheus.yml
prometheus-3.3.1.linux-amd64/promtool
root@MateBook-D-16:/mnt/d/Учеба/ЦК/pract# cd prometheus-3.3.1.linux-amd64/
```

Рисунок 33 – Распаковка архива

```
global:
  scrape_interval: 15s

scrape_configs:
- job_name: node
  static_configs:
  - targets: ['localhost:9100']
```

Рисунок 34 – Данные файла конфигурации

Запуск сервиса (Рисунок 35) обеспечил сбор и хранение метрик с Node Exporter.


```

root@MateBook-D-16:/mnt/d/Учеба/ЦК/pract/prometheus-3.3.1.linux-amd64# ./prometheus --config.file=./prometheus.yml
time=2025-05-25T12:59:35.189Z level=INFO source=main.go:1487 msg="updated GOGC" old=100 new=75
time=2025-05-25T12:59:35.192Z level=INFO source=main.go:625 msg="Leaving GOMAXPROCS=12: CPU quota undefined" component=automaxprocs
time=2025-05-25T12:59:35.192Z level=INFO source=main.go:666 msg="No time or size retention was set so using the default time retention" duration=15d
time=2025-05-25T12:59:35.192Z level=INFO source=main.go:713 msg="Starting Prometheus Server" mode=server version="(version=3.3.1, branch=HEAD, revision=3dcecabff6f52cc155290cb723706f4c324ed0ab)"
time=2025-05-25T12:59:35.192Z level=INFO source=main.go:718 msg="operational information" build_context="(go=go1.24.2, platform=linux/amd64, user=root@f0d295ca0db2, date=20250502-15:03:21, tags=netgo,builtinassets,stringlabels)" host_details="(Linux 5.10.16.3-microsoft-standard-WSL2 #1 SMP Fri Apr 2 22:23:49 UTC 2021 x86_64 MateBook-D-16 localdomain)" fd_limits="(soft=1048575, hard=1048576)" vm_limits="(soft=unlimited, hard=unlimited)"
time=2025-05-25T12:59:35.213Z level=INFO source=web.go:654 msg="Start listening for connections" component=web address=0.0.0.0:9090
time=2025-05-25T12:59:35.214Z level=INFO source=main.go:1231 msg="Starting TSDB ..."
time=2025-05-25T12:59:35.224Z level=INFO source=tls_config.go:347 msg="Listening on" component=web address=[::]:9090
time=2025-05-25T12:59:35.224Z level=INFO source=tls_config.go:350 msg="TLS is disabled." component=web http2=false address=[::]:9090
time=2025-05-25T12:59:35.265Z level=INFO source=head.go:638 msg="Replaying on-disk memory mappable chunks if any" component=tsdb
time=2025-05-25T12:59:35.269Z level=INFO source=head.go:725 msg="On-disk memory mappable chunks replay completed" component=tsdb duration=5.168µs
time=2025-05-25T12:59:35.273Z level=INFO source=head.go:733 msg="Replaying WAL, this may take a while" component=tsdb
time=2025-05-25T12:59:35.281Z level=INFO source=head.go:805 msg="WAL segment loaded" component=tsdb segment=0 maxSegment=0
time=2025-05-25T12:59:35.282Z level=INFO source=head.go:842 msg="WAL replay completed" component=tsdb checkpoint_replay_duration=3.349012ms wal_replay_duration=5.65035ms wbl_replay_duration=219ns chunk_snapshot_load_duration=0s mmap_chunk_replay_duration=5.168µs total_replay_duration=12.493684ms
time=2025-05-25T12:59:35.285Z level=INFO source=main.go:1252 msg="filesystem information" fs_type=1021997
time=2025-05-25T12:59:35.285Z level=INFO source=main.go:1255 msg="TSDB started"
time=2025-05-25T12:59:35.285Z level=INFO source=main.go:1440 msg="Loading configuration file" filename=./prometheus.yml
time=2025-05-25T12:59:35.288Z level=INFO source=main.go:1480 msg="Completed loading of configuration file" db_storage=1.466µs remote_storage=1.467µs web_handler=530ns query_engine=1.184µs scrape=192.38µs scrape_sd=36.408µs notify=1.409µs notify_sd=720ns rules=1.84µs tracing=5.963µs filename=./prometheus.yml totalDuration=3.04748ms
time=2025-05-25T12:59:35.288Z level=INFO source=main.go:1216 msg="Server is ready to receive web requests."
time=2025-05-25T12:59:35.288Z level=INFO source=manager.go:175 msg="Starting rule manager..." component="rule manager"

```

Рисунок 35 – Запуск Prometheus

Установлен пакет Grafana (Рисунки 36–37), сервис запущен и проверен (Рисунок 38).

```

root@MateBook-D-16:/mnt/d/Учеба/ЦК/pract# wget https://dl.grafana.com/enterprise/release/grafana-enterprise_12.0.1_amd64.deb
--2025-05-25 16:11:16-- https://dl.grafana.com/enterprise/release/grafana-enterprise_12.0.1_amd64.deb
Resolving dl.grafana.com (dl.grafana.com)... 146.75.118.217, 2a04:4e42:8d::729
Connecting to dl.grafana.com (dl.grafana.com)|146.75.118.217|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 182410844 (174M) [application/octet-stream]
Saving to: 'grafana-enterprise_12.0.1_amd64.deb'

grafana-enterprise_12.0.1_amd 100%[=====>] 173.96M 6.69MB/s in 21s

2025-05-25 16:11:38 (8.30 MB/s) - 'grafana-enterprise_12.0.1_amd64.deb' saved [182410844/182410844]

```

Рисунок 36 – Загрузка пакета Grafana

```

root@MateBook-D-16:/mnt/d/Учеба/ЦК/pract# sudo dpkg -i grafana-enterprise_12.0.1_amd64.deb
Selecting previously unselected package grafana-enterprise.
(Reading database ... 52797 files and directories currently installed.)
Preparing to unpack grafana-enterprise_12.0.1_amd64.deb ...
Unpacking grafana-enterprise (12.0.1) ...
Setting up grafana-enterprise (12.0.1) ...
info: Selecting UID from range 100 to 999 ...

info: Adding system user `grafana' (UID 106) ...
info: Adding new user `grafana' (UID 106) with group `grafana' ...
info: Not creating home directory `/usr/share/grafana'.
### NOT starting on installation, please execute the following statements to configure grafana to start automatically using systemd
sudo /bin/systemctl daemon-reload
sudo /bin/systemctl enable grafana-server
### You can start grafana-server by executing
sudo /bin/systemctl start grafana-server

```

Рисунок 37 – Установка Grafana

```
root@MateBook-D-16:/mnt/d/Учеба/ЦК/pract# sudo service grafana-server start
* Starting Grafana Server
root@MateBook-D-16:/mnt/d/Учеба/ЦК/pract# sudo service grafana-server status
* grafana is running
```

Рисунок 38 – Запуск Grafana и проверка статуса

Настроен источник данных Prometheus (Рисунок 39) и импортирован готовый дашборд (Рисунок 40). Итоговая визуализация метрик (Рисунок 41) показала работоспособность стека мониторинга.

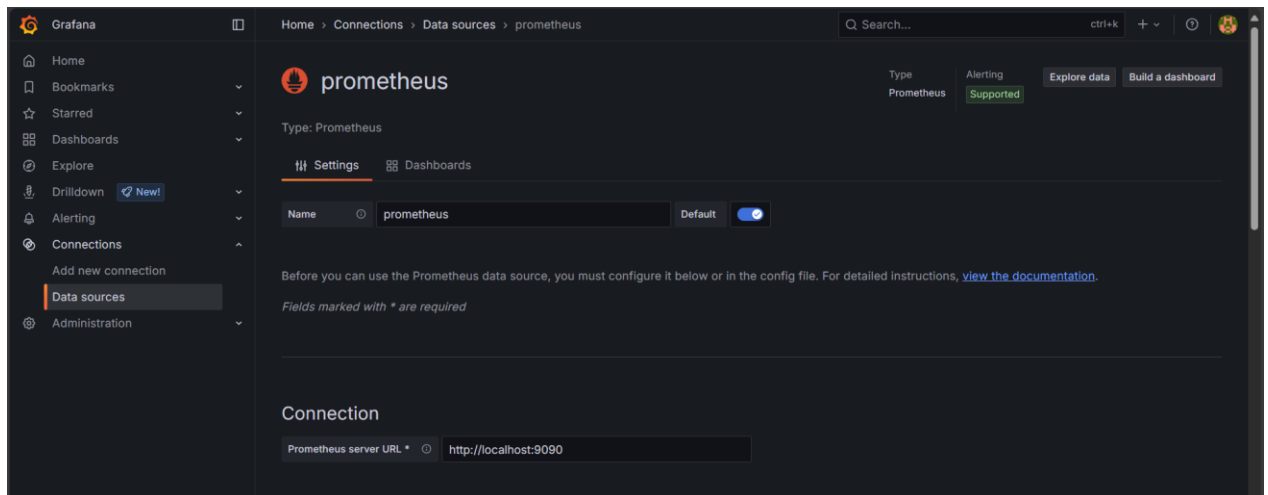


Рисунок 39 – Настройка источника данных

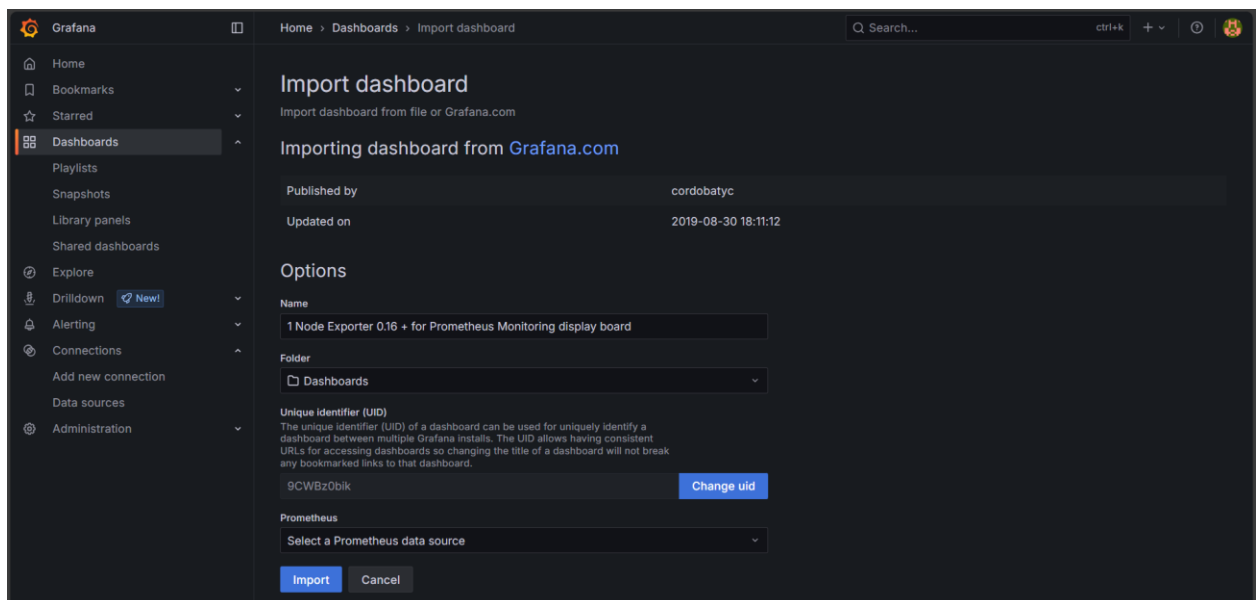


Рисунок 40 – Импорт готового дашборда

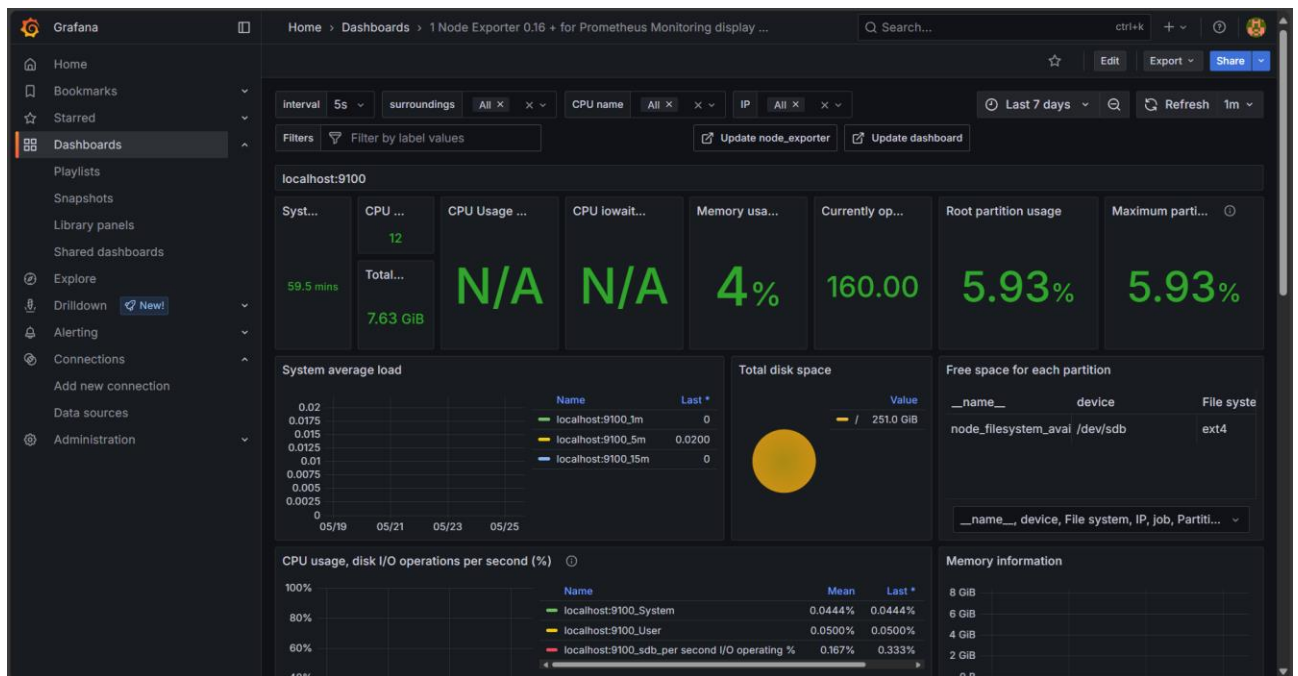


Рисунок 41 – Итоговые данные на дашборде

2.4 Модуль 9. Виртуализация в DevOps

На предварительно развернутой ВМ с Ubuntu 24.04 выполнено обновление пакетов системы (Рисунок 42) для обеспечения актуальности всех компонентов перед установкой дополнительных инструментов.

```
nikita@pavlov:~$ sudo apt-get update
[sudo] password for nikita:
Hit:1 http://ru.archive.ubuntu.com/ubuntu noble InRelease
Hit:2 http://ru.archive.ubuntu.com/ubuntu noble-updates InRelease
Hit:3 http://ru.archive.ubuntu.com/ubuntu noble-backports InRelease
Hit:4 http://security.ubuntu.com/ubuntu noble-security InRelease
Reading package lists... Done
```

Рисунок 42 – Обновление пакета установки

Установлен текстовый редактор vim (Рисунок 43) с проверкой его версии (Рисунок 44), что обеспечило удобное редактирование конфигурационных файлов в терминале.

```
nikita@pavlov:~$ sudo apt -y install vim
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  libsodium23 vim-runtime
Suggested packages:
  ctags vim-doc vim-scripts
The following NEW packages will be installed:
  libsodium23 vim vim-runtime
0 upgraded, 3 newly installed, 0 to remove and 111 not upgraded.
```

Рисунок 43 – Установка vim

```
nikita@pavlov:~$ vim --version
VIM - Vi IMproved 9.1(2024 Jan 02, compiled Apr 01 2025 20:12:31)
Included patches: 1-16, 647, 678, 697
Modified by team+vim@tracker.debian.org
Compiled by team+vim@tracker.debian.org
```

Рисунок 44 – Проверка версии vim

Инсталлирован файловый менеджер mc (Midnight Commander) (Рисунок 45) с подтверждением версии (Рисунок 46), значительно упрощающий навигацию по файловой системе.

```
nikita@pavlov:~$ sudo apt -y install mc
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  bzip2 libssh2-1t64 mailcap mc-data
```

Рисунок 45 – Установка mc

```
nikita@pavlov:~$ mc --version
GNU Midnight Commander 4.8.30
Built with GLib 2.80.0
Built with S-Lang 2.3.3 with terminfo database
Built with libssh2 1.11.0
With builtin Editor and Aspell support
With subshell support as default
With support for background operations
With mouse support on xterm and Linux console
With support for X11 events
With internationalization support
With multiple codepages support
With ext2fs attributes support
Virtual File Systems:
  cpiofs, tarfs, sfs, extfs, ext2undelfs, ftpfs, sftpfs, fish
Data types:
  char: 8; int: 32; long: 64; void *: 64; size_t: 64; off_t: 64;
```

Рисунок 46 – Проверка версии mc

Добавлен пакет net-tools (Рисунок 47), включающий утилиту ifconfig (Рисунок 48), что позволило проводить базовую диагностику сетевых подключений.

```
nikita@pavlov:~$ sudo apt -y install net-tools
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following NEW packages will be installed:
  net-tools
0 upgraded, 1 newly installed, 0 to remove and 111 not upgraded.
```

Рисунок 47 – Установка net-tools

```
nikita@pavlov:~$ ifconfig --version
net-tools 2.10
```

Рисунок 48 – Проверка версии ifconfig

2.5 Модуль 12. Конфигурационное управление. Что такое IaC

Установлена и проверена версия Ansible (Рисунок 49), что подтвердило готовность системы к автоматизации управления конфигурациями.

```
root@MateBook-D-16:/mnt/d/Учеба/ЦК/pract# ansible --version
ansible [core 2.18.6]
  config file = /etc/ansible/ansible.cfg
  configured module search path = ['/root/.ansible/plugins/modules', '/usr/share/ansible/plugins/modules']
  ansible python module location = /usr/lib/python3/dist-packages/ansible
  ansible collection location = /root/.ansible/collections:/usr/share/ansible/collections
  executable location = /usr/bin/ansible
  python version = 3.12.3 (main, Feb  4 2025, 14:48:35) [GCC 13.3.0] (/usr/bin/python3)
  jinja version = 3.1.2
  libyaml = True
```

Рисунок 49 – Установленная версия ansible

Проведена настройка базового конфигурационного файла Ansible (Рисунок 50), включая определение рабочих параметров и переменных среды.

```
GNU nano 7.2                                ansible.cfg *
inventory = /etc/ansible/hosts
host_key_checking = False
```

Рисунок 50 – Содержимое конфигурационного файла ansible


Выполнена верификация доступности управляемых узлов (Рисунок 51) с использованием инвентарного файла. Подтверждена корректность сетевых подключений и аутентификации между control-нодой и целевыми серверами

```
root@PAVLOV:/etc/ansible# ansible all -m ping
node2 | SUCCESS => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python3"
  },
  "changed": false,
  "ping": "pong"
}
node1 | SUCCESS => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python3"
  },
  "changed": false,
  "ping": "pong"
}
root@PAVLOV:/etc/ansible#
```

Рисунок 51 – Проверка доступности узлов

Создан комплексный playbook (Рисунки 52-53), содержащий:

- Настройку базовых системных параметров
- Управление пакетами и сервисами
- Конфигурацию сетевых компонентов
- Развертывание тестового окружения



```
GNU nano 7.2 group_vars/all.yml *
---
folders:
  - /opt/folder1
  - /opt/folder2

user_name: test
user_home: /home/home_dir_test
user_shell: /bin/bash
user_groups:
  - dev
  - test

packages:
  - vim
  - mc
```

Рисунок 52 – Содержимое playbook'а (часть 1)



```
GNU nano 7.2
--
- name: Setup Debian nodes
  hosts: debian_nodes
  become: yes

  tasks:
    - name: Create user groups
      group:
        name: "{{ item }}"
        state: present
      loop: "{{ user_groups }}"

    - name: Create user
      user:
        name: "{{ user_name }}"
        shell: "{{ user_shell }}"
        home: "{{ user_home }}"
        groups: "{{ user_groups | join(',') }}"
        append: yes
        state: present

    - name: Create folders
      file:
        path: "{{ item }}"
        state: directory
        owner: "{{ user_name }}"
        group: "{{ user_name }}"
        mode: '0755'
      loop: "{{ folders }}"

    - name: Install packages only on node2
      apt:
        name: "{{ packages }}"
        state: latest
        update_cache: yes
      when: inventory_hostname == "node2"
```

Рисунок 53 – Содержимое playbook'а (часть 2)

Playbook успешно применен к целевым узлам (Рисунок 54).

```
root@PAVLOV:~/ansible# ansible-playbook debian-setup.yml

PLAY [Setup Debian nodes] *****

TASK [Gathering Facts] *****
ok: [node1]
ok: [node2]

TASK [Create user groups] *****
ok: [node1] => (item=dev)
ok: [node2] => (item=dev)
ok: [node1] => (item=test)
ok: [node2] => (item=test)

TASK [Check if user exists] *****
changed: [node2]
changed: [node1]

TASK [Create user only if not exists] *****
skipping: [node1]
skipping: [node2]

TASK [Create folders] *****
changed: [node1] => (item=/opt/folder1)
changed: [node2] => (item=/opt/folder1)
changed: [node1] => (item=/opt/folder2)
changed: [node2] => (item=/opt/folder2)

TASK [Install packages only on node2] *****
skipping: [node1]
changed: [node2]

PLAY RECAP *****
node1      : ok=4    changed=2    unreachable=0    failed=0    skipped=2    rescued=0    ignored=0
node2      : ok=5    changed=3    unreachable=0    failed=0    skipped=1    rescued=0    ignored=0
```

Рисунок 54 – Результаты запуска playbook'a

2.6 Модуль 13. Системы контроля версий. Распределённая система управления версиями Git

Установлен и настроен Git (Рисунок 55), включая базовую конфигурацию пользователя (Рисунок 57). Создан системный пользователь git (Рисунок 56) для безопасного управления репозиториями.

```
nikita@pavlov:~$ sudo apt install git
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
git is already the newest version (1:2.43.0-1ubuntu7.2).
0 upgraded, 0 newly installed, 0 to remove and 112 not upgraded.
nikita@pavlov:~$ git --version
git version 2.43.0
```

Рисунок 55 – Установка Git

```
nikita@pavlov:~$ sudo adduser git
info: Adding user `git' ...
info: Selecting UID/GID from range 1000 to 59999 ...
info: Adding new group `git' (1001) ...
info: Adding new user `git' (1001) with group `git (1001)' ...
info: Creating home directory `/home/git' ...
info: Copying files from `/etc/skel' ...
New password:
Retype new password:
passwd: password updated successfully
Changing the user information for git
Enter the new value, or press ENTER for the default
    Full Name []: Nikita
    Room Number []:
    Work Phone []:
    Home Phone []:
    Other []:
Is the information correct? [Y/n] y
info: Adding new user `git' to supplemental / extra groups `users' ...
info: Adding user `git' to group `users' ...
```

Рисунок 56 – Создание пользователя Git

```
nikita@pavlov:~$ sudo su - git
git@pavlov:~$ git config --global user.email "nikniknik388@gmail.com"
git@pavlov:~$ git config --global user.name "n1k_pavlov"
```

Рисунок 57 – Предварительные настройки Git

Инициализирован новый репозиторий (Рисунок 58).

```

git@pavlov:~$ git init ~/tempgitsrv.git
hint: Using 'master' as the name for the initial branch. This default branch name
hint: is subject to change. To configure the initial branch name to use in all
hint: of your new repositories, which will suppress this warning, call:
hint:
hint:   git config --global init.defaultBranch <name>
hint:
hint: Names commonly chosen instead of 'master' are 'main', 'trunk' and
hint: 'development'. The just-created branch can be renamed via this command:
hint:
hint:   git branch -m <name>
Initialized empty Git repository in /home/git/tempgitsrv.git/.git/
git@pavlov:~$ cd ~/tempgitsrv.git/
git@pavlov:~/tempgitsrv.git$ touch README.md

```

Рисунок 58 – Инициализация репозитория и добавление файла

Отработана процедура добавления файлов в индекс и создания коммитов (Рисунок 59).

```

git@pavlov:~/tempgitsrv.git$ git commit -m "empty README.md"
[master (root-commit) 5a9d5db] empty README.md
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 README.md

```

Рисунок 59 – Коммит изменения

Выполнено клонирование репозитория (Рисунок 60), демонстрирующее типовой workflow разработчика.

```

git@pavlov:~/tempgitsrv.git$ mkdir ~/tempgit
git@pavlov:~/tempgitsrv.git$ cd ~/tempgit
git@pavlov:~/tempgit$ git clone git@github.com:DrTECHNIC/tempgitsrv.git tempgitsrv.git
Cloning into 'tempgitsrv.git'...
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Total 3 (delta 0), reused 3 (delta 0), pack-reused 0 (from 0)
Receiving objects: 100% (3/3), done.

```

Рисунок 60 – Клонирование Git локально

Создана и переключена на новую ветку (Рисунок 61). Выполнена отправка изменений в удаленный репозиторий (Рисунок 62). Отработаны операции слияния веток (Рисунок 64). Продемонстрировано отображение всех веток репозитория (Рисунок 63).

```
git@pavlov:~/tempgit$ cd tempgitsrv.git/
git@pavlov:~/tempgit/tempgitsrv.git$ git checkout -b readme
Switched to a new branch 'readme'
```

Рисунок 61 – Создание новой ветки

```
git@pavlov:~/tempgit/tempgitsrv.git$ echo '# Readme' > Readme.md
git@pavlov:~/tempgit/tempgitsrv.git$ git add Readme.md
git@pavlov:~/tempgit/tempgitsrv.git$ git commit -m "Readme.md"
[readme 5c1c081] Readme.md
 1 file changed, 1 insertion(+)
git@pavlov:~/tempgit/tempgitsrv.git$ git push --set-upstream origin readme
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Writing objects: 100% (3/3), 262 bytes | 262.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
remote:
remote: Create a pull request for 'readme' on GitHub by visiting:
remote:   https://github.com/DrTECHNIC/tempgitsrv/pull/new/readme
remote:
To github.com:DrTECHNIC/tempgitsrv.git
 * [new branch]      readme -> readme
Branch 'readme' set up to track remote branch 'readme' from 'origin'.
```

Рисунок 62 – Отправка изменений в репозиторий

```
git@pavlov:~/tempgit/tempgitsrv.git$ git branch -vv
* master 3f28c1d [origin/master] empty Readme.md
  readme 419da35 [origin/readme] Readme.md
```

Рисунок 63 – Список всех веток

```
git@pavlov:~/tempgit/tempgitsrv.git$ cat Readme.md
git@pavlov:~/tempgit/tempgitsrv.git$ git merge readme
Updating 3f28c1d..419da35
Fast-forward
  Readme.md | 1 +
  1 file changed, 1 insertion(+)
git@pavlov:~/tempgit/tempgitsrv.git$ cat Readme.md
# Readme
```

Рисунок 64 – Слияние веток

2.7 Модуль 14. Жизненный цикл ПО

Успешно выполнена установка Jenkins (Рисунки 65-66), включая:

- Настройку репозитория пакетов
- Установку необходимых зависимостей
- Запуск сервиса

```
nikita@pavlov:~$ sudo wget -O /etc/apt/keyrings/jenkins-keyring.asc \
> https://pkg.jenkins.io/debian-stable/jenkins.io-2023.key
[sudo] password for nikita:
--2025-06-01 09:01:41-- https://pkg.jenkins.io/debian-stable/jenkins.io-2023.key
Resolving pkg.jenkins.io (pkg.jenkins.io)... 199.232.174.133, 2a04:4e42:6b::645
Connecting to pkg.jenkins.io (pkg.jenkins.io)|199.232.174.133|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 3175 (3.1K) [application/pgp-keys]
Saving to: '/etc/apt/keyrings/jenkins-keyring.asc'

/etc/apt/keyrings/jenkins-key 100%[=====] 3.10K --.-KB/s in 0s
2025-06-01 09:01:41 (33.4 MB/s) - '/etc/apt/keyrings/jenkins-keyring.asc' saved [3175/3175]

nikita@pavlov:~$ echo "deb [signed-by=/etc/apt/keyrings/jenkins-keyring.asc]" \
> https://pkg.jenkins.io/debian-stable binary/ | sudo tee /etc/apt/sources.list.d/jenkins.list > /dev/null
nikita@pavlov:~$ sudo apt-get update
```

Рисунок 65 – Установка Jenkins (часть 1)

```
nikita@pavlov:~$ sudo apt-get install jenkins
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  net-tools
The following NEW packages will be installed:
  jenkins net-tools
0 upgraded, 2 newly installed, 0 to remove and 112 not upgraded.
Need to get 92.4 MB of archives.
After this operation, 95.2 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://ru.archive.ubuntu.com/ubuntu noble-updates/main amd64 net-tools amd64 2.10-0.1ubuntu4.4 [204 kB]
Get:2 https://pkg.jenkins.io/debian-stable binary/ jenkins 2.504.2 [92.2 MB]
Fetched 92.4 MB in 17s (5,304 kB/s)
Selecting previously unselected package net-tools.
(Reading database ... 150357 files and directories currently installed.)
Preparing to unpack .../net-tools_2.10-0.1ubuntu4.4_amd64.deb ...
Unpacking net-tools (2.10-0.1ubuntu4.4) ...
Selecting previously unselected package jenkins.
Preparing to unpack .../jenkins_2.504.2_all.deb ...
Unpacking jenkins (2.504.2) ...
Setting up net-tools (2.10-0.1ubuntu4.4) ...
Setting up jenkins (2.504.2) ...
Created symlink /etc/systemd/system/multi-user.target.wants/jenkins.service → /usr/lib/systemd/system/jenkins.service.
Could not execute systemctl: at /usr/bin/deb-systemd-invoke line 148.
Processing triggers for man-db (2.12.0-4build2) ...
```

Рисунок 66 – Установка Jenkins (часть 2)

```

nikita@pavlov:~$ systemctl cat jenkins
# /usr/lib/systemd/system/jenkins.service
#
# This file is managed by systemd(1). Do NOT edit this file manually!
# To override these settings, run:
#
#     systemctl edit jenkins
#
# For more information about drop-in files, see:
#
#     https://www.freedesktop.org/software/systemd/man/systemd.unit.html
#
[Unit]
Description=Jenkins Continuous Integration Server
Requires=network.target
After=network.target
StartLimitBurst=5
StartLimitIntervalSec=5m

```

Рисунок 67 – Просмотр подробностей

Создан административный пользователь (Рисунки 68, 75) для безопасного управления сервером.

```

nikita@pavlov:~$ id jenkins
uid=122(jenkins) gid=124(jenkins) groups=124(jenkins)

```

Рисунок 68 – Пользователь для Jenkins

Установлена Java (Рисунок 69) - обязательное требование для работы Jenkins. Проверена версия Java (Рисунок 70), что подтвердило корректность установки.

```

nikita@pavlov:~$ sudo apt install openjdk-11-jre
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  ca-certificates-java fonts-dejavu-extra java-common libatk-wrapper-java libatk-wrapper-java-jni
  openjdk-11-jre-headless
Suggested packages:
  default-jre fonts-ipafont-gothic fonts-ipafont-mincho fonts-wqy-microhei | fonts-wqy-zenhei fonts-indic
The following NEW packages will be installed:
  ca-certificates-java fonts-dejavu-extra java-common libatk-wrapper-java libatk-wrapper-java-jni openjdk-11-jre
  openjdk-11-jre-headless
0 upgraded, 7 newly installed, 0 to remove and 112 not upgraded.
Need to get 44.6 MB of archives.
After this operation, 184 MB of additional disk space will be used.
Do you want to continue? [Y/n] y

```

Рисунок 69 – Установка Java

```

nikita@pavlov:~$ java --version
openjdk 11.0.27 2025-04-15
OpenJDK Runtime Environment (build 11.0.27+6-post-Ubuntu-0ubuntu124.04)
OpenJDK 64-Bit Server VM (build 11.0.27+6-post-Ubuntu-0ubuntu124.04, mixed mode, sharing)

```

Рисунок 70 – Проверка версии Java

Сервер Jenkins успешно запущен (Рисунок 71).

```
nikita@pavlov:~$ sudo systemctl enable jenkins
Synchronizing state of jenkins.service with SysV service script with /usr/lib/systemd/systemd-sysv-install.
Executing: /usr/lib/systemd/systemd-sysv-install enable jenkins
nikita@pavlov:~$ sudo systemctl start jenkins
nikita@pavlov:~$ sudo systemctl status jenkins
● jenkins.service - Jenkins Continuous Integration Server
   Loaded: loaded (/lib/systemd/system/jenkins.service; enabled; vendor preset: enabled)
   Active: active (running) since Sun 2025-05-11 17:45:04 MSK; 2min 6s ago
     Main PID: 7517 (java)
       Tasks: 50 (limit: 6795)
      Memory: 736.1M
         CPU: 28.321s
        CGroup: /system.slice/jenkins.service
                └─7517 /usr/bin/java -Djava.awt.headless=true -jar /usr/share/java/jenkins.war --webroot=/v
```

Рисунок 71 – Запуск Jenkins

Проведена первоначальная конфигурация (Рисунок 72).

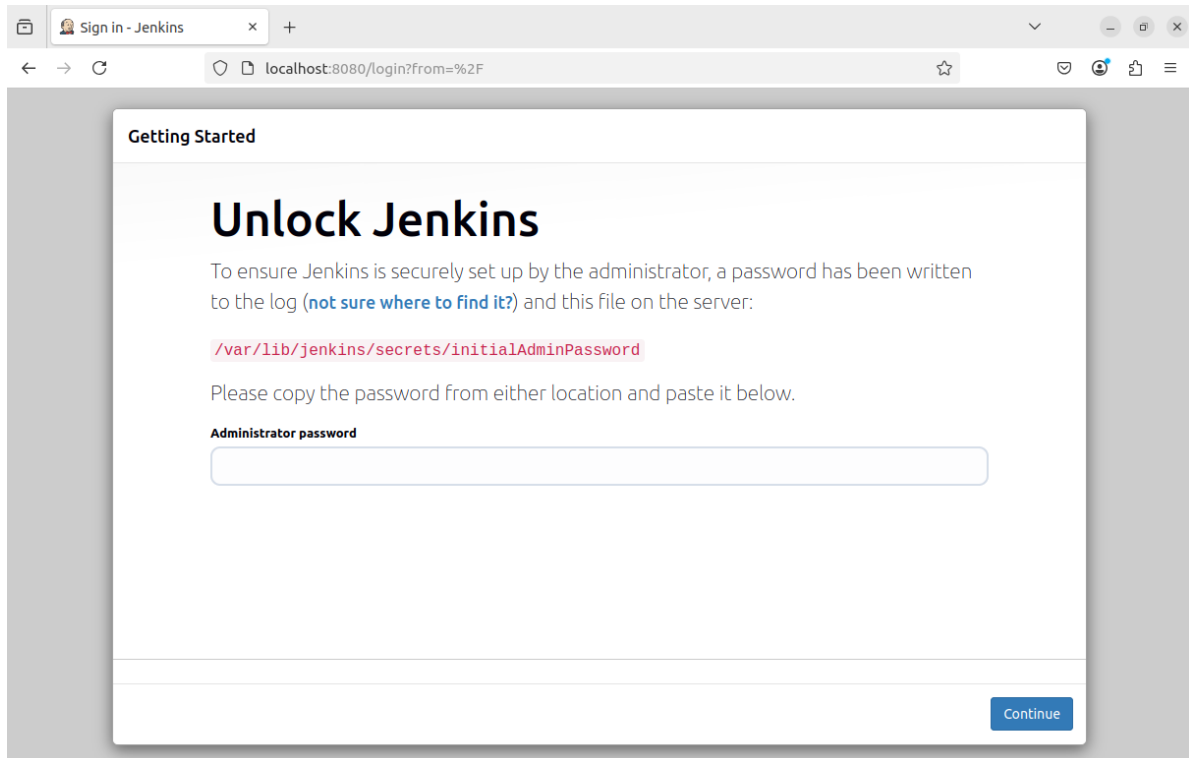


Рисунок 72 – Страница разблокировки Jenkins

Установлен базовый набор плагинов (Рисунки 73-74).

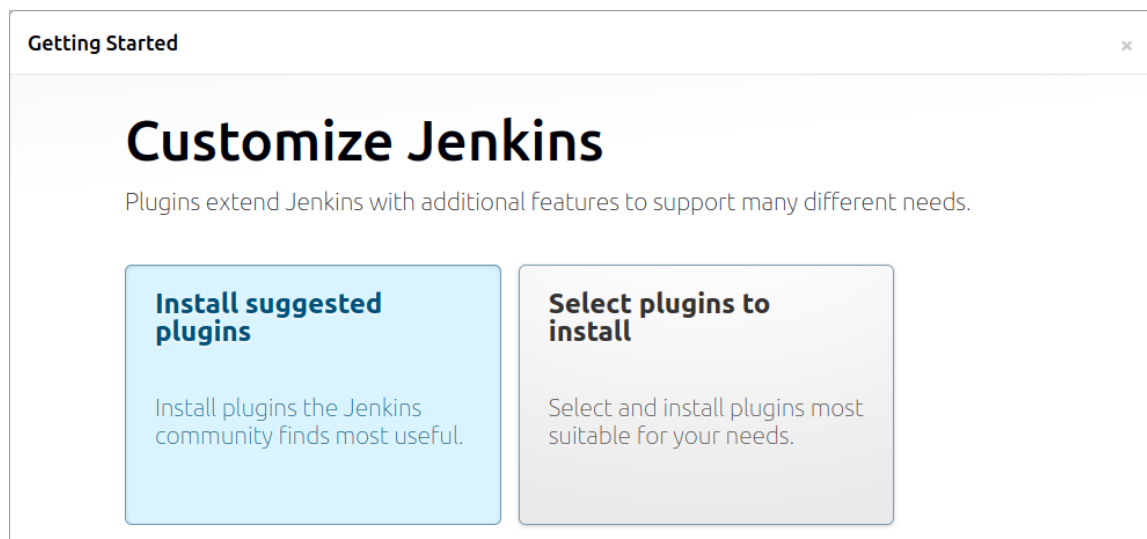


Рисунок 73 – Плагины для Jenkins

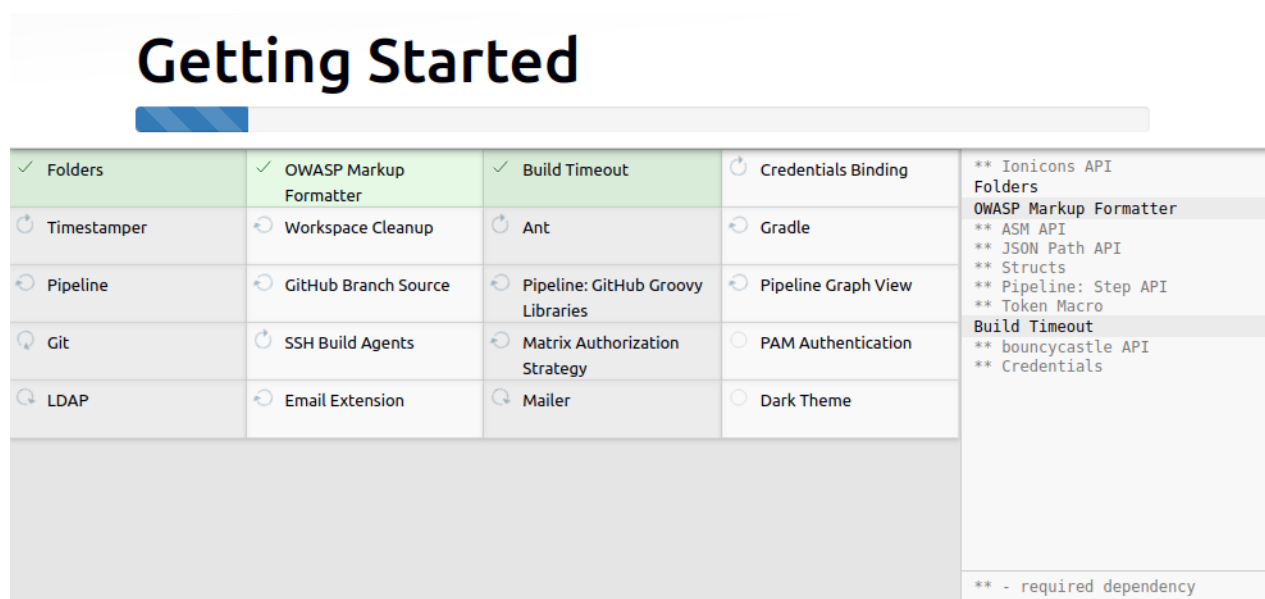


Рисунок 74 – Установка плагинов Jenkins

Create First Admin User

Имя пользователя

Nikita

Пароль

....

Повторите пароль

....

Ф.И.О.

Павлов Никита Сергеевич

Jenkins 2.504.1

[Skip and continue as admin](#)

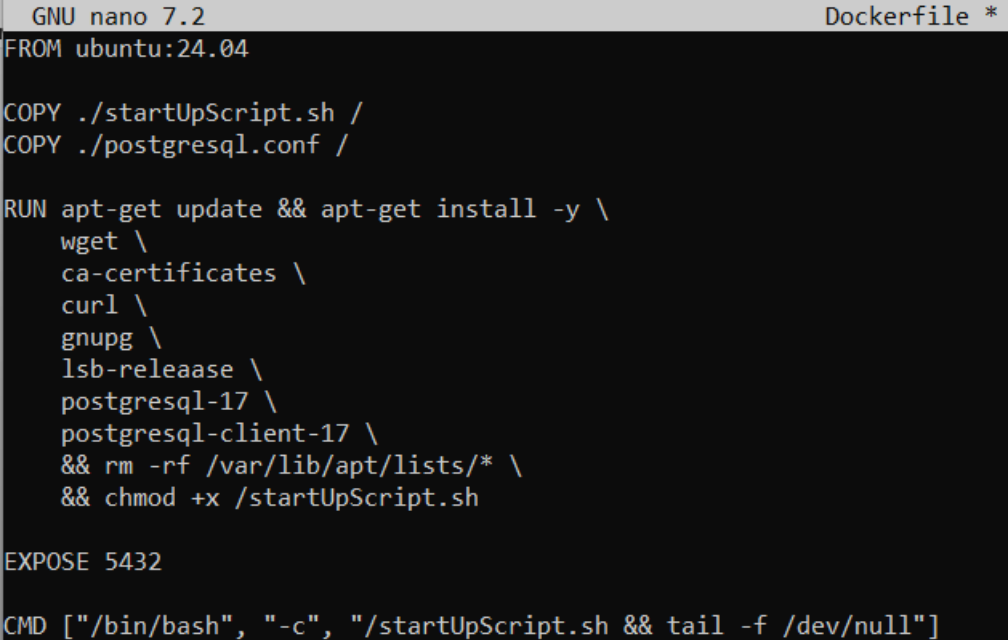
[Save and Continue](#)

Рисунок 75 – Создание первого пользователя администратора

2.8 Модуль 15. Практические навыки работы с Docker

Разработан Dockerfile (Рисунок 76), содержащий:

- Базовый образ на основе ubuntu
- Установку PostgreSQL и зависимостей
- Настройку конфигурации СУБД
- Определение точек монтирования и портов

A screenshot of a text editor window titled 'GNU nano 7.2' and 'Dockerfile *'. The content of the Dockerfile is as follows:

```
FROM ubuntu:24.04

COPY ./startUpScript.sh /
COPY ./postgresql.conf /

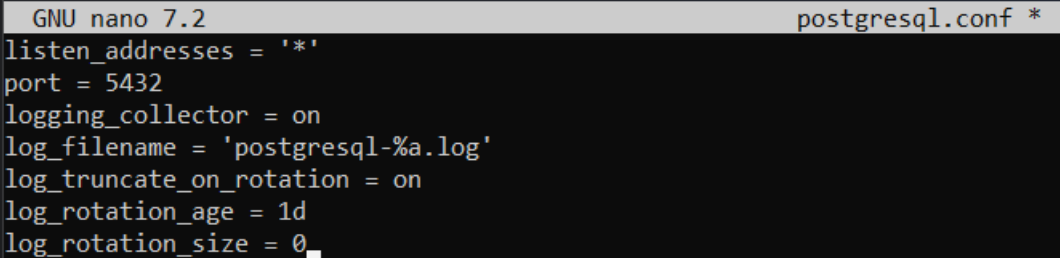
RUN apt-get update && apt-get install -y \
    wget \
    ca-certificates \
    curl \
    gnupg \
    lsb-release \
    postgresql-17 \
    postgresql-client-17 \
    && rm -rf /var/lib/apt/lists/* \
    && chmod +x /startUpScript.sh

EXPOSE 5432

CMD ["/bin/bash", "-c", "/startUpScript.sh && tail -f /dev/null"]
```

Рисунок 76 – Содержимое Dockerfile

Подготовлен конфигурационный файл PostgreSQL (Рисунок 77) для корректной работы в контейнере.

A screenshot of a text editor window titled 'GNU nano 7.2' and 'postgresql.conf *'. The content of the configuration file is as follows:

```
listen_addresses = '*'
port = 5432
logging_collector = on
log_filename = 'postgresql-%a.log'
log_truncate_on_rotation = on
log_rotation_age = 1d
log_rotation_size = 0
```

Рисунок 77 – Конфигурация PostgreSQL

Создан стартовый скрипт (Рисунок 78) для автоматической инициализации БД при запуске.

```

GNU nano 7.2                                                                    startUpScript.sh *
#!/bin/bash

su -l postgres -c /usr/pgsql-17/bin/initdb

echo "host all all 0.0.0.0/0 trust" >> /etc/postgresql/17/main/pg_hba.conf

cp ./postgresql.conf /etc/postgresql/17/main/pg_hba.conf

su -l postgres -c "/usr/pgsql-17/bin/pg_ctl -D /var/lib/postgresql/17/main -l /tmp/pg_logfile start"

psql -v ON_ERROR_STOP=1 --username postgres --dbname postgres <<EOSQL
CREATE USER tester;
CREATE DATABASE "tempdb-15";
GRANT ALL PRIVILEGES ON DATABASE "tempdb-15" TO tester;

CREATE TABLE bicycle (
    bicycle_id serial PRIMARY KEY,
    brand varchar(50) NOT NULL,
    model varchar(50) NOT NULL,
    year_of_release smallint NOT NULL,
    color varchar(30) NOT NULL,
    price_usd numeric(10,2),
    weight_kg numeric(4,2),
    suspension_mm varchar(20)
);

INSERT INTO bicycle (brand, model, year_of_release, color, price_usd, weight_kg, suspension_mm) VALUES
('Specialized', 'S-Works Epic EVO', 2024, 'Satin Carbon', 12500.00, 9.8, '120/110'),
('Trek', 'Supercaliber SLR 9.9 XX AXS', 2024, 'Matte Dnister Black', 11500.00, 10.1, '100/60'),
('Scott', 'Spark RC Team Issue AXS', 2025, 'Pearl White', 9999.00, 10.3, '120/120'),
('Cannondale', 'Scalpel HT Carbon 1', 2024, 'Black Inc.', 8500.00, 9.9, '100 (Hardtail)'),
('Orbea', 'Oiz M-LTD', 2025, 'Coral Red', 10500.00, 10.0, '120/120'),
('Santa Cruz', 'Blur CC X01 AXS RSV', 2024, 'Gloss Carbon', 11000.00, 10.5, '100/100'),
('Giant', 'Anthem Advanced Pro 0', 2025, 'Giant Black', 7500.00, 11.2, '110/100'),
('Canyon', 'Lux CFR LTD', 2024, 'Stealth Black', 8999.00, 9.7, '100 (Hardtail)'),
('Pivot', 'LES SL 29 XTR', 2025, 'Mach 1 Silver', 10500.00, 10.8, '120/120'),
('Yeti', 'SB120 T1 X01 Eagle', 2024, 'Turquoise', 9500.00, 11.0, '120/120');

EOSQL

```

Рисунок 78 – Стартовый скрипт для запуска

Выполнена сборка Docker-образа (Рисунок 79). Запущен контейнер (Рисунок 80) с пробросом портов и подключением volumes.

```

root@MateBook-D-16:/mnt/d/Учеба/ЦК/ pract/docker# docker build --rm -t local/c7-9-pg .
[+] Building 9.5s (8/8) FINISHED                                                                                               docker:default
=> [internal] load build definition from Dockerfile                                0.0s
=> => transferring dockerfile: 525B                                              0.0s
=> [internal] load metadata for docker.io/library/ubuntu:24.04                  0.9s
=> [internal] load .dockerignore                                                 0.0s
=> => transferring context: 2B                                                  0.0s
=> [1/4] FROM docker.io/library/ubuntu:24.04@sha256:6015f66923d7afbc53558d7ccffd325d43b4e249f41a6e93eef074c9505d 0.0s
=> => resolve docker.io/library/ubuntu:24.04@sha256:6015f66923d7afbc53558d7ccffd325d43b4e249f41a6e93eef074c9505d 0.0s
=> [internal] load build context                                                0.0s
=> => transferring context: 73B                                                0.0s
=> CACHED [2/4] COPY --chown=postgres:postgres ./startUpScript.sh /           0.0s
=> CACHED [3/4] COPY --chown=postgres:postgres ./postgresql.conf /            0.0s

```

Рисунок 79 – Сборка Docker-образа

```

root@MateBook-D-16:/mnt/d/Учеба/ЦК/ pract/docker# docker run -p 5432:5432 -d local/c7-9-pg
> 174eca877c9cdf75af879402e8015d580a04fb5f7fb497b16d92f3f49b7ddb75

```

Рисунок 80 – Запуск контейнера

3 ЗАКЛЮЧЕНИЕ

В ходе выполнения итоговой аттестационной работы по программе «Технологии Девопс» были успешно выполнены следующие практические задания:

1. Модуль 6. Администрирование баз данных

- Установка и настройка PostgreSQL.
- Создание пользователей и управление правами доступа.
- Создание базы данных и таблиц.
- Выполнение SQL-запросов (SELECT, INSERT, UPDATE).

2. Модуль 7. Масштабируемость и отказоустойчивость

- Настройка синхронизации времени между узлами.
- Генерация SSH-ключей и настройка безопасного доступа.
- Развертывание кластера PostgreSQL с использованием Pacemaker и Corosync.
- Проверка отказоустойчивости кластера.

3. Модуль 8. Мониторинг, логирование и оповещение событий

- Установка и настройка Node Exporter для сбора метрик.
- Развертывание Prometheus и настройка сбора данных.
- Установка Grafana, подключение источников данных и импорт дашбордов.
- Визуализация метрик производительности системы.

4. Модуль 9. Виртуализация в DevOps

- Развертывание VM на базе Ubuntu 24.04
- Установка и настройка дополнительных утилит (vim, mc, net-tools).
- Проверка работоспособности сетевых интерфейсов.

5. Модуль 12. Конфигурационное управление (IaC)

- Установка Ansible и настройка инвентарных файлов.
- Создание и выполнение playbook'ов для автоматизации задач.
- Проверка доступности узлов и управление конфигурациями.

6. Модуль 13. Системы контроля версий (Git)

- Установка и настройка Git.
- Создание локального репозитория, добавление файлов и коммиты.
- Работа с ветками: создание, слияние, отправка изменений.
- Организация распределённой работы с Git.

7. Модуль 14. Жизненный цикл ПО

- Установка и настройка Jenkins.
- Настройка плагинов и создание администратора.
- Интеграция с инструментами CI/CD.

8. Модуль 15. Практические навыки работы с Docker

- Создание Dockerfile для развертывания PostgreSQL.
- Сборка Docker-образа и запуск контейнера.
- Настройка конфигурации и проверка работоспособности.

В результате выполнения всех модулей были освоены ключевые технологии DevOps, включая управление базами данных, обеспечение отказоустойчивости, мониторинг, автоматизацию развертывания и работу с контейнеризацией. Практические задания продемонстрировали готовность к решению реальных задач в области DevOps, включая настройку CI/CD, управление инфраструктурой и обеспечение стабильности ИТ-систем.

СПИСОК ЛИТЕРАТУРЫ

1. Официальная документация PostgreSQL – [Электронный ресурс]. – URL: <https://www.postgresql.org/docs> (дата обращения: 01.05.2025).
2. Pacemaker и Corosync: High Availability Cluster Management – [Электронный ресурс]. – URL: <https://clusterlabs.org> (дата обращения: 08.05.2025).
3. Prometheus: документация – [Электронный ресурс]. – URL: <https://prometheus.io/docs> (дата обращения: 12.05.2025).
4. Grafana Labs: руководство пользователя – [Электронный ресурс]. – URL: <https://grafana.com/docs> (дата обращения: 12.05.2025).
5. Ansible: официальная документация – [Электронный ресурс]. – URL: <https://docs.ansible.com> (дата обращения: 15.05.2025).
6. Git: документация – [Электронный ресурс]. – URL: <https://git-scm.com/doc> (дата обращения: 17.05.2025).
7. Jenkins: User Handbook – [Электронный ресурс]. – URL: <https://www.jenkins.io/doc> (дата обращения: 25.05.2025).
8. Docker: официальная документация – [Электронный ресурс]. – URL: <https://docs.docker.com> (дата обращения: 28.05.2025).
9. DevOps: принципы, практики и инструменты – Ким Д., Хамбл Д., Дебуа П., Уиллис Д. – М.: Альпина Паблишер, 2024. – 480 с.
10. Автоматизация DevOps с помощью Ansible – Лорх Д. – СПб.: Питер, 2023. – 320 с.
11. Kubernetes и Docker: оркестрация контейнеров – Хайнеман Р. – М.: ДМК Пресс, 2025. – 412 с.
12. ГОСТ Р 7.0.100-2018. Библиографическая запись. Библиографическое описание. Общие требования и правила составления – М.: Стандартинформ, 2018. – 124 с.