



МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РФ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«ЛИПЕЦКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

Институт (факультет)
Кафедра

институт компьютерных наук
автоматизированные системы управления

ЛАБОРАТОРНАЯ РАБОТА №5

По дисциплине: «Интаро Софт»

На тему: «Работа с SSH»

Студент

АИ-23-1

Группа

подпись, дата

Кирнос Б.В.

фамилия, инициалы

Руководитель

кандидат тех. наук, доцент

ученая степень, ученое звание

Кургасов В.В.

фамилия, инициалы

Липецк 2025

Содержание

1.	Создание SSH-ключа.....	3
2.	Работы с базой данных MariaDB	5
Вывод		8

1. Создание SSH-ключа.

В рамках выполнения лабораторной работы была реализована настройка безопасного удалённого доступа к серверу, с этой целью на локальной машине был сгенерирован SSH-ключ, представляющий собой пару взаимосвязанных ключей: приватного и публичного. Приватный ключ хранится на стороне клиента и не подлежит передаче третьим лицам, тогда как публичный ключ предназначен для размещения на удалённом сервере и используется для аутентификации пользователя без ввода пароля.

После генерации ключевой пары публичный ключ был передан на удалённый сервер с использованием утилиты scp, работающей поверх защищённого протокола SSH. Передача ключа по зашифрованному каналу обеспечивает целостность и конфиденциальность данных в процессе копирования. На сервере полученный публичный ключ был добавлен в файл authorized_keys, что позволило связать данный ключ с учётной записью пользователя.

В результате выполненных действий была настроена аутентификация по SSH-ключу, обеспечивающая более высокий уровень безопасности по сравнению с традиционной парольной аутентификацией, а также повышающая удобство и автоматизацию при последующих подключениях к серверу.

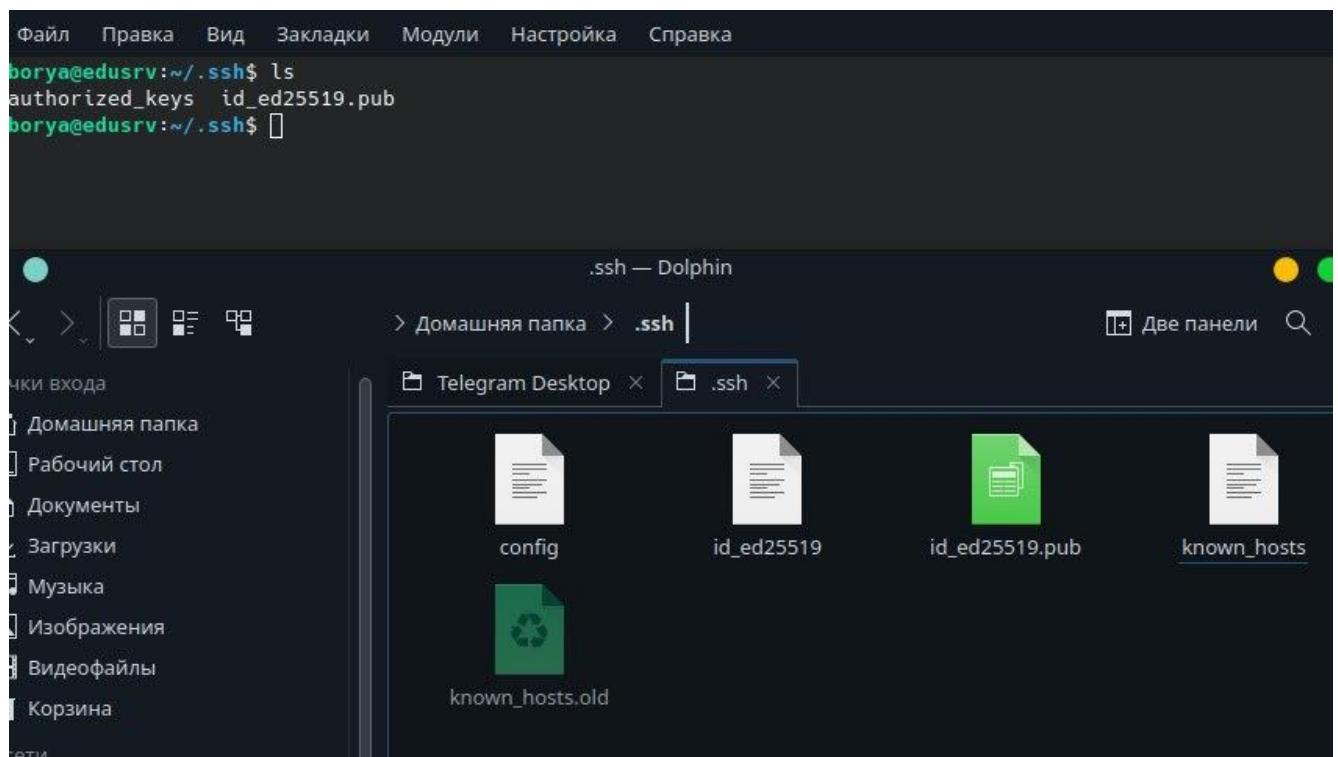
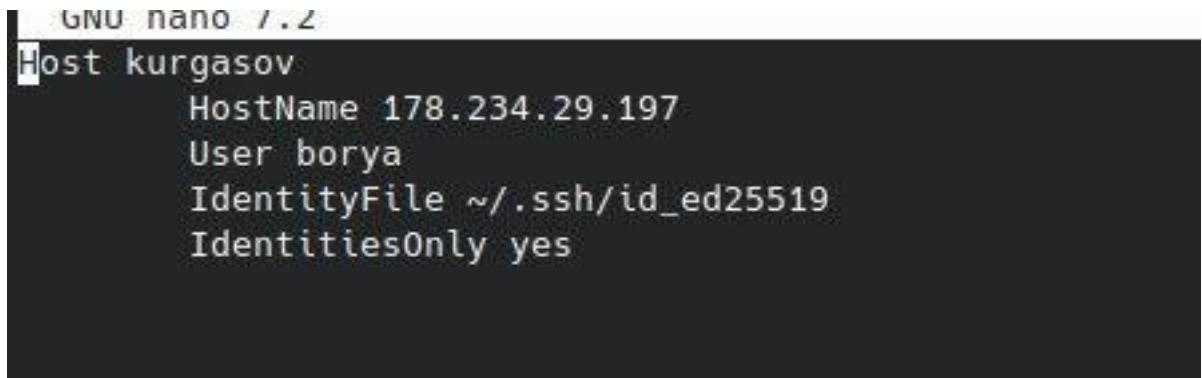


Рисунок 1. Структура SSH-ключей на локальной машине и сервере.

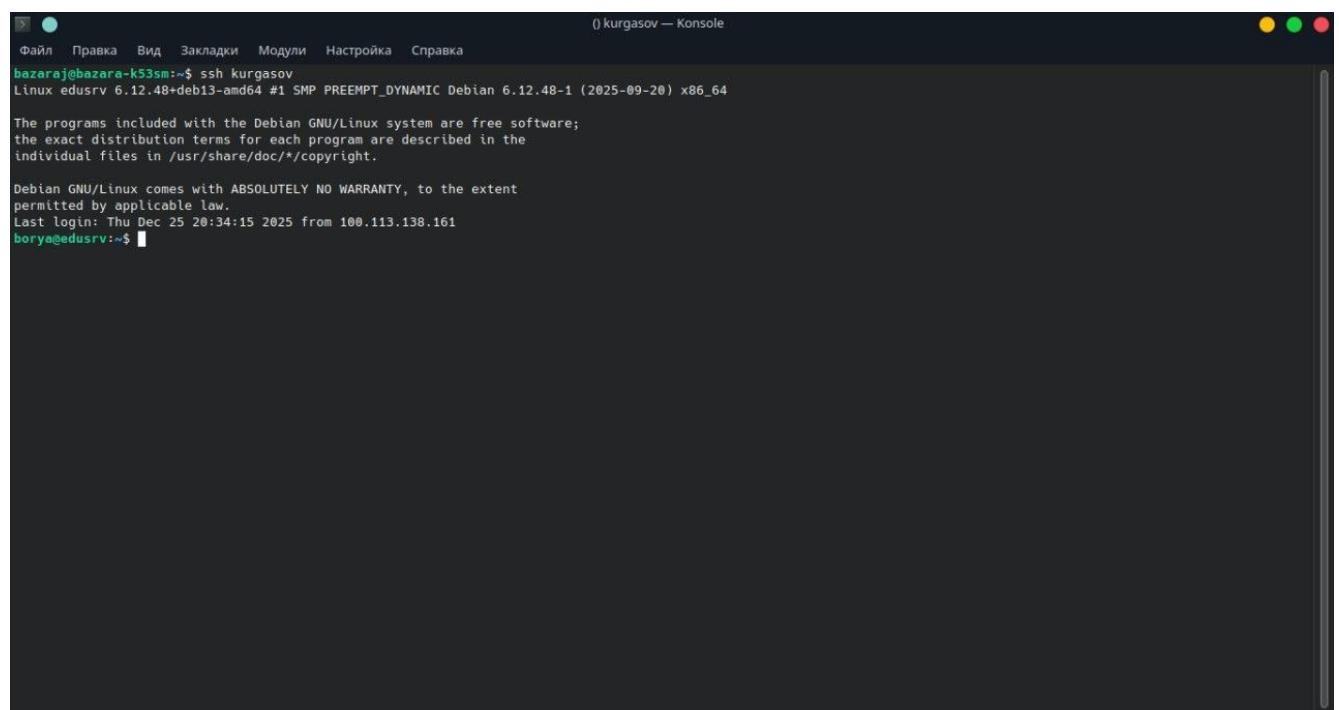
Как видно на рисунке 1, приватный ключ находится на нашем компьютере, а публичный на удаленном сервере.

Для удобства использования создадим файл конфигурации ssh



```
GNU nano 7.2
Host kurgasov
  HostName 178.234.29.197
  User borya
  IdentityFile ~/.ssh/id_ed25519
  IdentitiesOnly yes
```

Рисунок 2. Файл конфигурации



```
0 kurgasov — Konsole
Файл Правка Вид Закладки Модули Настройка Справка
bazarej@bazara-k53sm:~$ ssh kurgasov
Linux edusrv 6.12.48+deb13-amd64 #1 SMP PREEMPT_DYNAMIC Debian 6.12.48-1 (2025-09-20) x86_64
The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/*copyright.

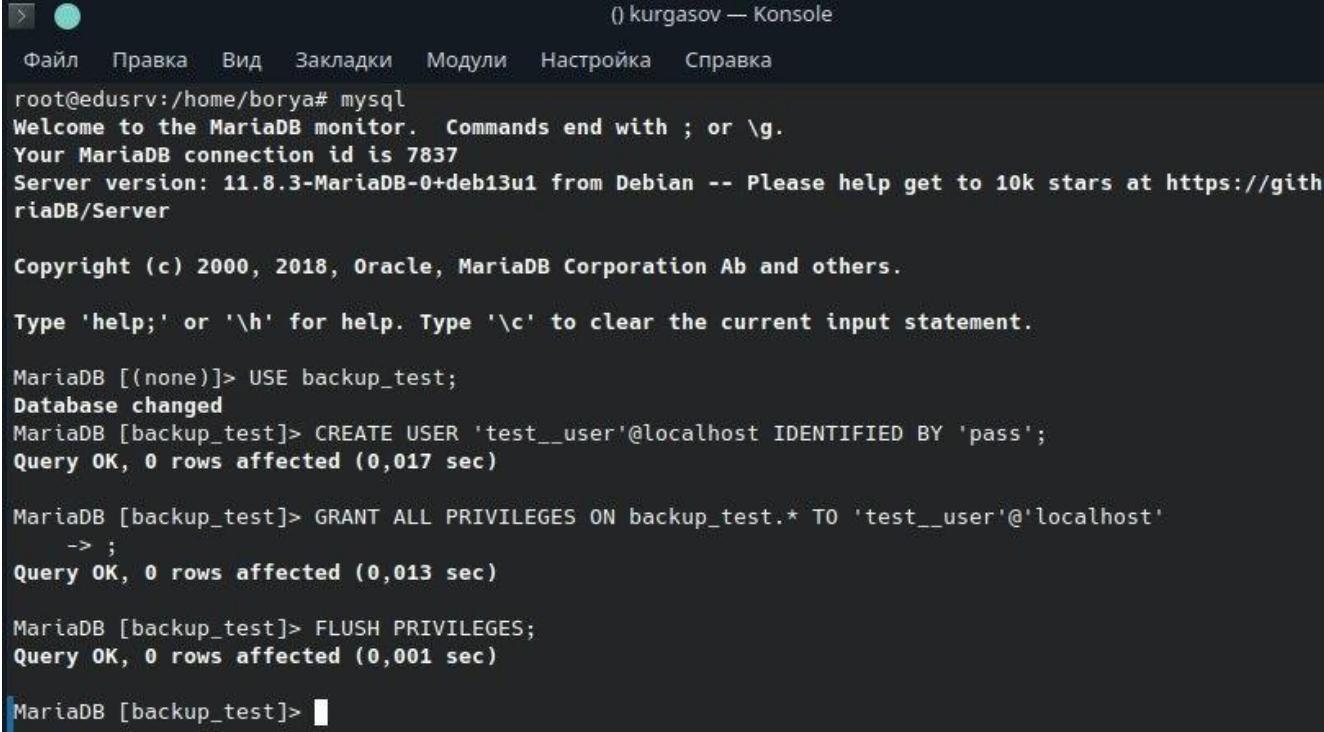
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Thu Dec 25 20:34:15 2025 from 100.113.138.161
borya@edusrv:~$
```

Рисунок 3. Подключение к серверу

В результате выполнения работы было успешно осуществлено подключение к удалённому серверу по протоколу SSH с использованием ключевой аутентификации. Настроенный способ доступа подтвердил свою работоспособность и надёжность, обеспечив безопасное соединение без использования пароля.

2. Работы с базой данных MariaDB

В ходе выполнения лабораторной работы была развернута и настроена база данных MariaDB. После создания базы данных и инициализации её структуры была выполнена процедура резервного копирования с использованием стандартных средств СУБД. Бэкап базы данных был сформирован в виде SQL-дампа, содержащего описание структуры и данных, что обеспечивает возможность полного восстановления состояния базы.



The screenshot shows a terminal window titled '0 kurgasov — Konsole'. The window contains the following text:

```
Файл Правка Вид Закладки Модули Настройка Справка
root@edusrv:/home/borya# mysql
Welcome to the MariaDB monitor. Commands end with ; or \g.
Your MariaDB connection id is 7837
Server version: 11.8.3-MariaDB-0+deb13u1 from Debian -- Please help get to 10k stars at https://gith
riaDB/Server

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> USE backup_test;
Database changed
MariaDB [backup_test]> CREATE USER 'test__user'@localhost IDENTIFIED BY 'pass';
Query OK, 0 rows affected (0,017 sec)

MariaDB [backup_test]> GRANT ALL PRIVILEGES ON backup_test.* TO 'test__user'@'localhost'
    -> ;
Query OK, 0 rows affected (0,013 sec)

MariaDB [backup_test]> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0,001 sec)

MariaDB [backup_test]> █
```

Рисунок 4. Настройка БД

В ходе работы была выбрана ранее созданная база данных `backup_test`, после чего в ней был создан новый пользователь `test__user` с аутентификацией по паролю. Далее этому пользователю были выданы все привилегии на объекты базы данных `backup_test`, что позволяет ему выполнять полный набор операций внутри данной базы. Завершающим шагом была выполнена команда `FLUSH PRIVILEGES`, обеспечивающая немедленное применение изменений в системе управления доступом.

Скриншот демонстрирует корректную настройку базы данных и прав доступа пользователей в MariaDB, что является важным этапом администрирования и подготовки базы к дальнейшей эксплуатации.

```
Database changed
MariaDB [backup_test]> SHOW TABLES;
+-----+
| Tables_in_backup_test |
+-----+
| users                 |
+-----+
1 row in set (0,001 sec)

MariaDB [backup_test]> SELECT * FROM users;
+----+-----+-----+
| id | username | email           |
+----+-----+-----+
| 1  | Nikita   | creativeEmail@mail.ru |
| 2  | Andrey    | notACreativeEmail@mail.ru |
+----+-----+-----+
2 rows in set (0,001 sec)

MariaDB [backup_test]> █
```

Рисунок 5. Заполнение БД

Создадим sql скрипт для заполнения базы данных.

```
root@edusrv:/home/borya# mysqldump -u root -p backup_test > backup_$(date +%Y%m%d).sql
Enter password:
root@edusrv:/home/borya# ls
backup_20251225.sql  script.sql
root@edusrv:/home/borya# █
```

Рисунок 6 Создание бэкапа

На данном скриншоте показан процесс создания резервной копии базы данных с использованием утилиты mysqldump. Выполним команду экспорта базы данных backup_test, указав сохранение дампа в файл с именем, содержащим текущую дату. Такой подход позволяет удобно идентифицировать резервные копии и упрощает их хранение и последующую ротацию.

```

MariaDB [(none)]> DROP DATABASE backup_test;
Query OK, 1 row affected (0,376 sec)

MariaDB [(none)]> SHOW DATABASES;
+-----+
| Database |
+-----+
| azuriom_db |
| gurovbd |
| information_schema |
| joomladb |
| modx_db |
| moodle |
| mysql |
| opencart |
| opencart_db |
| performance_schema |
| sys |
| tjttn_wp |
| webwork_db |
+-----+
13 rows in set (0,001 sec)

```

Рисунок 6. Удаление БД

Была выполнена команда удаления базы данных `backup_test`, что подтверждается сообщением об успешном выполнении запроса. Это действие имитирует потерю данных или очистку среды, что является типичным шагом при проверке процедуры восстановления из резервной копии.

Далее выполнена команда `SHOW DATABASES`, вывод которой демонстрирует перечень существующих баз данных на сервере. Отсутствие базы `backup_test` в списке подтверждает, что операция удаления прошла успешно.

```

root@edusrv:/home/borya# mysql -u root -p backup_test < /home/borya/backup_20251225.sql
Enter password:
root@edusrv:/home/borya# mysql -u root -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 7849
Server version: 11.8.3-MariaDB-0+deb13u1 from Debian -- Please help get to 10k stars at https://github.com/MariaDB/S
Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> USE backup_test; SELECT * FROM users;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
+----+-----+
| id | username | email           |
+----+-----+
| 1  | Nikita   | creativeEmail@mail.ru |
| 2  | Andrey    | notACreativeEmail@mail.ru |
+----+-----+
2 rows in set (0,000 sec)

```

Рисунок 7. Восстановление БД

На представленном скриншоте продемонстрирован процесс восстановления базы данных из ранее созданной резервной копии. В начале была выполнена команда импорта SQL-дампа в базу данных `backup_test`, что обеспечило восстановление её структуры и данных из файла резервного копирования.

После завершения процедуры восстановления повторно подключимся к серверу MariaDB и выберем восстановленную базу данных. Для проверки корректности выполненной операции был выполнен запрос к таблице `users`, результат которого отобразил сохранённые записи. Наличие данных в таблице подтверждает успешное восстановление базы данных из бэкапа и корректную работу механизма резервного копирования и восстановления.

Вывод

В ходе выполнения лабораторной работы была успешно настроена аутентификация по SSH с использованием ключевой пары, что позволило обеспечить безопасное и надёжное подключение к удалённому серверу без применения парольной аутентификации. Проверка подключения подтвердила корректность настройки и возможность практического использования данного механизма для администрирования системы.

Также была развернута и настроена база данных MariaDB. Выполнены операции создания базы данных и пользователей, резервного копирования и последующего восстановления данных. Корректность восстановления была подтверждена сохранением структуры и содержимого базы. Полученные результаты демонстрируют освоение базовых навыков администрирования Linux-систем и СУБД, а также понимание принципов обеспечения безопасности и сохранности данных.

