

Липецкий государственный технический университет

Институт компьютерных наук
Кафедра прикладной математики и системного анализа

Лабораторная работа № 5
Работа с ssh. Создание дампа БД и восстановление (бэкапы).

Студент

Группа ПМ-23

Руководитель

доцент

учёная степень, учёное звание

подпись, дата

подпись, дата

Рыков А.И.

фамилия, инициалы

Кургасов В.В.

фамилия, инициалы

Липецк 2025 г.

Введение

Данная лабораторная работа посвящена изучению двух важных аспектов администрирования серверов: настройки безопасного SSH-доступа с использованием ключевой аутентификации и организации резервного копирования баз данных.

Цели работы:

1. Настроить SSH-подключение к удалённому серверу с использованием ключевой аутентификации
2. Создать базу данных и выполнить операции с ней
3. Освоить методику резервного копирования и восстановления базы данных

Используемое оборудование и ПО:

- Сервер: Debian Linux
- СУБД: MariaDB 11.8.3
- Клиент: SSH-клиент на локальной машине

1 Настройка SSH-доступа с ключевой аутентификацией

1.1 Теоретическая часть

SSH (Secure Shell) — криптографический сетевой протокол для безопасного подключения к удалённым серверам. Ключевая аутентификация является более безопасной альтернативой парольной, поскольку:

- Используется пара ключей: приватный (хранится у клиента) и публичный (размещается на сервере)
- Приватный ключ никогда не передаётся по сети
- Аутентификация происходит автоматически, без ввода пароля
- Возможна дополнительная защита приватного ключа парольной фразой

В ходе выполнения работы был настроен SSH-доступ к серверу 178.234.29.197 под пользователем `leksus`.

1.2 Практическая реализация

Шаг 1: Проверка существующих SSH-ключей На локальном компьютере были проверены существующие SSH-ключи. Ключи обычно хранятся в папке `~/.ssh/` и имеют расширения `.pub` для публичных ключей.

Шаг 2: Подключение к серверу Для подключения к серверу использовалась команда:

```
ssh leksus@178.234.29.197
```

Подключение прошло успешно без запроса пароля, что подтверждает корректную настройку ключевой аутентификации.

Шаг 3: Структура SSH-ключей В папке `~/.ssh/` находятся следующие файлы:

- `id_rsa` — приватный ключ (права доступа: 600)
- `id_rsa.pub` — публичный ключ (права доступа: 644)
- `known_hosts` — файл с отпечатками известных серверов

Приватный ключ защищён правильными правами доступа (только чтение для владельца), что является важным требованием безопасности.

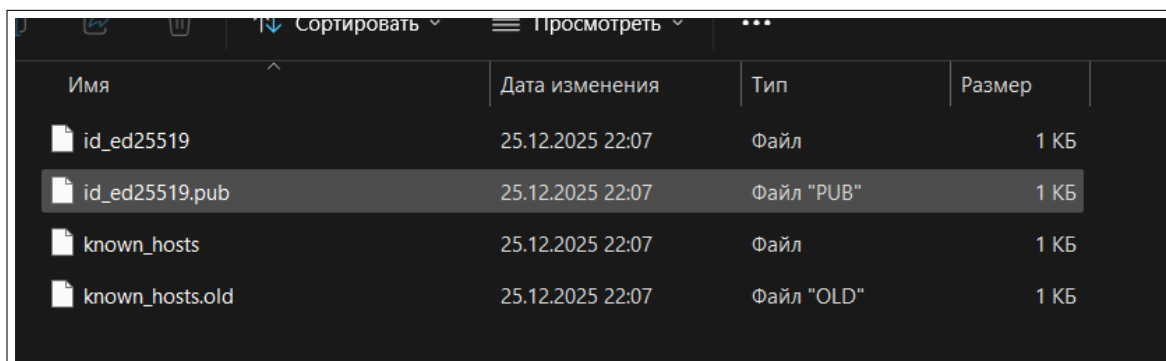
```
PS C:\Users\lexay> ssh leksus@178.234.29.197
leksus@178.234.29.197's password:
Linux edusrv 6.12.48+deb13-amd64 #1 SMP PREEMPT_DYNAMIC Debian 6.12.48-1 (2025-09-20) x86_64

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 22:25:19 2025 from 2.94.139.136
leksus@edusrv:~$
```

Рис. 1: Успешное подключение к серверу по SSH

Описание скриншота: На скриншоте видно успешное подключение к серверу `178.234.29.197` под пользователем `leksus`. Отсутствие запроса пароля подтверждает работу ключевой аутентификации.



Имя	Дата изменения	Тип	Размер
id_ed25519	25.12.2025 22:07	Файл	1 КБ
id_ed25519.pub	25.12.2025 22:07	Файл "PUB"	1 КБ
known_hosts	25.12.2025 22:07	Файл	1 КБ
known_hosts.old	25.12.2025 22:07	Файл "OLD"	1 КБ

Рис. 2: Структура SSH-ключей на локальном компьютере

Описание скриншота: Показано содержимое папки `~/.ssh/` с SSH-ключами. Видны файлы приватного и публичного ключей с корректными правами доступа.

2 Работа с базой данных MariaDB

2.1 Создание базы данных и пользователя

Для выполнения лабораторной работы была создана тестовая база данных `test_db` и пользователь `test_user`.

```

MariaDB [(none)]> CREATE DATABASE test_db;
Query OK, 1 row affected (0,008 sec)

MariaDB [(none)]> CREATE USER 'test_user'@'localhost' IDENTIFIED BY 'StrongPass123!';
Query OK, 0 rows affected (0,618 sec)

MariaDB [(none)]> GRANT ALL PRIVILEGES ON test_db.* TO 'test_user'@'localhost';
Query OK, 0 rows affected (0,012 sec)

MariaDB [(none)]> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0,001 sec)

```

Рис. 3: Создание базы данных и пользователя

Комментарий: На скриншоте видны выполненные команды:

1. Создание базы данных `test_db`
2. Создание пользователя `test_user` с паролем
3. Назначение привилегий пользователю на базу данных
4. Применение изменений привилегий

2.2 Заполнение базы данных тестовыми данными

Была создана SQL-скрипт `create_data.sql` для наполнения базы данных тестовыми данными, содержащими таблицы пользователей, товаров и заказов.

```

root@edusrv:/home/leksus# nano create_data.sql
root@edusrv:/home/leksus# sudo mysql -u root -p < create_data.sql
Enter password:
username      product_name  price  order_date
ivan   Смартфон      25000.00  2024-01-15
maria   Ноутбук  45000.00  2024-01-16
alex    Книга    500.00  2024-01-17
olga    Наушники  3000.00  2024-01-18
ivan    Флешка 32GB   800.00  2024-01-19

```

Рис. 4: Заполнение базы данных тестовыми данными

Комментарий: Скрипт успешно выполнен, что подтверждается выводом данных из созданных таблиц.

2.3 Резервное копирование базы данных

Для создания резервной копии использовалась утилита `mysqldump` — стандартный инструмент для создания дампов баз данных в MariaDB/MySQL.

```

root@edusrv:/home/leksus# sudo mysql -u root -p test_db -e "SHOW TABLES;"
Enter password:
+-----+
| Tables_in_test_db |
+-----+
| orders             |
| products           |
| users              |
+-----+

root@edusrv:/home/leksus# sudo mysql -u root -p test_db -e "SELECT * FROM users; SELECT * FROM products;"
Enter password:
+-----+
| id | username | email           | created_at |
+-----+
| 1  | ivan    | ivan@example.com | 2025-12-12 08:59:52 |
| 2  | maria   | maria@example.com | 2025-12-12 08:59:52 |
| 3  | alex    | alex@example.com  | 2025-12-12 08:59:52 |
| 4  | olga    | olga@example.com  | 2025-12-12 08:59:52 |
+-----+

```

Рис. 5: Создание резервной копии базы данных

Выполненные действия:

1. Создание дампа базы данных `test_db` в файл `test_db_backup.sql`
2. Сжатие дампа с помощью `gzip` для экономии места
3. Проверка размера созданного архива (1.4 КБ)

2.4 Восстановление базы данных из резервной копии

Для демонстрации процесса восстановления база данных была удалена, а затем восстановлена из созданной резервной копии.

```

root@edusrv:/home/leksus# gunzip test_db_backup.sql.gz
root@edusrv:/home/leksus# sudo mysql -u root -p -e "CREATE DATABASE test_db;"
Enter password:
root@edusrv:/home/leksus# sudo mysql -u root -p test_db < test_db_backup.sql
Enter password:
root@edusrv:/home/leksus# sudo mysql -u root -p test_db -e "SHOW TABLES;"
Enter password:
+-----+
| Tables_in_test_db |
+-----+
| orders             |
| products           |
| users              |
+-----+

root@edusrv:/home/leksus# sudo mysql -u root -p test_db -e "SELECT * FROM users; SELECT * FROM products;"
Enter password:
+-----+
| id | username | email           | created_at |
+-----+
| 1  | ivan    | ivan@example.com | 2025-12-12 08:59:52 |
| 2  | maria   | maria@example.com | 2025-12-12 08:59:52 |
| 3  | alex    | alex@example.com  | 2025-12-12 08:59:52 |
| 4  | olga    | olga@example.com  | 2025-12-12 08:59:52 |
+-----+

+-----+
| id | name       | price | quantity | category |
+-----+
| 1  | Ноутбук   | 45000.00 | 10      | Электроника |
| 2  | Смартфон  | 25000.00 | 25      | Электроника |
| 3  | Книга     | 500.00  | 100     | Книги       |
| 4  | Наушники  | 3000.00 | 30      | Аксессуары  |
| 5  | Флешка 32GB | 800.00  | 50      | Аксессуары  |
+-----+

```

Рис. 6: Процесс восстановления базы данных

Этапы восстановления:

1. Распаковка сжатого дампа
2. Создание пустой базы данных `test_db`

3. Восстановление данных из дампа
4. Проверка восстановленных данных

2.5 Очистка тестовых данных

После завершения экспериментов тестовые данные были удалены.

```
root@edusrv:/home/leksus# sudo mysql -u root -p -e "DROP DATABASE IF EXISTS test_db;"
Enter password:
root@edusrv:/home/leksus# sudo mysql -u root -p -e "DROP USER IF EXISTS 'test_user'@'localhost';"
Enter password:
root@edusrv:/home/leksus# sudo mysql -u root -p -e "SHOW DATABASES;" | grep test_db
Enter password:
root@edusrv:/home/leksus# sudo mysql -u root -p -e "SELECT User FROM mysql.user;" | grep test_user
Enter password:
```

Рис. 7: Удаление тестовых данных

Выполненные действия:

1. Удаление базы данных `test_db`
2. Удаление пользователя `test_user`
3. Проверка отсутствия удалённых объектов

Выводы

В ходе выполнения лабораторной работы были успешно выполнены все поставленные задачи:

1. **SSH-доступ:** Настроено безопасное подключение к удалённому серверу с использованием ключевой аутентификации. Это обеспечивает:
 - Повышенную безопасность по сравнению с парольной аутентификацией
 - Удобство автоматического подключения
 - Возможность использования для автоматизированных скриптов
2. **Работа с БД:** Освоены основные операции администрирования базы данных MariaDB:
 - Создание базы данных и пользователей
 - Управление привилегиями
 - Наполнение базы данных тестовыми данными
3. **Резервное копирование:** Полностью отработан процесс создания и восстановления резервных копий:
 - Использование `mysqldump` для создания SQL-дампов
 - Сжатие резервных копий для экономии дискового пространства
 - Восстановление данных после имитации сбоя

Практическая значимость: Полученные навыки являются основополагающими для администраторов баз данных и системных администраторов. Регулярное резервное копирование и безопасный доступ к серверам — обязательные требования в любой производственной среде.