

## Практическая работа 2-1. Основы Linux

### Цель

Познакомиться с основами работы в Linux, включая установку системы, базовую настройку и конфигурацию SSH.

### Задачи

1. Установить Ubuntu 24 в VirtualBox 7.0.x.
2. Выполнить предварительную настройку системы.
3. Настроить SSH-сервер.

### Ход работы

#### 1. Установка Ubuntu 24 в VirtualBox 7.0.x

- 1.1. Скачайте и установите VirtualBox 7.0.x с официального сайта.
- 1.2. Скачайте ISO-образ Ubuntu 24 с официального сайта Ubuntu.
- 1.3. Создайте новую виртуальную машину в VirtualBox:
  - Выберите тип ОС: Linux, версия: Ubuntu (64-bit).
  - Выделите минимум 2 ГБ оперативной памяти.
  - Создайте виртуальный жесткий диск объемом не менее 20 ГБ.
- 1.4. Запустите виртуальную машину и выполните установку Ubuntu, следуя инструкциям мастера установки.

#### 2. Предварительная настройка системы

- 2.1. Создание нового пользователя:

**bash**

```
sudo adduser newuser
```

- 2.2. Добавление пользователя в группу sudo:

**bash**

```
sudo usermod -aG sudo newuser
```

- 2.3. Создание новой группы:

**bash**

```
sudo groupadd newgroup
```

- 2.4. Добавление пользователя в новую группу:

**bash**

```
sudo usermod -aG newgroup newuser
```

- 2.5. Проверка членства пользователя в группах:

**bash**

```
groups newuser
```

#### 3. Настройка SSH

- 3.1. Установка SSH-сервера:

**bash**

```
sudo apt update
```

```
sudo apt install openssh-server
```

### 3.2. Проверка статуса SSH-сервера:

```
bash
```

```
sudo systemctl status ssh
```

### 3.3. Настройка файрвола:

```
bash
```

```
sudo ufw allow ssh
```

### 3.4. Изменение порта SSH (опционально):

- Отредактируйте файл конфигурации SSH:

```
bash
```

```
sudo nano /etc/ssh/sshd_config
```

- Найдите строку #Port 22 и измените её на Port 2222 (или любой другой порт)

- Сохраните изменения и перезапустите SSH-сервер:

```
bash
```

```
sudo systemctl restart ssh
```

### 3.5. Подключение к SSH с другого компьютера:

```
bash
```

```
ssh username@ip_address -p 2222
```

## Требования к выполнению и отчету

### Общие требования к выполнению

1. Все задания должны выполняться в операционной системе Ubuntu 24, установленной в VirtualBox 7.0.x.
2. Перед началом выполнения заданий убедитесь, что система обновлена до последней версии.
3. Все команды должны выполняться в терминале, если не указано иное.
4. При выполнении заданий следуйте принципам безопасности и не используйте пароли, которые вы используете в реальных системах.
5. Работать в программе [MobaXterm](#) или [PuTTY](#) для подключения по SSH.

## **Требования к отчету**

1. Отчет должен быть представлен в формате PDF.
2. Структура отчета:
  - Титульный лист с указанием ФИО, группы, номера варианта
  - Содержание
  - Введение (кратко опишите цель работы)
  - Основная часть (по каждому заданию)
  - Заключение (выводы по работе)
3. В основной части для каждого задания необходимо предоставить:
  - Формулировку задания.
  - Описание выполненных действий.
  - Используемые команды (в виде текста, который можно скопировать).
  - Скриншот выполнения каждого действия (см. требования к скриншотам ниже).
  - Результат выполнения (если применимо).
  - Краткое объяснение результата или ответ на вопрос задания.
4. Все скриншоты должны быть четкими и читаемыми.
5. Текст отчета должен быть грамотным и структурированным.

## **Требования к скриншотам**

1. Каждое действие должно быть подтверждено скриншотом.
2. На скриншоте должны быть видны:
  - Командная строка с введенной командой
  - Результат выполнения команды
  - Приглашение командной строки, показывающее имя пользователя и имя хоста
3. Если результат команды не помещается на один экран, сделайте несколько скриншотов, чтобы показать всю информацию.
4. Скриншоты должны быть пронумерованы и иметь подписи, объясняющие, что на них изображено.
5. Размер скриншотов должен быть достаточным для того, чтобы текст на них можно было легко прочитать.

## **Критерии оценки**

1. Правильность выполнения каждого задания: 60% от общей оценки
2. Полнота и качество отчета: 30% от общей оценки
3. Оформление отчета и скриншотов: 10% от общей оценки

## Варианты заданий

### Вариант 1

1. Создайте нового пользователя student1 и добавьте его в группу developers.
2. Настройте SSH для использования нестандартного порта 2345.
3. Создайте скрипт, который выводит информацию о использовании диска с помощью команды df -h.

### Вариант 2

1. Создайте группу с названием developers.
2. Добавьте пользователя student1 в группу developers.
3. Выведите список всех групп, в которых состоит пользователь student1.

### Вариант 3

1. Создайте файл hello.txt в домашнем каталоге с текстом "Hello, Linux!".
2. Измените права доступа к файлу hello.txt, чтобы только владелец мог его читать и изменять.
3. Выведите содержимое файла hello.txt в терминал.

### Вариант 4

1. Установите пакет htop с помощью менеджера пакетов.
2. Запустите htop и найдите процесс с наибольшим потреблением CPU.
3. Завершите работу htop.

### Вариант 5

1. Создайте символическую ссылку на файл /etc/hostname в вашем домашнем каталоге.
2. Выведите содержимое файла, используя созданную символическую ссылку.
3. Удалите созданную символическую ссылку.

### Вариант 6

1. Найдите все файлы с расширением .log в директории /var/log.
2. Выведите первые 10 строк файла /var/log/syslog.
3. Подсчитайте количество строк в файле /var/log/syslog.

### Вариант 7

1. Создайте файл script.sh со следующим содержимым:

```
bash
#!/bin/bash
echo "Текущая дата и время: $(date)"
```

2. Сделайте файл script.sh исполняемым.
3. Запустите скрипт script.sh.

### Вариант 8

1. Выведите информацию о занятом и свободном месте на диске.
2. Определите, сколько свободной памяти доступно в системе.
3. Выведите информацию о версии ядра Linux.

### **Вариант 9**

1. Создайте архив backup.tar.gz содержимого вашего домашнего каталога.
2. Выведите список файлов в созданном архиве.
3. Извлеките содержимое архива в директорию ~/restore.

### **Вариант 10**

1. Найдите свой IP-адрес с помощью команды ip или ifconfig.
2. Проверьте доступность сайта google.com с помощью команды ping.
3. Выведите список открытых портов на вашей системе с помощью команды ss или netstat.

### **Вариант 11**

1. Создайте файл todo.txt и добавьте в него три задачи на сегодня.
2. Выведите содержимое файла todo.txt в обратном порядке (последняя строка первой).
3. Подсчитайте количество слов в файле todo.txt.

### **Вариант 12**

1. Создайте директорию ~/scripts.
2. Переместите файл script.sh из домашнего каталога в директорию ~/scripts.
3. Добавьте директорию ~/scripts в переменную окружения PATH.

### **Вариант 13**

1. Найдите все файлы в вашем домашнем каталоге, измененные за последние 24 часа.
2. Выведите список всех скрытых файлов в вашем домашнем каталоге.
3. Создайте жесткую ссылку на файл todo.txt с именем todo\_link.txt.

### **Вариант 14**

1. Установите текстовый редактор nano с помощью менеджера пакетов.
2. Откройте файл todo.txt в редакторе nano, добавьте новую задачу и сохраните изменения.
3. Выведите строки из todo.txt, содержащие слово "важно" (без учета регистра).

### **Вариант 15**

1. Создайте новую переменную окружения MYVAR со значением "Hello, Linux!".
2. Выведите значение переменной MYVAR.
3. Создайте файл env.sh, который при выполнении будет выводить значение MYVAR.

### **Вариант 16**

1. Выведите список всех процессов, запущенных от имени вашего пользователя.
2. Найдите PID процесса с названием "bash".
3. Выведите информацию об использовании памяти и CPU для процесса с найденным PID.

### **Вариант 17**

1. Создайте файл numbers.txt с числами от 1 до 10, каждое на новой строке.
2. Отсортируйте числа в файле numbers.txt в обратном порядке.
3. Выведите только четные числа из файла numbers.txt.

### **Вариант 18**

1. Узнайте текущую рабочую директорию.
2. Создайте дерево директорий ~/test/dir1/dir2.
3. Переместитесь в директорию ~/test/dir1/dir2 одной командой.

### **Вариант 19**

1. Создайте файл permissions.txt.
2. Установите для файла permissions.txt права доступа, чтобы владелец мог читать и писать, группа только читать, а остальные не имели доступа.
3. Выведите права доступа файла permissions.txt в числовом формате.

### **Вариант 20**

1. Выведите список всех переменных окружения.
2. Найдите значение переменной окружения HOME.
3. Создайте файл my\_home.txt с текстом, содержащим значение переменной HOME.

### **Вариант 21**

1. Выведите календарь на текущий месяц.
2. Создайте файл date\_info.txt с информацией о текущей дате и времени.
3. Добавьте в файл date\_info.txt информацию о часовом поясе.

### **Вариант 22**

1. Выведите список всех установленных пакетов в системе.
2. Найдите пакет с названием, содержащим слово "python".
3. Выведите подробную информацию об одном из найденных пакетов.

### **Вариант 23**

1. Создайте файл test\_file.txt размером ровно 1MB, заполненный случайными данными.
2. Выведите тип файла test\_file.txt.
3. Создайте уменьшенную копию файла test\_file.txt размером 512KB.

### **Вариант 24**

1. Найдите все файлы в домашнем каталоге с расширением .txt.
2. Подсчитайте общее количество найденных файлов.
3. Выведите список найденных файлов, отсортированный по размеру.

### **Вариант 25**

1. Создайте простой файл index.html с текстом "Hello, Web!".
2. Запустите простой веб-сервер на Python для отображения этой страницы.
3. Проверьте доступность веб-страницы с помощью команды curl.

### **Вариант 26**

1. Выведите список всех активных сетевых соединений.
2. Определите, какой процесс использует порт 80 (если есть).
3. Проверьте статус фаервола UFW.

### **Вариант 27**

1. Создайте файл test.txt с текстом "Hello, sed!".
2. Используя sed, замените слово "Hello" на "Hi" в файле test.txt.
3. Выведите содержимое измененного файла.

### **Вариант 28**

1. Выведите список всех подключенных USB-устройств.
2. Определите модель вашего процессора.
3. Выведите информацию о занятом и свободном месте на всех подключенных дисках.

### **Вариант 29**

1. Создайте файл log.txt с несколькими строками текста.
2. Добавьте в начало каждой строки файла log.txt текущую дату.
3. Выведите только строки из log.txt, содержащие слово "error" (без учета регистра).

### **Вариант 30**

1. Выведите список всех системных служб и их статус.
2. Проверьте статус службы SSH.
3. Перезапустите службу SSH (если установлена и запущена).

## ПРИМЕР ВЫПОЛНЕНИЯ РАБОТЫ

### Предварительная настройка системы

#### 1. Создать пользователя **ansible** с домашней директорией **/home/ansible**

```
sudo adduser ansible --home /home/ansible  
pass Qq1234!Aa!
```

#### 2. Сменить пользователю **ansible** пароль

```
sudo passwd ansible  
!1234!Qq
```

#### 3. Сменить **root** пароль

```
sudo -i  
sudo passwd  
pass !1234!Qq
```

#### 4. Создать группу **wheel**

```
sudo groupadd wheel  
cat /etc/group    Чтобы просмотреть все созданные группы в Ubuntu
```

#### 5. Создать директорию **/admin**

```
sudo mkdir /admin
```

#### 6. Сделать пользователя **ansible** владельцем директории **/admin** и назначить группу **wheel**

```
sudo chown ansible:wheel /admin
```

#### 7. Установить права на директорию **/admin**: чтение и запись для пользователя **ansible**, только чтение для группы **wheel**

```
sudo chmod 640 /admin  
ls -ld /admin просмотреть права на каталог
```

Первая строка (drwxr-x---) показывает права доступа:

d: обозначает, что это директория.

rwX: права пользователя (владельца) на чтение, запись и выполнение.

r-x: права группы на чтение и выполнение.

---: права остальных пользователей (нет доступа).



**8. Отредактировать файл /etc/sudoers с помощью утилиты nano, чтобы разрешить группе wheel использовать sudo с паролем**

```
sudo nano /etc/sudoers
```

```
sudo visudo
```

```
%wheel ALL=(ALL:ALL) ALL
```

```
devops@devopsvm:~$ sudo su
[sudo] password for devops:
root@devopsvm:/home/devops# history
1  visudo
2  sudo passwd
3  sudo groupadd wheel
4  cat /etc/group
5  sudo mkdir /admin
6  ls
7  sudo chown ansible:wheel /admin
8  sudo chmod 640 /admin
9  ls -ld /admin
10 exit
11 history
```

## Настройка SSH на Ubuntu 24

### 1. Установить SSH-сервер

Сначала нужно установить SSH-сервер (если он еще не установлен):

```
bash
```

```
sudo apt update
```

```
sudo apt install openssh-server
```

### 2. Проверить статус службы SSH

Проверьте, работает ли служба SSH, и включена ли она для автоматического запуска при загрузке:

```
bash
```

```
sudo systemctl status ssh
```

Если служба не запущена, запустите ее:

```
bash
```

```
sudo systemctl start ssh
```

И включите ее для автоматического запуска:

```
bash
```

```
sudo systemctl enable ssh
```

### 3. Настроить конфигурацию SSH (опционально)

Файл конфигурации SSH находится в /etc/ssh/sshd\_config. Отредактируйте его для изменения параметров:

```
bash
```

```
sudo nano /etc/ssh/sshd_config
```

Некоторые ключевые настройки, которые можно изменить:

- **Port:** Порт, на котором работает SSH (по умолчанию 22). Для большей безопасности можете изменить порт.

**conf**

Port 2222

- **PermitRootLogin:** разрешает или запрещает вход под root (рекомендуется запретить).

**conf**

PermitRootLogin no

- **PasswordAuthentication:** разрешает аутентификацию с использованием пароля (можно отключить и использовать только ключи SSH).

**conf**

PasswordAuthentication yes

После внесения изменений, сохраните файл с помощью Ctrl + O, затем Enter, и выйдите с помощью Ctrl + X.

#### 4. Перезапустить SSH-сервис

После изменения конфигурации перезапустите SSH-сервис:

**bash**

sudo systemctl restart ssh

#### 5. Настроить брандмауэр (если установлен)

Если используется UFW (Uncomplicated Firewall), необходимо разрешить SSH:

**bash**

sudo ufw allow ssh

Или, если вы изменили порт SSH (например, на 2222), используйте:

**bash**

sudo ufw allow 2222/tcp

Проверьте статус UFW, чтобы убедиться, что правило применено:

**bash**

sudo ufw status

#### 6. Подключение к SSH

Теперь можно подключиться к серверу SSH с другого компьютера с помощью команды:

**bash**

ssh username@server\_ip\_address

Если порт SSH изменен, используйте:

**bash**

ssh -p 2222 username@server\_ip\_address

Где:

- username — имя пользователя для входа.

- server\_ip\_address — IP-адрес сервера Ubuntu.