

## ЛАБОРАТОРНАЯ РАБОТА №0

**Тема: Основы работы в ОС Linux (Ubuntu).**

**Цель работы:** Изучить основы работы в операционной системе Linux (Ubuntu), включая базовые команды для работы с файловой системой, управление пользователями и доступом к ресурсам.

Цель работы обуславливает постановку и решение следующих **задач**:

1. Изучение базовых команд оболочки Linux (Ubuntu) для навигации по файловой системе и управления файлами и директориями.
2. Освоение работы с текстовыми редакторами в командной строке, такими как vim.
3. Изучение команд для работы с пользователями и группами, таких как useradd, usermod, userdel, groupadd, groupmod, groupdel.
4. Рассмотрение принципов управления правами доступа к файлам и директориям, использование команд chmod, chown, chgrp.
5. Анализ системных журналов и изучение основ мониторинга процессов с помощью утилит: ps, top, htop, dmesg, journalctl.
6. Анализ результатов выполнения лабораторной работы и формулировка выводов о полученных знаниях и навыках.

### Ход выполнения лабораторной работы

1. Самостоятельно выполнить установку Linux (Ubuntu), используя инструкции на официальном сайте ОС: <https://help.ubuntu.com>. При этом допустимо использовать программное обеспечение для виртуализации, например VirtualBox или VMware Workstation Player, позволяющее создать виртуальную машину для установки и работы с Linux (Ubuntu) без необходимости изменять текущую конфигурацию вашего основного компьютера. Убедитесь, что вы правильно настроили виртуальную машину, выделили достаточное количество оперативной памяти и дискового пространства, а также установили все необходимые дополнения гостевой системы для корректной работы и взаимодействия с хостовой операционной системой.

1.1. Пример установки ОС Ubuntu в виртуальной машине, обеспечиваемой VMware Workstation Player в рабочем окружении ОС Windows 11.

1.1.1. Необходимо получить запускаемый файл Vmware-player-full-17.6.0-24238078.exe (415.07 MB). Для этого можно выполнить регистрационные действия на сайте [broadcom.com](https://www.broadcom.com) и скачать программное обеспечение, найдя его через поиск: <https://support.broadcom.com/group/ecx/productdownloads?subfamily=VMware+Workstation+Player>


VMware Workstation Player		Release 16.2.5	Release Level Info 208718		
File Name	Last Updated	SHA2		MD5	
VMware Workstation 16.2.5 Player for Windows 64-bit Operating Systems					
VMware-player-full-16.2.5-20904516.exe(584.35 MB) Build Number: 20904516	Dec 13, 2022 12.00AM	3da986cbc26afca63d9c7721621388089876dbd128d 607a50cc58777cc15891d		246a9a2e7dd957f6ac51bdb4768946bb	
					

Рис.1. Устанавливаемая программа VMware Workstation Player с указанием SHA2, MD5

Для упрощения последовательности действий предлагается файл VMware-player-full-17.6.0-24238078.exe скачать по **прямой ссылке**: <https://disk.yandex.ru/d/EgezruD8UZRbXQ>

Внимание! Перед запуском с целью безопасности и недопущения установки инфицированного ПО важно убедиться в том, что запускаемый файл соответствует по хешам (SHA2, MD5) файлу-оригиналу.

Например, следующим образом:

```
certutil -hashfile .\VMware-player-full-16.2.5-20904516.exe MD5
```

Дальнейший ход установочных действий интуитивен и не должен представлять существенных трудностей.

1.1.2. В целях унификации рекомендуется выполнить установку Linux (Ubuntu) с образом ubuntu-24.04.1-desktop-amd64.iso , который можно скачать с сайта <https://releases.ubuntu.com/>

2. Изучить базовые команды оболочки Linux (Ubuntu) для навигации по файловой системе и управления файлами и директориями, отработав следующие сценарии:

2.1. Откройте терминал и изучите основные команды для навигации по файловой системе:

2.1.1. Использование команды pwd для отображения текущего каталога.

2.1.2. Переход в другой каталог с помощью команды cd (например, cd /home/username/Documents).

2.1.3. Использование команды ls для просмотра содержимого текущего или указанного каталога (например, ls -l для детального списка файлов).

2.2. Работа с файлами и директориями:

2.2.1. Создание новых директорий с помощью команды mkdir (например, mkdir new\_directory).

2.2.2. Создание пустого файла: touch file.txt Если указанный файл не существует, touch создаст новый пустой файл с заданным именем.

2.2.3. Копирование файлов и директорий с помощью команды `cp` (например, `cp file.txt /home/username/Documents`).

2.2.4. Перемещение или переименование файлов и директорий с помощью команды `mv` (например, `mv file.txt new_name.txt` или `mv file.txt /home/username/Documents`).

2.2.5. Удаление файлов с помощью команды `rm` (например, `rm file.txt`) и директорий с помощью `rm -r` (например, `rm -r directory_name`).

2.2.6. Просмотр содержимого текстовых файлов с помощью команды `cat` (например, `cat file.txt`).

3. Освоение работы с текстовыми редакторами в командной строке, такими как `vim`, отработав следующие сценарии:

3.1. Открытие и создание файлов:

Открытие существующего файла в `vim` и создание нового, если файла не существует. Используйте команду `vim file.txt` для открытия или создания файла.

3.2. Редактирование текста:

3.2.1 Практикуйте вставку текста в файл. Для этого перейдите в режим вставки, нажав клавишу `i`, и введите произвольный текст.

3.2.2. Научитесь удалять, копировать и вставлять строки и слова, используя команды режима команд (`dd` для удаления строки, `yy` для копирования строки, `p` для вставки после курсора).

3.3. Сохранение изменений и выход из редактора:

3.3.1. Для сохранения изменений и выхода используйте команду `:wq`. Для выхода без сохранения изменений используйте команду `:q!`.

3.4. Поиск и замена текста:

3.4.1. Освойте поиск текста в документе с помощью команды `/search_term`, где `search_term` — искомое слово или фраза.

3.4.2. Научитесь заменять текст, используя команду `:s/old_text/new_text/g`, где `old_text` — заменяемое слово или фраза, а `new_text` — новый текст.

4. Изучить команды для работы с пользователями и группами: `useradd`, `usermod`, `userdel`, `groupadd`, `groupmod`, `groupdel`, отработав следующие сценарии:

4.1. Создание нового пользователя и группы

4.1.1. Создать нового пользователя `newuser` с домашней директорией `/home/newuser` (команда `useradd -m newuser`).

4.1.2. Создать новую группу newgroup (команда `groupadd newgroup`).

#### 4.2. Изменение пользователя

4.2.1. Добавить пользователя newuser в группу newgroup (команда `usermod -aG newgroup newuser`).

4.2.2. Изменить имя пользователя newuser на updateduser (команда `usermod -l updateduser newuser`).

#### 4.3. Удаление пользователя и его домашней директории

4.3.1. Удалить пользователя updateduser и его домашнюю директорию (команда `userdel -r updateduser`).

#### 4.4. Изменение группы

4.4.1. Переименовать группу newgroup в updatedgroup (команда `groupmod -n updatedgroup newgroup`).

#### 4.5. Удаление группы

4.5.1. Удалить группу updatedgroup (команда `groupdel updatedgroup`).

#### 4.6. Создание пользователя с определённой группой и настройкой пароля

4.6.1. Создать пользователя devuser и сразу добавить его в группу developers, а также задать пароль для этого пользователя (команды `useradd -m -G developers devuser` и `passwd devuser`).

#### 4.7. Настройка учетных записей с ограниченным временем действия.

4.7.1. Создать пользователя tempuser с учётной записью, срок действия которой истекает через 7 дней (команда `useradd -e $(date -d "+7 days" +%Y-%m-%d) tempuser`).

### 5. Рассмотреть принципы управления правами доступа к файлам и директориям, используя команды `chmod`, `chown`, `chgrp`, отработав следующие сценарии:

#### 5.1. Изменение прав доступа к файлу.

5.1.1. Установить права доступа для файла file.txt только на чтение для всех пользователей

```
chmod 444 file.txt
```

#### 5.2. Назначение прав доступа группе пользователей.

5.2.1. Создать группу developers и добавить в неё нескольких пользователей:

```
sudo groupadd developers
```

```
sudo usermod -aG developers user1  
sudo usermod -aG developers user2
```

5.2.2. Изменить группу владельца для директории `project_folder` на `developers`:

```
chgrp developers project_folder
```

5.3. Передача прав собственности на файл.

5.3.1. Создать файл `script.sh` и назначить его владельцем пользователя `user2`

```
chown user2 script.sh
```

5.4. Установка прав доступа.

5.4.1. Установить права доступа на выполнение для владельца и группы, и запретить все права для остальных пользователей для файла `run.sh`

```
chmod u+x,g+x,o-rwx run.sh
```

5.5. Управление доступом к директории.

5.5.1. Создать директорию `secure_data` и установить права доступа, позволяющие только владельцу просматривать и изменять её содержимое:

```
chmod 700 secure_data
```

6. Проанализировать системные журналы и изучить основы мониторинга процессов с помощью утилит: `ps`, `top`, `htop`, `dmesg`, `journalctl`, отработав следующие сценарии:

6.1. Просмотр активных процессов:

6.1.1. Используйте команду `ps aux` для вывода списка всех активных процессов в системе. Ознакомьтесь с информацией, которую предоставляет `ps`, включая идентификатор процесса (PID), имя пользователя, которому принадлежит процесс, и команду, запустившую процесс.

6.1.2. Откройте утилиту `top` для реального времени мониторинга процессов. Проанализируйте вывод и попробуйте отфильтровать процессы по использованию процессора или памяти, используя клавиши `P` (по CPU) и `M` (по памяти).

6.1.3. Установите и запустите `htop` для более удобного интерфейса мониторинга процессов. Научитесь сортировать процессы, завершать их с помощью клавиши `F9`, и изменять приоритеты процессов (`renice`) через меню.

6.2. Анализ системных сообщений и журналов:

6.2.1. Используйте команду `dmesg` для просмотра сообщений ядра. Проанализируйте последние записи и определите, какие события были зарегистрированы при загрузке системы и работе с устройствами.

6.2.2. Научитесь использовать утилиту `journalctl` для просмотра системных журналов. Попробуйте отобразить записи за последний час с помощью команды `journalctl --since "1 hour ago"` и за последние 10 минут с помощью `journalctl --since "10 minutes ago"`.

6.2.3. Отфильтруйте системные сообщения по уровню важности, используя `journalctl -p 3 -b` для вывода всех сообщений уровня "error" с последней загрузки системы.

### 6.3. Мониторинг использования ресурсов:

6.3.1. Используйте `top` и `htop` для мониторинга использования процессора, памяти и `swap`-файла. Определите, какие процессы занимают больше всего ресурсов, и проанализируйте их поведение.

6.3.2. Научитесь выводить статистику использования процессора по всем ядрам отдельно в `htop`, нажав клавишу F2 (setup) и выбрав соответствующий режим отображения.

6.3.3. Проанализируйте использование дискового пространства и активность ввода/вывода с помощью команды `iostat` или утилиты `iotor`, если она установлена.

## Контрольные вопросы

- 1) Какие базовые команды используются для навигации по файловой системе в Linux?
- 2) Как создать и удалить директорию в Linux? Приведите примеры команд.
- 3) Опишите процесс создания и редактирования текстовых файлов с помощью текстового редактора `vim`. Какие команды используются для сохранения изменений и выхода?
- 4) Какие команды используются для управления пользователями и группами в Linux?
- 5) Как изменить права доступа к файлам и директориям в Linux? Приведите примеры использования команд `chmod`, `chown`, и `chgrp`.
- 6) Какие утилиты используются для мониторинга процессов в Linux? Чем отличаются `ps`, `top`, и `htop`?
- 7) Как просмотреть системные журналы в Linux? Какие команды можно использовать для анализа сообщений ядра и системных журналов?
- 8) Какие права доступа могут быть назначены файлам и директориям в Linux? Объясните символическое и числовое представление прав доступа.
- 9) Опишите процесс создания пользователя с ограниченным сроком действия. Как установить дату истечения учётной записи?
- 10) Как просмотреть и фильтровать процессы по их использованию ресурсов, таких как CPU и память? Какие клавиши используются для фильтрации в утилитах `top` и `htop`?