

Федеральное государственное бюджетное образовательное учреждение
высшего образования

Бурятский Государственный университет имени Доржи Банзарова

Институт математики, физики и компьютерных наук

ПРАКТИЧЕСКИЙ КУРС

«Введение в Linux»

Авторы: И.С. Поломошнов,
старший лаборант ЛСТ;
А.Е. Шорников,
лаборант ЛСТ.

Улан-Удэ

2024

Оглавление

От авторов.....	3
Введение.....	4
История Linux.....	5
1. Вариант «Базовый»:.....	6
1.1. Виды систем Linux.....	6
1.2. Linux в России.....	7
1.3. Процесс установки и базовой настройки.....	8
1.4. Терминал. Основные команды Linux.....	9
1.5. Графические оболочки.....	12
1.6. Пакетные менеджеры.....	15
1.7. Терминал. Работа с дисками и внешними устройствами.....	17
2. Вариант «Продвинутый»:.....	19
2.1. Администрирование Linux.....	19
2.2. Терминал. Файловые системы.....	20
2.3. Настройка сети.....	21
2.4. Настройка сервера DHCP.....	24
2.5. Настройка сервера DNS.....	27
2.6. Настройка сервера Samba.....	28
2.7. Настройка сервера CUPS.....	30
2.8. Настройка сервера Web.....	32
2.9. Работа с wget, curl, dpkg.....	34
2.10. Практические задачи.....	35

От авторов.

Linux – это не операционная система, а всего лишь ее ядро. Все, что вы привыкли видеть и понимать как операционку (кнопки, окна и панели) - на самом деле ею не является. То, что вы видите - это GUI (Graphical user interface), то есть графический интерфейс пользователя, именуемый также графической средой или графической оболочкой. GUI – это программа, призванная упростить жизнь пользователю, чтобы он не мучился с консолью. На самом деле любая ОС выглядит как консоль, то есть как старый добрый DOS.

Виды операционных систем, построенных на ядре Linux сильно разнятся как по обеспечению рабочим окружением, так и по набору прикладных программ, доступных в ОС. Это отчасти потому, что часть программ поддерживают лишь некоторые GUI или зависят от других программ, которых нет в данной системе.

Итак, любой Linux начинается с консоли. DOS – аббревиатура всем более знакомая. И внешний вид DOS ещё не всеми забыт.

С windows все понятно - один разработчик, одна графическая среда. Ее качество зависит от способностей и стараний программистов Microsoft. Linux же – система открытая и свободная, поэтому каждый желающий может написать любую программу и графическую оболочку в том числе. Ее популярность будет зависеть от того, насколько она понравится пользователям в плане красоты, стабильности, удобства... Свободная конкуренция так сказать.

Введение.

Использование Linux систем довольно актуальная тема, по ряду различных причин. В первую очередь Linux имеет открытый системный код, что делает его довольно гибкой системой и позволяет пользователям, изменять и подстраивать ее под свои запросы. Во вторых это система считается наиболее безопасной по сравнению с другими ОС, большинство серверов в интернете работают на Linux, благодаря его стабильности и производительности. Также Linux имеет широчайший выбор дистрибутивов, способных удовлетворить запросы каждого, при том что вес их минимален. Использование Linux-систем остается актуальным по нескольким причинам.

Еще один плюс - Linux широко используется в встраиваемых системах, таких как маршрутизаторы, IoT-устройства и др.

Таким образом, Linux-системы продолжают оставаться важным инструментом в различных областях — от разработки ПО до серверного администрирования.

История Linux.

1991 год Линус Торвалдс - финский ученый начал разрабатывать собственное ядро операционной системы, вдохновляясь Minix — учебной версией Unix. Линус решил сделать Linux свободным и открытым программным обеспечением, что позволило другим разработчикам участвовать в его улучшении и распространении. Эмблемой Linux стал Такс (Tux) – пингвин.

Первая официальная версия Linux 1.0 вышла в 1994 году, вторая – в 1996 году. Товарный знак Linux был зарегистрирован на год раньше – в 1995. С ростом популярности Linux к проекту присоединились многие разработчики, что привело к созданию различных дистрибутивов, таких как Debian, Red Hat и Slackware. В середине 90-х годов (1993 - 1995) появились первые полноценные дистрибутивы Linux, которые включали в себя не только ядро, но и набор программного обеспечения, упрощающий установку и использование системы.

В 1998 году крупные компании, такие как IBM и Dell, начали поддерживать Linux, что способствовало его популяризации в бизнес-среде. В 2000-х Linux стал основой для серверов, суперкомпьютеров и встраиваемых систем. Появление таких технологий, как виртуализация и контейнеризация (например, Docker), также способствовало его распространению.

К 2000-м годам Linux уже занял значительную долю на серверном рынке благодаря своей стабильности, безопасности и низким затратам, несмотря на то, что Linux долгое время оставался менее популярным на настольных ПК, дистрибутивы, такие как Ubuntu, помогли изменить ситуацию. Вскоре Linux стал основой для Android — самой популярной мобильной операционной системы. На сегодняшний день Linux используется во множестве сфер: от серверов и облачных решений до персональных компьютеров и встраиваемых систем. Сообщество продолжает активно развивать ядро и сопутствующее программное обеспечение.

1. Вариант «Базовый»

1.1. Виды систем Linux.

Список родителей:

- Debian
Самый стабильный дистрибутив. Сообщество относится к разработке и стабильности очень ответственно, я бы даже сказал дотошно. Каждый релиз тестируется самым тщательным образом. Поэтому релизы выходят нечасто, а программы, входящие в его состав, далеко не последних версий.
- Slackware
Самый старый из долгожителей. Сообществу этого дистрибутива приписывают следующее высказывание: Если вы знаете Slackware – вы знаете GNU/Linux. Если вы знаете Red Hat, то всё, что вы знаете, – это Red Hat. Внушительный возраст определяет наличие богатого опыта, накопленного сообществом со всеми вытекающими...
- RedHat
Успешный коммерческий проект, имеющий в своем составе коммерческий дистрибутив Red Hat Enterprise и бесплатный – Fedora. Как следствие подобного успеха – широкое распространение и популяризация данного семейства дистрибутивов и, соответственно, очень большое сообщество пользователей.
- Gentoo
Дистрибутив, собирающийся из исходных кодов. Требует определенного уровня знаний от пользователя, однако при сборке из исходных кодов осуществляется наибольшая совместимость с конкретной машиной, на которой и для которой происходит сборка и компиляция компонентов ОС и ПО. В результате прирост производительности до 80% по сравнению с остальными дистрибутивами.

1.2. Linux в России.

Уход Microsoft из России простых пользователей не испугал. Windows с домашнего компьютера никуда не денется, а для всех остальных случаев есть сами знаете какие ресурсы. А вот обладателям корпоративных лицензий пришлось озаботиться: перспектива остаться без обновлений радует мало. Правда, и эти опасения были не слишком серьезными, было понятно, что помимо Windows есть множество клонов Linux. А у российских разработчиков уже были операционные системы, которые не только были готовы к применению, но и имели и опыт применения, и компетенции интеграторов, и развитые комьюнити.

Оказалось, что сделать свою востребованную операционку, тем более, клон Linux, совсем не сложно. Более того, российские софтверные компании выкатили даже оригинальные специализированные ОС. В реестре российского ПО имеются 62 (шестьдесят две) разные операционки.

1.3. Процесс установки и базовой настройки.

1. Загрузка образа Debian

- Перейдите на [официальный сайт Debian](<https://www.debian.org/distrib/>) и скачайте ISO-образ нужной версии (Stable, Testing или Unstable).
- Выберите архитектуру (обычно amd64 для современных компьютеров).

2. Создание загрузочного носителя

- Используйте программу, такую как Rufus (Windows), Etcher (кроссплатформенный) или dd (Linux), чтобы создать загрузочный USB-накопитель из ISO-образа.

3. Загрузка с USB

- Вставьте USB-накопитель в компьютер и перезагрузите его.
- Зайдите в BIOS/UEFI (обычно нажимая клавишу F2, F10, DEL или ESC) и выберите загрузку с USB.

4. Установка Debian

1. Выбор языка: После загрузки выберите язык установки.
2. Настройка сети: Установите сетевые параметры, если необходимо.
3. Разметка диска:
 - Выберите автоматическую или ручную разметку.
 - Если выбираете ручную, создайте разделы (например, /, swap, /home).
4. Выбор программного обеспечения:
 - Установите стандартные системные утилиты и графическую среду (например, GNOME, KDE).
5. Установка загрузчика: Установите GRUB на основной загрузочный диск.

5. Перезагрузка системы

- После завершения установки перезагрузите систему и извлеките USB-накопитель.

6. Первоначальная настройка системы

1. Вход в систему: Введите имя пользователя и пароль, созданные во время установки.
2. Обновление системы:

```
sudo apt update
sudo apt upgrade
```
3. Установка дополнительных пакетов:
 - Установите необходимые программы:

```
sudo apt install vim git curl wget
```


1.4. Основные команды Linux.

ip addr help

Мониторинг IP-адресов: Выведите все устройства с помощью следующей команды: **ip addr**

Чтобы вывести список всех сетевых интерфейсов и связанных IP-адресов, используйте команду: **ip addr show**

Вы также можете увидеть информацию об отдельной сети: **ip addr show dev [interface]**

Чтобы получить список всех IPv4 адресов, используйте: **ip -4 addr**

Чтобы получить список всех IPv6 адресов, используйте: **ip -6 addr**

lsblk - выводит информацию о дисках и созданных на них разделах, их размерах, точке монтирования.

Sudo — команда, которая наделяет правами суперпользователя. Ее используют прежде какой-либо иной команды в консоли для выполнения с правами администратора. Используется данная утилита довольно часто, особенно при обновлении и модификации пакетов ОС.

Команда apt: Синтаксис команды apt следующий:

sudo apt <действие> <пакет(ы)> <доп. параметры>

Обновить список пакетов

sudo apt update

Чтобы выполнить обновление пакетов, установленных в системе, используется команда: **sudo apt upgrade**

Данная команда только обновляет пакеты до новых версий, но никогда не удаляет и не устанавливает другие пакеты.

systemctl — это команда для управления системными службами и состоянием системы в Linux, использующая систему инициализации systemd. Вот описание основных команд:

1. status:

- Команда: **systemctl status <имя_сервиса>**
- Описание: Показывает текущее состояние указанного сервиса, включая его статус (активен/неактивен), последние логи и другую информацию.

2. stop:

- Команда: **systemctl stop <имя_сервиса>**
- Описание: Останавливает указанный сервис. Это может быть полезно для временной деактивации службы.

3. start:

- Команда: `systemctl start <имя_сервиса>`
- Описание: Запускает указанный сервис. Если служба была остановлена, эта команда активирует её.

4. restart:

- Команда: `systemctl restart <имя_сервиса>`
- Описание: Перезапускает указанный сервис. Это может быть полезно для применения изменений конфигурации или устранения неполадок.

5. poweroff:

- Команда: `systemctl poweroff`
- Описание: Выключает систему. Все запущенные процессы будут завершены, и система будет выключена.

6. suspend:

- Команда: `systemctl suspend`
- Описание: Приостанавливает работу системы, переводя её в спящий режим. В этом состоянии система сохраняет текущее состояние в оперативной памяти.

7. reboot:

- Команда: `systemctl reboot`
- Описание: Перезагружает систему. Все процессы будут завершены, и система перезапустится.

8. **top** — это стандартная утилита, которая отображает список активных процессов, их использование CPU, памяти и другие параметры в режиме реального времени.

- Основные функции:

- Отображает список процессов с информацией о PID (идентификатор процесса), пользователе, времени работы, использовании CPU и памяти.
- Обновляет информацию автоматически через заданные интервалы времени (по умолчанию 3 секунды).
- Позволяет сортировать процессы по различным критериям (например, по использованию CPU или памяти).
- Можно завершать процессы, используя комбинацию клавиш.

htop — это улучшенная версия top, предоставляющая более удобный интерфейс с цветной графикой и интерактивными возможностями.

- Основные функции:

- Визуально более привлекательный интерфейс с цветным отображением использования ресурсов.
- Позволяет использовать стрелки для навигации по списку процессов.
- Можно легко завершать или изменять приоритет процессов (nice) с помощью клавиш.

- Отображает использование CPU в виде графиков, а также информацию о загрузке системы.

- Поддерживает фильтрацию и поиск процессов.

Команда **kill** позволяет отправлять сигналы процессам, обычно для их завершения. По умолчанию она отправляет сигнал TERM, который запрашивает корректное завершение процесса.

Команда **killall** завершает процессы по имени или другим критериям, что делает ее более удобной для работы с несколькими процессами одновременно.

Команда **mkdir** используется для создания новых каталогов (папок).

Команда **chmod** изменяет права доступа к файлам и каталогам.

Команда **chown** изменяет владельца и/или группу файла или каталога.

Команда **rm** удаляет файлы и каталоги.

- Опции:

- -f: принудительно удалить без подтверждения.

- -d: удалить пустые каталоги.

- -r: рекурсивно удалить каталоги и их содержимое.

Команда **mv** используется для перемещения или переименования файлов и каталогов.

- -f: принудительно переместить, не спрашивая подтверждения.

- -d: разрешает перемещение каталогов (по умолчанию это разрешено).

1.5. Графические оболочки.

1. **GNOME** (GNU Network Object Model Environment) — одна из популярных сред рабочего стола в мире Linux. Она постоянно поддерживается и довольно легко интегрируется с аппаратным и программным обеспечением. Стоит отметить тот факт, что Canonical даже приняла GNOME для ОС Ubuntu в качестве среды рабочего стола по умолчанию. Однако надо помнить, что GNOME — не самый простой инструмент. Дизайн довольно сильно отличается от привычного виндового интерфейса, и для начального взаимодействия и простоты использования, вам придётся поколдовать с существующими расширениями. Например, если вам необходима панель задач, как в винде, для приложений или переключения окон, то вы можете попробовать для себя Dash to Panel. Нет смысла дублировать тут tutorial по установке, лучше воспользоваться инструкцией разработчика. Но помните, что с большим набором расширений оболочка начинает заметно подъедать системные ресурсы слабощных машинок.

2. **GNOME Flashback** (Ubuntu Classic/панель GNOME) — это базовая и классическая среда рабочего стола, которая даёт возможность ностальгировать о чистом виде системы Ubuntu. Быстрая и лёгкая, основана на ранней версии Ubuntu.

3. **KDE Plasma**. После GNOME — оболочка KDE (K Desktop Environment) Plasma является наиболее используемой и настраиваемой средой рабочего стола для ОС Linux. Эта оболочка предлагает о-о-очень широкий спектр опций. Например, если вы захотите сделать что-то windows- или macos-подобное, то при должных усилиях — у вас всё получится :). Если вам нравится возиться в настройках системы и вы готовы приложить некоторые усилия, то Plasma это то, что вам нужно. Её можно назвать «швейцарским ножом» в мире рабочих столов Linux. В отличие от Ubuntu Unity 3D, для бесперебойной работы требуется аппаратное 3d ускорение.

4. **XFCE**. Если у вас есть старая или не очень мощная машина,

например, старый нетбук, то XFCE станет отличным выбором. Он довольно лёгкий и обеспечивает баланс между лёгкими рабочими столами, такими как LXDE или LXQt, и полноценными, такими как KDE и GNOME, про которые написано выше. Оболочка малотребовательна к ресурсам компьютера и при этом на борту имеет много вариантов настройки. Всё это делает XFCE одной из самых чистых и простых в использовании, особенно когда требуется нересурсоёмкая среда без аппаратного ускорения.

5. **LXDE** (облегчённая среда рабочего стола X11). Это невероятно лёгкая среда рабочего стола Linux, которая ориентирована на высокую производительность, при этом требует ещё меньше ресурсов, чем XFCE. Для её работы не требуется никакого аппаратного ускорения. Очень быстрая, гибко настраиваемая и не занимает много места в оперативной памяти.

6. **Cinnamon** — это, конечно, одна из самых популярных сред рабочего стола. По умолчанию поставляется только в Linux Mint. Эта среда рабочего стола является ответвлением оболочки GNOME, так что тут стоит вспомнить требование про аппаратное 3D-ускорение. Хотя она и может работать с программным рендерингом, но будет сильно тормозить. Cinnamon хорошо подойдёт для тех, кто только что перешёл с ОС Windows, поскольку эта оболочка наиболее близка к ней по внешнему виду. Среда бережно относится к пожиранию ресурсов, что делает её идеальным кандидатом для чуть более устаревшей машины, которая уже не тянет винду.

7. **Pantheon** — это лёгкая и модульная среда рабочего стола, которая используется в elementary OS в качестве рабочего стола по умолчанию. Графический интерфейс этой оболочки идентичен Apple Mac OS. Его можно установить на любую машину с Ubuntu. Работа с этой средой почти такая же, как в macOS. В ней есть даже больше жестов и имеются функции сенсорной панели. Pantheon также является одним из самых доступных интерфейсов, которые подходят для дисплеев со сверхвысоким разрешением (HiDPI/Retina), поэтому её стоит рассматривать в качестве рабочего решения для замены

macOS. Эта среда автоматически определит разрешение вашего экрана и сделает изображение подходящим для вашего монитора.

8. **Budgie** вполне подойдёт для тех, кому нужен такой же настраиваемый рабочий стол, как в KDE Plasma, но без мороки. Budgie — это обычная рабочая среда, созданная для ОС Solus. Она предоставляет множество настроек, таких как виджеты рабочего стола, красивые темы и значки, режим тёмной темы и расширенные функции панели.

9. **Qtile** — это тайловый оконный менеджер, который написан только на языке Python. Qtile — это очень простая, маленькая и легко расширяемая среда. Она позволяет легко писать свои собственные раскладки, виджеты, встраиваемые команды. По сути, это оконный менеджер, который написан программистами и исключительно для программистов.

10. **i3wm** — фреймовый оконный менеджер. Компактный, лёгкий и простой в использовании. Его нельзя назвать полноценной средой рабочего стола. Скорее, он необходим для управления окнами экрана и предназначен в первую очередь для настоящих гиков Linux. Простому пользователю он вряд ли может быть интересен.

1.6. Пакетные менеджеры.

Пакетные менеджеры в Linux — это инструменты, которые упрощают установку, обновление и удаление программного обеспечения. Они управляют пакетами, которые содержат скомпилированные программы и их зависимости. Вот основные типы пакетных менеджеров и их особенности:

1. APT (Advanced Package Tool)

- Используется в: Debian, Ubuntu и их производных.
- Команды:
 - apt update — обновление списка доступных пакетов.
 - apt install <package> — установка пакета.
 - apt remove <package> — удаление пакета.
- Особенности: Поддерживает управление зависимостями, автоматическую установку необходимых библиотек.

2. YUM/DNF (Yellowdog Updater Modified / Dandified YUM)

- Используется в: Red Hat, CentOS, Fedora.
- Команды:
 - yum update или dnf update — обновление системы.
 - yum install <package> или dnf install <package> — установка пакета.
 - yum remove <package> или dnf remove <package> — удаление пакета.
- Особенности: DNF является более современным вариантом YUM и обеспечивает более эффективное управление зависимостями.

3. Pacman

- Используется в: Arch Linux и его производных.
- Команды:
 - pacman -Syu — обновление системы.
 - pacman -S <package> — установка пакета.
 - pacman -R <package> — удаление пакета.
- Особенности: Простой и мощный, использует бинарные пакеты и поддерживает управление зависимостями.

4. Zypper

- Используется в: openSUSE.
- Команды:
 - zypper refresh — обновление списка пакетов.

- `zypper install <package>` — установка пакета.
- `zypper remove <package>` — удаление пакета.
- Особенности: Поддерживает репозитории и управление зависимостями.

5. Snap

- Используется в: различных дистрибутивах Linux.
- Команды:
 - `snap install <package>` — установка snap-пакета.
 - `snap remove <package>` — удаление snap-пакета.
- Особенности: Позволяет устанавливать приложения с их зависимостями в изолированном окружении.

6. Flatpak

- Используется в: различных дистрибутивах Linux.
- Команды:
 - `flatpak install <remote> <package>` — установка flatpak-пакета.
 - `flatpak uninstall <package>` — удаление flatpak-пакета.
- Особенности: Обеспечивает изоляцию приложений и управление зависимостями.

1.7. Терминал. Работа с дисками и внешними устройствами.

Работа с дисками и внешними устройствами в терминале Linux включает в себя несколько ключевых команд и концепций. Вот основные аспекты, которые могут быть полезны:

1. Просмотр подключенных устройств

- `lsblk`: Показывает список всех блочных устройств (дисков, разделов и т.д.).

- `fdisk -l`: Показывает информацию о всех дисках и их разделах.

```
sudo fdisk -l
```

- `df -h`: Показывает информацию о файловых системах и их использовании.

```
df -h
```

2. Монтирование и размонтирование

- Монтирование устройства:

Чтобы смонтировать устройство, сначала создайте точку монтирования (каталог):

```
sudo mkdir /mnt/mydrive
```

Затем смонтируйте устройство:

```
sudo mount /dev/sdX1 /mnt/mydrive
```

Здесь `/dev/sdX1` — это устройство, которое вы хотите смонтировать (например, `/dev/sdb1`).

- Размонтирование устройства:

Чтобы размонтировать устройство, используйте команду:

```
sudo umount /mnt/mydrive
```

3. Форматирование дисков

- Форматирование устройства:

Чтобы отформатировать диск в файловую систему (например, `ext4`):

```
sudo mkfs.ext4 /dev/sdX1
```

- Другие файловые системы:

Для других файловых систем можно использовать:

- `mkfs.vfat` для FAT32

- `mkfs.ntfs` для NTFS

4. Работа с разделами

- Создание и изменение разделов:

Используйте `fdisk` или `parted` для работы с разделами.

Пример использования `fdisk`:

```
sudo fdisk /dev/sdX
```

- Команды внутри `fdisk`:

- `m` — показать помощь.
- `n` — создать новый раздел.
- `d` — удалить раздел.
- `w` — сохранить изменения и выйти.

5. Проверка и исправление файловых систем

- `fsck`: Проверка и исправление файловых систем.

```
sudo fsck /dev/sdX1
```

6. Управление RAID и LVM

- RAID: Используйте `mdadm` для управления RAID-массивами.

- LVM (Logical Volume Manager): Используйте команды `lvcreate`, `lvextend`, `lvreduce` для управления логическими томами.

2. Вариант «Продвинутый»

2.1. Администрирование *Linux*.

Linux — стабильная и отказоустойчивая система, именно эти качества данного «зверя» делают его незаменимым инструментом для обслуживания более чем 1 млн серверов по всему миру (90% мировых серверов работают на *Linux* системах).

Администрирование *Linux* требует от администратора прежде всего понимания работы железа с программным обеспечением, что играет важную роль в стабильности и качестве работы как отдельного сервера, так и всей инфраструктуры в целом.

Подавляющее большинство серверов *Linux* работают без GUI (Graphic User Interface (графический интерфейс пользователя)), либо на «жаргоне» «сисадминов» - «Без головы». Данный подход к работе системы позволяет добиться высокой производительности за счет отключения всех ненужных серверу инструментов поддержки графического ядра и всей графической периферии. Для работы в таком режиме необходимо знать, как минимум, основные команды терминала и их ключи, которые помогут выжать из серверного железа максимум.

Администрирование *Linux* можно производить и в графическом режиме, но следует помнить, что часть настроек так или иначе придется проводить через терминал, что накладывает на администратора дополнительную ответственность и внимательность, т. к. настройка системы через терминал прививает обслуживающему персоналу понимание работы системы, оборудования, а также открывает дополнительные возможности для отладки, конфигурирования и работы, что в свою очередь дает некую свободу в плане выбора софта: вы можете использовать как уже готовое ПО с открытым исходным кодом, так и внести в него свои правки, либо же написать такое ПО, которое будет идеально подходить для вашей ситуации.

2.2. Терминал. Файловые системы.

Особое место в Linux, как пожалуй и в других системах, занимают файловые системы, суть которых состоит в организации и управлении файлами на устройстве хранения.

Основными в работе файловыми системами являются системы: FAT, exFAT, ext3, ext4, иногда применяются BrFS, hfs, ahfs и другие. Все они имеют свои достоинства и недостатки. Зачастую в системах семейства Linux и на сетевых файловых хранилищах (NAS) можно встретить фс* типа Ext4, её достоинствами являются: пространственная (extent) запись файлов, что уменьшает фрагментацию, но повышает производительность; именно в этой фс максимальный объем одного раздела диска может достигать 1 эксбибайта (1152921504.6068 ГБ); в сравнении с ext3, которая не может содержать более, чем 32000 подкаталогов, ext4 снимает это ограничение и позволяет создавать неограниченное количество подкаталогов.

Основные команды для работы с фс:

`mkfs.fat -F32 /dev/sd**` - создает фс FAT32 на диске `/dev/sd**`

`mkfs.ext4 /dev/sd**` - создает фс Ext4 на диске `/dev/sd**`

`mkfs.ntfs -t /dev/sd**` - создает фс NTFS (windows) на диске `/dev/sd**`,

стоит заметить, что для работы с фс NTFS потребуется установить пакет «ntfs-3g».

*ФС(фс) — Файловая система.

2.3. Настройка сети.

Для корректной работы нашего сервера в сети стоит задать ему статический ip адрес, следующие шаги помогут вам решить эту задачу: включаем терминал и вводим команду ip a, она покажет вам какие сетевые подключения имеет ваш сервер: приводим выдержку из терминала (рис 2.1)

```
[root@altsrv ~]# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:41:2b:e0 brd ff:ff:ff:ff:ff:ff
    inet 10.0.2.15/24 brd 10.0.2.255 scope global dynamic noprefixroute enp0s3
        valid_lft 86397sec preferred_lft 86397sec
    inet6 fe80::a00:27ff:fe41:2be0/64 scope link
        valid_lft forever preferred_lft forever
3: enp0s8: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:8a:f3:fd brd ff:ff:ff:ff:ff:ff
    inet6 fe80::d63:aa0f:4697:d1b7/64 scope link noprefixroute
        valid_lft forever preferred_lft forever
[root@altsrv ~]#
```

Рис.2.1

Как мы видим, наш адрес от провайдера это 10.0.2.15 на интерфейсе enp0s3, его мы трогать не будем, оставим как есть. Нас интересует интерфейс внутренней сети на интерфейсе с названием enp0s8, его мы и будем настраивать. Переходим в терминал и текстовым редактором NANO открываем файл настройки нашего enp0s8 по адресу /etc/net/ifaces/enp0s8/options : (рис.2.2)

```
GNU nano 7.2 /etc/net/ifaces/enp0s8/options Изменён
NM_CONTROLLED=yes
DISABLED=yes

Справка      Записать   Поиск      Вырезать   Выполнить   Позиция
Выход        ЧитФайл   Замена     Вставить   Выровнять   К строке
```

Рис.2.2

Открыв
данный файл, мы не

увидим ничего полезного для нас, поэтому, очищаем файл и приводим его к следующему виду: (рис.2.3)

```

GNU nano 7.2 /etc/net/ifaces/enp0s8/options
BOOTPROTO=static
TYPE=enp0s8
NM_CONTROLLED=yes
DISABLED=yes
CONFIG_WIRELESS=no
SYSTEMD_BOOTPROTO=static
CONFIG_IPV4=yes
SYSTEMD_CONTROLLED=no
CONFIG_IPV6=no

```

Рис.2.3

```

Прочитано 9 строк
^G Справка      ^O Записать     ^W Поиск        ^K Вырезать     ^T Выполнить    ^C Позиция
^X Выход        ^R ЧитФайл     ^N Замена       ^U Вставить     ^J Выводить     ^_/ К строке

```

Этот набор указателей задет протокол настройки сети на интерфейсе enp0s8 на статический, отключает Wi-Fi, т.к. его там и нет, отключает использование ipv6. Сохраняем файл, закрываем. Перезагружаем работу сети командами: `systemctl restart network && systemctl restart NetworkManager`

Протокол указали, теперь необходимо задать ip адрес сервера. Для начала создадим профиль интерфейса enp0s8, назовем его «*native_enp0s8*» и зададим ему статический адрес: «`nmcli connection add con-name "native_enp0s8" type ethernet ifname enp0s8`», после чего система проинформирует вас: (рис.2.4)

```

[root@altsrv ~]# nmcli connection add con-name "native_enp0s8" type ethernet ifname enp0s8
Подключение «native_enp0s8» (d8a4352f-d359-458c-a2b3-9730ea6574b7) успешно добавлено.

```

Рис.2.4

После чего зададим статику командой: «`nmcli connection modify native_enp0s8 connection.autoconnect yes ipv4.method manual ipv4.address 192.168.62.1/24`» и поднимем (запустим) наше подключение командой: «`nmcli connection up native_enp0s8`». В свою очередь, система проинформирует вас: (рис.2.5)

```

[root@altsrv ~]# nmcli connection modify native_enp0s8 connection.autoconnect yes ipv4.method manual ipv4.address 192.168.62.1/24
[root@altsrv ~]# nmcli connection up native_enp0s8
Подключение успешно активировано (активный путь D-Bus: /org/freedesktop/NetworkManager/ActiveConnection/6)
[root@altsrv ~]#

```

Рис.2.5

На этом можно считать настройку статического ip адреса завершенной. Проверить корректность настроек можно командой «`ip a`» (рис.2.6)

```

[root@altsrv ~]# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
   link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
   inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
   inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
   link/ether 08:00:27:41:2b:e0 brd ff:ff:ff:ff:ff:ff
   inet 10.0.2.15/24 brd 10.0.2.255 scope global dynamic noprefixroute enp0s3
       valid_lft 85073sec preferred_lft 85073sec
   inet6 fe80::a00:27ff:fe41:2be0/64 scope link
       valid_lft forever preferred_lft forever
3: enp0s8: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
   link/ether 08:00:27:8a:f3:fd brd ff:ff:ff:ff:ff:ff
   inet 192.168.62.1/24 brd 192.168.62.255 scope global noprefixroute enp0s8
       valid_lft forever preferred_lft forever
   inet6 fe80::140c:d5fb:e45f:b363/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
[root@altsrv ~]# _

```

Рис.2.6

2.4. Настройка сервера DHCP.

Этап 1. Подготовка менеджера пакетов apt-get.

*Важно учитывать, что после установки `alt server` ваш пользователь по умолчанию не будет иметь прав на внесение каких-либо изменений в системные файлы. Есть несколько способов исправления данной ситуации, так например можно внести необходимые изменения в файл по адресу: `/etc/sudoers`, либо использовать учетную запись суперпользователя «root», что мы собственно и будем проделывать ниже: открываем терминал и вводим команду «`su`», после чего система потребует от вас ввода пароля суперпользователя, который вы настраивали при установке системы. Если при вводе пароля у вас ничего не отображается (что-то вроде символов `****` или `#####`), то это совершенно нормальное явление. Стоит помнить, что используя учетную запись суперпользователя вы можете нанести вред вашей системе, поэтому следует быть предельно внимательным и аккуратным!*

Прежде, чем начать установку самого сервера динамических адресов, сделаем обновление пакетов системы для того, чтобы обновить все устаревшие пакеты программного обеспечения системы.

Вводим в терминале команду: **apt-get update**, если система покажет необходимость обновления пакетов, то вводим команду на обновление найденных пакетов: **apt-get upgrade**. На этом процесс подготовки завершен.

Этап 2. Установка и настройка dhcp-сервера.

Начинаем процесс установки, вводим в терминал команду: `apt-get install dhcp-server`, после обнаружения данного пакета в репозиториях, система предложит вам установить самую последнюю версию найденного пакета, с чем мы соглашаемся, вводя символ «Y» с клавиатуры, что означает «YES» или «ДА».

Процесс установки завершен. Теперь необходимо убедиться в том, что сервер dhcp выключен: `systemctl status dhcpd` данная команды покажет статус сервиса, после установки, по умолчанию сервер будет выключен (`inactive(dead)`) и его включение будет невозможным до тех пор, пока вы не внесете правки в конфигурационный файл сервера, что мы сейчас и сделаем. Прежде, чем вносить конфигурацию, нужно создать сам конфигурационный файл, обычно он называется `dhcpd.conf.example`, т.е. конфиг для примера, в нем есть много полезной информации, которую мы настоятельно рекомендуем внимательно изучить, а мы продолжаем. Введем в терминал команду: `cp /etc/dhcp/dhcpd.conf.example /etc/dhcp/dhcpd.conf`, данной командой мы скопируем файл примера и сделаем из него основной, тогда, при вводе команды `nano /etc/dhcp/dhcpd.conf` мы получим следующее: (Рис.2.7)


```

GNU nano 7.2 /etc/dhcp/dhcpd.conf
# dhcpd.conf
#
# Sample configuration file for ISC dhcpd
#
# option definitions common to all supported networks...
option domain-name "example.org";
option domain-name-servers ns1.example.org, ns2.example.org;

default-lease-time 600;
max-lease-time 7200;

# Use this to enable / disable dynamic dns updates globally.
#ddns-update-style none;

# If this DHCP server is the official DHCP server for the local
# network, the authoritative directive should be uncommented.
#authoritative;

# Use this to send dhcp log messages to a different log file (you also
# have to hack syslog.conf to complete the redirection).
log-facility local7;

# No service will be given on this subnet, but declaring it helps the

```

Рис.2.7

Приведем конфигурационный файл к следующему виду: (рис.2.8), где: subnet 192.168.62.0 — сеть для клиентов нашей сети, а subnet 10.0.2.0 — сеть, получаемая нами от провайдера. Подробный разбор данного файла показан в *приложении по dhcp*.

```

GNU nano 7.2 /etc/dhcp/dhcpd.conf
log-facility local7;

subnet 192.168.62.0 netmask 255.255.255.0 {
  range 192.168.62.2 192.168.62.254;
  option domain-name-servers 192.168.62.1;
  option domain-name "testdomain.net";
  option routers 192.168.62.1;
  option broadcast-address 192.168.62.255;
  default-lease-time 600;
  max-lease-time 7200;
}

subnet 10.0.2.0 netmask 255.255.255.0 {
}

```

Рис.2.8

Этап 3. Запуск и проверка dhcp-сервера.

Чтобы применить все проделанные ваши действия перезапустим сервис «dhcp» и проверим его статус:

1. `systemctl restart dhcpd` — команда перезапустит сервис средствами системы `systemd`
2. `systemctl status dhcpd` — команда покажет статус сервиса, если все настроено правильно, то данная команда покажет следующее: (рис.2.9)

```

[root@altsrv ~]# systemctl restart dhcpd
[root@altsrv ~]# systemctl status dhcpd
● dhcpd.service - DHCPv4 Server Daemon
   Loaded: loaded (/lib/systemd/system/dhcpd.service; disabled; vendor preset: disabled)
   Active: active (running) since Mon 2024-03-11 10:47:52 +08; 1s ago
     Docs: man:dhcpd(8)
           man:dhcpd.conf(5)
   Process: 6246 ExecStartPre=/etc/chroot.d/dhcpd.all (code=exited, status=0/SUCCESS)
  Main PID: 6364 (dhcpd)
    Tasks: 1 (limit: 1149)
   Memory: 4.5M
      CPU: 111ms
   CGroup: /system.slice/dhcpd.service
           └─6364 /usr/sbin/dhcpd -4 -f --no-pid

map 11 10:47:52 altsrv dhcpd[6364]: Copyright 2004-2021 Internet Systems Consortium.
map 11 10:47:52 altsrv dhcpd[6364]: All rights reserved.
map 11 10:47:52 altsrv dhcpd[6364]: For info, please visit https://www.isc.org/software/dhcp/
map 11 10:47:52 altsrv dhcpd[6364]: Listening on LPF/enp0s8/08:00:27:8a:f3:fd/192.168.62.0/24
map 11 10:47:52 altsrv dhcpd[6364]: Sending on   LPF/enp0s8/08:00:27:8a:f3:fd/192.168.62.0/24
map 11 10:47:52 altsrv dhcpd[6364]: Listening on LPF/enp0s3/08:00:27:41:2b:e0/10.0.2.0/24
map 11 10:47:52 altsrv dhcpd[6364]: Sending on   LPF/enp0s3/08:00:27:41:2b:e0/10.0.2.0/24
map 11 10:47:52 altsrv dhcpd[6364]: Sending on   Socket/fallback/fallback-net
map 11 10:47:52 altsrv dhcpd[6364]: Wrote 0 leases to leases file.
map 11 10:47:52 altsrv dhcpd[6364]: Server starting service.
[root@altsrv ~]#

```

Рис.2.9

На этом можно считать настройку роли dhcp сервера завершенной.

2.5. Настройка сервера DNS.

DNS Кэширующий.

Установка сервера: `apt install bind9`

Файл конфигурации: `nano /etc/bind/named.conf.options`, приводится к виду:

```
options {  
    directory "/var/cache/bind";  
  
    forwarders {  
        77.88.8.8; 8.8.8.8;  
    };  
    dnssec-validation auto;  
    auth-nxdomain no;  
    listen-on {192.168.1.2; 127.0.0.1; };  
    allow-query { any; };  
    allow-query-cache { any; };  
};
```

Где:

forwarders - адреса внешних DNS серверов. У меня указаны DNS Яндекса и Гугл, вы можете указать провайдерские;

listen-on - на каких адресах слушать запросы;

allow-query - с каких ip адресов разрешить запросы;

allow-query-cache - тоже самое, только для кэша.

После сделанных действий, сохраняем файл конфигурации, выполняем перезагрузку сервиса bind9 командой: **systemctl restart bind9**

2.6. Настройка сервера Samba.

1. Установка Samba

Для начала установим Samba. В зависимости от вашего дистрибутива используйте одну из следующих команд:

Для Debian/Ubuntu:

```
sudo apt update  
sudo apt install samba
```

Для CentOS/RHEL:

```
sudo yum install samba samba-client samba-common
```

2. Настройка Samba

После установки необходимо отредактировать конфигурационный файл Samba, который обычно находится по пути `/etc/samba/smb.conf`.

```
sudo nano /etc/samba/smb.conf
```

3. Основные параметры конфигурации

Добавьте или измените следующие секции в файле `smb.conf`:

Пример настройки общего каталога

```
[shared]  
path = /srv/samba/shared  
browsable = yes  
writable = yes  
guest ok = yes  
read only = no  
create mask = 0755  
directory mask = 0755
```

- `[shared]` — имя общего ресурса.
- `path` — путь к каталогу, который вы хотите сделать общим.
- `browsable` — делает ресурс видимым в сети.
- `writable` — позволяет запись в общий каталог.
- `guest ok` — разрешает доступ без аутентификации.
- `create mask` и `directory mask` — устанавливают права доступа для создаваемых файлов и директорий.

4. Создание общего каталога

Создайте каталог, который вы указали в конфигурации:

```
sudo mkdir -p /srv/samba/shared
```

Установите нужные права доступа:

```
sudo chmod 0777 /srv/samba/shared
```

5. Перезапуск Samba

После внесения изменений в конфигурацию необходимо перезапустить службы Samba:

```
sudo systemctl restart smbd
```

```
sudo systemctl restart nmbd
```

6. Настройка брандмауэра (если требуется)

Если у вас включен брандмауэр, откройте порты для Samba:

Для UFW (Ubuntu):

```
sudo ufw allow samba
```

Для firewalld (CentOS/RHEL):

```
sudo firewall-cmd --permanent --add-service=samba
```

```
sudo firewall-cmd --reload
```

7. Проверка конфигурации

Вы можете проверить конфигурацию Samba на наличие ошибок:

```
testparm
```

8. Подключение к Samba с Windows

1. Откройте проводник Windows.

2. В адресной строке введите \\IP_адрес_сервера, например, \\192.168.1.100.

3. Должен открыться общий ресурс, который вы настроили.

9. Дополнительные настройки (по желанию)

Если вы хотите настроить доступ с аутентификацией, создайте пользователя Samba:

```
sudo smbpasswd -a username
```

Затем измените параметры в smb.conf, убрав `guest ok = yes` и установив `valid users = username`.

2.7. Настройка сервера CUPS.

1. Установка CUPS

Для начала установим CUPS. В зависимости от вашего дистрибутива используйте одну из следующих команд:

Для Debian/Ubuntu:

```
sudo apt update  
sudo apt install cups
```

Для CentOS/RHEL:

```
sudo yum install cups
```

2. Запуск и настройка CUPS

После установки необходимо запустить службу CUPS и включить её автозапуск:

```
sudo systemctl start cups  
sudo systemctl enable cups
```

3. Настройка доступа к веб-интерфейсу CUPS

CUPS предоставляет веб-интерфейс для управления принтерами. Для доступа к нему откройте файл конфигурации:

```
sudo nano /etc/cups/cupsd.conf
```

Измените следующие строки:

- Найдите строку Listen localhost:631 и измените её на:

```
Port 631
```

- Найдите секцию <Location /> и измените параметры доступа:

```
<Location />  
Order allow,deny  
Allow @LOCAL  
</Location>
```

- Найдите секцию <Location /admin> и измените параметры доступа:

```
<Location /admin>  
Order allow,deny  
Allow @LOCAL  
</Location>
```

Эти изменения позволят доступ к CUPS с локальной сети.

4. Перезапуск CUPS

После внесения изменений в конфигурацию необходимо перезапустить службу CUPS:

```
sudo systemctl restart cups
```

5. Добавление принтера

1. Откройте веб-браузер и перейдите по адресу: <http://localhost:631>.
2. Перейдите в раздел "Administration".
3. Нажмите на "Add Printer".
4. Вам будет предложено ввести имя пользователя и пароль (если вы используете аутентификацию).
5. Выберите принтер из списка и следуйте инструкциям для его настройки.

6. Настройка принтера для общего доступа

Чтобы сделать принтер доступным для других пользователей в сети:

1. В веб-интерфейсе CUPS выберите ваш принтер.
2. Перейдите в раздел "Administration".
3. Убедитесь, что опция "Share printers connected to this system" включена.

7. Проверка и управление принтерами

Вы можете управлять принтерами через веб-интерфейс CUPS, где доступны функции, такие как:

- Печать тестовой страницы.
- Изменение настроек принтера.
- Просмотр очереди печати.

8. Настройка брандмауэра (если требуется)

Если у вас включен брандмауэр, откройте порт для CUPS:

Для UFW (Ubuntu):

```
sudo ufw allow 631/tcp
```

Для firewalld (CentOS/RHEL):

```
sudo firewall-cmd --permanent --add-port=631/tcp
```

```
sudo firewall-cmd --reload
```

2.8. Настройка сервера Web.

1. Установка Apache

Установите веб-сервер Apache:

```
sudo apt install apache2
```

2. Проверка статуса сервера

После установки проверьте, работает ли Apache:

```
sudo systemctl status apache2
```

Вы должны увидеть статус "active (running)". Если сервер не запущен, запустите его:

```
sudo systemctl start apache2
```

3. Настройка брандмауэра

Если у вас включен брандмауэр, разрешите HTTP и HTTPS трафик:

```
sudo ufw allow 'Apache Full'
```

4. Проверка работы сервера

Откройте веб-браузер и введите IP-адрес вашего сервера (или `http://localhost`, если вы работаете на локальной машине). Вы должны увидеть страницу приветствия Apache.

5. Создание веб-сайта

1. Создание директории для сайта:

Создайте новую директорию для вашего сайта:

```
sudo mkdir /var/www/html/ваш_сайт
```

2. Настройка прав доступа:

Установите правильные права доступа к директории:

```
sudo chown -R www-data:www-data /var/www/html/ваш_сайт
```

```
sudo chmod -R 755 /var/www/html/ваш_сайт
```

3. Создание файла индекса:

Создайте файл `index.html` для тестирования:


```
echo "<h1>Привет, мир!</h1>" | sudo tee /var/www/html/ваш_сайт/index.html
```

6. Настройка виртуального хоста

1. Создайте файл конфигурации для вашего сайта:

```
sudo nano /etc/apache2/sites-available/ваш_сайт.conf
```

Добавьте следующее содержимое:

```
<VirtualHost *:80>
    ServerAdmin admin@ваш_домен
    ServerName ваш_домен
    ServerAlias www.ваш_домен
    DocumentRoot /var/www/html/ваш_сайт

    <Directory /var/www/html/ваш_сайт>
        Options Indexes FollowSymLinks
        AllowOverride None
        Require all granted
    </Directory>

    ErrorLog ${APACHE_LOG_DIR}/error.log
    CustomLog ${APACHE_LOG_DIR}/access.log combined
</VirtualHost>
```

2. Включите новый виртуальный хост:

```
sudo a2ensite ваш_сайт.conf
```

3. Отключите конфигурацию по умолчанию (опционально):

```
sudo a2dissite 000-default.conf
```

4. Перезагрузите Apache:

```
sudo systemctl reload apache2
```

2.9. Работа с *wget*, *curl*, *dpkg*.

1. Работа с *wget*. *wget* — это утилита для загрузки файлов из сети. Основные команды:

- Загрузка файла:
`wget http://example.com/file.zip`
- Загрузка файла с изменением имени:
`wget -O newname.zip http://example.com/file.zip`
- Загрузка всех файлов с сайта:
`wget -r http://example.com/`
- Загрузка с продолжением:
`wget -c http://example.com/file.zip`

2. Работа с *curl*. *curl* — это инструмент для работы с URL, который поддерживает множество протоколов. Основные команды:

- Загрузка файла:
`curl -O http://example.com/file.zip`
- Сохранение файла под другим именем:
`curl -o newname.zip http://example.com/file.zip`

3. Работа с *dpkg*. *dpkg* — это низкоуровневая утилита для управления пакетами в Debian и его производных (например, Ubuntu).

Основные команды:

- Установка пакета:
`sudo dpkg -i package.deb`
- Удаление пакета:
`sudo dpkg -r package_name`
- Просмотр информации о пакете:
`dpkg -s package_name`
- Список всех установленных пакетов:
`dpkg -l`

3. Практические задачи.

№ 1. Установить Debian 12 используя установщик типа «netinst», образ брать по ссылке <https://ilspo.ru/beta/ftp/iso/linux/debian-12.4.0-amd64-netinst.iso>;

№2. Используя штатные средства ОС Debian 12, получить файл-инструкцию используя следующую ссылку: <https://ilspo.ru/beta/ftp/txt/samba.txt>;

№3. Установить любой web сервер и запустить на хостинг «свой» сайт (заранее следует создать файл index.html и style.css для проверки работоспособности).

№4. Используя предложенные инструкции по ссылке (<https://ilspo.ru/lin>), выполнить настройку корпоративной сети с использованием linux сервера (требования: автоматическая выдача адресов клиентам сети, разрешение имен сайтов на уровне локальной сети, создать сетевое файловое хранилище с использованием сервера samba).

№5*. Используя штатные средства системы и установленные пакеты mdadm, parted, exfat-progs выполнить настройку raid массива типа 0 и 1 для организации надежного файлового хранилища для web сервера.

№6*. Используя штатные средства системы, пакеты samba-ad-dc, bind9, isc-dhcp-server, организовать работу контроллера домена (с разрешением внутримоменных имен и выдачей ip адресов клиентам) с применением консольной утилиты samba-tool.

№7*. Выполнить техническую смену пароля пользователя root на производных системах ubuntu (по умолчанию пароль для root не задается, а попасть в него, используя штатные средства не так и просто, следует применить смекалку).