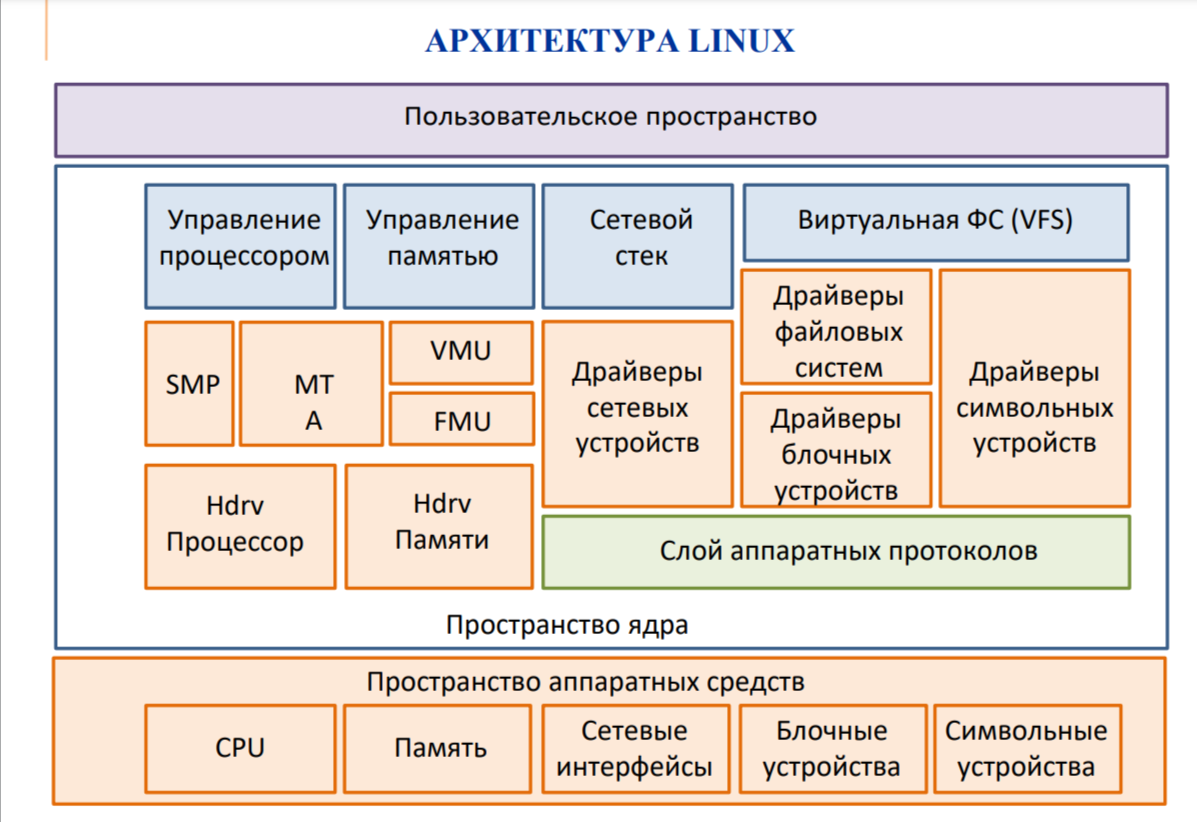
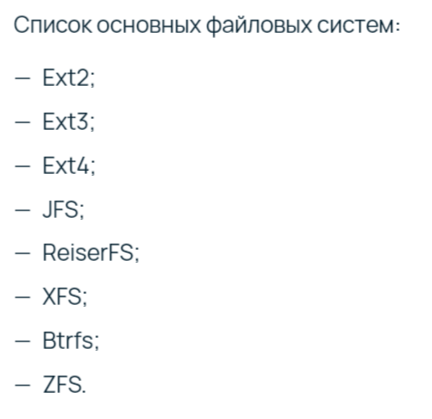
1. Введение
2. Краткая история
3. Введение в установку
4. Структура ОС



1. Файловая система Linux
2. Обзор файловых систем Linux и их достоинства и недостатки



**Ext2, Ext3, Ext4**

Первая группа ФС — Extended Filesystem (Ext2, Ext3, Ext4) — является стандартом для Linux. Как следствие, это самые распространенные системы. Они редко обновляются, но зато стабильны.

**ReiserFS**

Главные особенности: увеличенная производительность и более широкие возможности. Изменяющийся размер блока дает пользователю возможность объединять небольшие файлы в один блок, таким образом удается избежать фрагментации и повысить качество работы ФС в целом. Размер разделов можно менять прямо в процессе работы, однако эта ФС может показать нестабильные результаты и потерять данные, например, при отключении энергии.

**XFS**

Однако, в отличие от аналогов, в логи записывает исключительно те изменения, которые претерпевают метаданные. Важные особенности: быстро работает с файлами сравнительно большого размера, умеет выделять место в отложенном режиме, а также менять размеры разделов в процессе работы. Часто встречается в дистрибутивах на основе Red Hat. Минусы: нельзя уменьшить размер разделов, сложно восстанавливать данные и можно потерять информацию при отключении питания.

**Btrfs**

Современная ФС, главной особенностью которой является высокая отказоустойчивость. Из дополнительных «бонусов»: удобна для сисадминов и поддерживает сравнительно простой процесс восстановления данных. Поддерживает подтома, разрешает менять размеры разделов в динамическом режиме и позволяет делать снапшоты. Отличается высокой производительность. Применяется как ФС, установленная по умолчанию, в OpenSUSE и SUSE Linux. Главный минус — нестабильность (нарушена обратная совместимость, сложная для поддержки и так далее).

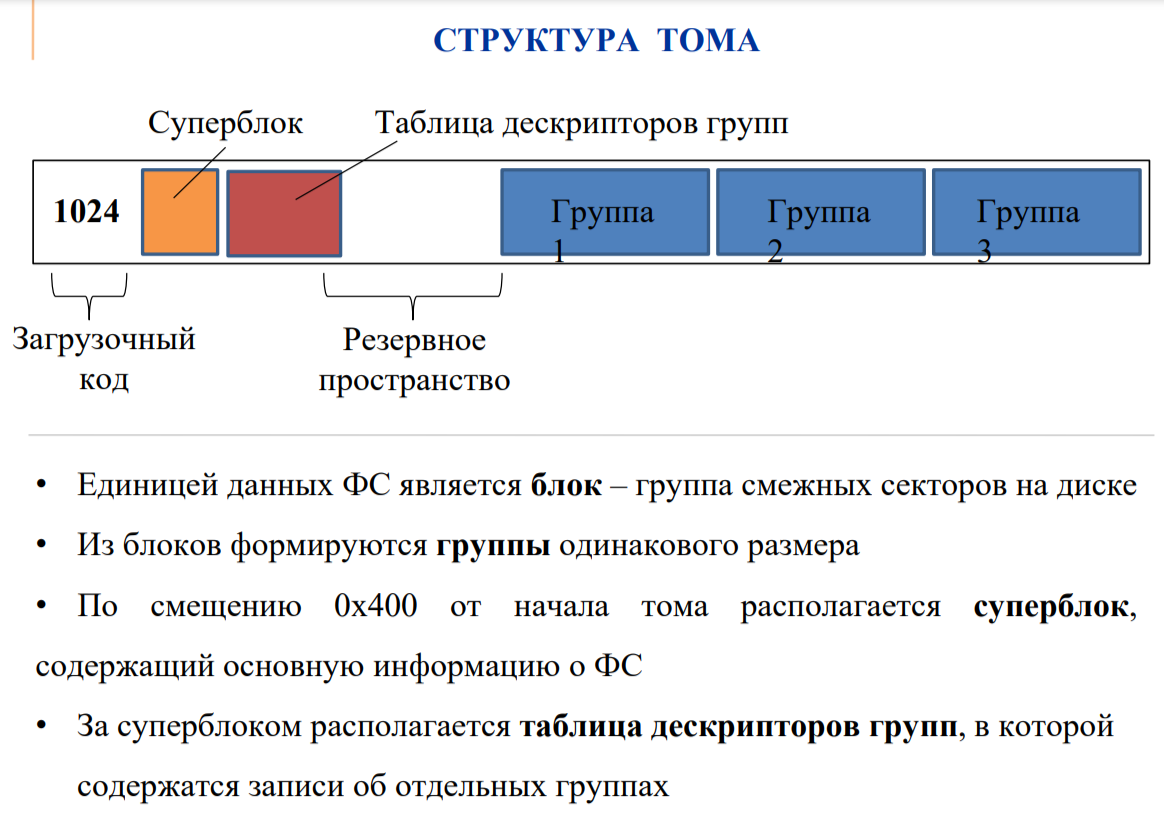
**F2FS**

Flash-Friendly File System входит в состав ядра ОС Linux и предназначена для использования с хранилищем на основе флеш-памяти.

**OpenZFS**

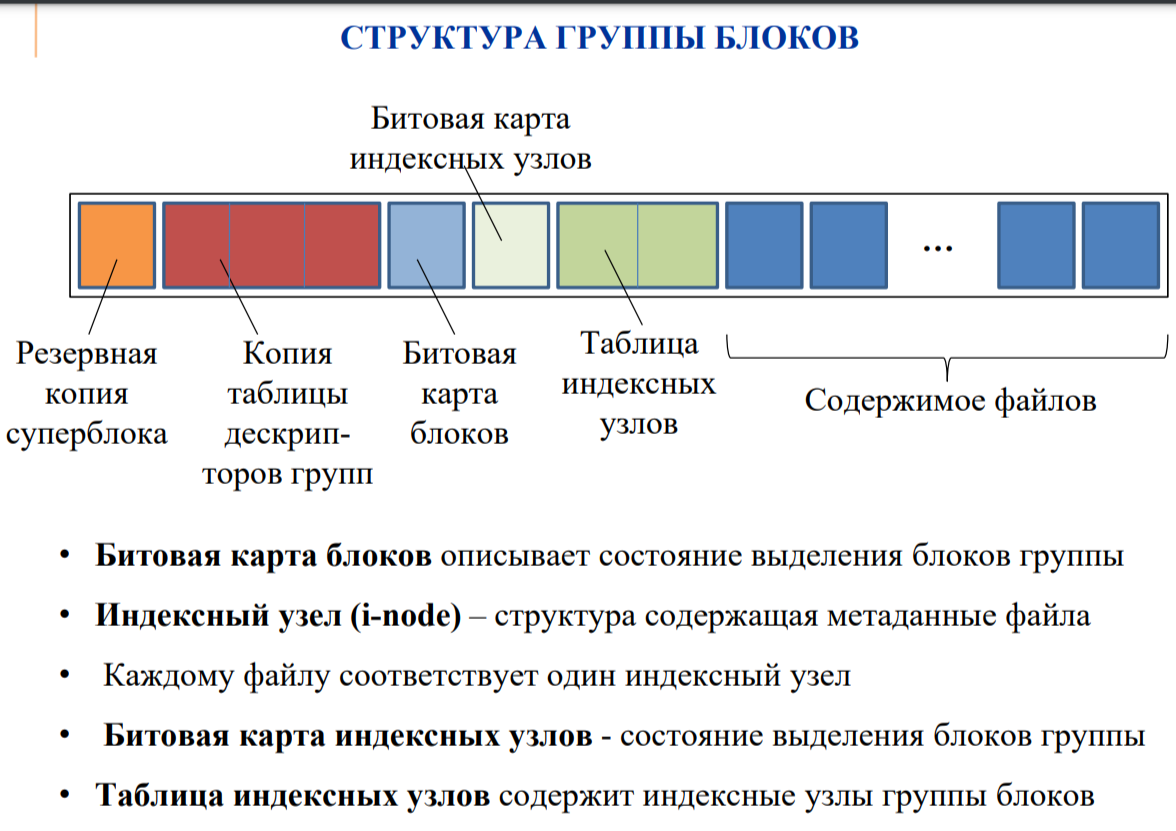
Главные плюсы: защита от повреждения данных, поддержка больших файлов и автоматическое восстановление.

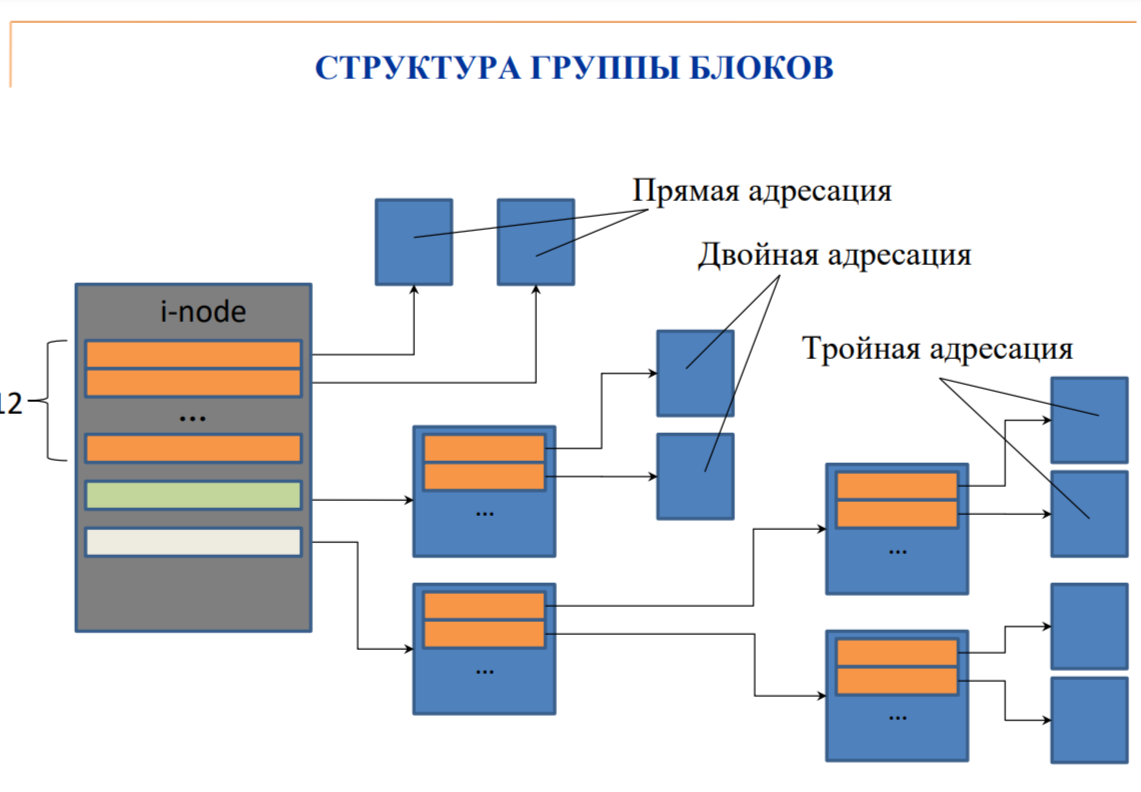
1. Характеристика и сравнение Ext2/Ext3/Ext4
2. Низкоуровневая структура тома Ext3



Суперблок — основной элемент файловой системы ext2. Он содержит общую информацию о файловой системе

1. Структура группы блоков





1. Структура i-node и методы адресации блоков

Inode

C каждым файлом в ОС Unix связана особая структура данных - индексный дескриптор (inode), хранящий метаинформацию файла (владелец, права доступа и т.п.).

Индексные дескрипторы в оригинальной UnixFS объединялись в последовательно нумерованный (индексированный) массив, что и дало название структуре. В современных ФС эта структура может иметь разные размеры и набор полей или отсутствовать вовсе. Соответственно, классические утилиты мониторинга ФС могут выдавать неверные данные о количестве занятых и свободных inode.

В реализации API доступа к ФС inode – это стандартизованная структура данных для обобщённого представления атрибутов файла. В оперативной памяти индексный дескриптор может быть представлен в виртуальном виде – vnode. Для ФС , не хранящих индексные дескрипторы, vnode создаётся на основе других подходящих источников данных.

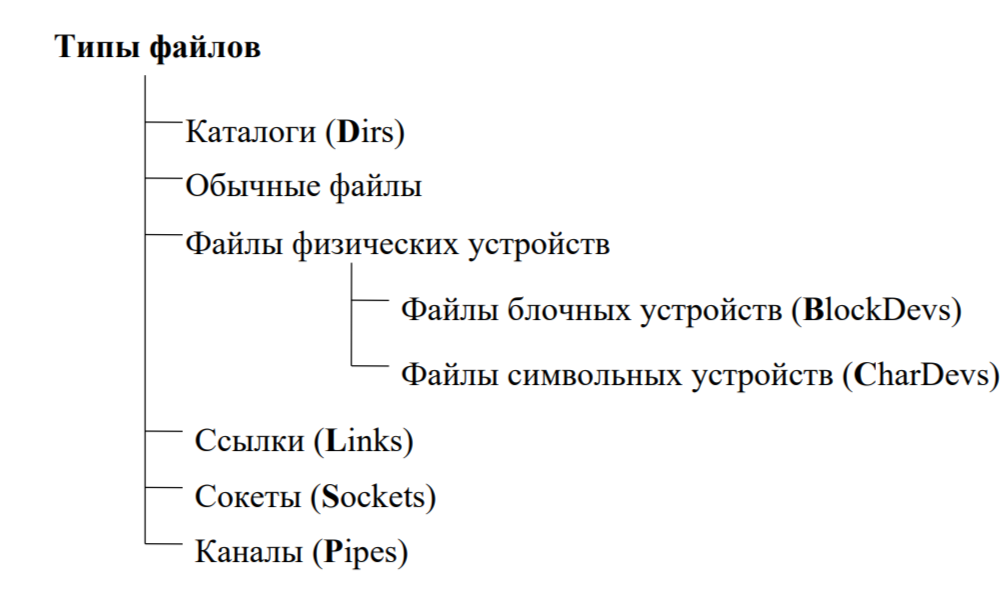
Номер индексного дескриптора уникален в рамках одной ФС, однако, при монтировании нескольких ФС в одно дерево номера индексных дескрипторов будут повторяться. Поэтому vnode хранит номер индексного дескриптора плюс идентификатор ФС, в которой он находится. Для дисковых ФС в Linux номером ФС является число, составленное из мажора и минора блочного устройства, на котором ФС расположена. Для NFS, похоже, номер ФС определяется порядком монтирования и последовательно возрастает начиная с 1Ah.

Блоки данных

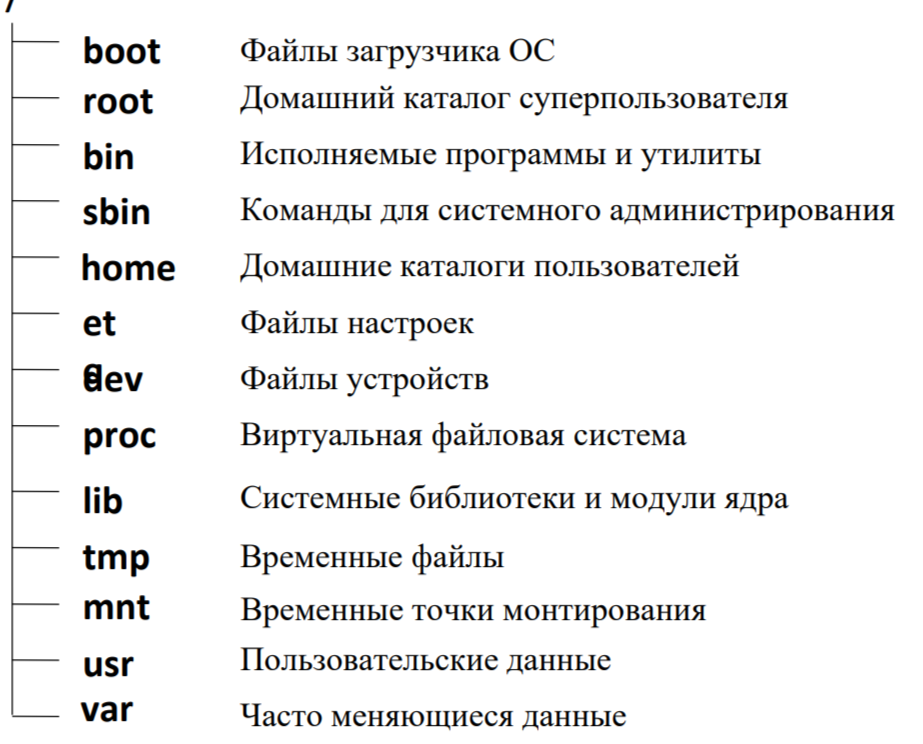
Размещение данных файла в блоках описывается ссылками, хранящимися в inode файла



1. Типы файлов Linux



1. Низкоуровневая структура файла каждого типа
2. Обзор основных системных каталогов и их назначение



1. Команды для работы с файловой системой
2. Основные приемы работы в командном интерпретаторе

CD – для смены текущего каталога

Ls – просмотр содержимого каталога

Touch - создание пустого файла

Ln – создание жестких и сивольных ссылок

Mkdir – создание директории

Cp – создание файлов и директории

Mv – перемещение файлов и директорий

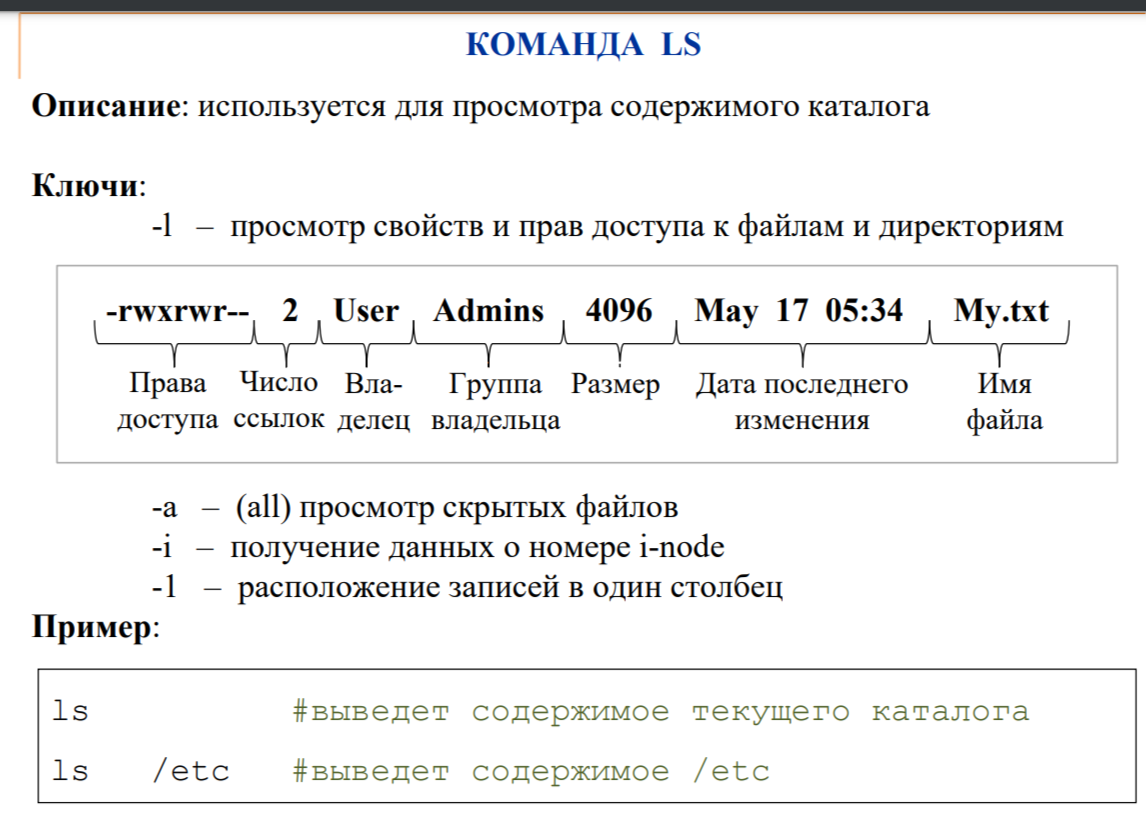
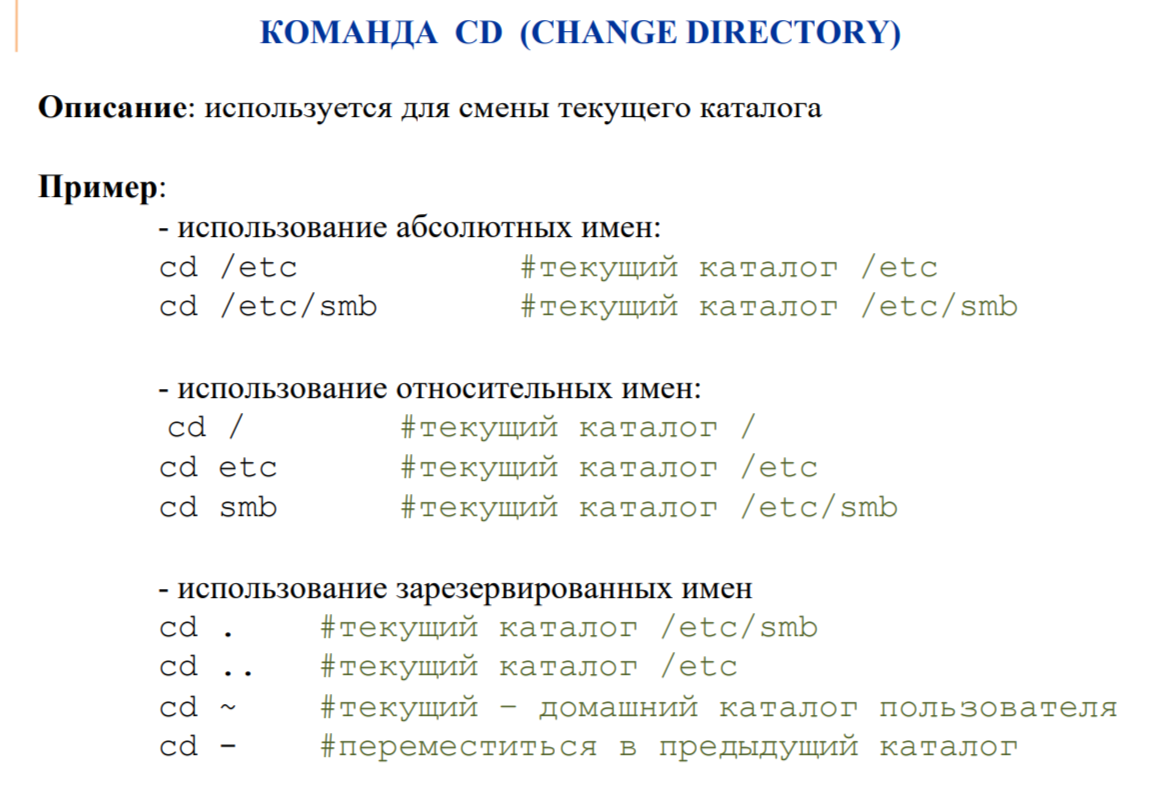
Rm – удаление файлов и директорий

Car – конкатенация файла и вывод в поток стандартного вывода

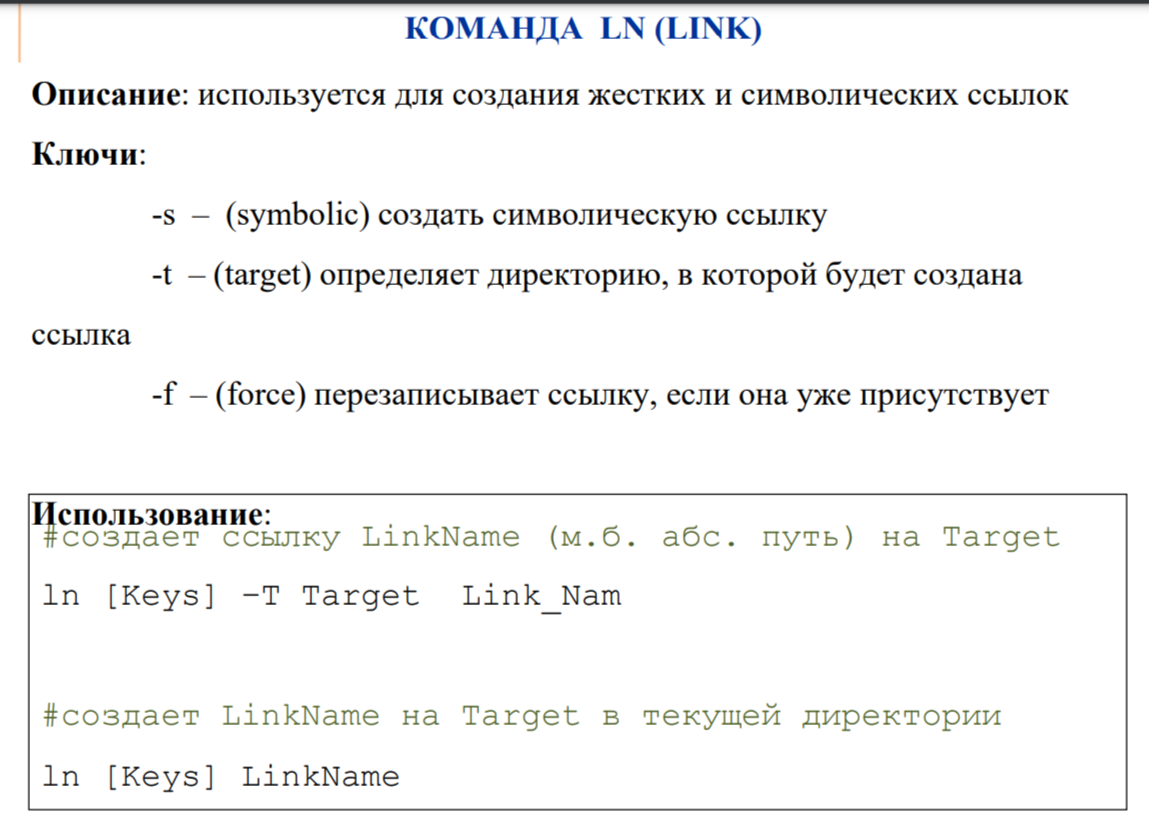
Dd – копирование и конвентирование файла

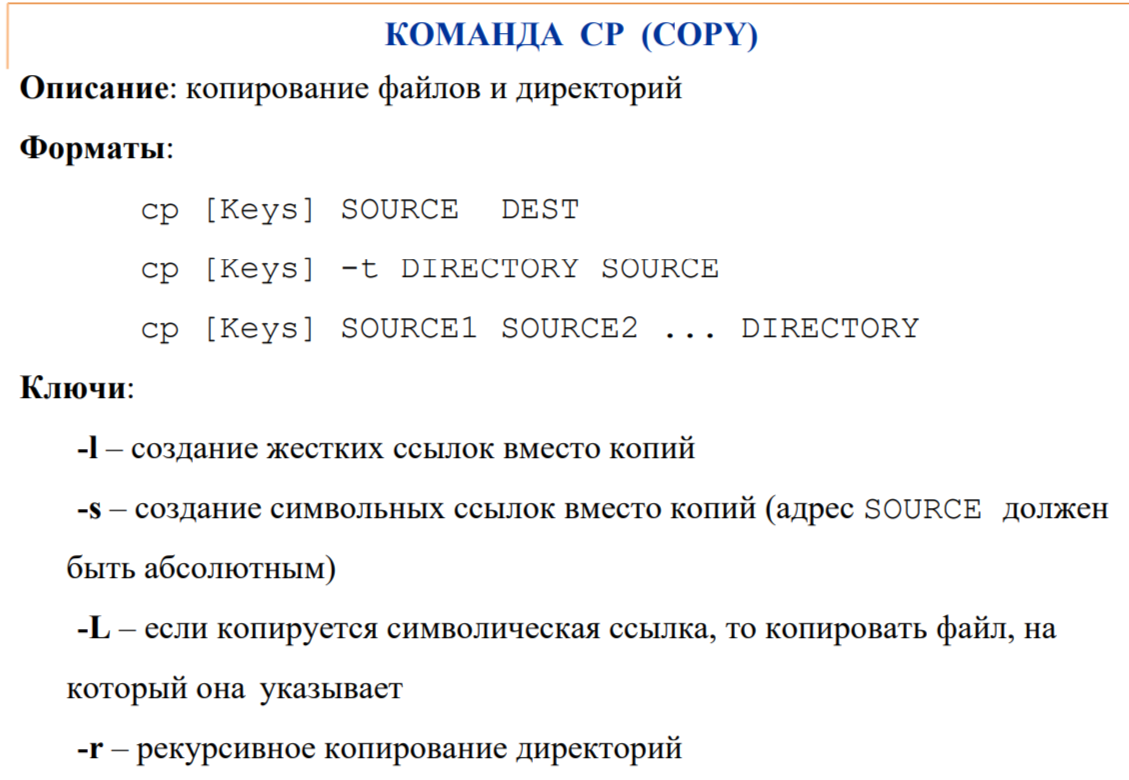
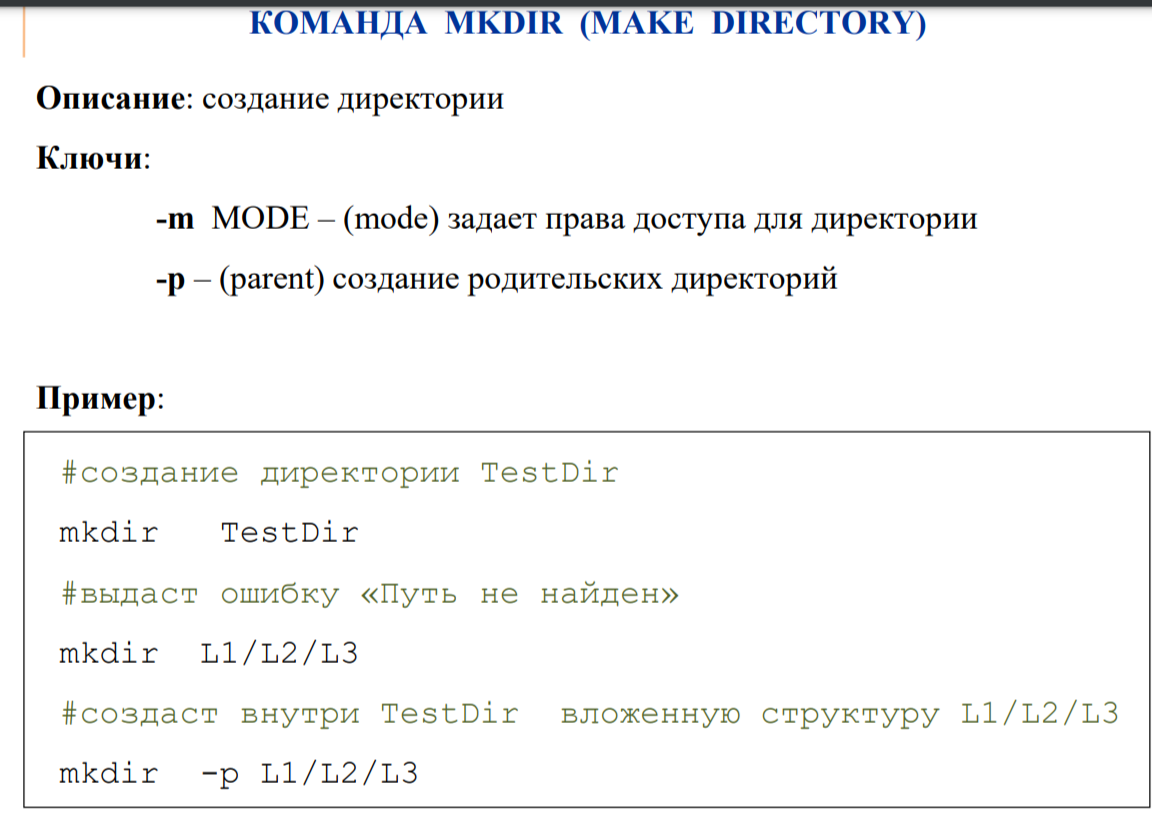
Mkfs – получение информации о файловой системе

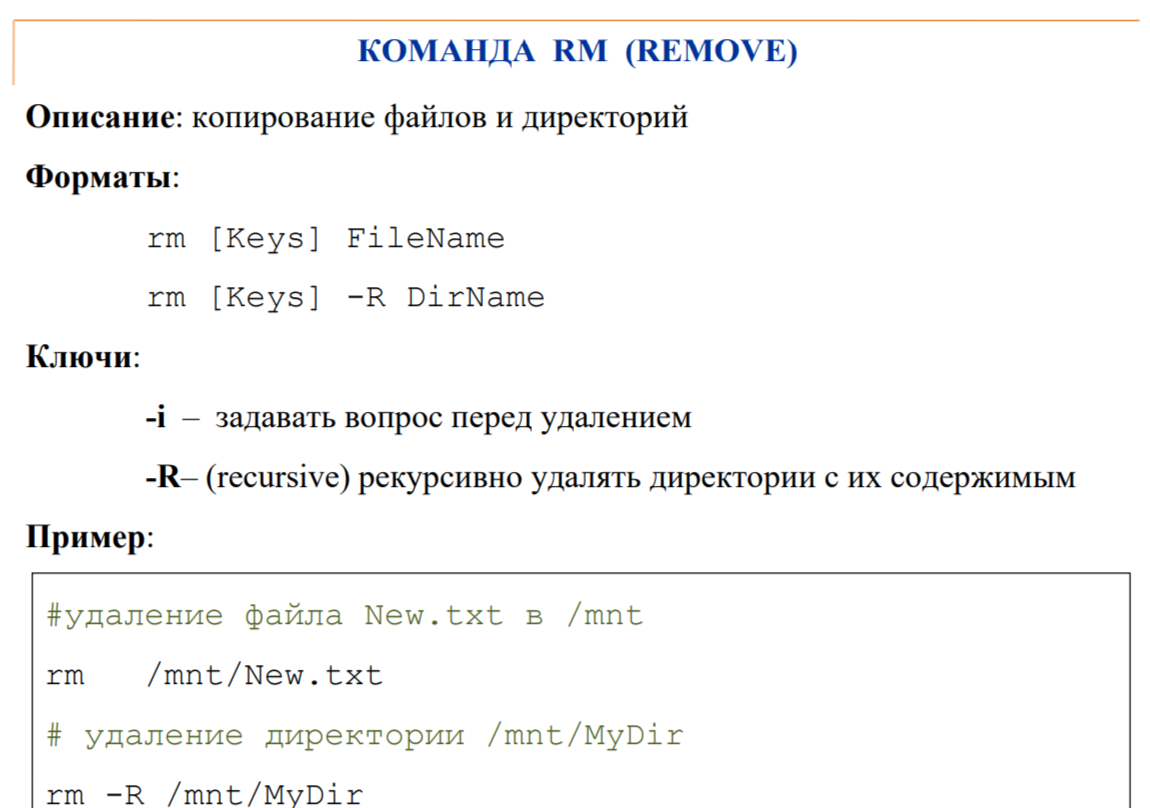
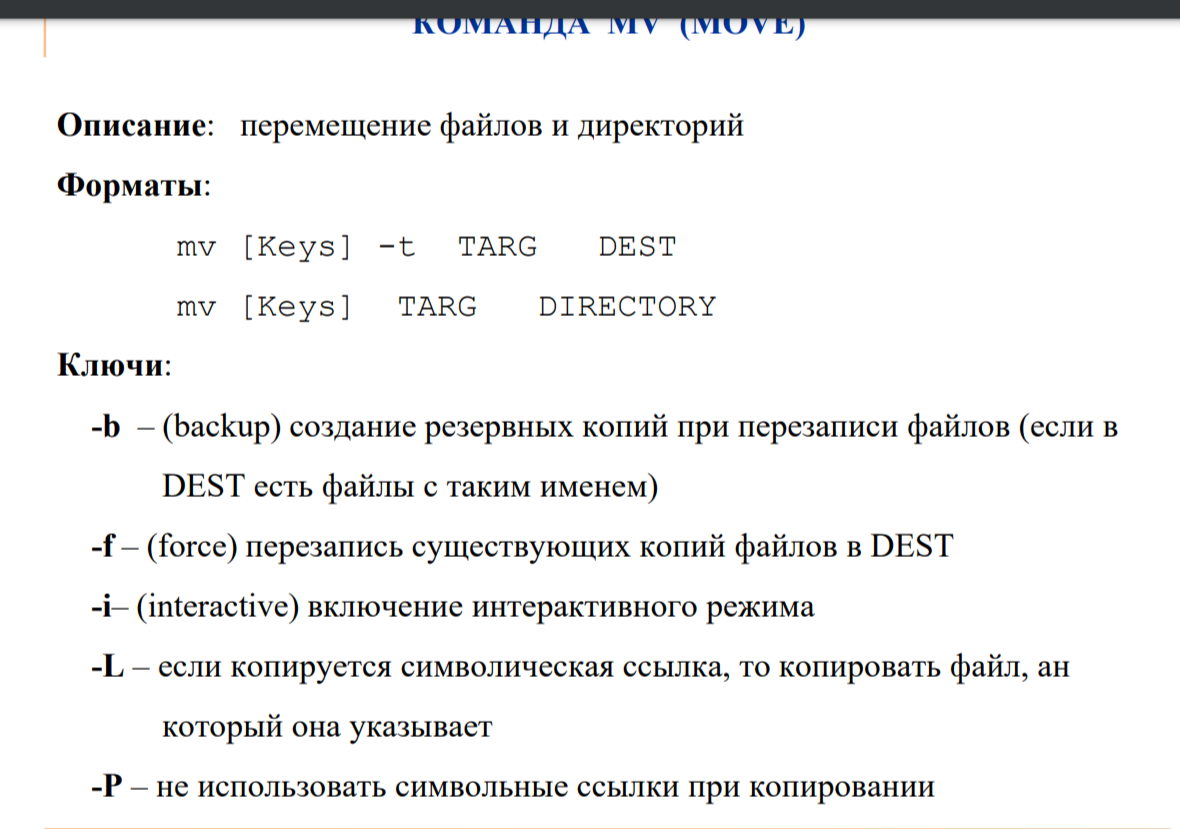
1. Команды обзора файловой структуры (cd, ls, pwd, whereis, find)

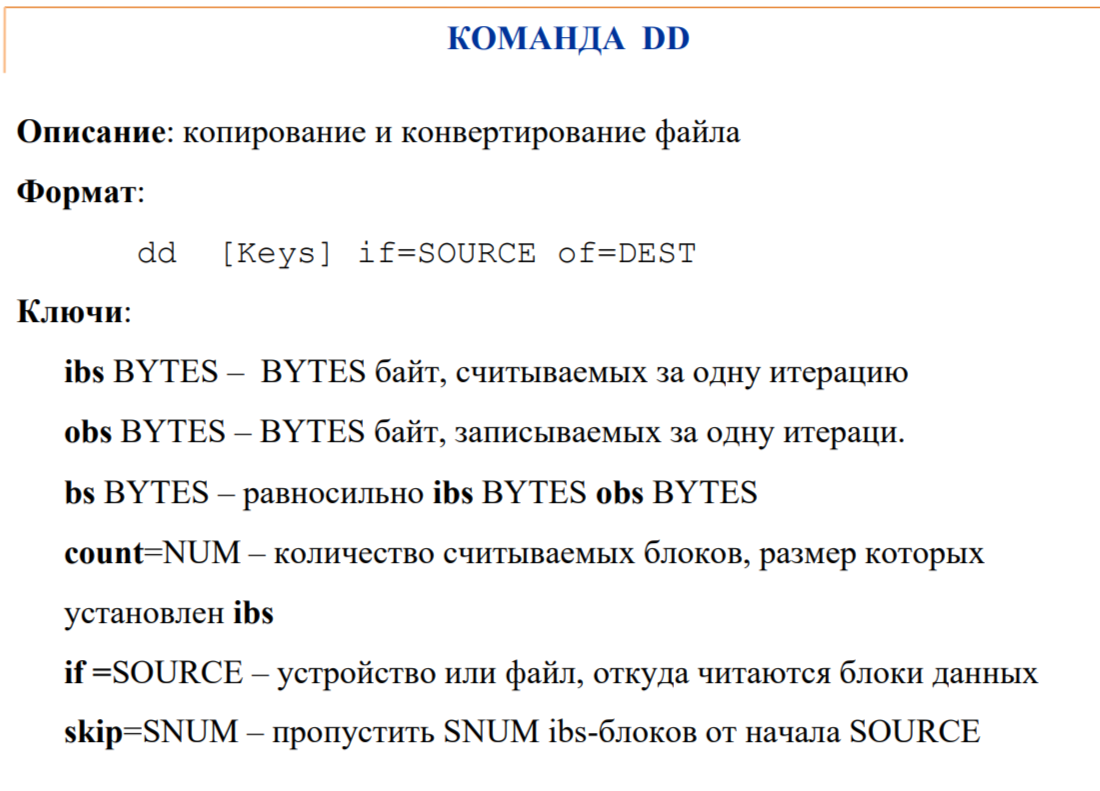
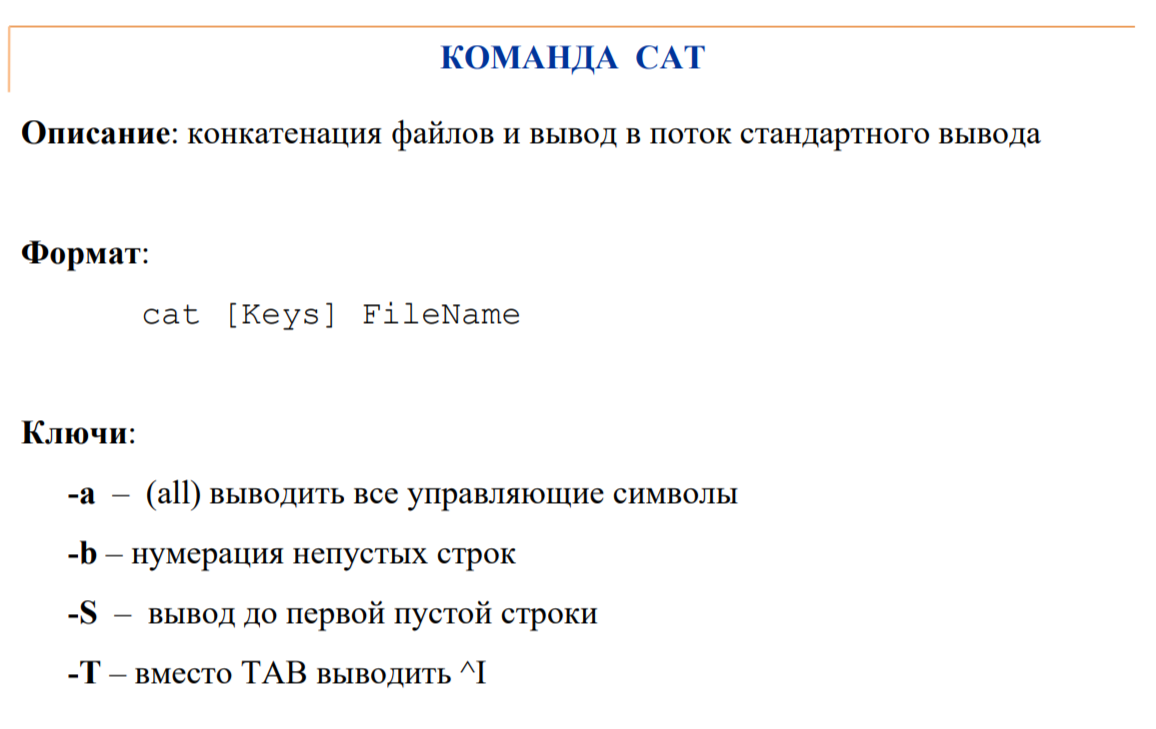


1. Команды для работы с файлами и директориями (touch, ln, mkdir, rmdir, cp, mv, rm, cat, dd)

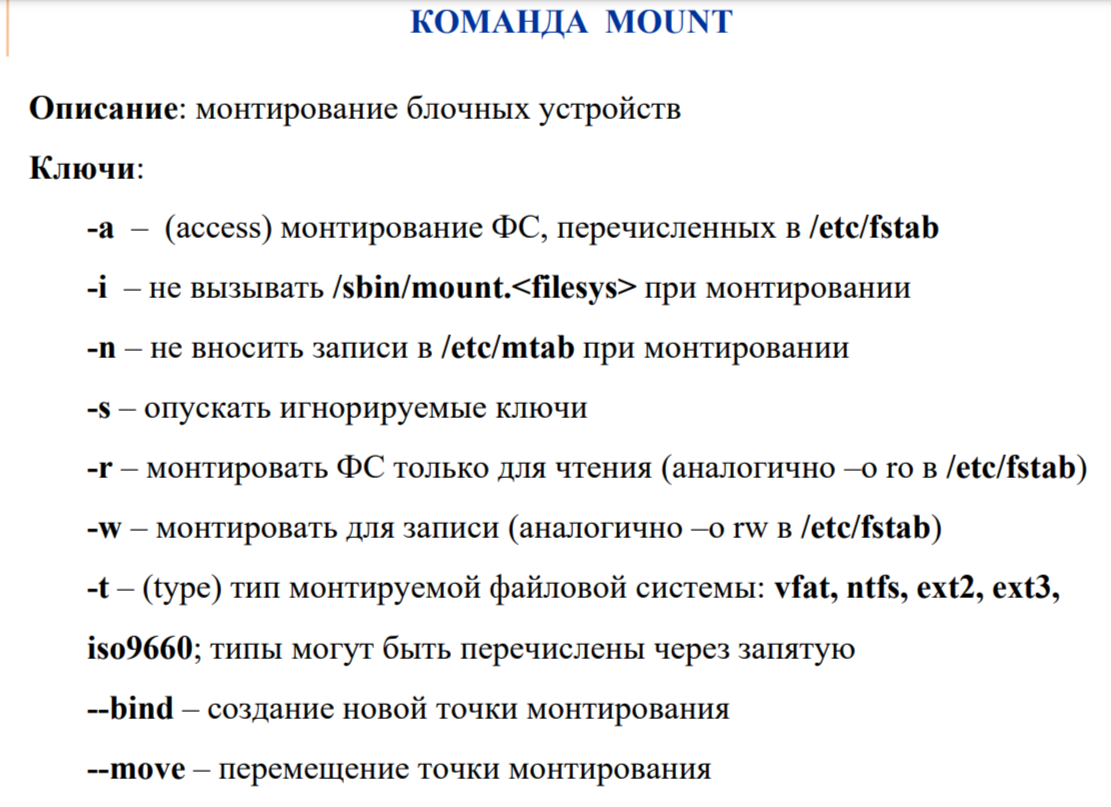


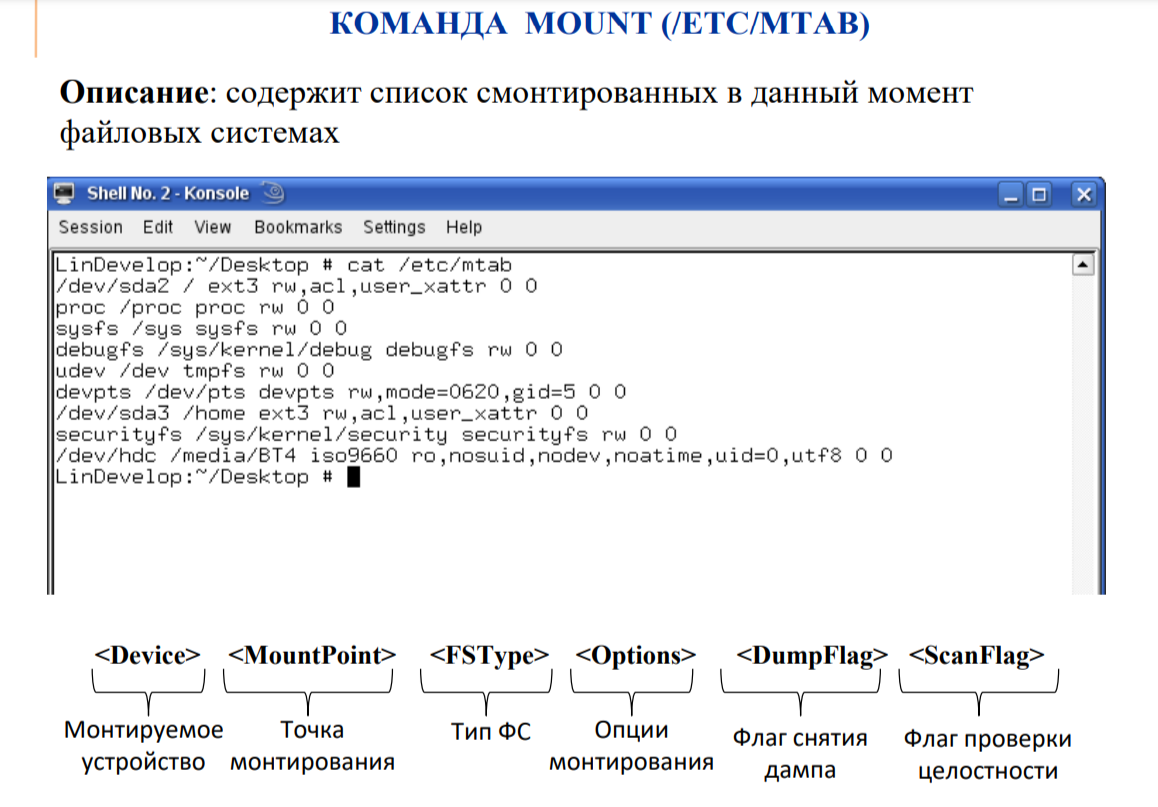
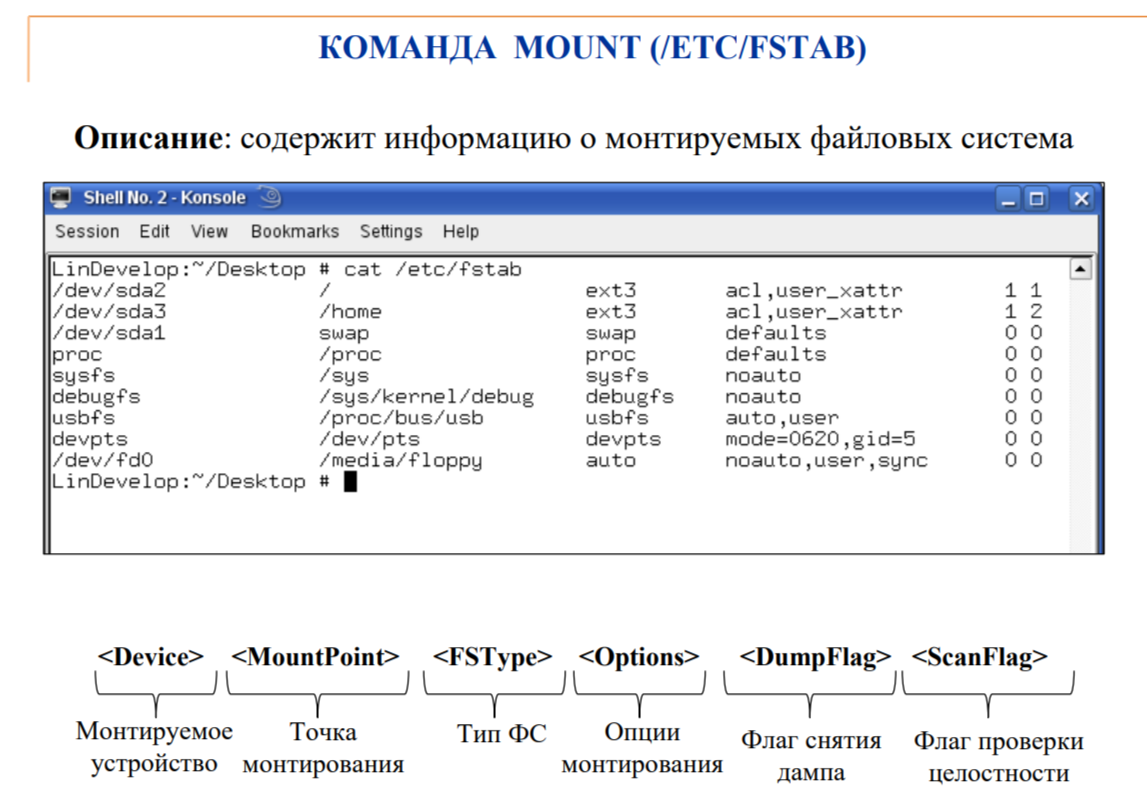




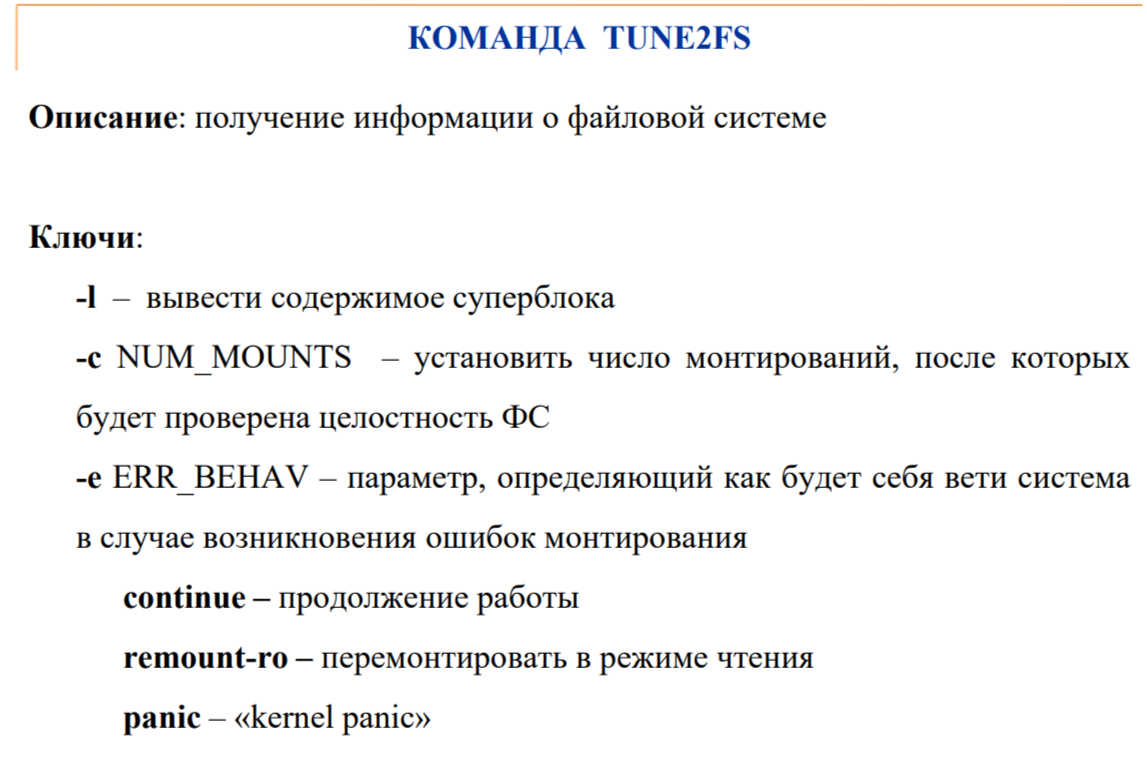
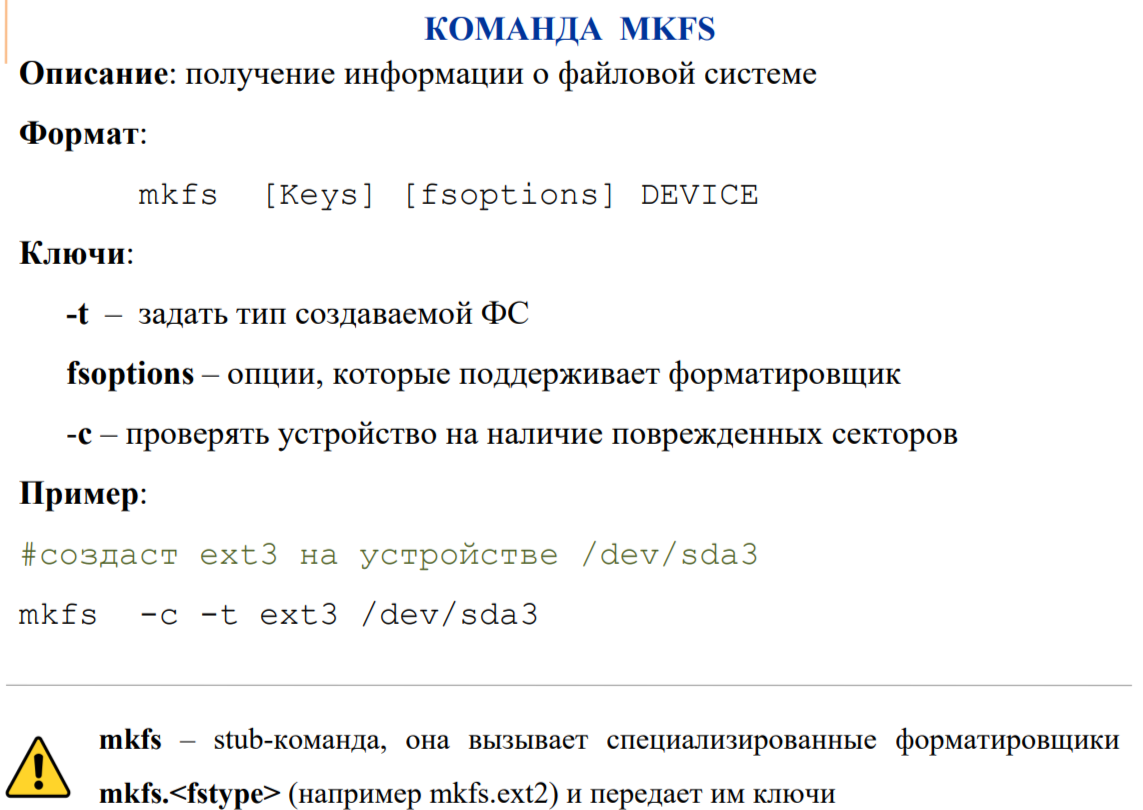


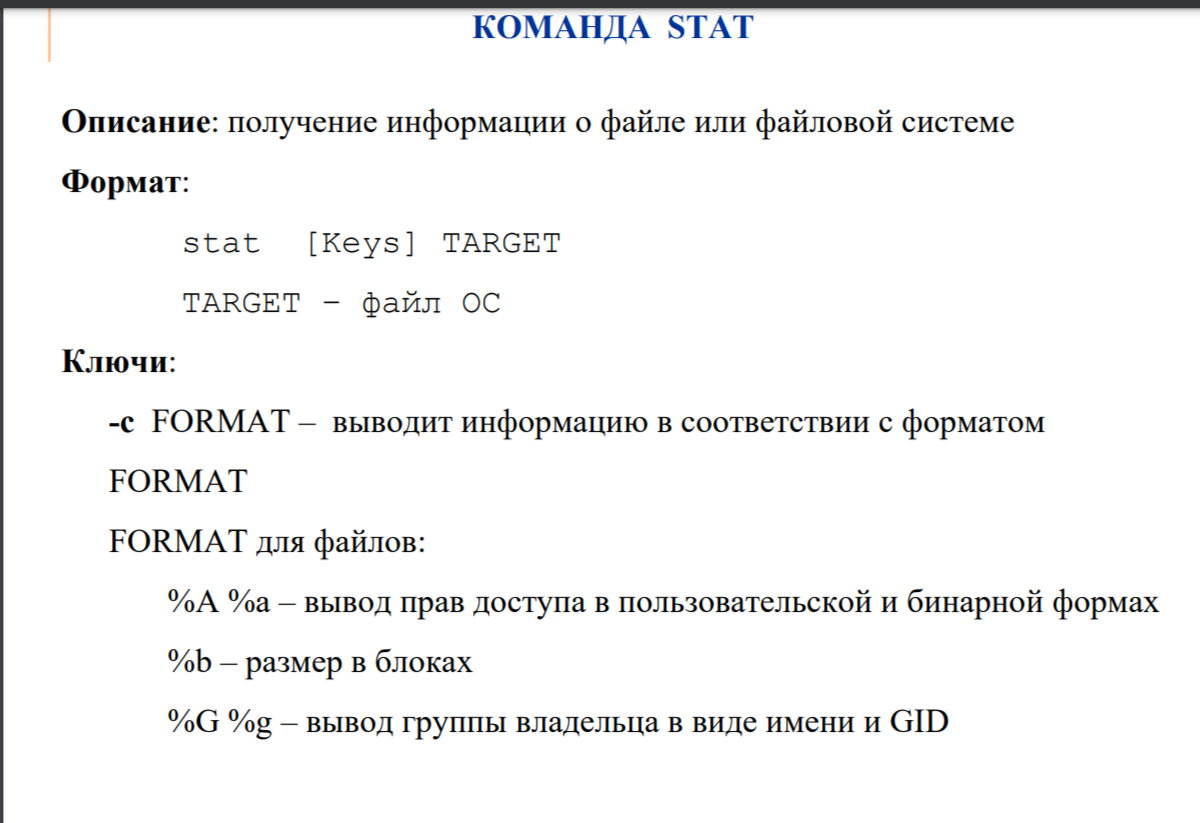
1. Монтирование ФС (mount, umount), механизм монтирования на уровне ФС, структура файлов монтирования (fstab, mtab, mounts, filesystems)



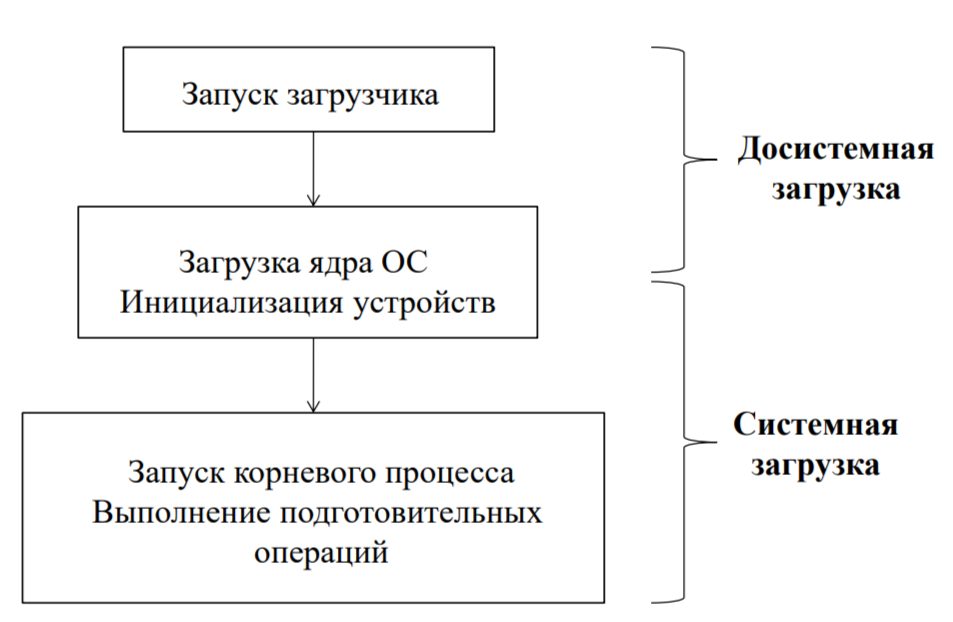


1. Команды работы с файловой системой (mkfs, tune2fs, stat)

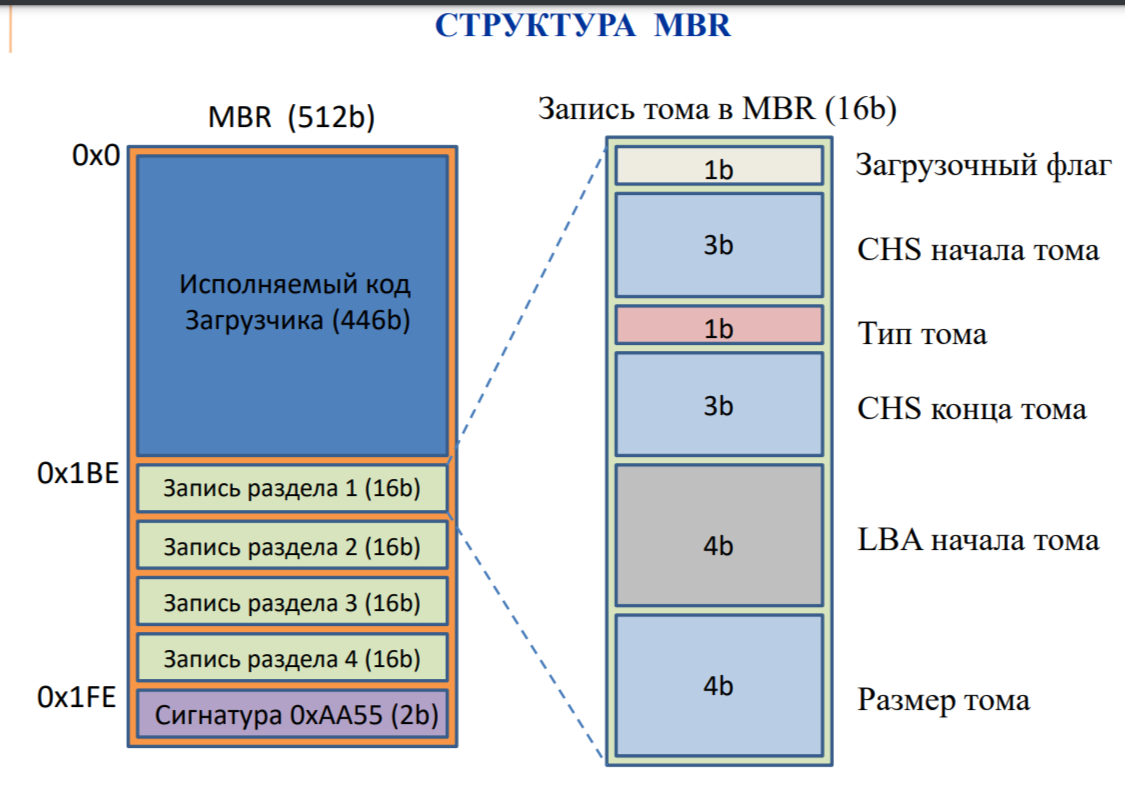




1. Загрузка Linux
2. Этапы загрузки ОС



1. Классификация томов и их расположение на HDD
2. Структура MBR

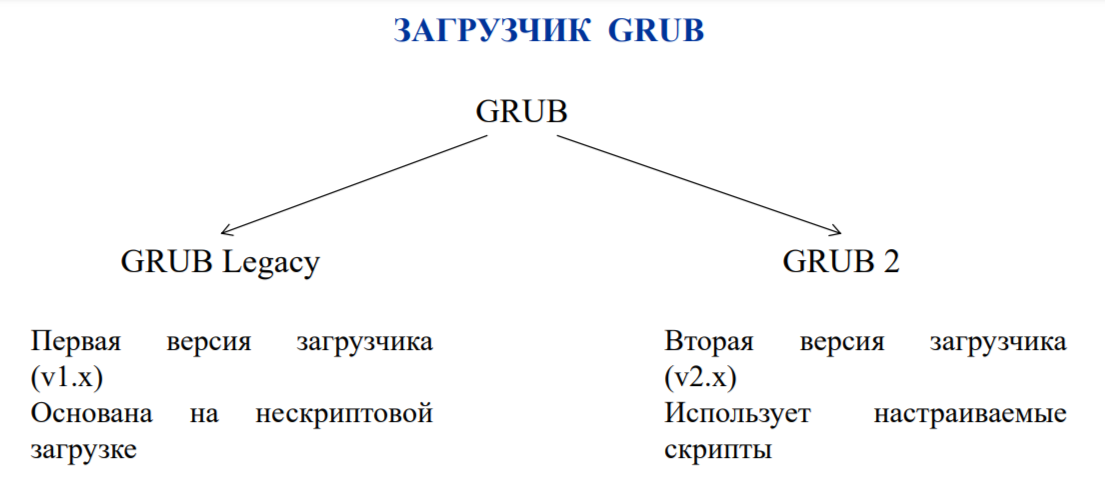


1. Загрузчик LILO: достоинства и недостатки, установка , конфигурационные файлы и их структура

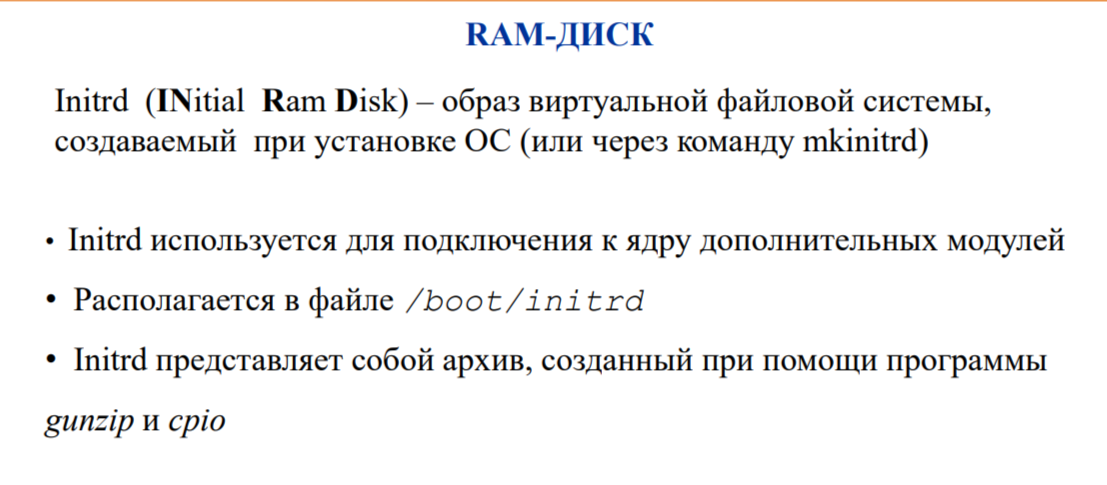


LILO (LInux LOader) был стандартным загрузчиком для многих дистрибутивов Linux до тех пор, пока не стал более популярным GRUB. LILO разрабатывался и поддерживался Вернером Альмесбергером (Werner Almesberger) с 1992 по 1998 год, Джоном Коффманом (John Coffman) с 1999 по 2007 год, и Иоахимом Вьедорном (Joachim Wiedorn) с 2010 года. Проект до сих пор активен. Преимуществом LILO над многими другими загрузчиками является отсутствие требований к типу файловой системы. Это значит, что операционная система может использовать любую файловую систему (NTFS, EXT4, BTRFS, FAT32, HFS+, UFS, JFS и т.д.), и она все равно будет загружена. LILO также может загружать ядро Linux с гибких дисков и внешних жестких дисков, и способен работать с шестнадцатью установленными на компьютер операционными системами. Их количество можно увеличить, используя загрузку по цепочке. LILO может быть установлен в загрузочный сектор раздела или в Master Boot Record (MBR).

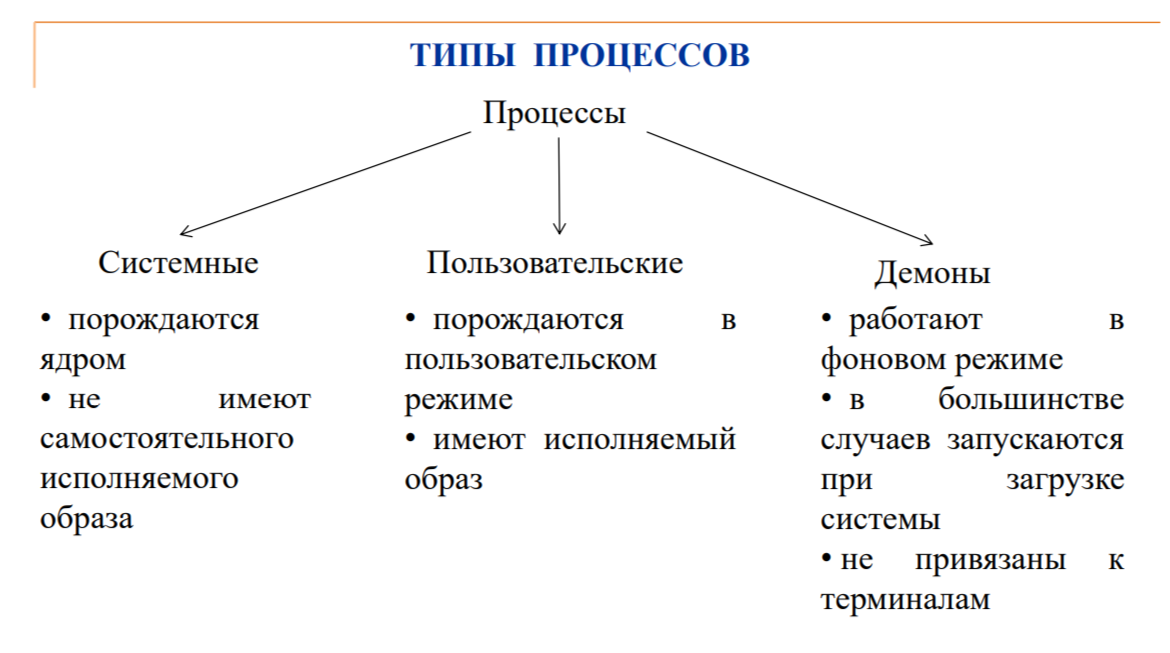
1. Загрузчик GRUB: достоинства и недостатки, установка, конфигурационные файлы и их структура



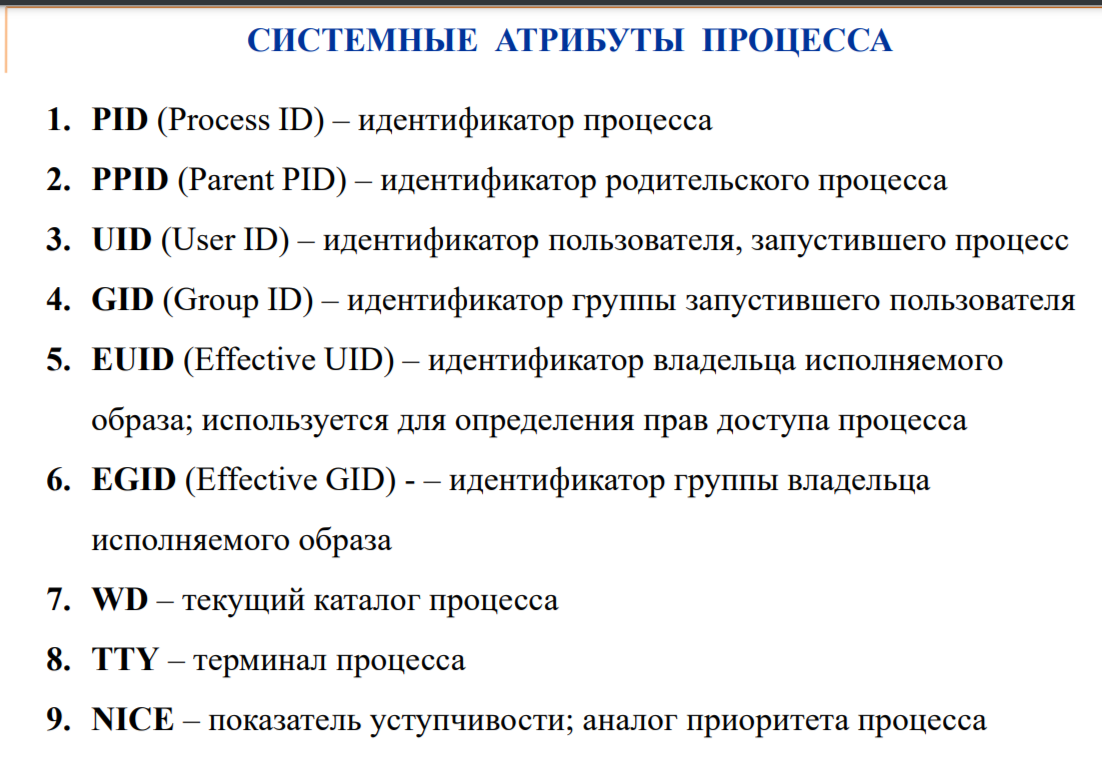
1. RAM-диск: назначение, структура



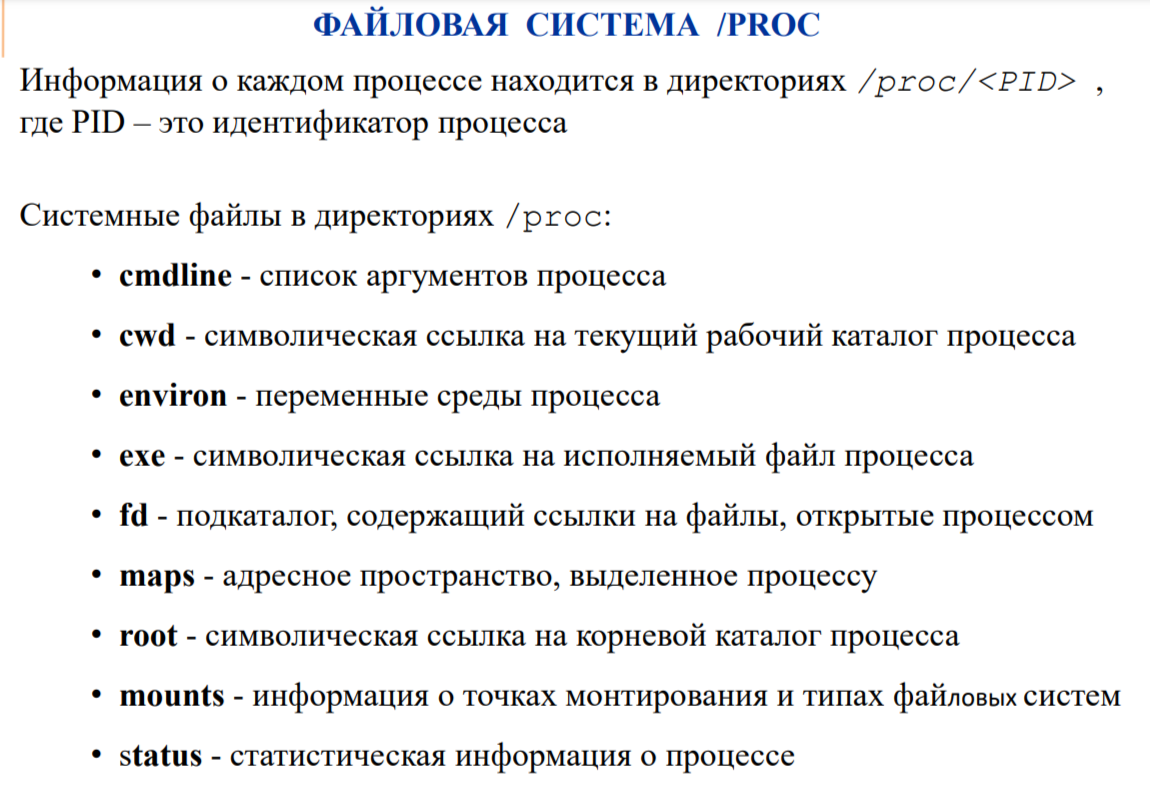
1. Запуск корневого процесса: конфигурационный файл, скрипты второго уровня
2. Процессы Linux
3. Виды процессов и их особенности



1. Механизм ветвления процессов
2. Системные атрибуты процесса и их назначение



1. Получение информации о процессе через виртуальную ФС /proc



1. Команды для работы с процессами (ps, top, kill, lsof, fstat, pstree)

4 лекция

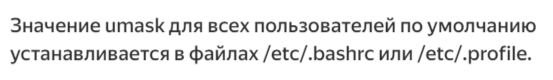
1. Разграничение доступа
2. Элементы системы разграничения доступа в Linux

Механизм разграничения доступа в Linux базируется на именах пользователей и названиях групп пользователей, состав которых определяет суперпользователь (root).

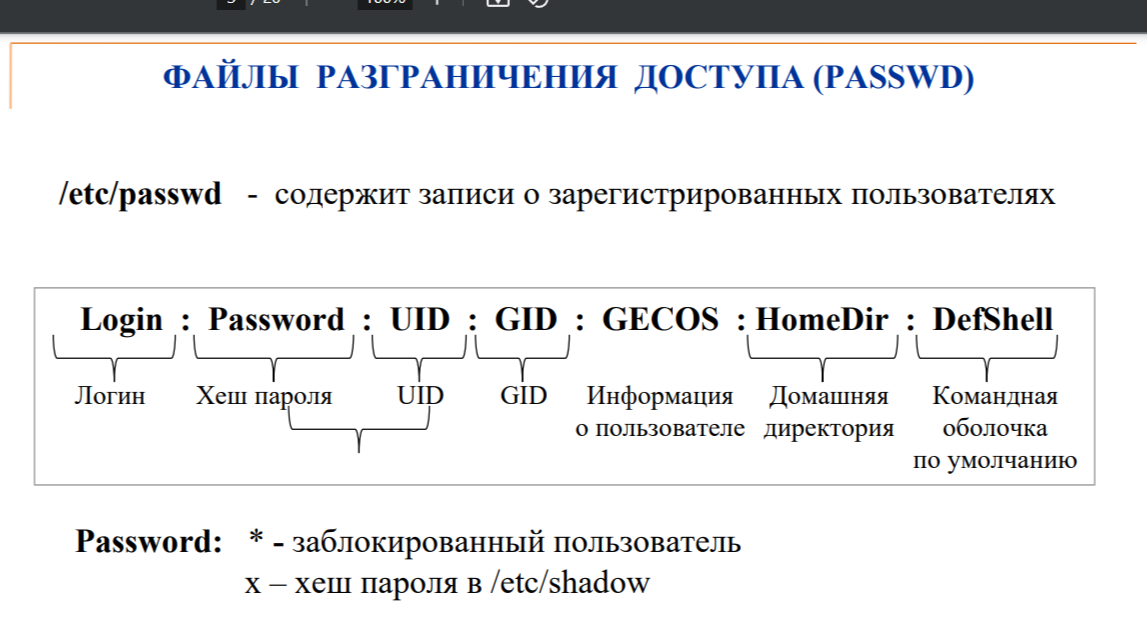
Пользователь может входить в одну или несколько групп и иметь права доступа в зависимости от группы, в которую он входит.

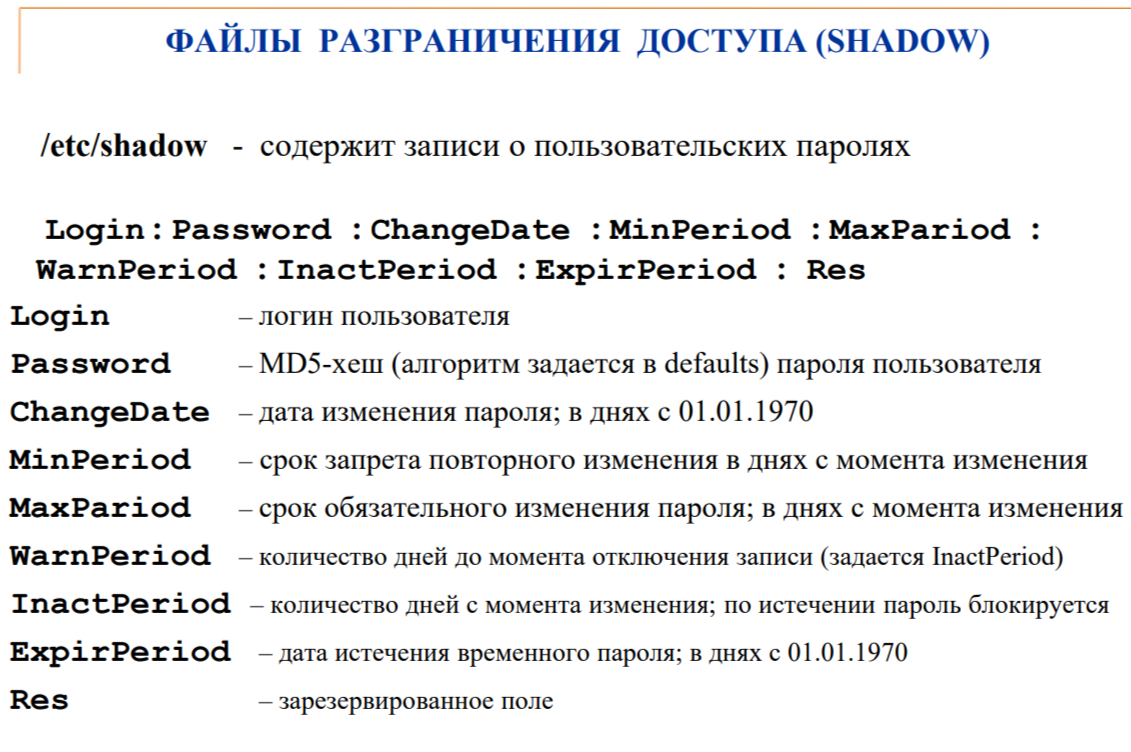
При создании файла у него появляется владелец, то есть пользователь, который запустил процесс его создания. Также определяется группа, которая будет иметь права на этот файл.

1. Маски прав доступа к файлам и их расположение

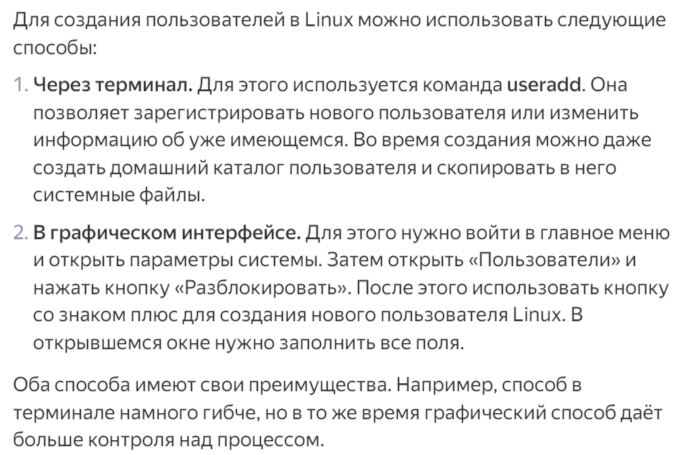


1. Пользователи Linux





1. Командное и ручное создание пользователей



1. Конфигурационные файлы системы разграничения прав
2. Создание и редактирование записей о пользователях и группах вручную

Получение информации о пользователях

1. Список пользователей можно посмотреть в файле **/etc/passwd** командой:

less /etc/passwd

Данная команда вернет всех пользователей, в том числе служебных. Для отображения только учетных записей пользователей можно ввести команду:

users

2. Проверить существование пользователя и увидеть его идентификатор можно с помощью команды id:

id username

Мы должны увидеть что-то на подобие:

uid=1001(username) gid=1001(username) groups=1001(username),27(sudo)

*\* учетная запись****username****существует, она находится в основной группе****username****и дополнительной****sudo****. Ее идентификатор****1001****.*

3. Также мы можем получить строку о пользователе из файла **/etc/passwd**:

getent passwd username

Мы увидим что-то такое:

username:x:1001:1001::/home/username:/bin/sh

*\* наш пользовател username имеют идентификатор пользователя и группы****1001****, его домашняя директория****/home/username****, командная оболочка****/bin/sh****.*

Создание

Синтаксис:

useradd <имя пользователя> [опции]

*\* опции не являются обязательными при создании пользователя.*

Пример:

useradd dmosk

*\* в данном примере создается учетная запись****dmosk****.*

Для учетной записи стоит сразу создать пароль:

passwd dmosk

*\* после ввода, система попросит ввести пароль дважды.*

Редактирование

Синтаксис:

usermod <имя пользователя> [опции]

Пример:

usermod dmosk -G wheel

*\* эта команда добавит пользователя dmosk в группу****wheel***

Удаление

Синтаксис:

userdel <имя пользователя> [опции]

Пример:

userdel dmosk

Блокировка

1. Можно заблокировать пользователя, не удаляя его из системы:

usermod -L <имя пользователя>

Пример:

usermod -L dmosk

Чтобы разблокировать пользователя, вводим:

usermod -U <имя пользователя>

2. В некоторых системах Linux может использоваться pam-модуль tally. Он осуществляет подсчет количества неудачных попыток входа в систему. Также он может блокировать доступ при превышении данного количества.

Чтобы посмотреть счетчик для пользователя, вводим команду:

pam\_tally2 --user=admin

В версия постарея:

pam\_tally --user=admin

Для сброса счетчика (и блокировки, если она есть) вводим:

pam\_tally2 --user=admin --reset

Работа с группами

1. Добавление группы:

groupadd <группа> [опции]

2. Редактирование:

groupmod <группа> [опции]

3. Удаление группы:

groupdel <группа> [опции]

4. Добавление пользователя в группу:

Выполняется через команду usermod:

usermod -a -G <группы через запятую> <пользователь>

5. Удаление из группы:

Выполняется с помощью gpasswd:

gpasswd --delete <пользователь> <группы через запятую>

1. Команды для работы с системой разграничения доступа (useradd, userdel, usermod, passwd, chown, chgrp, chmod, su, sudo, login)

Лекция 6

1. Сборка и установка программ ч.1
2. Варианты установки программ
3. Бинарные пакетные файлы и их структура
4. Определение программных зависимостей
5. Команды для работы с бинарными пакетами
6. Сборка и установка программ ч.2
7. Сборка и установка программ из исходных текстов
8. Назначение и структура сборочных
9. Архиваторы и компрессоры
10. Скрипты автозагрузки и их структура
11. Команды сборки и установки ПО
12. Язык командного интерпретатора Shell ч.1
13. Служебные символы
14. Переменные и параметры
15. Внутренние переменные
16. Конструкции проверки условий
17. Язык командного интерпретатора Shell ч.2
18. Косвенные ссылки
19. Циклы и ветвления
20. Потоки ввода/вывода и их перенаправление
21. Массивы
22. Устройство и конфигурирование ядра Linux
23. Разновидности ядер
24. Модульная архитектура ядра
25. Динамические параметры ядра
26. Параметры смонтированных ФС
27. Параметры планировщика
28. Параметры RAM-диска