Министерство Образования, Культуры и Исследований

Молдавский Государственный Университет

Факультет Математики и Информатики

Департамент Информатики

Операционные системы

**Индивидуальная работа**

Тема:

**«Администрирование файловой системы в операционной системе UNIX»**

Проверила:

Eleonora Seiciuc

Выполнил:

студент группы IA2302

Михайлов Пётр

Кишинев, 2022

**ОГЛАВЛЕНИЕ**

[Два кита UNIX: файлы и процессы 3](#_Toc102990276)

[Особенности файловой системы 4](#_Toc102990277)

[Специальный файл устройства 6](#_Toc102990278)

[Именованный канал 6](#_Toc102990279)

[Символическая ссылка 6](#_Toc102990280)

[Сокет 7](#_Toc102990281)

[Дерево каталогов 7](#_Toc102990282)

[Корневой каталог 9](#_Toc102990283)

[/bin 9](#_Toc102990284)

[/boot 9](#_Toc102990285)

[/dev 10](#_Toc102990286)

[/etc 10](#_Toc102990287)

[/home (необязательно) 10](#_Toc102990288)

[/lib 10](#_Toc102990289)

[/mnt 10](#_Toc102990290)

[/root (необязательно) 10](#_Toc102990291)

[/sbin 11](#_Toc102990292)

[/usr 11](#_Toc102990293)

[/var, /tmp 11](#_Toc102990294)

[РЕЗЮМЕ 11](#_Toc102990295)

[БИБЛИОГРАФИЯ 12](#_Toc102990296)

Несмотря на многообразие версий UNIX, основой всего семейства являются принципиально одинаковая архитектура и ряд стандартных интерфейсов (в UNIX стандартизовано почти всё — от расположения системных папок и файлов, до интерфейса системных вызовов и списка драйверов базовых устройств). Опытный администратор без особого труда сможет обслуживать другую версию, тогда как для пользователей переход на другую систему и вовсе может оказаться незаметным. Для системных же программистов такого рода стандарты позволяют полностью сосредоточиться на программировании, не тратя время на изучение архитектуры и особенностей конкретной реализации системы.

В системе UNIX может одновременно выполняться множество процессов (задач), причем их число логически не ограничивается, и множество частей одной программы может одновременно находиться в системе. Благодаря специальному механизму управления памятью, каждый процесс развивается в своем защищенном адресном пространстве, что гарантирует безопасность и независимость от других процессов. Различные системные операции позволяют процессам порождать новые процессы, завершают процессы, синхронизируют выполнение этапов процесса и управляют реакцией на наступление различных событий.

## Два кита UNIX: файлы и процессы

Существует два основных объекта операционной системы UNIX, с которыми приходиться работать пользователю — файлы и процессы. Эти объекты сильно связаны друг с другом, и в целом организация работы с ними как раз и определяет архитектуру операционной системы.

Все данные пользователя хранятся в файлах; доступ к периферийным устройствам осуществляется посредством чтения и записи специальных файлов; во время выполнения программы, операционная система считывает исполняемый код из файла в память и передает ему управление.

С другой стороны, вся функциональность операционная определяется выполнением соответствующих процессов. В частности, обращение к файлам на диске невозможно, если файловая подсистема операционной системы (совокупность процессов, осуществляющих доступ к файлам) не имеет необходимого для этого кода в памяти.

Термин файловая система по историческим причинам обозначает одновременно и иерархию каталогов и файлов, и часть ядра, управляющую доступом к каталогам и файлам.

## Особенности файловой системы

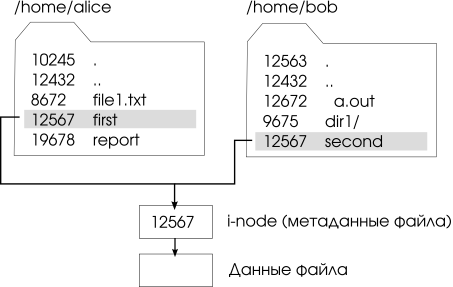
Первое значение термина упирается в рассмотрение структур, в которые могут быть организованы файлы на носителях данных. Существует несколько видов таких структур: линейные, древовидные, объектные и другие, но в настоящее время широко распространены только древовидные структуры.

Каждый файл в древовидной структуре расположен в определенном хранилище файлов – каталоге, каждый каталог, в свою очередь, также расположен в некотором каталоге. По принципу вложения элементов файловой системы (файлов и каталогов) друг в друга строится дерево, вершинами которого являются непустые каталоги, а листьями – файлы или пустые каталоги. Корень такого дерева имеет название корневой каталог и обозначается каким-либо специальным символом или группой символов (например, «C:» в операционной системе Windows). Каждому файлу соответствует некоторое имя, определяющее его расположение в дереве файловой системы. Полное имя файла состоит из имен всех вершин дерева файловой системы, через которые можно пройти от корня до данного файла (каталога), записывая их слева-направо и разделяя специальными символами-разделителями.

В настоящее время существует огромное количество файловых систем, каждая из которых используется для определенной цели: для быстрого доступа к данным, для обеспечения целостности данных при сбоях системы, для простоты реализации, для компактного хранения данных, и т.д. Однако среди всего множества файловых систем можно выделить такие, которые обладают рядом схожих признаков, а именно:

Файлы и каталоги идентифицируются не по именам, а по индексным узлам (i-node) – индексам в общем массиве файлов для данной файловой системы. В этом массиве хранится информация об используемых блоках данных на носителе, а также – длина файла, владелец файла, права доступа и другая служебная информация под общим названием «метаданные о файле». Логические же связки типа «имя–i-node» – есть не что иное, как содержимое каталогов.

Таким образом, каждый файл характеризуется одним i-node, но может быть связан с несколькими именами – в UNIX это называют жёсткими ссылками. При этом, удаление файла происходит тогда, когда удаляется последняя жёсткая ссылка на этот файл.



**Скриншот 1.** Пример жесткой ссылки

Важной особенностью таких файловых систем является то, что имена файлов зависят от регистра, другими словами файлы test.txt и TEST.txt отличаются (т.е. являются разными строками в файле директории).

В определенных (фиксированных для данной файловой системы) блоках физического носителя данных находится т.н. суперблок. Суперблок – это наиболее ответственная область файловой системы, содержащая информацию для работы файловой системы в целом, а также – для ее идентификации. В суперблоке находится идентификатор файловой системы, отличающий её от других файловых систем, список свободных блоков, список свободных i-node'ов и некоторая другая служебная информация.

Помимо каталогов и обычных файлов для хранения информации, Файловая Система может содержать следующие виды файлов:

### Специальный файл устройства

Обеспечивает доступ к физическому устройству. При создании такого устройства указывается тип устройства (блочное или символьное), старший номер – индекс драйвера в таблице драйверов операционной системы и младший номер – параметр, передаваемый драйверу, поддерживающему несколько устройств, для уточнения о каком «подустройстве» идет речь (например, о каком из нескольких IDE-устройств или COM-портов).

### Именованный канал

Используется для передачи данных между процессами, работает по принципу двунаправленной очереди (FIFO). Является одним из способов обмена между изолированными процессами.

### Символическая ссылка

Особый тип файла, содержимое которого – не данные, а имя какого-либо другого файла. Для пользователя такой файл неотличим от того, на который он ссылается.

Символическая ссылка имеет ряд преимуществ по сравнению с жёсткой ссылкой: она может использоваться для связи файлов в разных файловых системах (ведь номера индексных узлов уникальны только в рамках одной файловой системы), а также более прозрачно удаление файлов – ссылка может удаляться совершенно независимо от основного файла.

### Сокет

Предназначен для взаимодействия между процессами через специальное API, схожее с TCP/IP-сокетами.

Такие файловые системы наследуют особенности оригинального UNIX. К ним можно отнести, например: s5 (используемая в версиях UNIX System V), ufs (BSD UNIX), ext2, ext3, reiserfs (Linux), qnxfs (QNX). Все эти файловые системы различаются форматами внутренних структур, но совместимы с точки зрения основных концепций.

## Дерево каталогов

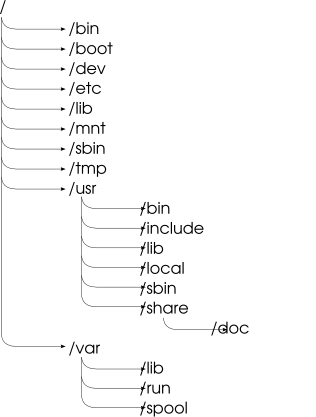
Рассмотрение второго значения термина ФС приводит нас к уже обозначенной ранее совокупности процедур, осуществляющих доступ к файлам на различных носителях. Особенностью операционных систем семейства UNIX является существование единого дерева файловой системы для любого количества носителей данных с одинаковыми или разными типами файловых систем на них. Это достигается путем монтирования – временной подстановкой вместо каталога одной файловой системы дерева другой файловой системы, вследствие чего система имеет не несколько деревьев никак не связанных друг с другом, а одно большое разветвленное дерево с единым корневым каталогом.

Файловая подсистема операционной системы UNIX имеет уникальную систему обработки запросов к файлам – переключатель файловых систем или виртуальная файловая система (VFS). VFS предоставляет пользователю стандартный набор функций (интерфейс) для работы с файлами, вне зависимости от места их расположения и принадлежности к разным файловым системам.

В мире стандартов UNIX определено, что корневой каталог единого дерева файловой системы должен иметь имя /, как и символ-разделитель при формировании полного имени файла. Тогда полное имя файла может быть, например, /usr/share/doc/bzip2/README. Задача VFS – по полному имени файла найти его местоположение в дереве файловой системы, определить её тип в этом месте дерева и «переключить», т.е. передать файл на дальнейшую обработку драйверу конкретной файловой системы. Такой подход позволяет использовать практически неограниченное количество различных файловых систем на одном компьютере под управлением одной операционной системы, а пользователь даже не будет знать, что файлы физически находятся на разных носителях информации.

Использование общепринятых имен основных файлов и структуры каталогов существенно облегчает работу в операционной системе, её администрирование и переносимость. Некоторые из этих структур используются при запуске системы, некоторые – во время работы, но все они имеют большое значение для ОС в целом, а нарушение этой структуры может привести к неработоспособности системы или ее отдельных компонентов.

Все каталоги можно разделить на две группы: для статической (редко меняющейся) информации – /bin, /usr и динамической (часто меняющейся) информации – /var, /tmp. Исходя из этого администраторы могут разместить каждый из этих каталогов на собственном носителе, обладающем соответствующими характеристиками.



**Скриншот 2.** Стандартные каталоги в файловой системе UNIX

### Корневой каталог

Корневой каталог / является основой любой ФС UNIX. Все остальные каталоги и файлы располагаются в рамках структуры (дерева), порождённой корневым каталогом, независимо от их физического местонахождения.

### /bin

В этом каталоге находятся часто употребляемые команды и утилиты системы общего пользования. Сюда входят все базовые команды, доступные даже если была примонтирована только корневая файловая система. Примерами таких команд являются: ls, cp, sh и т.п.

### /boot

Директория содержит всё необходимое для процесса загрузки операционной системы: программу-загрузчик, образ ядра операционной системы и т.п.

### /dev

Каталог содержит специальные файлы устройств, являющиеся интерфейсом доступа к периферийным устройствам. Наличие такого каталога не означает, что специальные файлы устройств нельзя создавать в другом месте, просто достаточно удобно иметь один каталог для всех файлов такого типа.

### /etc

В этом каталоге находятся системные конфигурационные файлы. В качестве примеров можно привести файлы /etc/fstab, содержащий список монтируемых файловых систем, и /etc/resolv.conf, который задаёт правила составления локальных DNS-запросов. Среди наиболее важных файлов – скрипты инициализации и деинициализации системы. В системах, наследующих особенности UNIX System V, для них отведены каталоги с /etc/rc0.d по /etc/rc6.d и общий для всех файл описания – /etc/inittab.

### /home (необязательно)

Директория содержит домашние директории пользователей. Её существование в корневом каталоге не обязательно и её содержимое зависит от особенностей конкретной UNIX-подобной операционной системы.

### /lib

Каталог для статических и динамических библиотек, необходимых для запуска программ, находящихся в директориях /bin и /sbin.

### /mnt

Стандартный каталог для временного монтирования файловых систем – например, гибких и флэш-дисков, компакт-дисков и т.п.

### /root (необязательно)

Директория содержит домашнюю директорию суперпользователя. Её существование в корневом каталоге не обязательно.

### /sbin

В этом каталоге находятся команды и утилиты для системного администратора. Примерами таких команд являются: route, halt, init и т.п. Для аналогичных целей применяются директории /usr/sbin и /usr/local/sbin.

### /usr

Эта директория повторяет структуру корневой директории – содержит каталоги /usr/bin, /usr/lib, /usr/sbin, служащие для аналогичных целей.

Каталог /usr/include содержит заголовочные файлы языка C для всевозможных библиотек, расположенных в системе.

Каталог /usr/local является следующим уровнем повторения корневого каталога и служит для хранения программ, установленных администратором в дополнение к стандартной поставке операционной системы.

Каталог /usr/share хранит неизменяющиеся данные для установленных программ. Особый интерес представляет каталог /usr/share/doc, в который добавляется документация ко всем установленным программам.

### /var, /tmp

Используются для хранения временных данных процессов – системных и пользовательских соответственно.

## РЕЗЮМЕ

Все файловые системы семейства UNIX обладают схожей структурой. Основным элементом файловой системы является индексный узел, каталог связывает файл с именем. Помимо файлов и директорий существуют специальные виды файлов: устройства, каналы, символические ссылки и сокеты. Стандарт на файловую систему описывает основные директории иерархической файловой системы UNIX. Для объединения файловых систем нескольких устройств используется механизм монтирования.

## БИБЛИОГРАФИЯ

1. [Введение в администрирование UNIX/Введение в UNIX](https://ru.wikibooks.org/wiki/%D0%92%D0%B2%D0%B5%D0%B4%D0%B5%D0%BD%D0%B8%D0%B5_%D0%B2_%D0%B0%D0%B4%D0%BC%D0%B8%D0%BD%D0%B8%D1%81%D1%82%D1%80%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5_UNIX/%D0%92%D0%B2%D0%B5%D0%B4%D0%B5%D0%BD%D0%B8%D0%B5_%D0%B2_%D0%AE%D0%BD%D0%B8%D0%BA%D1%81)
2. [Введение в администрирование UNIX/Архитектура UNIX](https://ru.wikibooks.org/wiki/%D0%92%D0%B2%D0%B5%D0%B4%D0%B5%D0%BD%D0%B8%D0%B5_%D0%B2_%D0%B0%D0%B4%D0%BC%D0%B8%D0%BD%D0%B8%D1%81%D1%82%D1%80%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5_UNIX/%D0%90%D1%80%D1%85%D0%B8%D1%82%D0%B5%D0%BA%D1%82%D1%83%D1%80%D0%B0_UNIX#%D0%A4%D0%B0%D0%B9%D0%BB%D0%BE%D0%B2%D0%B0%D1%8F_%D1%81%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0_UNIX)