

МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ  
УНИВЕРСИТЕТ им. Н.Э. БАУМАНА

Факультет «Информатика и системы управления»

Кафедра «Автоматизированные системы обработки информации и управления»



**Сёмкин П.С., Сёмкин А.П.**

Методические материалы к лабораторным работам  
по дисциплине  
«Операционные системы»

Лабораторная работа № 10  
**«OS Alt Linux. Файловые системы. Основные права доступа  
к каталогам и файлам»**

**Москва**

**2023 г.**

## ОГЛАВЛЕНИЕ

<b>1 ЦЕЛЬ РАБОТЫ .....</b>	<b>3</b>
<b>2 ТЕОРЕТИЧЕСКАЯ ЧАСТЬ .....</b>	<b>3</b>
2.1 Логическая организация файловых систем ОС Alt Linux .....	3
2.1.1 Понятие файловой системы .....	3
2.1.2 Понятие файла .....	3
2.1.3 Типы файлов .....	4
2.1.4 Каталоги .....	4
2.1.5 Архитектура файловой системы ОС Alt Linux .....	5
2.2 Безопасность файлов .....	5
2.2.1 Администрирование основных прав доступа к файлам и папкам .....	6
2.2.2 Задание основных прав доступа при создании файла или каталога .....	7
2.2.3 Изменение прав доступа .....	8
2.2.4 Изменение владельца файла или каталога .....	8
<b>3 ВЫПОЛНЕНИЕ РАБОТЫ .....</b>	<b>8</b>
3.1 Задание .....	8
3.2 Порядок выполнения работы .....	9
<b>4 КОНТРОЛЬНЫЕ ВОПРОСЫ .....</b>	<b>9</b>
<b>5 ЛИТЕРАТУРА .....</b>	<b>10</b>
<b>6 ПРИЛОЖЕНИЕ. ....</b>	<b>11</b>
6.1 Отображение информации о файлах и каталогах .....	11
6.2 Переход по файловой системе .....	11
6.3 Создание каталога .....	11
6.4 Удаление каталога .....	11
6.5 Создание файла .....	11
6.6 Изменение владельца и группы каталога или файла .....	12
6.7 Способы представления прав доступа .....	12
6.8 Изменение прав доступа в командной строке .....	13
6.9 Изменение прав доступа к каталогам и файлам с использованием графического интерфейса .....	14

## 1 Цель работы

Целью работы является знакомство с архитектурой, объектами файловой системы ОС Alt Linux и администрированием основных прав доступа к файлам и каталогам.

## 2 Теоретическая часть

### 2.1 Логическая организация файловых систем ОС Alt Linux

#### 2.1.1 Понятие файловой системы

Файловая система определяет способ организации данных на диске или на каком-нибудь ином носителе данных и принципы доступа к данным, организованным в файлы.

Файловая система предоставляет возможность пользователям ФС работать с логическим уровнем структуры данных и операциями, выполняемыми над структурами данных в процессе их обработки.

Файловая подсистема является одной из основных компонентов любой операционной системы. Данная подсистема включает в себя комплекс программных модулей, обеспечивающих работу с файловой системой в конкретной операционной системе.

#### 2.1.2 Понятие файла

Под файлом понимают некоторый набор данных (**данные файла**), связанные с этим набором **атрибуты** (имя, размер и т.д.) и множество допустимых операций над атрибутами и данными

**Данные файла на логическом уровне** представляет собой совокупность записей одинаковой структуры (однородных записей) в некотором формате и которые могут быть обработаны некоторой прикладной программой

**Атрибуты файла** определяют его характеристики. Список атрибутов файлов зависит от конкретной файловой системы. При создании файла ему присваивается основной атрибут – его имя. По имени осуществляют доступ к файлу. Атрибуты файлов хранятся в специальных объектах файловой системы – каталогах и индексных узлах(i-node)

### 2.1.3 Типы файлов

ОС Alt Linux поддерживает следующие типы файлов:

- **Обычный файл (regular file)** - содержит данные в некотором формате. Интерпретация содержимого производится прикладной программой. Для ОС это просто последовательность байтов;
- **Каталог (directory)** - содержит имена файлов и указатели на индексные узлы(номера inode) файлов или вложенных каталогов;
- **Специальный файл устройства (special device file)** - обеспечивает доступ к физическому устройству путем открытия, чтения и записи в специальный файл устройства;
- **Символьный файл(character)** - для небуфферизированного обмена данными с символьными устройствами;
- **Блочный файл (block)** - для обмена данными с блочными устройствами в виде пакетов фиксированной длины – блоков;
- **Именованный канал (named pipe)** - используется для связи между процессами;
- **Связь(link)** -Позволяет косвенно адресовать файл (символическая связь);
- **Сокет(socket)** - предназначен для организации взаимодействия между процессами

### 2.1.4 Каталоги

**Каталоги** представляют собой объекты файловой системы, связывающие имена файлов с номерами индексных узлов файлов при помощи **элементов каталогов** (directory entry, dentry).

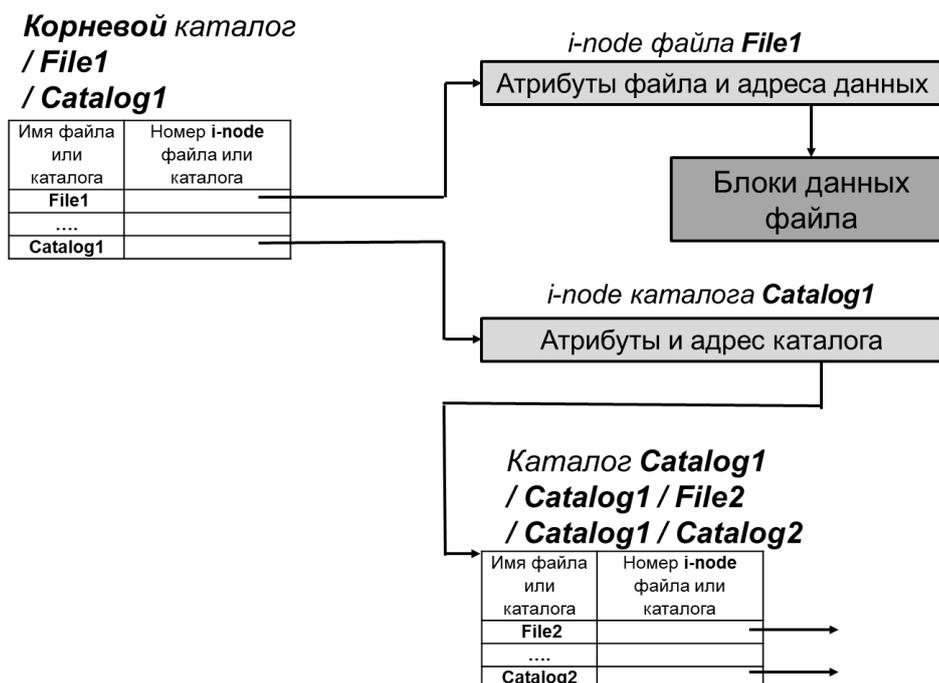
Каждый элемент каталога включает:

- номер индексного узла,
- длину элемента каталога,
- длину имени файла,
- тип файла
- имя файла

### 2.1.5 Архитектура файловой системы ОС Alt Linux

Файловая система ОС организована в виде единого дерева с одной исходной вершиной, которая называется корнем (записывается: "/");

- Каждая вершина в древовидной структуре файловой системы, кроме листьев, является **каталогом**.
- Листья соответствуют **файлам** соответствующего типа.



**Рис.1** Логическая организация хранения атрибутов и данных в файловых системах Alt Linux

### 2.2 Безопасность файлов

Индексный узел каждого каталога или файла содержит информацию, используемую ядром для поддержки политик контроля доступа к соответствующему файлу.

В файловых системах **ext**, индексные узлы включают два атрибута, связанные с безопасностью:

- **файловые разрешения (file permission)**
- **файловые атрибуты (file attribute);**

**Файловые разрешения** определяют права на чтение, запись и исполнение для трех категорий пользователей:

- **владелец файла** (обычно пользователь, создавший файл, но он может быть изменён),
- **группа пользователей**, которые могут иметь доступ к файлу (обычно, группа, к которой принадлежит пользователь, создавший файл, но не обязательно),
- **остальные пользователи.**

**Файловые атрибуты** определяют возможность модификации данных.

Например, файловый атрибут «**только добавление**» (append-only) означает, что пользователи могут добавлять данные к файлу, но не могут модифицировать данные, которые уже в нем присутствуют.

Файловая система **ext** позволяет расширять перечень файловых атрибутов для поддержки других функций безопасности. Например, в **ext** в дополнительных файловых атрибутах хранятся **метаданные контроля доступа, предназначенные для реализации списков контроля доступа POSIX.**

### **2.2.1 Администрирование основных прав доступа к файлам и папкам**

Возможности доступа пользователей к файлам, выполнения команд и перехода в тот или иной каталог можно ограничить путем настройки основных прав доступа для различных категорий пользователей.

Каждая категория пользователей имеет три вида прав, причём эти права имеют отличия для файлов и каталогов.

Для файлов:

- **r(read)** – чтение файла.
- **w(write)** – изменение файла.
- **x(execute)** – выполнение файла, как программы.

Для каталогов:

- **r(read)** – чтение списка файлов в каталоге.

- **w(write)** – изменение и создание файлов в каталоге.
- **x(execute)** – открытие файлов в каталоге.

При выводе на экран длинного списка (**ls -l**) файлов и каталогов в Linux первые десять отображаемых символов являются индикаторами, что представляет собой соответствующий элемент (файл, каталог и т. д.) и возможности чтения, записи и/или выполнения этого элемента для соответствующих категорий пользователей.

После создания каталога и файла **первый символ** в соответствующем длинном листинге выступает в роли индикатора:

**каталог(d); файл(-); символьное устройство(c); блочное устройство(b); символьная ссылка(l); именованный канал(p); сокет(s).**

Следующие девять символов представляют права доступа, настроенные для каталога или файла.

Права доступа можно представить посредством **восьмеричного числа (0-7), либо букв (rwx).**

Права доступа с возможностью чтения позволяют просматривать содержимое каталога, с возможностью записи - изменять (добавлять или модифицировать) содержимое каталога, а с возможностью выполнения - переходить (получать доступ) в каталог.

### **2.2.2 Задание основных прав доступа при создании файла или каталога**

Стандартные права по умолчанию, принятые в Linux, в восьмеричном формате определяются для файла как **0666**, а для каталога **0777**.

При создании файла или каталога используется набор прав доступа, получаемый из стандартного на основе значения стандартной маски **umask**. Для получения прав, используемых по умолчанию, из стандартных прав вычитается стандартная маска.

Информацию о стандартной маске прав доступа можно получить, выполнив команду **umask**.

При использовании стандартной маски **0022** по умолчанию

- для создаваемого файла назначаются права **0644(rw- r - - r - -)**

- для создаваемого каталога назначаются права **0755(rwx r-x r-x.**

Используя команду **umask**, можно настроить права доступа, отличные от значения по умолчанию.

### **2.2.3 Изменение прав доступа**

Права доступа в отношении файлов или каталогов пользователя-владельца можно изменить с помощью команды командной строки **chmod**.

Используя параметр **-R**, можно рекурсивно изменять права доступа ко всем файлам каталога, начиная с той или иной точки в файловой системе.

Права доступа можно изменять как используя интерфейс командной строки, так и используя графическую оболочку.

### **2.2.4 Изменение владельца файла или каталога**

При создании файла или каталога, пользователь становится владельцем этого файла или каталога. То же происходит и в отношении первичной группы. Можно изменить владельца (пользователя) и группу, присвоенную файлу, на другого пользователя и/или группу с помощью команд **chown** и **chgrp**.

## **3 Выполнение работы**

### **3.1 Задание.**

Установить права доступа пользователей и групп пользователей к каталогам и файлам файловой системы:

1. К каталогу **КАФЕДРА** пользователь **admin\_kaf** должен иметь полный доступ. Остальным пользователям доступ запрещён;
2. К каталогу **ИНФОРМАЦИЯ / РАСПОРЯЖЕНИЯ** пользователь **admin\_kaf** должен иметь полный доступ. Остальные пользователи должны иметь возможность чтения;
3. К каталогу **ИНФОРМАЦИЯ / ОБЪЯВЛЕНИЯ** должны иметь полный доступ пользователи **admin\_kaf** и **admin\_stud**. Остальные пользователи должны иметь возможность чтения;

4. К каталогу **СТУДЕНТЫ** и ко всем вложенным каталогам пользователь **admin\_stud** должен иметь полный доступ.

5. К каталогам **СТУДЕНТЫ/ПЕРВЫЙ\_КУРС – СТУДЕНТЫ/ТРЕТИЙ КУРС** разрешить полный доступ для пользователей групп **stud\_1k – stud\_3k** соответственно. Прочим пользователям доступ запретить.

**Примечание.** Пользователи, группы пользователей и каталоги, созданные в системе ранее:

- пользователи: **admin\_kaf, admin\_stud, stud\_51, stud\_52, stud\_53, stud\_54, stud\_55**

- группа пользователей **stud\_3k (stud\_51, stud\_52, stud\_53, stud\_54, stud\_55)**

- каталоги файловой системы:

**/ home / admin\_kaf / ИНФОРМАЦИЯ**

**/ home / admin\_stud / СТУДЕНТЫ**

### **3.2 Порядок выполнения работы.**

1. Войти в систему под учётной записью **stud\_XX** (XX –индекс группы).
2. Запустить программу виртуализации **Oracle VM VirtualBox**
3. Запустить виртуальную машину **Alt-10**
4. Войти в систему под учётной записью **root/adminroot**
5. Создать необходимые каталоги
6. Установить разрешения для файлов и каталогов в соответствии с заданием.
7. Проверить правильность установки разрешений для файлов и каталогов.
8. Ответить на контрольные вопросы.

## **4 Контрольные вопросы**

1. Какие типы файлов поддерживает ОС Alt Linux?
2. Что такое файловые разрешения и файловые атрибуты?
3. Какие категории пользователей определяют файловые разрешения?

## 5 Литература

1. Сёмкин П.С., Аксёнов А.Н. Файловые системы. Логическая организация и физическая реализация. Сборник учебно-методических работ кафедры «Системы обработки информации и управления» (бакалавры). Учебное пособие. Вып. 1./Под ред: В.М. Черненко. –М: «АртКом», 2013. – стр. 95-120
2. Сёмкин П.С., Семкин А.П. Файловые системы операционных систем Windows и Unix. Сборник учебно-методических работ кафедры «Системы обработки информации и управления» (бакалавры). Учебное пособие. Вып. 2./Под ред. В.М. Чёрненко. –М: «АртКом», 2014. – стр. 160-189
3. Негус К. Ubuntu и Debian Linux для продвинутых. 2-е изд. – СПб.: Питер, 2014. -384 с.: ил.
4. Семкин П.С., Семкин А.П., Горячкин Б.С. Лабораторный практикум по дисциплине «Операционные системы». Часть 1. ОС Alt Linux. Управление пользователями. Администрирование дисковой подсистемы: Учебно-методическое пособие. –М.: Издательство «Спутник+», 2023. -78 с.
5. Документация для ОС «Альт Рабочая станция». Режим доступа: <https://www.basealt.ru/alt-workstation/docs>

## 6 Приложение.

### 6.1 Отображение информации о файлах и каталогах

**\$ pwd** - вывод пути к текущему каталогу

**\$ ls** - вывод списка файлов и каталогов текущего каталога

**\$ ls -l** - вывод списка файлов и каталогов в форматированном виде с отображением прав доступа

**\$ ls -la** - вывод списка файлов и каталогов в форматированном виде, в том числе начинающихся с точки

**\$ ls -li** - вывод информации о файле (включая информацию о индексном узле)

### 6.2 Переход по файловой системе

**\$ cd** - переход в домашний каталог пользователя

**\$ cd /home** - переход в каталог **/home**

**\$ cd ..** - переход в родительский каталог данного каталога

**\$ cd /** - переход в корневой каталог файловой системы

### 6.3 Создание каталога

**# mkdir <имя каталога>** - создание каталога

### 6.4 Удаление каталога

**# rmdir <имя каталога>** - удаление каталога

**# rm -R <имя каталога>** - удаление непустого каталога

### 6.5 Создание файла

**# touch <путь> <имя файла>** - создание пустого файла

**# cp** – копирование файла

**# cat** - вывести на экран файл (**cat <имя файла>**)

**# cat >> <имя файла>** ввод с консоли (остановить ввод - **Ctrl+Z**)

## 6.6 Изменение владельца и группы каталога или файла

**# chown <владелец>: <имя каталога или файла>** - изменение владельца каталога или файла

**# chgrp <владелец>: <имя каталога или файла>** - изменение группы каталога или файла

**# chown <владелец>:<группа> <имя каталога или файла>** - изменение владельца и группы каталога или файла

## 6.7 Способы представления прав доступа

### 1. Для файла

Символьное представление	Двоичное представление	Восьмеричное представление	Права доступа
<b>r w x</b>	<b>1 1 1</b>	<b>7</b>	чтение, запись и выполнение
<b>r w -</b>	<b>1 1 0</b>	<b>6</b>	чтение и запись
<b>r - x</b>	<b>1 0 1</b>	<b>5</b>	чтение и выполнение
<b>r - -</b>	<b>1 0 0</b>	<b>4</b>	только чтение
<b>- w x</b>	<b>0 1 1</b>	<b>3</b>	запись и выполнение
<b>- w -</b>	<b>0 1 0</b>	<b>2</b>	только запись
<b>- - x</b>	<b>0 0 1</b>	<b>1</b>	только выполнение
<b>- - -</b>	<b>0 0 0</b>	<b>0</b>	нет прав

### 2. Для каталога

Символьное представление	Двоичное представление	Восьмеричное представление	Права доступа
<b>r w x</b>	<b>1 1 1</b>	<b>7</b>	чтение списка файлов, изменение и создание файлов, открытие файлов в каталоге
<b>r w -</b>	<b>1 1 0</b>	<b>6</b>	чтение списка, изменение и создание
<b>r - x</b>	<b>1 0 1</b>	<b>5</b>	чтение и открытие
<b>r - -</b>	<b>1 0 0</b>	<b>4</b>	только чтение списка

- w x	0 1 1	3	изменение и создание, открытие
- w -	0 1 0	2	только изменение
- - x	0 0 1	1	только открытие
- - -	0 0 0	0	нет прав

### 3. Примеры записи в восьмеричном представлении:

**[права пользователя][права группы][права остальных]**

- **744** - разрешить всё для пользователя-владельца, а остальным только чтение;
- **755** - всё для пользователя-владельца, остальным только чтение и выполнение;
- **764** - всё для пользователя-владельца, чтение и запись для группы, и только чтение для остальных;
- **777** - всем разрешено всё.

### 4. Примеры записей в символьном представлении:

**[категории пользователей][операция][права]**

**категории пользователей:** **u**(user) – пользователь, **g**(group) – группа, **o** (other) – остальные пользователи

в качестве операции могут использоваться знаки

"+" - включить или "-" - отключить.

#### Примеры

- **u+x** - разрешить выполнение для владельца;
- **ugo+x** - разрешить выполнение для всех;
- **ug+w** - разрешить запись для владельца и группы;
- **o-x** - запретить выполнение для остальных пользователей;
- **ugo+rwx** - разрешить все для всех;

## 6.8 Изменение прав доступа в командной строке

**# chmod [параметры] [права] [имя файла]**

**параметры:**

- **-c** - выводить информацию обо всех изменениях;
- **-f** - не выводить сообщения об ошибках;
- **-v** - выводить максимум информации;
- **--preserve-root** - не выполнять рекурсивные операции для корня "/";
- **--reference** - взять маску прав из указанного файла;
- **-R** - включить поддержку рекурсии;
- **--version** - вывести версию утилиты;

### Примеры

**# chmod 0777 /home/** - изменение права доступа к каталогу для **/home. 0777**

– разрешение на чтение/запись/исполнение для всех категорий пользователей;

**# chmod -R 0777 /home/** - рекурсивное изменение прав доступа к каталогу

**/home. 0777** – разрешение на чтение/запись/исполнение для всех групп. Все вложенные каталоги и файлы будут иметь такие же права **0777**;

## 6.9 Изменение прав доступа к каталогам и файлам с использованием графического интерфейса

- В контекстном меню файла или каталога перейти по команде **свойства**.
- В отрывшемся окне выбрать закладку **Права**
- Установить права доступа для соответствующих категорий пользователей

### Значения поля **Доступ**

- **Нет**

Пользователь даже не сможет увидеть, какие файлы содержатся в каталоге.

- **Только перечисление файлов**

Пользователь сможет увидеть, какие файлы содержатся в каталоге, но не сможет открывать, создавать или удалять их.

- **Доступ к файлам**

Пользователь сможет открывать файлы в каталоге (если это позволяют права доступа к данному конкретному файлу), но не сможет удалять файлы или создавать новые файлы.

- **Создание и удаление файлов**

Пользователь будет иметь полный доступ к каталогу, включая открытие, создание и удаление файлов.

Можно также установить права доступа для всех файлов в каталоге, нажав

- **Изменить права на вложенные файлы.**

Можно использовать выпадающие списки для настройки прав доступа к вложенным файлам или каталогам, затем нажать «**Изменить**». Права доступа будут установлены как для файлов и каталогов, так и для вложенных каталогов до любой глубины вложенности.