

УТВЕРЖДЕН  
КШДС.10514-01 34 01-ЛУ

**ОПЕРАЦИОННАЯ СИСТЕМА**

**«АЛЬТ ЛИНУКС СПТ 7.0»**

**Руководство оператора**

**КШДС.10514-01 34 01**

**Книга № 4**

**Листов 35**

Изн. № подл.	Подп. и дата	Взам. изн. №	Изн. № дубл.	Подп. и дата

2017

Литера

**АННОТАЦИЯ**

Настоящий документ является руководством оператора программного комплекса «Операционная система «Альт Линукс СПТ 7.0» КШДС.10514-01.

В документе представлены общие сведения о назначении и условиях выполнения программного комплекса «Операционная система «Альт Линукс СПТ 7.0» КШДС.10514-01, о его функциональных возможностях, а также приведен порядок работы и список действий оператора в ходе работы с программным комплексом.

## СОДЕРЖАНИЕ

<b>1. Назначение программы .....</b>	<b>4</b>
<b>2. Условия выполнения программы .....</b>	<b>6</b>
2.1. Минимальный состав аппаратных средств .....	6
2.2. Требования и условия организационного и технологического характера .....	6
<b>3. Выполнение программы.....</b>	<b>7</b>
3.1. Загрузка ОС «Альт Линукс», установленной на жесткий диск.....	7
3.1.1. Вход и работа в системе в консольном режиме .....	9
3.1.2. Вход и работа в системе в графическом режиме.....	9
3.2. Работа с наиболее часто используемыми компонентами .....	11
3.2.1. Работа с командной оболочкой и основные команды .....	11
3.2.2. Текстовый редактор Vi.....	19
3.2.3. Редактор Vim.....	22
3.2.4. Служба xinetd .....	24
3.2.5. Служба Crontab.....	26
3.2.6. Служба передачи файлов FTP .....	27
3.2.7. Защищенный интерпретатор команд SSH.....	28
3.3. Работа со справочной информацией .....	29
3.3.1. Наиболее часто используемые опции команды man.....	29
3.3.2. Управление справочными страницами.....	29
3.3.3. Печать справочных страниц .....	30
3.4. Завершение работы ОС .....	30
3.4.1. Завершение работы ОС с помощью консоли.....	31
3.4.2. Завершение работы ОС с помощью инструментов графической оболочки.....	32
<b>4. Сообщения оператору .....</b>	<b>33</b>
4.1. Первичная идентификация.....	33
4.2. Предупреждение о неуспешных идентификации и аутентификации.....	33
4.3. Оповещение пользователя после успешного входа .....	33
<b>Перечень сокращений .....</b>	<b>34</b>

## 1. НАЗНАЧЕНИЕ ПРОГРАММЫ

Программный комплекс «Операционная система «Альт Линукс СПТ 7.0» (далее – ОС «Альт Линукс») предназначен для группового и корпоративного использования в качестве средства автоматизации информационных, конструкторских и производственных процессов предприятий (организаций, учреждений). ОС «Альт Линукс» обеспечивает выполнение программ в защищенной среде, обработку, хранение и передачу информации.

ОС «Альт Линукс» поддерживает клиент-серверную архитектуру и может обслуживать процессы как в пределах одной компьютерной системы, так и процессы на других персональных электронных вычислительных машинах (далее – ПЭВМ) через каналы передачи данных или сетевые соединения.

ОС «Альт Линукс» обладает следующими функциональными характеристиками:

- обеспечивается возможность обработки, хранения и передачи информации в защищенной программной среде;
- обеспечивается возможность запуска пользовательского программного обеспечения в сертифицированном окружении;
- обеспечивается возможность функционирования в многозадачном режиме (одновременное выполнение множества процессов);
- обеспечивается возможность масштабирования системы: возможна эксплуатация операционной системы как на одной ПЭВМ, так и в информационных системах различной архитектуры;
- обеспечивается многопользовательский режим эксплуатации;
- обеспечивается поддержка мультипроцессорных систем;
- обеспечивается поддержка виртуальной памяти;
- обеспечивается поддержка запуска виртуальных машин;
- обеспечивается сетевая обработка данных.

Для поддержки выполнения описанных функций в ОС «Альт Линукс» реализованы следующие возможности:

- управление процессами;
- управление памятью;
- управление файлами и внешними устройствами;
- управление доступом к обрабатываемой информации;
- администрирование;
- поддержка интерфейса прикладного программирования;
- поддержка пользовательского интерфейса.

Комплекс средств защиты ОС «Альт Линукс» (далее – КСЗ) предназначен для выполнения функций защиты информации в объеме требований класса защищенности 4 в соответствии с руководящим документом «Средства вычислительной техники. Защита от несанкционированного доступа к информации. Показатели защищенности от несанкционированного доступа к информации» (Гостехкомиссия России, 1992)

(далее – РД НСД) при соблюдении условий и указаний по эксплуатации в документе «Операционная система «Альт Линукс СПТ 7.0». Формуляр» КШДС.10514-01 30 01.

КСЗ соответствует требованиям уровня 3 контроля отсутствия недеklarированных возможностей в соответствии с документом «Защита от несанкционированного доступа к информации. Часть 1. Программное обеспечение средств защиты информации. Классификация по уровню контроля отсутствия недеklarированных возможностей» (Гостехкомиссия России, 1999 г.) (далее – РД НДВ).

С учетом ограничений по эксплуатации ОС «Альт Линукс» может использоваться для проектирования и построения автоматизированных систем с классом защищенности до «1В» включительно.

## 2. УСЛОВИЯ ВЫПОЛНЕНИЯ ПРОГРАММЫ

### 2.1. Минимальный состав аппаратных средств

Для функционирования ОС «Альт Линукс» необходима ПЭВМ, обладающая следующими необходимыми характеристиками:

- аппаратная платформа – ПЭВМ типа IBM PC;
- аппаратная платформа – процессор архитектуры x86-64 (Intel или совместимый с ним процессор, включая AMD, при этом для версии ОС x86 процессор должен поддерживать технологию PAE);
- оперативная память – не менее 512 МБ (рекомендуется 1 ГБ и более);
- объем доступного пространства накопителя на жестких магнитных дисках – не менее 2 ГБ (рекомендуется 15 ГБ и более);
- периферийные устройства ввода/вывода – устройство чтения и записи компакт-дисков.

### 2.2. Требования и условия организационного и технологического характера

К пользователю ОС «Альт Линукс» предъявляется следующее требование: базовые навыки работы с ОС семейства «Linux».

### 3. ВЫПОЛНЕНИЕ ПРОГРАММЫ

#### 3.1. Загрузка ОС «Альт Линукс», установленной на жесткий диск

Вызов ОС «Альт Линукс», установленной на жесткий диск ПЭВМ, происходит автоматически и выполняется после запуска ПЭВМ и отработки набора программ BIOS. ОС вызывает специальный загрузчик.

Загрузчик настраивается автоматически и включает в свое меню все системы, установку которых на ПЭВМ он определил. Поэтому загрузчик также может использоваться для вызова других ОС, если они установлены на компьютере.

Примечание. При наличии на компьютере нескольких ОС (или при наличии нескольких вариантов загрузки), оператор будет иметь возможность выбрать необходимую ОС (вариант загрузки). В случае, если оператором ни один вариант не был выбран, то по истечении заданного времени будет загружена ОС (вариант загрузки), заданные по умолчанию.

Для перенастройки или регенерации загрузочного меню и работы загрузчика необходимо воспользоваться следующей командой:

```
grub-mkconfig -o /boot/grub/grub.cfg
```

которая генерирует конфигурационный файл на основе содержимого каталога /etc/grub.d/.

При стандартной установке ОС «Альт Линукс» в начальном меню загрузчика доступны несколько вариантов загрузки (Рис. 1): обычная загрузка, загрузка с дополнительными параметрами (например, «recovery mode» – загрузка с минимальным количеством драйверов), загрузка в программу проверки оперативной памяти (memtest).

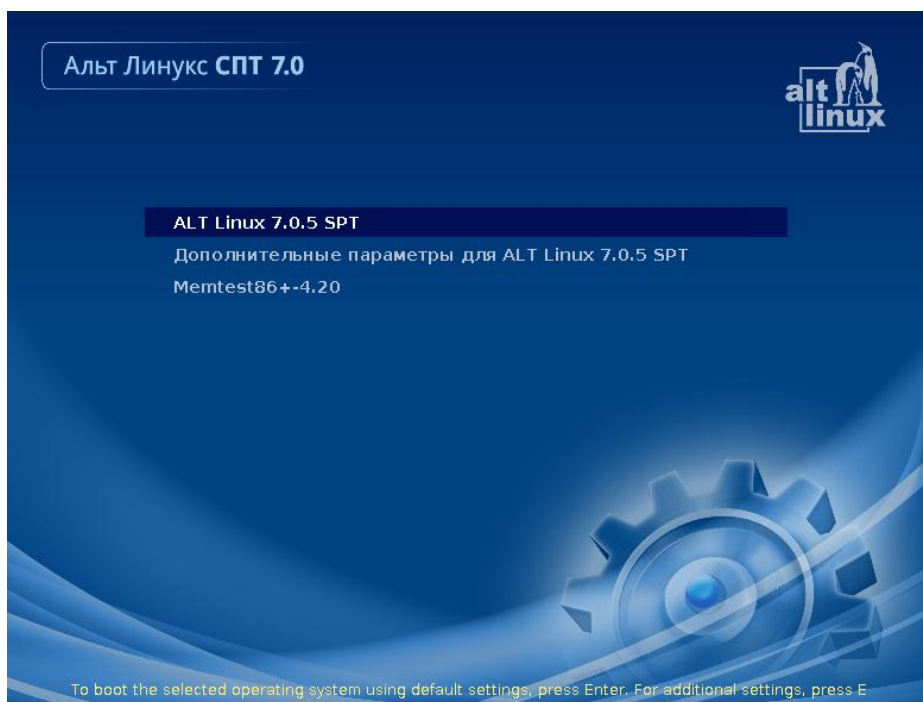


Рис. 1

Параметры, которые будут переданы ядру ОС «Альт Линукс» при загрузке, можно задать по нажатию клавиши <E> (Рис. 2).

```

GNU GRUB, версия 2.00-alt20.M70C.2

setparams 'Memtest86+-4.20'

    insmod part_msdos
    insmod ext2
    set root='hd0,msdos2'
    if [ x${feature_platform_search_hint} = xy ]; then
        search --no-floppy --fs-uuid --set=root --hint-bio\
s=hd0,msdos2 --hint-efi=hd0,msdos2 --hint-baremetal=ahci0,ms\
dos2 1efbd513-034b-4bd6-ab0c-3c5fe54b4043
    else
        search --no-floppy --fs-uuid --set=root 1efbd513-0\
34b-4bd6-ab0c-3c5fe54b4043
    fi
    linux16 /boot/memtest-4.20.bin

```

Поддерживается несколько Emacs-подобных команд редактирования на экране. Есть списки дополнений по TAB. Нажмите Ctrl-x или F10 для загрузки, Ctrl-c или F2 для получения командной строки или ESC для отмены изменений и возврата в меню GRUB.

Рис. 2

В процессе загрузки ОС «Альт Линукс» оператор может следить за информацией процесса загрузки, которая отображает этапы запуска различных служб и программных серверов в виде отдельных строк (Рис. 3), на экране монитора.

```

Starting kernel logger service: [ DONE ]
Starting fbsetfont: [ DONE ]
Starting keytable: [ DONE ]
Starting console mouse service: [ DONE ]
Adjusting environment for slapd: ^[[^[[^[[
Starting slapd service: [ DONE ]
Starting crond service: [ DONE ]
Starting smartd service: [ FAILED ]
Starting anacron service: [ DONE ]
Starting automount service: [ DONE ]
Starting Avahi mDNS/DNS-SD service: [ DONE ]
Starting xinetd service: [ DONE ]
Starting dovecot service: [ DONE ]
Starting sshd service: [ DONE ]
Starting CUPS service: [ DONE ]
Exporting NFS file systems: [ DONE ]
Starting NFS quotas service: [ DONE ]
Starting NFS daemon: [ DONE ]
Starting NFS mount service: [ DONE ]
Stopping rpc.idmapd service: [ DONE ]
Starting rpc.idmapd service: [ DONE ]
Starting bacula-fd service: [ DONE ]
Starting bacula-sd service: [ DONE ]
Starting spand service: [ DONE ]
Adjusting environment for mysqld: [ DONE ]
Starting mysqld service: [ DONE ]
Adjusting environment for postfix: [ DONE ]
Checking postfix configuration: [ DONE ]
Starting postfix: [ DONE ]
Starting httpd2 service: [ DONE ]
Starting bacula-dir service: [ DONE ]
Starting ahttpd service: [ DONE ]
Starting alterator service: [ DONE ]
Deactivating splash [ FAILED ]

```

Рис. 3

При этом каждая строка (см. Рис. 3) заканчивается словом вида [XXXXXXXX] (FAILED или DONE), являющегося признаком нормального или ненормального завершения этапа загрузки.



Слово XXXXXXXX=FAILED (авария) свидетельствует о неуспешном завершении этапа загрузки, что требует вмешательства и специальных действий администратора системы.

После окончания работы загрузчика на экране появляется запрос на ввод имени учетной записи (login). Для прохождения процедуры идентификации и аутентификации необходимо ввести корректное имя учетной записи (login), а затем корректный пароль (Password), соответствующий учетной записи.

### 3.1.1. Вход и работа в системе в консольном режиме

Стандартная установка ОС «Альт Линукс» включает базовую систему, работающую в консольном режиме.

При загрузке в консольном режиме работа загрузчика ОС «Альт Линукс» завершается запросом на ввод логина и пароля учетной записи (Рис. 4).

```
Welcome to ALT Linux 7.8.4 SPT (silo) / tty1
alt_08 login:
```

Рис. 4

Для дальнейшего входа в систему необходимо ввести логин и пароль учетной записи пользователя.

В случае успешного прохождения процедуры аутентификации и идентификации будет выполнен вход в систему. ОС «Альт Линукс» перейдет к штатному режиму работы и предоставит дальнейший доступ к консоли (Рис. 5).

```
Welcome to ALT Linux 7.8.4 SPT (silo) / tty1
alt_08 login: alt
Password:
You have new mail in folder /var/mail/alt.
[alt@alt_08 ~]$\
```

Рис. 5

### 3.1.2. Вход и работа в системе в графическом режиме

В состав базовой ОС «Альт Линукс» также может входить графическая оболочка МАТЕ. Графическая оболочка состоит из набора различных программ и технологий, используемых для управления ОС и предоставляющих оператору удобный графический интерфейс для работы в виде графических оболочек и оконных менеджеров.

При загрузке в графическом режиме работа загрузчика ОС заканчивается переходом к окну входа в систему, в котором необходимо ввести логин учетной записи пользователя (Рис. 6) и нажать кнопку «Войти».

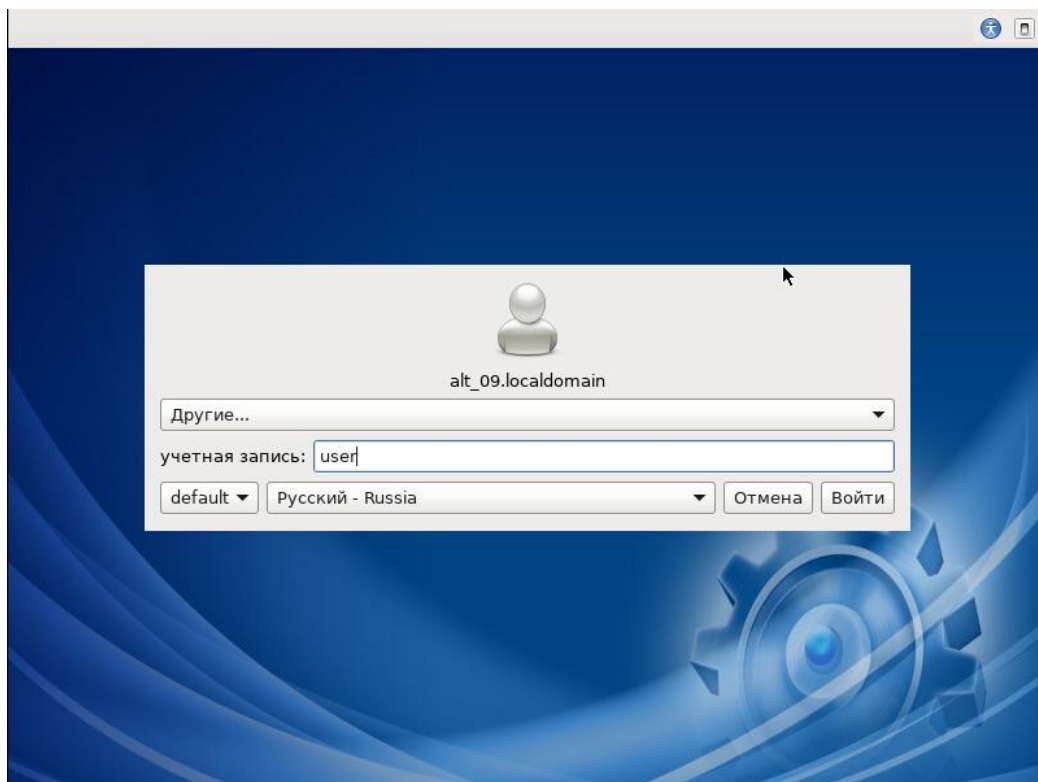


Рис. 6

В открывшемся окне (Рис. 7) необходимо ввести пароль, соответствующий учетной записи пользователя, после чего снова нажать кнопку «Войти».

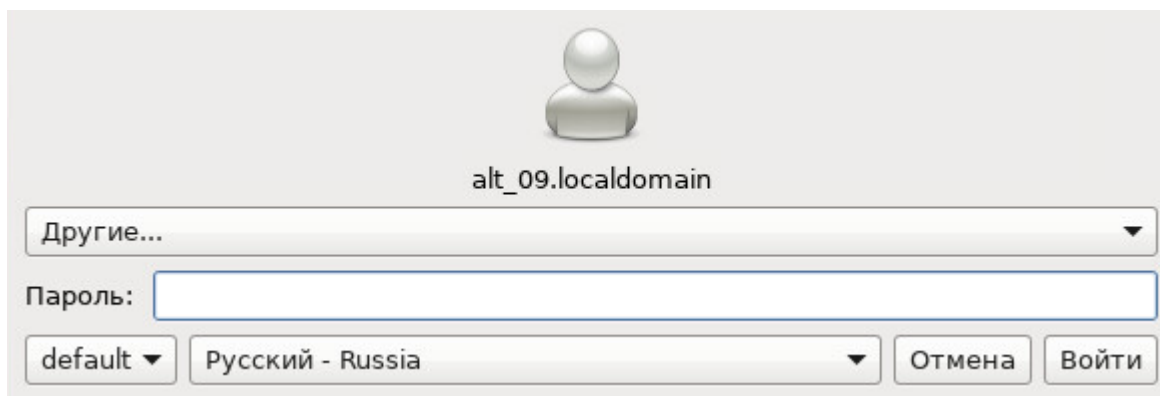


Рис. 7

В случае успешного прохождения процедуры аутентификации и идентификации будет выполнен вход в систему. ОС «Альт Линукс» перейдет к штатному режиму работы и предоставит дальнейший доступ к графическому интерфейсу (Рис. 8).



Рис. 8

### 3.2. Работа с наиболее часто используемыми компонентами

#### 3.2.1. Работа с командной оболочкой и основные команды

Для управления ОС используются командные оболочки (интерпретаторы).

Командный интерпретатор предназначен для передачи команд пользователю ОС. При помощи командных интерпретаторов можно писать небольшие программы – сценарии (скрипты). В ОС «Альт Линукс» доступны следующие командные оболочки:

- bash;
- pdksh.

Оболочкой по умолчанию является Bash (Bourne Again Shell) – самая распространённая оболочка под Linux, которая ведёт историю команд и предоставляет возможность их редактирования.

В дальнейшем описании работы с ОС «Альт Линукс» будут использоваться примеры с использованием этой оболочки.

В Bash имеется несколько приемов для работы со строкой команд. Например, можно использовать следующие сочетания клавиш:

- <Ctrl> + <A> – переход на начало строки;
- <Ctrl> + <U> – удаление текущей строки;
- <Ctrl> + <C> – остановка текущей задачи;

– `<Ctrl> + <R>` – поиск конкретной команды в списке набранных без пролистывания всей истории.

Для ввода нескольких команд одной строкой можно использовать разделитель «;». По истории команд можно перемещаться с помощью клавиш `<↑>` («вверх») и `<↓>` («вниз»).

Команды, присутствующие в истории, отображаются в списке пронумерованными. Для того, чтобы запустить конкретную команду необходимо набрать:

! номер команды

При вводе «!» запустится последняя из набранных команд.

В Bash имеется возможность самостоятельного завершения имен команд из общего списка команд, что облегчает работу в случае, если имена программ и команд слишком длинны. При нажатии клавиши `<Tab>` Bash завершает имя команды, программы или каталога, если не существует нескольких альтернативных вариантов. Например, чтобы использовать программу декомпрессии `bunzip2`, можно набрать следующую команду:

```
bu
```

Затем нажать клавишу `<Tab>`. Так как в данном случае существует несколько возможных вариантов завершения команды, то необходимо повторно нажать клавишу `<Tab>`, чтобы получить список имен, начинающихся с «`bu`».

В предложенном примере можно получить следующий список:

```
$ bu buildhash builtin bunzip2
```

В случае, если ввести команду «`bun`» (`bunzip` – это единственное имя, третьей буквой которого является «`n`»), а затем нажать клавишу табуляции `<Tab>`, то оболочка самостоятельно дополнит имя.

Программы, вызываемые из командной строки, Bash ищет в каталогах, определяемых в системной переменной `PATH`. По умолчанию в этот перечень каталогов не входит текущий каталог, обозначаемый «`./`» (если только не выбран один из двух самых слабых уровней защиты). Поэтому для запуска программы `prog` из текущего каталога необходимо использовать следующую команду:

```
./prog
```

### 3.2.1.1. Учетные записи пользователей

#### 3.2.1.1.1. Команда `su`

Команда `su` позволяет получить права администратора. При вводе команды `su`, будет запрошен пароль суперпользователя (`root`), и, в случае ввода корректного пароля, оператор получит права администратора. Чтобы вернуться к правам оператора, необходимо ввести следующую команду:

```
exit
```

## 3.2.1.1.2. Команда id

Команда id позволяет получить имя пользователя, его id, списка групп, в которых он состоит.

## 3.2.1.2. Основные операции с файлами и каталогами

## 3.2.1.2.1. Команда ls

Команда ls предназначена для вывода списка файлов в каталоге.

Доступные опции команды ls приведены в таблице (Таблица 1).

Таблица 1 – Опции команды ls

Опция	Описание
-a	Выводится список всех файлов, включая файлы, названия которых начинаются с точки (по умолчанию такие файлы скрываются)
-l	Осуществляется подробный вывод, включая атрибуты файлов
-s	Добавляется размер файла в блоках
-R	При выводе содержимого каталога выводится рекурсивно дерево подкаталогов с их содержимым
-t	Выводится содержимое каталога с сортировкой по дате и времени
-S	Выводится содержимое каталога с сортировкой по размеру файла

Следующий пример использования команды ls выводит список файлов в заданных каталогах dir1, dir2, dir3:

```
ls dir1 dir2 dir3
```

## 3.2.1.2.2. Команда cp

Команда cp предназначена для копирования файлов или каталогов.

Доступные опции команды cp приведены в таблице (Таблица 2).

Таблица 2 – Опции команды cp

Опция	Описание
-R	Осуществляется рекурсивное копирование
-p	Копируется не только содержимое файла, но и права на файл и временные метки, информация о его владельце и группе
-i	Копирование в интерактивном режиме, при котором выдаются предупреждения о перезаписи уже существующего файла
-f	Перезапись файла без предупреждения
-v	Предоставление подробной информации о действиях выполняемых командой

Следующий пример использования команды cp демонстрирует копирование файла srcfile1 в каталог dest\_dir:

```
cp srcfile1 dest_dir
```

### 3.2.1.2.3. Команда mv

Команда mv предназначена для переименования или перемещения файлов или каталогов.

Доступные опции команды mv приведены в таблице (Таблица 3).

Таблица 3 – Опции команды mv

Опция	Описание
-i	Интерактивный режим, при котором выдается предупреждение, прежде чем перезаписать существующий файл
-f	Перезапись файлов без предупреждений

Следующий пример использования команды mv переименовывает файл original\_file\_name:

```
mv original_file_name new_file_name
```

Следующий пример использования команды mv перемещает файлы в каталог destination\_dir:

```
mv file1 file2 file3 destination_dir
```

### 3.2.1.2.4. Команда rm

Команда rm предназначена для удаления файлов или каталогов.

Доступные опции команды rm приведены в таблице (Таблица 4).

Таблица 4 – Опции команды rm

Опция	Описание
-i	Интерактивный режим, при котором выдается предупреждение, прежде чем удалить какой-либо файл
-f	Удаление файлов, игнорируя любые ошибки и предупреждения
-r	Рекурсивное удаление каталога (т.е. удаление, включая подкаталоги и скрытые файлы)
-v	Предоставление подробной информации о действиях выполняемых командой

### 3.2.1.2.5. Команда cd

Команда cd предназначена для смены каталога. Команда работает как с абсолютными, так и с относительными путями. Абсолютный путь начинается от верха файловой системы, с корневого каталога /, также называемого просто корень (root), и продолжается до запрашиваемого каталога; относительный путь начинается от текущего каталога, где бы он ни находился. Использование абсолютного пути позволяет переходить в нужный каталог от корневого каталога /, но для этого надо знать полный путь до него. Использование относительных путей позволяет изменять каталог относительно текущего каталога, что бывает удобно, если необходимо перейти в каталог, который является подкаталогом текущего.

Например, для перехода из домашнего каталога в его подкаталог docs/, необходимо указать относительный путь:

```
cd docs/
```

Для перехода в каталог /usr/bin, необходимо указать абсолютный путь:

```
cd /usr/bin/
```

### 3.2.1.2.6. Команда pwd

Команда pwd выводит абсолютный путь к каталогу.

### 3.2.1.2.7. Команда mkdir

Команда mkdir предназначена для создания каталогов.

Доступные опции команды mkdir приведены в таблице (Таблица 5).

Таблица 5 – Опции команды mkdir

Опция	Описание
-p	При указании пути к каталогу (а не просто имени каталога), команда создаст все необходимые каталоги автоматически
-m	Создание каталога с заданными правами

Следующий пример использования команды mkdir создаст каталоги /one /one/two и /one/two/three если они не существуют, а затем и сам каталог /one/two/three/dir:

```
mkdir -p /one/two/three/dir
```

### 3.2.1.2.8. Команда rmdir

Команда rmdir предназначена для удаления каталога, при условии, что он пуст.

Для команды mkdir доступна опция «-p» – при указании пути к каталогу (а не просто имени каталога), команда удалит все каталоги автоматически.

Команда rmdir часто заменяется командой «rm -rf», которая позволяет удалять каталоги, даже если они не пусты.

## 3.2.1.3. Просмотр файлов

### 3.2.1.3.1. Команда cat

Команда cat позволяет просмотреть файл целиком, копируя файлы в стандартный поток вывода и объединяя их.

Доступные опции команды cat приведены в таблице (Таблица 6).

Таблица 6 – Опции команды cat

Опция	Описание
-T	Вывод знака табуляции в виде ^
-E	Вывод символа новой строки в виде \$
-n	Присоединение в начале каждой строки ее номера
-b	Присоединение в начале каждой непустой строки ее номера
-s	Замена каждой последовательности пустых строк одной пустой строкой

## 3.2.1.3.2. Команда less

Команда less позволяет просматривать текст постранично.

Доступные опции команды less приведены в таблице (Таблица 7).

Таблица 7 – Опции команды less

Опция	Описание
-c	Очистка экран перед тем, как отобразить следующую страницу
-m	Вывод информации о том, какая часть файла выведена на данный момент (в процентах)
-N	Вывод номеров строк
-r	Вывод управляющих (непечатаемых) символов
-s	Объединение несколько пустых строк в одну
-S	Урезание длинных строк до длины экрана вместо переноса

## 3.2.1.3.3. Команда grep

Команда grep предназначена для поиска текста соответствующего регулярному выражению в файлах или потоке вывода.

Команда grep обладает следующим синтаксисом:

```
grep [ опции ] шаблон_поиска [файл]
```

Доступные опции команды grep приведены в таблице (Таблица 8).

Таблица 8 – Опции команды grep

Опция	Описание
-r	Рекурсивный поиск во всех каталогах
-n	Вывод номеров строк, в которых найдено совпадение
-l	Вывод списка файлов, содержащих шаблон
-v	Поиск строк не содержащих шаблон (инверсия)
-i	Поиск с игнорированием регистра

## 3.2.1.4. Создание и редактирование файлов

Создание и редактирование файлов выполняется командой touch, которая устанавливает время последнего изменения и доступа в текущее системное время у заданного файла. Если файл не существует – он создается.

Следующий пример использования команды touch создает файл «myfile.txt»:

```
touch myfile.txt
```



### 3.2.1.5. Поиск файлов

#### 3.2.1.5.1. Команда find

Команда `find` предназначена для поиска всех файлов, начиная с корневой директории. Поиск может осуществляться по имени, типу или владельцу файла.

Ключи для поиска:

- `name` – поиск по имени файла;
- `type` – поиск по типу `f`=файл, `d`=каталог, `l`=ссылка(lnk);
- `user` – поиск по владельцу (имя или UID).

Доступные опции `find` приведены в таблице (Таблица 9).

Таблица 9 – Опции команды `find`

Опция	Описание
<code>-delete</code>	Удаление найденных файлов
<code>-exec command {} \;</code>	Выполнение над найденным файлом указанной команды
<code>-print</code>	Вывод на экран найденных файлов
<code>-mount</code>	Исключение из поиска каталогов, которые ведут на другие разделы диска. Эта опция должна быть указана перед опциями с параметрами (например « <code>-type</code> » и « <code>-name</code> »)
Примечание. Критерии поиска могут соединяться логическим «или» (опция « <code>-o</code> ») и логическим «и» (опция « <code>-a</code> »).	

Далее приводятся примеры поисковых запросов с помощью команды `find`:

```
find . \( -name "my*" -o -name "qu*" \) -print
find . \( -type d -a -name "qu*" \) -print
```

#### 3.2.1.5.2. Команда whereis

Команда `whereis` сообщает путь к исполняемому файлу программы, ее исходным файлам (если есть) и соответствующим страницам справочного руководства.

Доступные опции `whereis` приведены в таблице (Таблица 10).

Таблица 10 – Опции команды `whereis`

Опция	Описание
<code>-b</code>	Вывод информации только об исполняемых файлах
<code>-m</code>	Вывод информации только о страницах справочного руководства
<code>-s</code>	Вывод информации только об исходных файлах

### 3.2.1.6. Сжатие и упаковка файлов

Сжатие и упаковка файлов выполняется с помощью команды `tar`, которая преобразует файл или группу файлов в архив без сжатия (tarfile).

Упаковка файлов в архив чаще всего выполняется следующей командой:

`tar -cf` [имя создаваемого файла архива] [упаковываемые файлы и/или директории]

Далее приводится пример использования команды упаковки архива:

```
user@linux$ tar -cf moi_dokumenti.tar Docs project.tex
```

Распаковка содержимого архива в текущий каталог выполняется следующей командой:

```
tar -xf [имя файла архива]
```

Далее приводится пример использования команды распаковки архива:

```
user@linux$ tar -xf moi_dokumenti.tar
```

Для сжатия файлов используются специальные программы сжатия: «gzip», «bzip2» и «7z».

### 3.2.1.7. Сравнение файлов

#### 3.2.1.7.1. Команда `cmp`

Команда `cmp` сравнивает два файла, и если они различаются, сообщает о первом байте и строке, где было обнаружено различие.

Синтаксис:

```
cmp OPTIONS... FROM-FILE [TO-FILE]
```

Если команды после своего выполнения возвращает «0» – различий не найдено, если «1» – найдены, если «2» – обнаружена ошибка.

Доступные опции команды `cmp` приведены в таблице (Таблица 11).

Таблица 11 – Опции команды `cmp`

Опция	Описание
-c	Печать различающихся символов
-l	Печать смещения (десятичного) и значения (восьмеричного) всех различающихся байтов
-s	Возвращение выходного статуса, показывающего отличаются ли файлы

#### 3.2.1.7.2. Команда `diff`

Команда `diff` сравнивает содержимое двух файлов и возвращает в стандартный выходной поток список изменений, необходимых для преобразования первого файла во второй. Если различий не найдено, команда `diff` ничего не возвращает.

Также команда `diff` может сравнивать каталоги.

Следующий пример использования демонстрирует использование команды `diff` для сравнения двух каталогов «directory1» и «directory»:

```
diff directory1 directory
```

### 3.2.1.8. Мониторинг и управление процессами

#### 3.2.1.8.1. Команда ps

Команда ps отображает список текущих процессов. Колонка команд указывает имя процесса, колонка PID (идентификатор процесса) – номер процесса (этот номер используется, для операций с процессом).

Для просмотра всех запущенных процессов в системе используется ключ «-A».

#### 3.2.1.8.2. Команды kill и killall

Команда kill принудительно завершает процесс с указанным идентификатором. Команда killall завершает все процессы с данным именем.

Команда kill обладает следующим синтаксисом:

```
kill [идентификатор процесса]
killall [имя процесса]
```

Далее приводится пример использования kill:

```
kill 6583
killall httpd
```

Команда kill посылает сигнал KILL процессу, в результате чего процесс должен этот сигнал получить и соответственно повести себя. По умолчанию процессу посылается сигнал TERM, однако это не всегда помогает завершить процесс принудительно. В этих случаях имеет смысл использовать команду kill с ключом «-9».

Пример использования:

```
# kill -9 2395
```

### 3.2.2. Текстовый редактор Vi

Текстовый редактор – это программа, которая предназначена для редактирования (составления и изменения) файлов, содержащих только текст.

Vi (visual editor) является стандартным текстовым редактором Unix для редактирования текста.

#### 3.2.2.1. Запуск vi

Vi может быть запущен из командной строки разными способами:

1) запуск в командном режиме:

```
vi
```

при этом будет запущен vi с пустым буфером;

2) запуск для существующего файла:

```
vi filename
```

где filename – имя файла, который надо редактировать;

3) запуск на определенной строке файла:

```
vi +47 /usr/src/linux/init/main.c
```

vi запускается на 47-й строке /usr/src/linux/init/main.c. При этом vi покажет на экране указанный файл и поместит курсор на заданной строке. В случае, если указана строка за пределами конца файла, vi поместит курсор на последней строке.

#### 3.2.2.2. Режимы работы vi

vi работает в различных режимах, которые используются для выполнения различных задач:

- «Командный режим» – выполнение различных команд для работы с текстом, перемещения по файлу, сохранения, выхода и изменения режимов;
- «Режим ввода текста» – вставка и замена текста;
- «Режим строчного редактора» – используется для управления файлами.

#### 3.2.2.3. Открытие/создание файла

Для открытия или создания нового файла в командном режиме необходимо ввести команду:

```
:e filename
```

Для сохранения файлов используется ряд команд.

Следующая команда сохраняет файл с существующим именем:

```
:w
```

Следующая команда сохраняет файл с заданным именем:

```
:sav filename
```

В случае попытки выполнить запись в файл, владельцем которого является другой пользователь, операция сохранения не может выполняться, о чем будет выдано соответствующее предупреждение.

#### 3.2.2.4. Перемещение по файлу

Перемещение по файлу происходит с помощью клавиш со стрелками (в случае правильного описания терминала) либо с помощью следующих клавиш:

- <h> – перемещение на позицию влево;
- <j> – перемещение вниз;
- <k> – перемещение вверх;
- <l> – перемещение на позицию вправо;
- <Ctrl>+<F> – перемещение на страницу вниз;
- <Ctrl>+<B> – перемещение на страницу вверх.

Также можно использовать следующие быстрые клавиши:

<O> – перемещение в начало текущей строки;

<\$> – перемещение в конец текущей строки;

<w> – перемещение на слово вправо;

<b> – перемещение на слово влево;

<g> – перемещение в начало файла;

<G> – перемещение в конец файла.

### 3.2.2.5. Редактирование файла

Для редактирования текста в файле необходимо перейти в режим ввода. Чтобы перейти из командного режима в режим ввода текста необходимо воспользоваться командой:

i

после чего можно приступать к вводу текста.

Для возврата к командному режиму нужно использовать клавишу <ESC>.

При редактировании файла допускается использовать следующие команды:

- R – замена текста под курсором;
- I – переход в режим ввода с начала текущей строки;
- o – переход в режим ввода с новой строки под курсором;
- O – переход в режим ввода с новой строки над курсором;
- a – переход в режим ввода после курсора;
- x – стирание символа под курсором;
- X – стирание символа перед курсором;
- dd – стирание текущей строки;
- dw – удаление слова, на которое указывает курсор;
- d<число>d – стирание числа строк начиная с текущей;
- yy – копирование текущей строки в неименованный буфер;
- y<число>y – копирование числа строк начиная с текущей в неименованный буфер;
- p – вставка строки из неименованного буфера под курсор;
- P – вставка строки из неименованного буфера над курсором;
- J – слияние текущей строки со следующей;
- u – отмена последней команды;
- . – повтор последней команды.

### 3.2.2.6. Сохранение и выход

При работе в режиме ввода необходимо предварительно нажать <ESC> для перехода в командный режим.

Для выхода из редактора vi без сохранения изменений необходимо воспользоваться следующей командой:

q!

Для выхода из редактора с сохранением изменений, сделанных в файле, используется следующей команда:

:wq

Чтобы сохранить файл, но не выходить из редактора vi, используется следующей команда:

:w

### 3.2.3. Редактор Vim

Редактор Vim – свободный режимный текстовый редактор, созданный на основе более старого vi. Одна из главных особенностей редактора – применение двух основных, вручную переключаемых, режимов ввода: командного (после запуска редактор находится в нём) и текстового (режим непосредственного редактирования текста).

#### 3.2.3.1. Режимы работы

В Vim существуют четыре основных режима работы:

- основной;
- режим непосредственного редактирования текста;
- режим командной строки;
- визуальный режим.

По умолчанию Vim начинает свою работу в основном режиме, который также называют командным. Нажатие клавиш в этом режиме воспринимается как команды (копирования, удаления, перемещения текста и других команд).

Основной режим предназначен для просмотра файлов, ввода команд и перехода из него в другие режимы. Из любого режима в командный режим можно перейти, нажав (в некоторых случаях дважды) клавишу <Esc>.

При нажатии клавиши <:> происходит переход в режим командной строки Vim, в которой можно вводить команды. За двоеточием следует сложная команда (например, поиска или замены), которая после ввода передаётся на исполнение нажатием клавиши <Enter>. После выполнения команды редактор возвращается в нормальный режим. К этому режиму также относятся команды поиска дальше по тексту </>, поиск назад по тексту <(?)> и команда-фильтр <!> для передачи данных внешнему фильтру.

Другие примеры команд:

- команда выхода quit (Vim принимает сокращения, поэтому можно ввести просто q);
- команда сохранения write (или w), параметром которой может быть имя файла;
- команда вызова справки help (или h).

Для перехода из командного режима в режим непосредственного редактирования текста можно нажать клавишу <i> (для начала вставки текста на месте курсора) или клавишу <a> (для начала вставки текста после курсора).

В этом режиме по умолчанию набранные символы не воспринимаются как команды, а вставляются в существующий текст. Однако даже в этом режиме можно задать особые действия редактора, выполняемые при нажатии определённых клавиш, или их сочетаний.

Для вставки, удаления, изменения и автодополнения текста могут использоваться клавиатурные сочетания вида <Ctrl>+<R>.

Визуальный режим предназначен в первую очередь для выделения блоков текста. Для перехода используются следующие сочетания клавиш:

- <v> – для посимвольного выделения текста;
- <Shift>+<v> – для построчного выделения текста;
- <Ctrl>+<v> – для блочного выделения текста.

К выделенным фрагментам текста затем можно применить команды нормального режима (например, удаление выделенного текста или его замена).

### 3.2.3.2. Основные возможности

Перечисленные ниже команды вводятся в основном режиме. Все они имеют команднострочные аналоги и могут быть легко переопределены.

#### 3.2.3.2.1. Переходы

Для перехода на строку с номером n необходимо воспользоваться командой:

G

Для перехода к началу текста необходимо воспользоваться командой:

1G

Для перехода к концу текста необходимо воспользоваться командой:

§G

Для перехода на n символов в нужную вам сторону можно использовать клавиши со стрелками.

#### 3.2.3.2.2. Метки

Используются для отметки позиции и быстрого к ней перехода. Метки нижнего регистра действительны в пределах данного файла, метки верхнего регистра действуют во всех открытых файлах. Список всех меток можно получить следующей командой:

marks

#### 3.2.3.2.3. Сессии

При ведении группы проектов нередко желательно сохранить текущее состояние и настройки редактора, чтобы в дальнейшем продолжить работу с того же места. Для этого предназначены сессии, которые можно создать следующей командой:

:mksession /path/to/Session.vim

Сессии читаются следующей командой:

```
:so /path/to/Session.vim
```

Сохранение текущего контекста (например, положение курсора в тексте) выполняется следующей командой:

```
:mkview
```

Для открытия сохраненного состояния используется следующая команда:

```
:loadview
```

### 3.2.4. Служба xinetd

Служба xinetd запускает процессы, которые предоставляют различные сервисы Интернет. В отличие от сервисов, которые запускаются во время инициализации системы и находятся в режиме ожидания запросов, xinetd представляет собой только один процесс, который прослушивает все порты сервисов, перечисленных в файле конфигурации xinetd.conf. При поступлении запроса производит xinetd запуск соответствующего сервера.

Сервисы, перечисленные в конфигурационном файле xinetd, можно разделить на две группы. Сервисы из первой группы называются multi-threaded (многопоточными) и они требуют разветвления нового серверного процесса для каждого нового запроса на соединение. Далее соединением управляет новый сервер. Для таких сервисов xinetd продолжает прослушивать сеть для приема новых запросов, чтобы вызвать новые серверы. Вторая группа включает службы single-threaded (однопоточные), для которых xinetd не будет управлять новыми запросами, пока сервер не завершит свою работу. Службы в этой группе обычно основаны на передаче данных через датаграммы (UDP).

Служба xinetd применяется в основном для того, чтобы сохранить системные ресурсы через недопущение разветвления огромного числа процессов, которые могут бездействовать в течение большей части времени своей работы. В то же время, выполняя эту функцию, xinetd работает согласно идее суперсервера, предоставляя такие возможности, как управление доступом и протоколирование. Кроме того, xinetd не ограничена сервисами, перечисленными в файле /etc/services, поэтому данная служба может использоваться для запуска сервисов специального назначения.

#### 3.2.4.1. Опции xinetd

Доступные опции xinetd приведены в таблице (Таблица 12).

Таблица 12 – Опции службы xinetd

Опция	Описание
-d	Активирует режим отладки, при котором появляется большое количество отладочных сообщений, дающих возможность использовать отладчик на xinetd
-syslog syslog_facility	Опция разрешает протоколирование создаваемых xinetd сообщений через syslog с заданным syslog facility (данная опция неэффективна в режиме отладки, так как все необходимые сообщения отправляются на терминал)



Опция	Описание
-filelog файл_журнала	Сообщения, создаваемые xinetd, будут добавляться в указанный уже существующий файл, а если файл не существует, то он будет создан (данная опция неэффективна в режиме отладки, так как все необходимые сообщения отправляются на терминал)
-f файл_настроек	Задаёт файл, который xinetd использует для настройки, по умолчанию это /etc/xinetd.conf
-pidfile pid_файл	Указывает файл, в который записывается идентификатор процесса (данная опция неэффективна в режиме отладки)
-dontfork	Дает команду xinetd оставаться в интерактивном режиме, вместо отключения от терминала (эта опция автоматически устанавливает опцию – stayalive)
-stayalive	Дает команду xinetd оставаться запущенной, даже если не задано никаких служб
-remlock	Удаляет файл блокировки (по умолчанию /var/lock/subsys/xinetd) при выходе
-limit proc_limit	Устанавливает ограничение на количество одновременно запущенных процессов, которые может запустить xinetd (предназначена для предотвращения переполнения таблицы процессов)
-logprocs limit	Устанавливает ограничение на количество одновременно запущенных серверов на один идентификатор удаленного пользователя
-version	Предоставляет информацию версии xinetd
-inetd_compat	Заставляет xinetd считывать /etc/inetd.conf в дополнение к стандартным конфигурационным файлам (/etc/inetd.conf будет прочитан после стандартных конфигурационных файлов xinetd)
-cc interval	Выполнение периодических проверок внутреннего состояния xinetd с периодичностью, заданной параметром interval (в секундах)
Примечание. Опции syslog и filelog являются взаимно исключаящими. Если ни одна из них не задана, то по умолчанию используется syslog с daemon facility.	

### 3.2.4.2. Управление xinetd

При получении определенных сигналов xinetd выполняет определенные действия. Эти действия зависят от заданных сигналов и могут быть переопределены путем правки файла config.h и перекомпиляции.

#### 3.2.4.2.1. Сигнал SIGHUP

Заставляет выполнить жесткую перенастройку, которая означает, что xinetd перечитает файл с настройками и завершит работу серверов для тех служб, которые больше не доступны. Управление доступом выполняется снова на уже запущенные серверы через проверку удаленных подключений, времени доступа и копий серверов. Если количество копий серверов уменьшается, то некоторые произвольно выбранные серверы будут остановлены, чтобы соблюсти ограничение (это случится после завершения работы тех серверов, которые попадают под ограничение доступа с удаленных адресов или ограничение времени доступа). Также, если флаг INTERCEPT был сброшен и происходит его установка, то будет завершена работа любых запущенных серверов для служб с этим флагом.

#### 3.2.4.2.2. Сигнал SIGQUIT

Приводит к завершению работы.

### 3.2.4.2.3. Сигнал SIGTERM

Завершает работу всех запущенных серверов перед завершением работы xinetd.

### 3.2.4.2.4. Сигнал SIGHUP

Приводит к снятию образа внутреннего состояния. По умолчанию имя файла образа внутреннего состояния – /var/run/xinetd.dump, а чтобы изменить данное имя файла нужна правка config.h и перекомпиляция).

### 3.2.4.2.5. Сигнал SIGIOT

Производит внутреннюю проверку структуры данных, используемых программой, на повреждения. После завершения проверки xinetd сгенерирует сообщение об успешности или неуспешности проверки.

При перенастройке файлы журналов закрываются и открываются заново. Это позволяет удалить старые файлы журналов.

## 3.2.5. Служба Crontab

Crontab – служба таблиц, управляющих работой службы cron. Crontab управляет доступом пользователя к службе cron путем копирования, создания, выдачи содержимого и удаления файлов crontab, таблиц заданий.

При вызове без опций crontab копирует указанный файл или стандартный входной поток (если файл не указан) в каталог, в котором хранятся пользовательские таблицы заданий cron.

Для создания, изменения и удаления файлов cron следует использовать специальную утилиту crontab.

### 3.2.5.1. Формат записи файла crontab

Файл crontab состоит из строк, содержащих шесть полей. Поля разделяются пробелами или символами табуляции. Первые пять полей – целочисленные шаблоны, задающие:

- минуту (со значениями от «0» до «59»);
- час (со значениями от «0» до «23»);
- день месяца (со значениями от «1» до «31»);
- месяц года (со значениями от «1» до «12»);
- день недели (со значениями от «0» до «6», где «0» – воскресенье).

Дни можно указывать в двух полях (день месяца и день недели). Оба поля учитываются, если заданы в виде списка элементов.

Для задания полей «месяц» и «день\_недели» можно использовать имена. Диапазоны или списки имён не разрешены.

Допустимо указание нескольких значений (и диапазонов через тире) через запятую. Диапазон указывается как два числа, разделенных дефисом. Указываемые числа включаются в диапазон.

Символ звездочка («\*») без шага соответствует полному диапазону значений.

Шестое поле в строке файла crontab – строка, выполняемая командным интерпретатором в указанные моменты времени. Символ «%» в этом поле, если он не замаскирован символом «\», преобразуется в символ новой строки. Только первая строка (до символа «%» или до конца строки) поля команды выполняется командным интерпретатором. Другие строки передаются команде как стандартный входной поток. Любая строка, начинающаяся символом «#», считается комментарием и игнорируется. Файл не должен содержать пустых строк.

Служба cron запускает команды, когда значения полей «минута», «час», «месяц» и хотя бы одно из полей «число» и «день\_недели» совпадают с текущим временем. Служба cron сверяет директивы с текущим временем раз в минуту.

Командный интерпретатор вызывается из начального каталога соответствующего пользователя с аргументом arg0, равным sh. Пользователи, желающие, чтобы был выполнен их файл .profile, должны явно это указать в команде в файле crontab. Служба cron предоставляет каждому командному интерпретатору стандартную среду, задавая переменные HOME, LOGNAME, SHELL(=/bin/sh), TZ и PATH. Стандартное значение переменной PATH для пользовательских заданий cron – /usr/bin, а для заданий cron пользователя root – /usr/sbin:/usr/bin. Стандартное значение переменной PATH можно задать в файле /etc/default/cron; см. cron(1M).

Если стандартный выходной поток и стандартный поток ошибок команд не перенаправлены, любые сгенерированные результаты или сообщения об ошибках будут отправлены пользователю по электронной почте.

### 3.2.5.2. Опции

Доступные опции crontab приведены в таблице (Таблица 13).

Таблица 13 – Опции службы crontab

Опция	Описание
-e	Позволяет редактировать копию файла crontab текущего пользователя или создает пустой файл для редактирования, если соответствующего файла crontab не существует. Когда редактирование завершается, файл устанавливается в качестве пользовательского файла crontab. Если задано имя пользователя, то редактируется файл crontab заданного, а не текущего пользователя. Переменная среды EDITOR задает редактор, вызываемый при указании опции «-e»
-l	Показывает файл crontab вызвавшего пользователя
-r	Удаляет пользовательский файл crontab из каталога заданий cron

### 3.2.6. Служба передачи файлов FTP

В ОС «Альт Линукс» передача файлов обеспечивается с помощью программы lftp. Данная команда реализует протокол передачи файлов FTP. Для копирования файлов необходимо знать имя и пароль пользователя, которому принадлежат файлы на сервере службы FTP.

Для запуска lftp необходимо в консоли ввести команду:

lftp

После появления приглашения lftp :~> становятся доступными для использования внутренние команды lftp.

Основные внутренние команды lftp:

- open – подключение к серверу;
- user – идентификация при удаленном подключении;
- close – отключение от сервера;
- ls – просмотр списка файлов;
- lcd – смена локального каталога;
- mkdir – создание нового каталога;
- lpwd – просмотр имени каталога на локальном компьютере;
- get – копирование файла с сервера;
- put – копирование файла на сервер;
- help – просмотр списка доступных команд и справки по ним;
- exit – выход из lftp.

### 3.2.7. Защищенный интерпретатор команд SSH

Защищенный интерпретатор команд SSH – клиент-серверная система для организации защищенных туннелей для удаленного доступа к другим компьютерам.

SSH реализует соединение с удаленным компьютером, которое позволяет защититься от следующих угроз:

- прослушивание данных, передаваемых по этому соединению;
- манипулирование данными на пути от клиента к серверу;
- подмена клиента либо сервера путем манипулирования IP-адресами, DNS либо маршрутизацией.

Для создания защищенного туннеля используется программа ssh. Инициировать соединение с сервером можно командой:

```
ssh <имя_клиента>@IP_addr
```

где IP\_addr – IP-адрес компьютера с запущенной службой sshd.

При использовании идентификации по паролю на сервере должна существовать учетная запись с указанным именем клиента.

Параметры, относящиеся к способу аутентификации, а так же все прочие настройки ssh указываются в конфигурационном файле /etc/ssh/ssh\_config.

Конфигурационные файлы разбиты на разделы, установки которых относятся к отдельному компьютеру, группе компьютеров или ко всем компьютерам, при этом установки разных разделов могут конфликтовать друг с другом. Предпочтение в данном случае будет отдаваться тому параметру, который указан раньше.

### 3.3. Работа со справочной информацией

В состав ОС «Альт Линукс» входят справочные страницы, предназначенные для представления подробной информации по функциям ОС и командам, которые позволяют работать с функциями ОС, а также по конфигурационным файлам или библиотекам.

Для получения подробной справочной информации об интересующей программе или команде оператору необходимо вызвать соответствующую справочную страницу. Вызов справочных страниц осуществляется с консоли с помощью команды `man` в следующем формате:

```
man command_name
```

где `command_name` – имя нужной программы или команды.

По умолчанию после вывода первой найденной справочной страницы команда `man` завершит свою работу. Для изменения работы команды `man` могут применяться дополнительные опции.

#### 3.3.1. Наиболее часто используемые опции команды `man`

Доступные опции `man` приведены в таблице (Таблица 14).

Таблица 14 – Опции команды `man`

Опция	Описание
-a	Выводит все доступные для запрашиваемой программы или команды справочные страницы. Например: <code>man -a command_name</code>
-f	Ищет и выводит краткое описание всех справочных страниц, где имеются ссылки на запрашиваемую программу или команду. Например: <code>man -f command_name</code>
--warning	Включает предупреждения
-I	Включает чувствительность к регистру
-h	Выводит справку по опциям командной строки и завершает работу
-H	Активирует вывод в HTML и просмотр в браузере, который определен в <code>\$BROWSER</code> или определен по умолчанию во время компиляции. Например: <code>man -H[browser]</code>
Примечание. Для получения полной справочной информации по использованию команды <code>man</code> и ее опциям необходимо выполнить команду « <code>man man</code> ».	

#### 3.3.2. Управление справочными страницами

Так как большинство справочных страниц состоят более чем из одной страницы, для их вывода на экран используется команда `less`, которая позволяет перелистывать страницы с помощью клавиш `<PgUp>` и `<PgDown>`, или `<Space>` и `<Esc> + <V>`.

Выйти из справочной страницы можно в любое время с помощью клавиши `<q>`.

Поиск в справочной странице вызывается с помощью клавиши `</>`, каждое последующее найденное вхождение можно просмотреть с помощью клавиши `<n>`, а вернуться к предыдущему вхождению можно с помощью комбинации клавиш `<Shift>+ <n>`.

Справочные страницы команд можно находить по категориям, к которым они принадлежат. Для этого используется команда `argoros`, где цифрами можно задать категорию, к которой принадлежит команда.

Категории команд:

- «Команда 1» – исполняемые программы и команды оболочки;
- «Команда 2» – системные вызовы;
- «Команда 3» – библиотечные вызовы;
- «Команда 4» – файлы устройств (обычно расположены в `/dev`);
- «Команда 5» – форматы файлов;
- «Команда 6» – игры;
- «Команда 7» – макропакеты и соглашения;
- «Команда 8» – программы системного администрирования;
- «Команда 9» – процедуры ядра.

На справочные страницы принято ссылаться по имени, с указанием номера категории в скобках. Часто существуют сразу несколько справочных страниц с одинаковыми именами, но в разных категориях, например `man(1)` и `man(7)`. В таком случае, команде `man` необходимо передать номер конкретной категории перед именем справочной страницы, например:

```
man 5 passwd
```

отобразит справочную страницу по файлу `/etc/passwd` вместо утилиты `passwd`.

### 3.3.3. Печать справочных страниц

Если необходимо распечатать справочную страницу, то предварительно нужно удалить форматирование страницы с помощью команды `col`.

Например, чтобы напечатать страницу руководства по `man`, необходимо выполнить следующую команду:

```
man man | col -b | lpr
```

Команда, приведенная выше, пропускает выводимую информацию через фильтр `col`, который форматирует текст для принтера. Затем данные, выводимые `col`, посылаются на принтер.

## 3.4. Завершение работы ОС

Для корректного завершения работы ОС (перезагрузки) во время ее работы запрещается выключать питание компьютера или перезагружать компьютер нажатием на кнопку «Reset», так как для корректного завершения работы требуется размонтирование файловой системы.

Перед окончанием работы с ОС необходимо завершить все работающие программы.

Для завершения работы ОС можно воспользоваться несколькими различными способами остановки системы:

- нажать комбинацию клавиш <Ctrl>+<Alt>+<Del>, что на рабочей станции приведет к вызову диалога завершения работы системы, а на сервере – к перезагрузке системы, при этом необходимо дождаться появления на экране сообщения «Reboot» (перезагрузка) и выключить питание системы;
- воспользоваться специальной командой shutdown, доступной пользователю с правами root (суперпользователь);
- при наличии графической оболочки следует воспользоваться диалогом доступным в меню «Система».

### 3.4.1. Завершение работы ОС с помощью консоли

Для завершения работы ОС в консольном режиме рекомендуется использовать команду shutdown, вызывающую остановку, выключение и (или) перезагрузку ПЭВМ.

Синтаксис:

```
shutdown [опции...] [время] [сообщение...]
```

Доступные опции shutdown приведены в таблице (Таблица 15).

Таблица 15 – Опции команды shutdown

Опция	Описание
--help	Краткая справка по использованию команды
-H, --halt	Остановка компьютера
-P, --poweroff	Выключение компьютера (по умолчанию)
-r, --reboot	Перезагрузка компьютера
-h	То же что и --poweroff, если не указана опция --halt
-k	Не останавливать, не выключать и не перезагружать компьютер, а просто напечатать сообщение
--no-wall	Не печатать сообщение перед остановкой/выключением/перезагрузкой
-c	Отменить запланированное выключение. Эта опция может быть использована для отмены эффекта ранее выполненной команды «shutdown» с указанием времени отличным от «+0» или «now»

Аргумент [время] может быть строкой времени (в том числе, используется значение «now»). Время может быть указано в формате «чч:мм» для задания часа и минуты в формате 24 часа.

Также допускается использовать синтаксис «+м», указывая количество минут от текущего момента. «now» является синонимом для «+0» и может быть использовано для немедленного выключения компьютера. Значение времени по умолчанию (если не задано явно) считается «+1». Необходимо учитывать, что для задания сообщения также потребуется явно указать время. Если задана строка времени, то за пять минут до выключения системы будет создан файл /run/nologin для того чтобы гарантировать, что никакие новые пользователи не войдут в систему.

После указания времени допускается указать аргумент [сообщение] и ввести текстовую строку, которая будет напечатана во всех терминальных сессиях.

### 3.4.2. Завершение работы ОС с помощью инструментов графической оболочки

Для завершения работы ОС в графическом режиме необходимо перейти в меню «Система» и выбрать пункт «Выключить...» (Рис. 9).

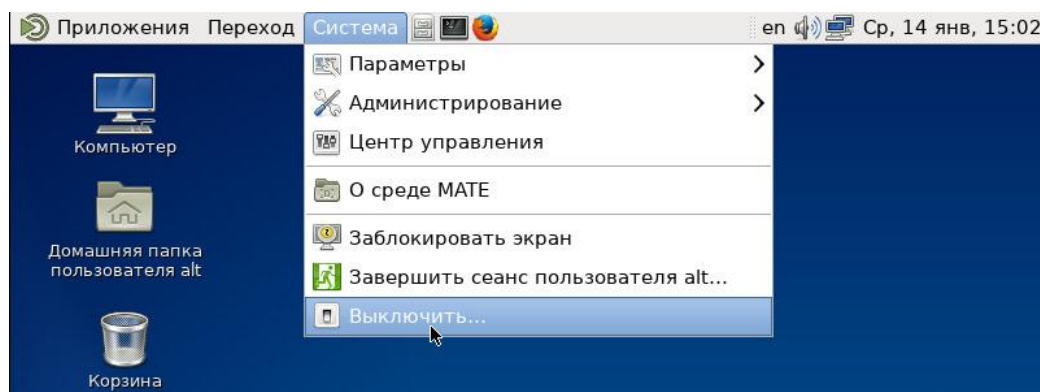


Рис. 9

Далее откроется окно, в котором предоставляется выбор дальнейших действий (Рис. 10).

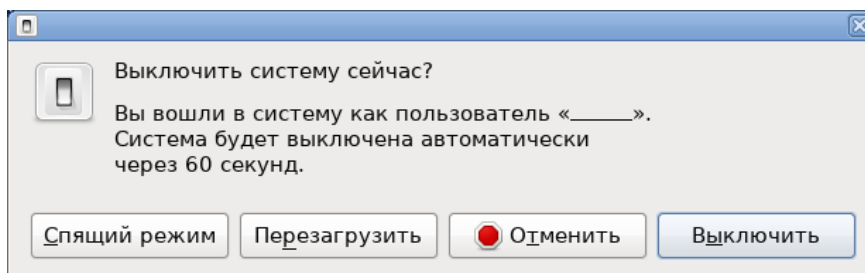


Рис. 10

После выбора действий и нажатия кнопки ОС перейдет в спящий режим, выполнит перезагрузку или выключение ПЭВМ соответственно.



#### 4. СООБЩЕНИЯ ОПЕРАТОРУ

##### 4.1. Первичная идентификация

При первом входе в систему и отсутствии выбранных предпочтительных учетных записей откроется окно предварительного выбора учетной записи (Рис. 11).

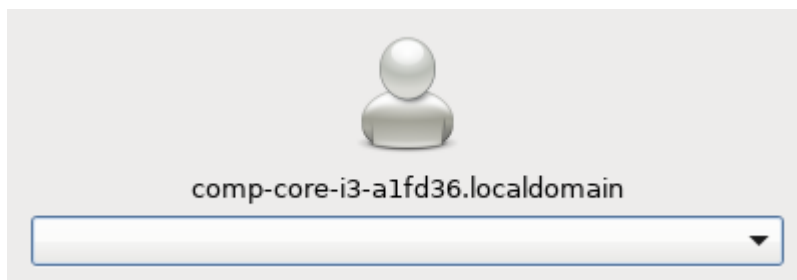


Рис. 11

##### 4.2. Предупреждение о неуспешных идентификации и аутентификации

В случае неуспешного прохождения процедуры идентификации и аутентификации при входе в ОС в графическом режиме будет выдано предупреждение (Рис. 12).

**Неверный пароль, попробуйте ввести ещё раз**

Рис. 12

В режиме консоли будет выдано предупреждение (Рис. 13).

**Login incorrect**

Рис. 13

##### 4.3. Оповещение пользователя после успешного входа

После успешного входа в информационную систему в режиме консоли на экране на какое-то время будет отображена информация о времени предыдущего входа в систему (Рис. 14).

**Last login: Wed Apr 20 11:23:22 MSK 2016 from localhost on tty1**

Рис. 14

**ПЕРЕЧЕНЬ СОКРАЩЕНИЙ**

DNS	– Domain Name System (система доменных имён);
FTP	– File Transfer Protocol (протокол передачи файлов);
HTML	– HyperText Markup Language (язык гипертекстовой разметки);
SSH	– Secure Shell (сетевой протокол прикладного уровня, позволяющий производить удаленное управление операционной системой);
UDP	– User Datagram Protocol (протокол пользовательских датаграмм);
КСЗ	– комплекс средств защиты;
ОС	– операционная система;
ПЭВМ	– персональная электронная вычислительная машина;
РД	– руководящий документ.

*Лист регистрации изменений*

Изм.	Номера листов (страниц)				Всего листов (страниц) в документе	№ документа	Входящий № сопроводительного документа и дата	Подпись	Дата
	измененных	замененных	новых	аннулированных					
1	-	Все	-	-	35	КШДС.420-2017	-		20.01.2017
2	-	2, 4	-	-	35	КШДС.421-2017	-		10.06.2017