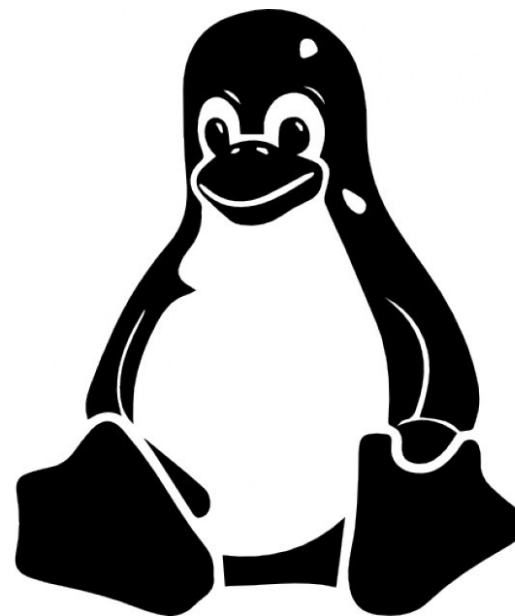


Команды для Linux терминала
Оболочка Bash



Разраб: Трунин Е.Е.

v 0.1

Краткий конспект книги С.Граннемана Необходимый код и команды. Карманный справочник Linux.

Содержание

Глава I — Простейшие команды

Глава II — Объединение команд

Глава III — Потоки ввода-вывода

Глава IV — Отображение содержимого файлов

Глава V — Вывод на печать

Глава VI — Владельцы файлов и права доступа

Глава VII — Создание архивов и сжатие данных

Глава VIII — Поиск данных

Глава IX — Работа с оболочкой

Глава X — Контроль использования системных ресурсов

Глава XI — Инсталяция программного обеспечения

Глава XII — Сетевое взаимодействие

Глава XIII — Работа в сети

///
///Простейшие//команды///
///Глава/I//

///LS//

ls [путь_к_каталогу] = выводит список файлов и каталогов в данной директории

Параметры :

- R = Просмотр содержимого подкаталогов
- l = Вывод содержимого каталога в один столбец
- m = Вывод содержимого Kаpfkjuf с запятыми в качестве разделителей
- a = Отображение скрытых файлов и каталогов
- F = Отображение информации о типах файлов
- l = Информации о правах доступа и владельцах файлов
- r = ВЫВОД информации В обратном порядке
- x = сортировка содержимого каталога по суффиксам имен файлов
- t = сортировка по дате и времени
- s = сортировка содержимого по размеру файлов
- h = предоставление размеров файлов в килобайтах, мегабайтах или гигабайтах

///PWD//

pwd = определение пути к текущему каталогу

///CD//

cd [путь_к_каталогу] = переход к другому каталогу

Параметры :

- (~) = переход в рабочий каталог (cd ~)
- (-) = переход к предыдущему каталогу

/////////////////////////////////TOUCH/////////////////////////////////

touch [имя_файла_для_изменения_времени] = изменение сведений о времени файла

Параметры :

-a = изменение времени доступа
-m = изменение времени модификации
-t [[CC]YY]MMDDhhmm[.ss]= установка произвольного времени для файла (touch -t 197002160701 name.log)

/////////////////////////////////MKDIR/////////////////////////////////

mkdir [имя_нового_каталога] = создание нового каталога

Параметры :

-p = создание нового каталога и необходимых подкаталогов (mkdir -p pictures/personal/family)
-v = информация о действиях выполняемых командой mkdir

/////////////////////////////////CP/////////////////////////////////

cp [имя_файла_для_копирования] [имя_нового_файла]
= копирование файлов

Параметры :

-v = вывод подробной информации о копировании файла
-i = предотвращение копирования файлов поверх существующих
-R = копирование каталогов
-a = создание резервных копий

/////////////////////////////////RM/////////////////////////////////

rm [имя_файла_для_удаления] = удаление файлов

Параметры :

-v = вывод подробной информации при удалении файлов
-i = предотвращение удаления важных файлов

-l[уровень_глубины] = указывает глубину рекурсивного поиска файлов
-pr = запрещает осуществлять рекурсивный поиск в родительском каталоге (26)

-w [значение_задержки]= задает небольшую задержку перед копированием каждого файла

-A.[формат_фалов] = указывает, что необходимо скачивание только файлов определенного формата

-R.[формат_фалов] = указываем тип файлов, которые не нужно копировать

-c = возобновление прерванного копирования

-E = при копировании WEB узла преобразует каждую страницу в html файл

-k = при копировании WEB узла указывает, что надо преобразовывать ссылки в состав WEB-страницы

-p = при копировании WEB узла позволяет корректно отображать скачанный узел, соединяя страницу со стилями и скриптами

culpr = аналог wget, главным отличием которого является возможность указывать последовательность и множества файлов для копирования

```
-ssh-keygen -t dsa
-ssh-copy-id -i ~/.ssh/id_dsa.pub
[имя_учетной_записи]@[IP_адрес_машины]
```

////////////////////////////////SFTP////////////////////////////////

sftp [имя_учетной_записи]@[IP_адрес_компьютера] =
организация защищенного взаимодействия с другим
компьютером средствами SFTP

Основные команды SFTP :

```
-cd = перейти в другой каталог
-exit = закрыть соединение с удаленным SSH сервером
-get = скопировать указанный файл на локальную машину
-help = отобразить справочную информацию
-lcd = перейти в другой каталог на локальной машине
-ls = отобразить список файлов на локальной машине
-ls = отобразить список файлов в данном каталоге удаленной
машины
-put = скопировать указанный файл на удаленный сервер
-rm = удалить указанный файл с удаленного сервера
-scp [имя_учетной_записи1]@[IP_адрес_машины1]:файл_1
[имя_учетной_записи2]@[IP_адрес_машины2]: файл_2 =
защищенное копирование файлов между машинами
-rsync -v = создание защищенной и удаленной резервной
копии
```

////////////////////////////////WGET////////////////////////////////

wget [параметры] [ссылка_на_источник] = копирование
файлов из WEB

Параметры :

-r = задает следование по ссылкам и просмотр каталогов в поисках файлов
-Rf = удаление файлов и каталогов содержащих данные

////////////////////////////////MV////////////////////////////////

mv [путь_и_имя_файла_копирования]
[путь_куда_копируем_и_имя_файла]= перемещение и переименование
файлов

////////////////////////////////RMDIR////////////////////////////////

rmdir [имя_пустого_каталога] = удаление пустого каталога

////////////////////////////////SU////////////////////////////////

su [имя_пользователя] = позволяет выполнять действия от имени
другого пользователя

Параметры :

-l = использование не только имени другого пользователя, но и его
переменное окружение

////////////////////////////////WHOAMI////////////////////////////////

whoami = сообщает имя пользователя с точки зрения оболочки

////////////////////////////////MAN////////////////////////////////

man [название_команды] = получение информации о команде

Параметры :

-k = поиск команды по выполняемым ею действиям
-f = получение кратких сведений о команде
-u = формирование базы данных команд
-e = вывести справочную информацию на печать

////////////////////////////////INFO////////////////////////////////

info [название_команды] = получение информации о команде,
организованная в виде разделов

/////////////////////////////////WHEREIS////////////////////////////////

whereis [название_команды_или_файла] = определяет путь к исполняемым, исходным файлам и страницам справочного руководства

Параметры :

- b = путь только к исполняемым файлам
- m = путь лишь к страницам справочного руководства
- s = путь к исходным файлам

/////////////////////////////////WHATIS////////////////////////////////

whatis [название_команды] = выводит сведения о команде (аналог man -f)

Параметры :

- w = поиск с применением групповых операций
- r = для работы с регуляторными выражениями

/////////////////////////////////APROPOS////////////////////////////////

argpros [название_команды] = поиск информации о команде по выполняемым ею действиям (аналог man -k)

Параметры :

- w = поиск с применением групповых операций
- r = для работы с регуляторными выражениями
- e = для точного совпадения слова или фразы при поиске

/////////////////////////////////WHICH////////////////////////////////

which [название_файла] = показывает местоположение файла который будет запущен командой из строки

host [IP_адрес] = поиск доменного имени по IP адресу

/////////////////////////////////IWCONFIG////////////////////////////////

iwconfig = получение информации о состоянии сетевого интерфейса беспроводной сети

/////////////////////////////////IFUP////////////////////////////////

ifup [сетевое_устройство] = активизация сетевого соединения

/////////////////////////////////IFDOWN////////////////////////////////

ifdown [сетевое_устройство] = разъединение сетевого соединения

/////////////////////////////////ROUTE////////////////////////////////

route = отображение таблицы маршрутизации

В данном руководстве представлены лишь, базовые команды. Не забывайте читать о командах в руководстве man.

/////////////////////////////////Работа//в//сети////////////////////////////////

/////////////////////////////////Глава//XIII////////////////////////////////

/////////////////////////////////SSH////////////////////////////////

ssh [имя_учетной_записи]@[IP_адрес_машины] = организация защищенного взаимодействия с другим компьютером средствами SSH

Чтобы установить соединение с другой машиной, необходимо выполнить следующие команды :

-upgrade = обновление установленных пакетов
-clean = удаление ненужных инсталляционных пакетов из системы

apt-cache search [ключевое_слово_поиска] = поиск пакетов доступных для копирования в систему

/////////////////////////////////
/////////////////////////////////Сетевое/взаимодействие/////////////////////////////////
/////////////////////////////////Глава/XII////////////////////////////////

/////////////////////////////////IFCONFIG/////////////////////////////////
ifconfig = определение состояния сетевых интерфейсов

ifconfig [имя_сетевого_устройства] [новый_IP_адрес] =
изменение IP адреса Ethernet карты

ifconfig [имя_сетевого_устройства] -promisc =
настройка сетевой карты на прием всех пакетов в поле зрения

/////////////////////////////////PING////////////////////////////////

ping [адрес_сети] = проверка способности компьютера
принимать запросы

/////////////////////////////////TRACEROUTE////////////////////////////////

tracert [адрес_узла] = контроль прохождения пакета
между двумя узлами

/////////////////////////////////HOST////////////////////////////////

host [доменное_имя] = нахождение IP адреса,
соответствующего данному доменному имени

/////////////////////////////////
/////////////////////////////////Объединение/команд/////////////////////////////////
/////////////////////////////////Глава/II////////////////////////////////

Чтобы выполнить последовательно несколько команд, необходимо
использовать символ (;)

Пр: Команда_1 ; Команда_2 ; Команда_3 ...

При этом вначале выполняется Команда_1, затем последующие.

Чтобы выполнить последовательно несколько команд с проверкой на их
корректность(смогут ли они выполняться), необходимо разделять команды
символом (&&)

Пр: Команда_1 && Команда_2 && Команда_3 ...

При этом команды также выполняются последовательно, но управление
следующей команде переходит только в случае, если предыдущая
завершилась без ошибки.

Чтобы выполнить последовательно несколько команд с проверкой на их
корректность(смогут ли они выполняться), необходимо разделять команды
символом (&&)

Пр: Команда_1 && Команда_2 && Команда_3 ...

При этом команды также выполняются последовательно, но управление
следующей команде переходит только в случае, если предыдущая
завершилась без ошибки.

Чтобы выполнить команду при условии, что предыдущая завершилась
успешно, используется символ (||)

Пр: Команда_1 || Команда_2

При этом Команда_2 выполнится только при условии, что не выпонилась Команда_1

Использование выходных данных одной команды при вызове другой команды осуществляется с помощью символов (\$())

Пр: Команда_1 \$(Команда_2)
Пример : mkdir \$(date "+%Y-%m-%d")

////////////////////////////////////
////////////////////////////////////Потоки//ввода-вывода////////////////////////////////////
////////////////////////////////////Глава//III////////////////////////////////////

Идентифика тор	Название	Сокращение	Используй вание
0	Стандартны й входной поток	stdin	клавиатура
1	Стандартны й выходной поток	stdout	терминал
2	Стандартны й поток ошибок	stderr	терминал

Для передачи выходных данных одной команды на вход другой команды используется символ (|)

Пр: Команда_1 | Команда_2
При этом Команда_1 будет выступать в качестве источника данных, которые передаются на вход Команда_2.
Пример : ls -l | less

-e = удаление программных пакетов
////////////////////////////////////YUM////////////////////////////////////

yum [команда] [название_пакета] = установка программных пакетов с зависимостями.

Параметры :
-install = установка пакета
-remove = удаление пакета
-update = поиск обновлений для управляемых пакетов
-search [ключево_слово] = поиск доступных к установке программ, содержащих ключевое слово в названии или описании.(не рекомендуем)
-list available = вывод списка всех доступных пакетов

////////////////////////////////////DEB////пакеты////////////////////////////////////
////////////////////////////////////DPKG////////////////////////////////////

dpkg [команда] [название_пакета].deb = установка Debian пакетов без зависимостей

Параметры :
-i = установка пакета
-r = удаление пакета

////////////////////////////////////APT////////////////////////////////////

apt-get [команда] [название_пакета] = установка пакетов с зависимостями.

Параметры :
-update = загрузка списка текущих программных пакетов с серверов и поиск новых версий пакетов
-install = устанавливает выбранный пакет
-remove = удаление пакета

//////////////////////////////////FREE//////////////////////////////////

free = отображение информации об оперативной памяти системы

//////////////////////////////////DF//////////////////////////////////

df = отображение информации об использовании дискового пространства

Параметры :

-h = отображение информации в понятной форме

//////////////////////////////////DU//////////////////////////////////

du = отображение размера области дискового пространства, занятого определенным каталогом.

//////////////////////////////////

//////////////////////////////////Инсталляция//программного//обеспечения//////////////////////////////////

//////////////////////////////////Глава//XI//////////////////////////////////

//////////////////////////////////RPM//пакеты//////////////////////////////////

//////////////////////////////////RPM//////////////////////////////////

rpm [u-i] [название_пакета].rpm = установка rpm пакетов(без уставки связей)

Параметры :

-i = инсталляция

-u = обновление до последней версии и инсталляция

-h = отображение хода процесса установки

-v = вывод подробной информации о выполняемых действиях

Для перенаправления выходных данных в файл используется символ (>)

Пр: Команда_1 > имя_файла

При этом выходные данные команды Команда_1 будут записаны в файл имя_файла

Пример : ls -lF > datafiles.txt

Чтобы предотвратить перезапись файла при перенаправлении вывода, нужно использовать команду noclobber

Она вызывается так : set -o noclobber

После её вызова, перенаправление в существующий файл будет производиться только с нашего разрешения.

Чтобы выключить функцию : set +o noclobber

Чтобы перенаправить выходные данные и записать их в конец файла используются символы (>>)

Команда_1 >> имя_файла

При этом выходные данные будут записаны в конец файла, если файла не существует, он будет создан. В отличие от (>) файл не будет каждый раз перезаписываться, а пополняться.

Чтобы использовать содержимое файла в качестве входных данных используется символ (<)

Команда_1 < имя_файла

При этом содержимое файла будет выступать в качестве входных данных для команды Команда_1

//////////////////////////////////

//////////////////////////////////Отображение//содержимого//файлов//////////////////////////////////

//////////////////////////////////Глава//IV//////////////////////////////////

////////////////////////////////CAT////////////////////////////////

cat [имя_файла] = выводит содержимое файла в консоль, а затем возвращает управление оболочке

cat [имя_файла1] [имя_файла2] [имя_файла3 ...] = выводит содержимое сразу нескольких файлов без разделителей.

cat [имя_файла1] [имя_файла2] > [имя_файла3] = объединяет 1 и 2 файл и сохраняет их в файл имя_файла3

Параметры :

-n = выводит содержимое файлов с нумерацией строк

////////////////////////////////LESS////////////////////////////////

less [имя_файла] = организует страничный вывод текста

Имеет множество параметров и комбинаций для внутреннего управления, которые я в данном руководстве рассматривать не буду. Используйте map или info.

////////////////////////////////HEAD////////////////////////////////

head [имя_файла] = просмотр первых десяти строк файла

head [имя_файла1] [имя_файла2] = просмотр первых десяти строк файла1 и файла2

Параметры :

-n [число_строк] = позволяет отображать указанное количество строк файла

-с [число_байтов] = позволяет отобразить указанное количество байтов файла

////////////////////////////////TAIL////////////////////////////////

tail [имя_файла] = просмотр последних десяти строк файла

Поддерживает все функции head.

////////////////////////////////KILL////////////////////////////////

kill [номер_процесса] = завершение выполняющегося процесса. Номер процесса можно найти используя команду ps aux.

Параметры :

-1 == -HUP = прекращение выполнения процесса

-15 == -TERM = "мягкое завершение"

-9 == -KILL = прекратить все выполняющиеся действия и завершить работу

////////////////////////////////TOP////////////////////////////////

top = предоставляет динамические данные о происходящих процессах в реальном времени. Чтобы завершить процесс из top, необходимо нажать клавишу k, затем выбрать номер процесса.

////////////////////////////////LSOF////////////////////////////////

lsof = получение списка открытых файлов. Т.к в UNIX системах все является файлом, данная команда выведет тысячи открытых файлов.

Параметры :

-u [имя_пользователя] = отображение файлов открытых конкретным пользователем.

-с [имя_программы] = отображение сведений о процессах, соответствующих конкретной программе.

lsof [имя_файла] = отображение имени пользователя, открывшего определенный файл

alias [имя_псевдонима] = просмотр команды
определенной данным псевдонимом

alias [псевдоним] = '[команда]' = создание временного
псевдонима и присвоение ему определенной команды.
Псевдоним, созданный таким способом действует
лишь до завершения сеанса работы с оболочкой.

alias [псевдоним] = '[команда]' = для создания
постоянного псевдонима, необходимо вписать такую команду
в файл .bashrc

unalias [псевдоним] = удаление временных
псевдонимов

```
////////////////////////////////////  
/////////Контроль//использования//системных//ресурсов////////  
////////////////////////////////////Глава/X//
```

```
/////////////////////////////////PS////////////////////////////////
```

ps aux = вывод информации о процессах, выполняемых
в системе. В столбце STAT содержится несколько различных
букв. Вот основные из них

(R = выполняется
S = спит
T = остановлен
z = зомби(завис))

ps axjf = просмотр дерева процессов

ps [имя_пользователя] = отображение процессов,
принадлежащих конкретному пользователю

Параметры :

-f = не завершает функцию и постоянно отображает последние строки файла,
позволяя видеть последние изменения.

```
////////////////////////////////////  
/////////Вывод//на//печать////////  
/////////////////////////////////Глава/V////////
```

```
/////////////////////////////////LPSTAT////////////////////////////////
```

lpstat -p = получение списка доступных принтеров
lpstat -d = определение принтера установленного по умолчанию
lpstat -s = определение расположения принтеров и принтера по
умолчанию
lpstat -t = получение полной информации о принтерах

```
/////////////////////////////////LPR////////////////////////////////
```

lpr [имя_файла_на_печать] = выводит указанный файл на печать. Для
распечатки можно использовать только форматы: txt, PDF, PostScript.

lpr -p [имя_принтера] [имя_файла_на_печать] = вывод печати на
указанный принтер

lpr -# [число_копий] [имя_файла_на_печать] = распечатка указанного
числа копий

lprq = получение списка заданий на печать

```
/////////////////////////////////LPRM////////////////////////////////
```

lprm = отмена задания, переданного на принтер по умолчанию

lprm [идентификатор_задания] = отмена задания переданного на произвольный принтер

lprm - = отмена всех заданий на печать

/////////////////////////////////
Владелец//файлов//и//права//доступа/////////////////////////////////
Глава//VI////////////////////////////////

d|rw|x|r-x|r-x| = разделение информации о правах доступа

- 1) обозначение типа файл (- = обычный каталог
 - = исполняемый файл
 - d = каталог
 - l = символьная ссылка
 - s = сокет
 - b = блочное устройство
 - c = символьное устройство
 - p = именованный канал)

- 2) права владельца файла
- 3) права группы
- 4) права любого другого пользователя

Категории пользователей и их обозначения

Категория пользователей	Сокращение
Владелец	u
Группа	g
Прочие пользователи	o

c - специальный файл символьного типа
p — FIFO
s - сокет)

- a = (and) позволяет объединить несколько условий для поиска
- o = (or) отображение результатов всех в отдельности условий поиска
- n = (not) отображение результатов не соответствующих условиям поиска
- exec = выполнение действий над каждым найденным файлом
- fprint [имя_файла_для_вывода_данных]= вывод результатов поиска в файл

/////////////////////////////////
Работа//с//оболочкой/////////////////////////////////
Глава//IX////////////////////////////////

////////////////////////////////HISTORY////////////////////////////////

history = просмотр списка команд введенных ранее. Сохраняются последние 500 команд.

!! = выполнение последней команды

![номер_команды] = выполнение предыдущей команды путем указания её номера

![название_команды] = выполняет команду из истории, путем указания её названия

Также стрелочками вверх и вниз на клавиатуре, можно выбирать команды из списка history

////////////////////////////////ALIAS////////////////////////////////

alias = отображение псевдонимов команд

Поиск слов в выходных данных других команд

Пример : `ls -l | grep 1960` (из данного списка команды `ls` будут выбраны только те файлы, которые содержат число 1960 в названии)

Поиск слов в результатах поиска

Пример : `ls -l | grep 196[6-7] | grep -v live` (указывает список файлов выдаваемых функцией `ls` содержащей цифры 1966 или 1967 и не содержащих слова `live`).

//////////////////////////////////FIND//////////////////////////////////

`find [путь_к_каталогу_поиска] [команда]`
[название_файла]= команда также производящая поиск данных на компьютере, но в отличие от `locate` она непосредственно ищет файлы по заданным условиям.

Параметры :

-name = поиск файлов по имени или его части
-user = поиск файлов принадлежащих пользователю
-group = поиск файлов принадлежащих группе
-size [размер_файла]= поиск файлов по размеру. В конце необходимо указать приставку размерности (b - 512 байтовые блоки

c - байты
k - килобайты
M - мегабайты
G - гигабайты)

-type [знак_типа]= поиск файлов по типу. В конце необходимо указать приставку типа файла (f - обычный каталог

d - каталог
l - символьная ссылка
b - специальный файл блочног типа

Атрибуты, определяющие доступ

Атрибут	Сокращение	Значение для файла	Значение для каталога
Чтение	r	Можно читать	Можно просматривать файл
Запись	w	Можно редактировать	Можно удалять, переименовывать или добавлять новые файлы
Выполнение	x	Можно запускать на выполнение	Можно читать файлы и каталоги и выполнять файлы
suid	s	Любой пользователь может запустить файл на выполнение с правами владельца	Не применяется
sgid	s	Любой пользователь может запустить файл на выполнение с правами группы	Все файлы, вновь созданные в каталоге, принадлежат группе, владеющей каталогом

"sticky bit"	t	Сообщает системе о том, что файл часто используется и должен находиться в области подкачки для быстрого обращения	Удалять или переименовывать файлы, находящиеся в каталоге имеют право только их владельцы или владелец каталога
--------------	---	---	---

cat /etc/passwd = выводит имена существующих пользователей системы, а также группы в которые они входят

/////////////////////////////////CHGRP/////////////////////////////////

chgrp [имя_каталога_или_файла] = изменение групп доступа для файлов и каталогов

Параметры :

-R = рекурсивное изменение принадлежности каталога к группе

-v = получить сведения о действиях команды chgrp

-c = получить сведения только о реальных изменениях командой chgrp

/////////////////////////////////CHOWN/////////////////////////////////

chown [имя_нового_владельца] [имя_файла] = изменение владельцев файлов или каталогов

chown [имя_нового_владельца:имя_новой_группы имя_файла] = изменение одновременно и владельца и группы для файлов и каталогов

locate [имя_файла] = ищет файлы и каталоги по заданному имени на основании базы данных, содержащей имена файлов и каталогов.

Параметры :

-i = поиск в базе имен файлов без учета регистра.

-n [количество_выводимых_результатов] = указываем кол-во результатов

slocate [имя_файла] = не ищет файлов в тех каталогах, которые не доступны данному пользователю.

updatedb = обновляет базу данных, для программы locate

/////////////////////////////////GREP/////////////////////////////////

grep [ключевое_слово] [имя_файла_для_поиска] = поиск фрагмента в заданном файле. В качестве файла можно ввести * и поиск будет произведен во всех файлах. Данная команда имеет достаточно гибкое устройство поиска, т.к имеет возможность использовать регуляторные выражения. В данном пособии регуляторные выражения рассмотрены не будут.

Параметры :

-R = рекурсивный поиск фрагментов текста в файлах

-i = поиск фрагментов текста без учета регистра букв

-w = поиск слов в файлах (производит поиск конкретного слова в файлах, без учета похожих слов)

-n = отображение номеров строк

-A = отображения последующих элементов контекста

-B = отображение предыдущих элементов контекста

-C = отображение всех элементов контекста

-v = отображение строк, не содержащих указанных слов

-l = отображение списка файлов, содержащих указанное слово

/////////////////////////////////BZIP2/////////////////////////////////

bzip2 [имя_сжимаемого_файла] = также как и gzip удаляет файл, оставляя лишь сжатый архив.

Параметры :

-c = не удаляет файл, который подлежит сжатию, а выводит в стандартный выходной поток. Для сохранения в файл нужно использовать перенаправление. Пример : bzip2 -c book.pdf > bookg.bz2

-[0-9] = задание коэффициента сжатия

bunzip2 [имя_архива].bz2 = команда для разархивации файлов .bz2

Параметры :

-c = не удаляет файл, который подлежит сжатию, а выводит в стандартный выходной поток. Для сохранения в файл нужно использовать перенаправление. Пример : gzip -c book.pdf > bookg.gz

-t = проверка файлов архива на ошибки разархивации

/////////////////////////////////TAR/////////////////////////////////

tar -cf [имя_архива].tar [имена_файлов_для_архива или название_каталога] = tar только объединяет файлы в архив.

/////////////////////////////////

/////////////////////////////////Поиск/данных/////////////////////////////////

/////////////////////////////////Глава/VIII/////////////////////////////////

/////////////////////////////////LOCATE/////////////////////////////////

chmod [ugo][+=[rwx] [имя_файла] = изменение прав доступа к файлам и каталогам с использованием символьных обозначений

Пример : chmod g+w libby.jpg

chmod [0-7][0-7][0-7] [имя_файла] = изменение прав доступа к файлам и каталогам с использованием числовых обозначений

Пояснелочка: здесь 0 = ---

1 = --x

2 = -w-

3 = -wx

4 = r--

5 = r-x

6 = rw-

7 = rwx

Пример : chmod 664 libby.jpg

Параметры :

-R = рекурсивное изменение прав (к нескольким файлам одновременно)
!Если ввести chmod 000, то производить какие-либо действия с файлом или каталогом можно будет только от имени пользователя root

chmod u[+]=s [имя_файла] = установка и сброс suid
Установленный признак suid означает, что пользователь может запустить файл на выполнение с правами пользователя, владеющего им, т.е. так, как будто этот файл был выполнен самим владельцем

chmod g[+]=s [имя_файла] = установка и сброс признака sgid

Признак `sgid` похож на `suid`, однако он применим не только к файлам, но и к каталогам. Для файлов признак `sgid` действует так же, как `suid`, за исключением того, что пользователь запускает файл на выполнение не правами владельца, а правами группы.

`chmod [+t] [имя_файла] = установка и сброс признака "sticky bit"`

////////////////////////////////////
////////////////////////////////////Создание//архивов//и//сжатие//данных////////////////////////////////////
////////////////////////////////////Глава/VII////////////////////////////////////

`zip` - самый популярный инструмент для архивирования и сжатия на всех платформах.

`gzip` - имеется практически на каждом UNIX компьютере, но нет на Mac и Windows.

`bzip2` - инструмент являющийся лучшим аналогом `gzip`, сжимает сильнее, но дольше.

`tar` - инструмент предназначенный только для архивирования. Очень популярен на серверах

////////////////////////////////////ZIP////////////////////////////////////

`zip [новое_имя_файла].zip`
`[имя_файла_который_будет_сжат] = команда для архивации и сжатия файлов`

Параметры :

- `-[0-9]` = указываем коэффициент сжатия от 0 до 9
- `-p [пароль]` = защита архива паролем (не рекомендуем)
- `-e` = шифрует архив и требует ввода пароля (рекомендуем)

`unzip [имя_архива].zip` = разархивирует `zip` архив

Параметры :

- `-v` = показывает подробную информацию о разархивировании
- `-l` = показывает содержимое архива прямо в командной строке
- `-t` = проверка файлов архива на ошибки разархивации

////////////////////////////////////GZIP////////////////////////////////////

`gzip [имя_сжимаемого_файла]` = команда которая удаляет файл и оставляет только сжатую копию его.

Параметры :

- `-c` = не удаляет файл, который подлежит сжатию, а выводит в стандартный выходной поток. Для сохранения в файл нужно использовать перенаправление. Пример : `gzip -c book.pdf > bookg.gz`
- `-r` = рекурсивная обработка файлов посредством программы `gzip`
- `-[0-9]` = задание коэффициента сжатия.

`gunzip [имя_архива].gz` = распаковка файлов, сжатых посредством `gzip`. Также удаляет архив, оставляя один распакованный файл.

Параметры :

- `-c` = не удаляет файл, который подлежит сжатию, а выводит в стандартный выходной поток. Для сохранения в файл нужно использовать перенаправление. Пример : `gzip -c book.pdf > bookg.gz`
- `-t` = проверка файлов архива на ошибки разархивации