## Front matter

title: "Отчёт по лабораторной работе №5" subtitle: "Анализ файловой структуры UNIX. Команды для работы с файлами и каталогами" author: "Абу Сувейлим Мухаммед Мунифович"

## Generic otions

lang: ru-RU toc-title: "Содержание"

## Bibliography

bibliography: bib/cite.bib csl: pandoc/csl/gost-r-7-0-5-2008-numeric.csl

## Pdf output format

toc: true # Table of contents toc-depth: 2 lof: true # List of figures lot: true # List of tables fontsize: 12pt linestretch: 1.5 papersize: a4 documentclass: scrreprt

## I18n polyglossia

polyglossia-lang: name: russian options: - spelling=modern - babelshorthands=true polyglossia-otherlangs: name: english

## I18n babel

babel-lang: russian babel-otherlangs: english

## Fonts

mainfont: PT Serif romanfont: PT Serif sansfont: PT Sans monofont: PT Mono mainfontoptions: Ligatures=TeX romanfontoptions: Ligatures=TeX sansfontoptions: Ligatures=TeX,Scale=MatchLowercase monofontoptions: Scale=MatchLowercase,Scale=0.9

## Biblatex

biblatex: true biblio-style: "gost-numeric" biblatexoptions: parentracker=true

backend=biber hyperref=auto language=auto autolang=other\* citestyle=gost-numeric

## Pandoc-crossref LaTeX customization

figureTitle: "Рис." tableTitle: "Таблица" listingTitle: "Листинг" lofTitle: "Список иллюстраций" lotTitle: "Список таблиц" lolTitle: "Листинги"

## Misc options

indent: true header-includes:

\usepackage{indentfirst}

\usepackage{float} # keep figures where there are in the text

\floatplacement{figure}{H} # keep figures where there are in the text

# Цель работы

Ознакомление с файловой системой Linux, её структурой, именами и содержанием каталогов. Приобретение практических навыков по применению команд для работы с файлами и каталогами, по управлению процессами (и работами), по проверке использования диска и обслуживанию файловой системы.

# Задание

1. Выполните все примеры, приведённые в первой части описания лабораторной работы.
2. Выполните следующие действия, зафиксировав в отчёте по лабораторной работе используемые при этом команды и результаты их выполнения:
   1. Скопируйте файл /usr/include/sys/io.h в домашний каталог и назовите его equipment. Если файла io.h нет, то используйте любой другой файл в каталоге /usr/include/sys/ вместо него.
   2. В домашнем каталоге создайте директорию ~/ski.plases.
   3. Переместите файл equipment в каталог ~/ski.plases.
   4. Переименуйте файл ~/ski.plases/equipment в ~/ski.plases/equiplist.
   5. Создайте в домашнем каталоге файл abc1 и скопируйте его в каталог ~/ski.plases, назовите его equiplist2.
   6. Создайте каталог с именем equipment в каталоге ~/ski.plases.
   7. Переместите файлы ~/ski.plases/equiplist и equiplist2 в каталог ~/ski.plases/equipment.
   8. Создайте и переместите каталог ~/newdir в каталог ~/ski.plases и назовите его plans.
3. Определите опции команды chmod, необходимые для того, чтобы присвоить перечисленным ниже файлам выделенные права доступа, считая, что в начале таких прав нет:
   1. drwxr--r-- ... australia
   2. drwx--x--x ... play

3.3. -r-xr--r-- ... my\_os

3.4. -rw-rw-r-- ... feathers При необходимости создайте нужные файлы.

1. Проделайте приведённые ниже упражнения, записывая в отчёт по лабораторной работе используемые при этом команды:
   1. Просмотрите содержимое файла /etc/password.
   2. Скопируйте файл ~/feathers в файл ~/file.old.
   3. Переместите файл ~/file.old в каталог ~/play.
   4. Скопируйте каталог ~/play в каталог ~/fun.
   5. Переместите каталог ~/fun в каталог ~/play и назовите его games.
   6. Лишите владельца файла ~/feathers права на чтение.
   7. Что произойдёт, если вы попытаетесь просмотреть файл ~/feathers командой cat?
   8. Что произойдёт, если вы попытаетесь скопировать файл ~/feathers?
   9. Дайте владельцу файла ~/feathers право на чтение.
   10. Лишите владельца каталога ~/play права на выполнение.
   11. Перейдите в каталог ~/play. Что произошло?
   12. Дайте владельцу каталога ~/play право на выполнение.
2. Прочитайте man по командам mount, fsck, mkfs, kill и кратко их охарактеризуйте, приведя примеры.

# Выполнение лабораторной работы

1. Выполните все примеры, приведённые в первой части описания лабораторной работы.

Я скопировал файл "~/abc1" в файл "april" и в файл "may".

Рис.1: Копирование файла в текущем каталоге

После создании каталога "monthly", я скопировал файлы "april" и "may" в каталог "monthly".

Рис.2: Копирование нескольких файлов в каталог

Я скопировал файл monthly/may в файл с именем june. Рис.3: Копирование файлов в произвольном каталоге

После создании каталога "monthly.00", я скопировал каталог "monthly" в каталог "monthly.00".

Рис.4: Копирование каталогов в текущем каталоге

Я скопировал каталог "monthly.00" в каталог "/tmp".

Рис.5: Копирование каталогов в произвольном каталоге

Я изменил название файла "april" на "july" в домашнем каталоге.

Рис.6: Переименование файлов в текущем каталоге

Я переместил файл "july" в каталог "monthly.00". Рис.7: Перемещение файлов в другой каталог Я переименовал каталог "monthly.00" в "monthly.01".

Рис.8: Переименование каталогов в текущем каталоге

После создании каталога "reports", я переместил каталог "monthly.01" в каталог "reports".

Рис.9: Перемещение каталога в другой каталог

Я переименовал каталог "reports/monthly.01" в "reports/monthly".

Рис.10: Переименование каталога, не являющегося текущим

Я создал файл "~/may" с правом выполнения для владельца. Рис.11: chmod u+x may А тепер наборот, надо лишить владельца файла ~/may права на выполнение

Рис.12: chmod u-x may

У каталога "monthly" запретил доступ на чтение для членов группы и всех остальных пользователей. Рис.13: chmod g-r, o-r monthly

У файла "abc1" есть доступ к записи. Рис.14: chmod g+w abc1

Для просмотра используемых в операционной системе файловых систем можно воспользоваться командой "mount" без параметров. В результате её применения можно получить примерно следующее. Рис.15: mount

Другой способ определения смонтированных в операционной системе файловых систем — просмотр файла "/etc/fstab". Сделать это можно например с помощью команды "cat".

Рис.16: cat /etc/fstab

Для определения объёма свободного пространства на файловой системе можно воспользоваться командой df, которая выведет на экран список всех файловых систем в соответствии с именами устройств, с указанием размера и точки монтирования. Рис.17: df

С помощью команды "fsck" можно проверить (а в ряде случаев восстановить) целостность файловой системы. Рис.18: fsck

1. Выполните следующие действия, зафиксировав в отчёте по лабораторной работе используемые при этом команды и результаты их выполнения.

Я скопировал файл "/usr/include/sys/io.h" в домашний каталог и назовал его "equipment".

Рис.19: cp io.h equipment

В домашнем каталоге я создал директорию "~/ski.plases". Рис.20: mkdir ski.plases

Я переместил файл "equipment" в каталог "~/ski.plases". Рис.21: mv equipment ~/ski.plases Я переименовал файл "equipment" в "equiplist". Рис.22: mv equipment ~/ski.plases/equiplist Я скопировал файл "abc1" в каталог "~/ski.plases", и назовал его "equiplist2".

Рис.23: cp ~/abc1 ~/ski.plases/equiplist2

Я создал каталог с именем "equipment" в каталоге "~/ski.plases".

Рис.24: mkdir ~/ski.plases/equipment

Я переместил файлы "equiplist" и "equiplist2" в каталог "~/ski.plases/equipment".

Рис.25: mv ~/ski.plases/equiplist ~/ski.plases/equipment

Я переместил каталог "newdir" в каталог "~/ski.plases" и назовал его "plans".

Рис.26: mv ~/newdir ~/ski.plases/plans

1. Определите опции команды chmod, необходимые для того, чтобы присвоить перечисленным ниже файлам выделенные права доступа, считая, что в начале таких прав нет:

Я создал каталоги "australia" и "play", а также файлы "my\_os" и "feathers" в домашнем каталоге.

Рис.27: mkdir australia play и tocuh my\_os feathers

Используя команду "ls -l" можно посмотреть дооступ каталогов и файлов, которые мы создали.

Рис.28: ls -l

Используя команду "chmod" я изминел доступ каталогов и файлов. Рис.29: chmod

1. Проделайте приведённые ниже упражнения, записывая в отчёт по лабораторной работе используемые при этом команды

На виртуальной машине нет файла "password", поэтому я просмотрил содержимое файла "/etc/passwd".

Рис.30: ls- a /etc/passwd

Я скопировал файл "~/feathers" в файл "file.old" Рис.31: cp ~/feathers ~/file.old Я переместил файл "~/file.old" в каталог "play" Рис.32: mv ~/file.old ~/play

Я скопировал каталог "~/play" в каталог "fun". Рис.33: cp ~/play ~/fun

Я переместил каталог "~/fun" в каталог "play" и назовите его "games".

Рис.34: mv ~/fun ~/play/games

У файла "feathers" такие права: -rw-rw-r--. Рис.35: ls -l

Я лишил владельца файла "~/feathers" права на чтение (--w-rw-r--). Рис.36: chmod u-r feathers Я не смог просмотреть файл "~/feathers" командой "cat", т.к. у меня нет права на чтение.

Рис.37: cat ~/feathers

Я не смог скопировать файл "~/feathers" в каталог "play", т.к. у меня нет права на чтение и выполнение. Рис.38: cp ~/feathers ~/play

Я дал владельца файла "~/feathers" права на чтение (-rw-rw-r--) Рис.39: chmod u+r feathers

Я лишил владельца каталога "~/play" права на выполнение (drw---x--x). Рис.40: chmod u-x play Я не смог перейдить в каталог "~/play", т.к. у меня нет права на выполнение. Рис.41: cd ~/play

1. Прочитайте man по командам mount, fsck, mkfs, kill и кратко их охарактеризуйте, приведя примеры.

I mounted the file work. If you check (using the sudo fsck command) the file, it will say it is mounted.

Рис.42: mount -a ~/work

I checked the filesystem "/dev/sda1", and it is mounted. Рис.43: sudo fsck /dev/sda1

Tapping the Tab bottun twice after writing mkfs without any sapces will show you the created filesystems. Рис.44: mkfs

kill -l will show you all the processes that are active right now. Kill -1 is the same as kill -SIGHUP. Рис.45: kill -1 -1

# Выводы

Я ознакомлся с файловой системой Linux, её структурой, именами и содержанием каталогов. Также, я приобретел практических навыков по применению команд для работы с файлами и каталогами, по

управлению процессами (и работами), по проверке использования диска и обслуживанию файловой системы.

# Контрольные вопросы

1.

proc: папка proc linux содержит файлы нулевого размера. Все поддиректории, файлы и хранящаяся в них информация генерируется ядром на лету, как только вы ее запрашиваете. sysfs: виртуальная файловая система в операционной системе GNU/Linux.

devtmpfs: это файловая система с автоматическими узлами устройств, заполняемыми ядром. ssecurityfs: эта файловая система предназначена для использования модулями безопасности, некоторые из которых в противном случае создавали свои собственные файловые системы; он должен быть смонтирован в /sys/kernel/security.

tmpfs: временное файловое хранилище во многих Unix-подобных ОС.

devpts: файловая система специального типа devpts смонтирована в каталоге /dev/pts. Эта файловая система не связана ни с каким аппаратным устройством, создается ядром Linux и напоминает файловую систему /proc.

cgroup2: компонент ядра Linux, обеспечивающий механизм изоляции, измерения и управления распределением ресурсов для набора процессов на сервере.

pstore: файловая система постоянного хранения, pstore, может хранить записи об ошибках, когда ядро умирает (или перезагружается, или отключается).

bpf: позволяет запускать произвольный код, предоставляемый пользователем, в пространстве ядра Linux и новая архитектура оказалась настолько удачной, что нам потребуется еще с десяток статей, чтобы описать все ее применения.

btrfs: файловая система для Linux, основанная на структурах B-деревьев и работающая по принципу «копирование при записи» (copy-on-write).

selinuxfs: реализация системы принудительного контроля доступа, которая может работать параллельно с классической избирательной системой контроля доступа. Входит в стандартное ядро Linux.

autofs: новая система, работающая с помощью ядра; это означает, что код ядра, отвечающий за работу с файловыми системами, знает, где расположены точки подключения в файловой системе, и система автоподключения узнает их с помощью этого кода.

mqueue: Файловая система mqueue обеспечивает необходимую поддержку ядра для библиотеки пользовательского пространства, которая реализует интерфейсы очереди сообщений POSIX. hugetlbfs: Эта псевдо-файловая система резервируется по требованию системного администратора с помощью записи количества страниц, которое необходимо зарезервировать, в

/proc/sys/vm/nr\_hugepages.

debugfs: используется для отладочных целей, в первую очередь для разработки ядра Linux. ext4: одна из основных файловых систем, используемая преимущественно в операционных системах на ядре Linux. Максимальный размер файла от 16 ГБ до 16 ТБ.

ext3: Файловая система Ext3 была представлена в 2001 году и была интегрирована в ядро 2.4.15 с функцией логирования, которая повышает надежность и устраняет необходимость проверять файловую систему после некорректного завершения работы. Максимальный размер файла 16 ГБ

– 2 ТБ

ext2: Файловая система Ext2 была представлена в 1993 году и была разработана Remy Card. Это была первая файловая система по умолчанию в нескольких дистрибутивах Linux, таких как RedHat и Debian. Максимальный размер файла составляет 16 ГБ – 2 ТБ.

1. В ОС Linux файловая система представлена единым корневым каталогом, обозначаемым как слэш (/). Соответственно, при данной файловой структуре не диски содержат каталоги, а каталог — диски.

: Описание некоторых каталогов файловой системы GNU Linux {#tbl:std-dir}

**Имя каталога**

**Описание каталога**

/ Корневая директория, содержащая всю файловую

Основные системные утилиты, необходимые как в однопользовательском режиме, так и

/bin

/etc

/home

при обычной работе всем пользователям

Общесистемные конфигурационные файлы и файлы конфигурации установленных программ

Содержит домашние директории пользователей, которые, в свою очередь, содержат персональные настройки и данные пользователя

/media Точки монтирования для сменных носителей

/root Домашняя директория пользователя root

/tmp Временные файлы

/usr Вторичная иерархия для данных пользователя

Более подробно об Unix см. в [@gnu- doc:bash;@newham:2005:bash;@zarrelli:2017:bash;@robbins:2013:bash;@tannenbaum:arch- pc:ru;@tannenbaum:modern-os:ru].

1. Монтирование тома.
2. Отсутствие синхронизации между образом файловой системы в памяти и ее данными на диске в случае аварийного останова может привести к появлению следующих ошибок:

Один блок адресуется несколькими mode (принадлежит нескольким файлам). Блок помечен как свободный, но в то же время занят (на него ссылается onode).

Блок помечен как занятый, но в то же время свободен (ни один inode на него не ссылается). Неправильное число ссылок в inode (недостаток или избыток ссылающихся записей в каталогах). Несовпадение между размером файла и суммарным размером адресуемых inode блоков.

Недопустимые адресуемые блоки (например, расположенные за пределами файловой системы). "Потерянные" файлы (правильные inode, на которые не ссылаются записи каталогов).

Недопустимые или неразмещенные номера inode в записях каталогов.

1. mkfs - позволяет создать файловую систему Linux.
2. Cat - выводит содержимое файла на стандартное устройство вывода
3. Cp – копирует или перемещает директорию, файлы.
4. Mv - переименовать или переместить файл или директорию.
5. Права доступа к файлу или каталогу можно изменить, воспользовавшись командой chmod.