

Лабораторная работа №5

**Анализ файловой системы Linux. Команды для работы с
файлами и каталогами**

Емельянов Антон НПМбв-01-21в

Содержание

1	Цель работы	5
2	Задание	6
3	Теоретическое введение	7
4	Выполнение лабораторной работы	9
5	Выводы	13
	Список литературы	14

Список иллюстраций

4.1	Повторяем примеры из первой части работы.	9
4.2	Провожу множество базовых операции с файлами.	10
4.3	Команда <code>chmod</code> позволяет менять права доступа к объектам, сейчас мы её опробуем.	10
4.4	Читаем файл.	11
4.5	Вот такая вот матрёшка получилась.	11
4.6	Не читается.	11
4.7	Не открыть нам неоткрываемое.	12
4.8	Проверил все команды.	12

Список таблиц

3.1	Описание некоторых каталогов файловой системы GNU Linux . .	7
-----	---	---

1 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.

2 Задание

Здесь приводится описание задания в соответствии с рекомендациями методического пособия и выданным вариантом.

3 Теоретическое введение

Здесь описываются теоретические аспекты, связанные с выполнением работы.

Например, в табл. 3.1 приведено краткое описание стандартных каталогов Unix.

Таблица 3.1: Описание некоторых каталогов файловой системы GNU Linux

Имя каталога	Описание каталога
/	Корневая директория, содержащая всю файловую
/bin	Основные системные утилиты, необходимые как в однопользовательском режиме, так и при обычной работе всем пользователям
/etc	Общесистемные конфигурационные файлы и файлы конфигурации установленных программ
/home	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 про Unix см. в [1–4].

4 Выполнение лабораторной работы

Для начала повторим все примеры из первой части лабораторной работы: проверим команды для работы с файлами и каталогами и поиграемся с правами доступа (рис. 4.1)

```
[avemel'yanov@avemel'yanov report]$ cd ..
[avemel'yanov@avemel'yanov lab05]$ touch abs1
[avemel'yanov@avemel'yanov lab05]$ cp abs1 april
cp: не удалось выполнить stat для 'abs1': Нет такого файла или каталога
[avemel'yanov@avemel'yanov lab05]$ cp abs1 april
[avemel'yanov@avemel'yanov lab05]$ cp abs1 may
[avemel'yanov@avemel'yanov lab05]$ ls
abs1 april may presentation report
[avemel'yanov@avemel'yanov lab05]$ mkdir monthly
[avemel'yanov@avemel'yanov lab05]$ cp april may monthly
[avemel'yanov@avemel'yanov lab05]$ ls
abs1 april may monthly presentation report
[avemel'yanov@avemel'yanov lab05]$ cd mounthly
-bash: cd: mounthly: Нет такого файла или каталога
[avemel'yanov@avemel'yanov lab05]$ cd monthly
[avemel'yanov@avemel'yanov monthly]$ ls
april may
[avemel'yanov@avemel'yanov monthly]$ cd ..
[avemel'yanov@avemel'yanov lab05]$ cp monthly/may monthly/june
[avemel'yanov@avemel'yanov lab05]$ ls monthly
april june may
[avemel'yanov@avemel'yanov lab05]$ mkdir monthly.00
[avemel'yanov@avemel'yanov lab05]$ cp -r monthly monthly.00
[avemel'yanov@avemel'yanov lab05]$ ls
abs1 april may monthly monthly.00 presentation report
[avemel'yanov@avemel'yanov lab05]$ cp -r monthly.00 /tmp
```

Рисунок 4.1: Повторяем примеры из первой части работы.

Далее переходим к практической работе с файлами: копируем файлы, создаём директории, переименовываем директории и файлы, создаём по несколько файлов за раз и меняем названия файлов не в своей директории (рис. 4.2)

```

[avemeljyanov@avemeljyanov study_2025-2026_os-intro]$ ls
CHANGELOG.md  equipment  Makefile      project-personal  README.md
config        labs       prepare       README.en.md      template
COURSE        LICENSE    presentation  README.git-flow.md
[avemeljyanov@avemeljyanov study_2025-2026_os-intro]$ mkdir ski.places
[avemeljyanov@avemeljyanov study_2025-2026_os-intro]$ mv equipment ski.places
[avemeljyanov@avemeljyanov study_2025-2026_os-intro]$ ls
CHANGELOG.md  LICENSE    project-personal  ski.places
config        Makefile    README.en.md      template
COURSE        prepare     README.git-flow.md
labs          presentation README.md
[avemeljyanov@avemeljyanov study_2025-2026_os-intro]$ ls ski.places
ls: невозможно получить доступ к 'ski.places': Нет такого файла или каталога
[avemeljyanov@avemeljyanov study_2025-2026_os-intro]$ ls ski.places
equipment
[avemeljyanov@avemeljyanov study_2025-2026_os-intro]$ mv ski.places/equipment ski.places/equiplist
[avemeljyanov@avemeljyanov study_2025-2026_os-intro]$ ls ski.places
equiplist

```

Рисунок 4.2: Провожу множество базовых операции с файлами.

Далее изучем опции команды `chmod`, ведь в следующем пункте нам с ней работать (рис. 4.3)

```

При задании --reference, установить режим
указанных ФАЙЛОВ как у ФАЙЛА.

-c, --changes      как verbose, но выводить только при выполнении
                  изменений
-f, --silent, --quiet не выводить большинство сообщений об ошибках
-v, --verbose      выводить диагностические сообщения по каждому файлу
--dereference      affect the referent of each symbolic link,
                  rather than the symbolic link itself
-h, --no-dereference affect each symbolic link, rather than the referent
--no-preserve-root не обрабатывать «/» особым образом (по умолчанию)
--preserve-root    отказываться рекурсивно обрабатывать «/»
--reference=RFILE  use RFILE's mode instead of specifying MODE values.
                  RFILE is always dereferenced if a symbolic link
-R, --recursive    рекурсивно изменять файлы и каталоги

The following options modify how a hierarchy is traversed when the -R
option is also specified. If more than one is specified, only the final
one takes effect. -H is the default.

-H                if a command line argument is a symbolic link
                  to a directory, traverse it
-L                traverse every symbolic link to a directory
                  encountered
-P                do not traverse any symbolic links

--help            показать эту справку и выйти
--version         показать информацию о версии и выйти

Каждый РЕЖИМ задаётся в форме «[ugoa]*([-+]=)([rwxXst]*[ugo])|[-+]=[0-7]+».

Страница справки по GNU coreutils: <https://www.gnu.org/software/coreutils/ls/>
Об ошибках в переводе сообщений сообщайте по адресу <https://translation.project.org/team/ru.html>
Полная документация: <https://www.gnu.org/software/coreutils/chmod>
или доступная локально: info '(coreutils) chmod invocation'

```

Рисунок 4.3: Команда `chmod` позволяет менять права доступа к объектам, сейчас мы её опробуем.

Но перед этим прочтём содержимое файла (рис. 4.4)

```

geoclue:x:999:999:User for geoclue:/var/lib/geoclue:/sbin/nologin
systemd-oom:x:998:998:systemd Userspace OOM Killer:/:usr/bin/nologin
polkitd:x:114:114:User for polkitd:/:sbin/nologin
sstpc:x:997:996:Secure Socket Tunneling Protocol(SSTP) Client:/var/run/s
stpc:/sbin/nologin
rtkit:x:172:172:RealtimeKit:/:sbin/nologin
chrony:x:995:995:chrony system user:/var/lib/chrony:/sbin/nologin
systemd-coredump:x:994:994:systemd Core Dumper:/:usr/bin/nologin
systemd-timesync:x:993:993:systemd Time Synchronization:/:usr/bin/nolog
in
rpc:x:32:32:Rpcbind Daemon:/var/lib/rpcbind:/usr/bin/nologin
systemd-resolve:x:193:193:systemd Resolver:/:usr/bin/nologin
pipewire:x:992:992:PipeWire System Daemon:/run/pipewire:/usr/bin/nologin
sssd:x:991:991:User for sssd:/run/sss:/sbin/nologin
unbound:x:990:990:Unbound DNS resolver:/var/lib/unbound:/sbin/nologin
nm-openconnect:x:989:989:NetworkManager user for OpenConnect:/:sbin/nol
ogin
wsdd:x:988:988:Web Services Dynamic Discovery host daemon:/:sbin/nologi
n
openvpn:x:987:987:OpenVPN:/etc/openvpn:/sbin/nologin
nm-openvpn:x:986:986:Default user for running openvpn spawned by Network
Manager:/:sbin/nologin
colord:x:985:985:User for colord:/var/lib/colord:/sbin/nologin
abrt:x:173:173:/:etc/abrt:/sbin/nologin
setroubleshoot:x:984:984:SELinux troubleshoot server:/var/lib/setroubles
hoot:/usr/bin/nologin
sddm:x:983:983:SDDM Greeter Account:/var/lib/sddm:/usr/bin/nologin
rpcuser:x:29:29:RPC Service User:/var/lib/nfs:/sbin/nologin
vboxadd:x:981:1:/:var/run/vboxadd:/usr/bin/nologin
usbmuxd:x:113:113:usbmuxd user:/:usr/bin/nologin
sshd:x:74:74:Privilege-separated SSH:/usr/share/empty.sshd:/usr/bin/nolo
gin
dnsmasq:x:980:980:Dnsmasq DHCP and DNS server:/var/lib/dnsmasq:/usr/bin/
nologin
systemd-network:x:192:192:systemd Network Management:/:usr/bin/nologin
passim:x:979:979:Local Caching Server:/usr/share/empty:/usr/bin/nologin
tcpdump:x:72:72:tcpdump:/:usr/bin/nologin
avemel'yanov:x:1000:1000:avemel'yanov:/home/avemel'yanov:/bin/bash
[avemel'yanov@avemel'yanov etc]$

```

Рисунок 4.4: Читаем файл.

После этого проведём пару операций вложения, копирования и переименовывания (рис. 4.5)

```

[avemel'yanov@avemel'yanov study_2025-2026_os-intro]$ cp feathers file.o
ld
[avemel'yanov@avemel'yanov study_2025-2026_os-intro]$ mv file.old play
[avemel'yanov@avemel'yanov study_2025-2026_os-intro]$ cp play fun
cp: не указан -i; пропускается каталог 'play'
[avemel'yanov@avemel'yanov study_2025-2026_os-intro]$ cp -r play fun
[avemel'yanov@avemel'yanov study_2025-2026_os-intro]$ mv fun play
[avemel'yanov@avemel'yanov study_2025-2026_os-intro]$ mv play/fun play/g
ames
[avemel'yanov@avemel'yanov study_2025-2026_os-intro]$ ls play/games
play
[avemel'yanov@avemel'yanov study_2025-2026_os-intro]$ ls play/games/play
file.old

```

Рисунок 4.5: Вот такая вот матрёшка получилась.

Далее начинаем работать с правами доступа к файлам - лишаем себя возможности читать файл и как итог НЕ можем его прочитать, потом всё возвращаем как было (рис. 4.6)

```

[avemel'yanov@avemel'yanov study_2025-2026_os-intro]$ chmod u-r feathers
[avemel'yanov@avemel'yanov study_2025-2026_os-intro]$ cat feathers
cat: feathers: Отказано в доступе
[avemel'yanov@avemel'yanov study_2025-2026_os-intro]$ cp feathers play
cp: невозможно открыть 'feathers' для чтения: Отказано в доступе
[avemel'yanov@avemel'yanov study_2025-2026_os-intro]$ chmod u+r feathers

```

Рисунок 4.6: Не читается.

А теперь поробуем лишить себя права выполнения каталога и затем откроем его. Не получилось. Вернём права как было (рис. 4.7)

```
[avemeljyanov@avemeljyanov study_2025-2026_os-intro]$ chmod u-x play
[avemeljyanov@avemeljyanov study_2025-2026_os-intro]$ cd play
-bash: cd: play: Отказано в доступе
[avemeljyanov@avemeljyanov study_2025-2026_os-intro]$ chmod u+x play
```

Рисунок 4.7: Не открыть нам неоткрываемое.

Ну а теперь изучим команды mount, fsck, mkfs и kill. Если коротко, то mount монтирует файловую систему, fsck проверяет и восстанавливает файловые системы, mkfs создаёт файловую систему на устройстве, а kill завершает процесс. (рис. 4.8)

```
[avemeljyanov@avemeljyanov study_2025-2026_os-intro]$ man mount
[avemeljyanov@avemeljyanov study_2025-2026_os-intro]$ man fsck
[avemeljyanov@avemeljyanov study_2025-2026_os-intro]$ man mkfs
[avemeljyanov@avemeljyanov study_2025-2026_os-intro]$ man kill
```

Рисунок 4.8: Проверил все команды.

5 Выводы

В ходе выполнения работы были получены навыки работы с файлами, файловыми системами, а также правами доступа. Теперь я умею лишать себя прав. Хорошо..

Список литературы

1. *Newham C.* Learning the bash Shell: Unix Shell Programming. — O'Reilly Media, 2005. — 354 p. — (In a Nutshell). — ISBN 0596009658. — URL: <http://www.amazon.com/Learning-bash-Shell-Programming-Nutshell/dp/0596009658>.
2. *Robbins A.* Bash Pocket Reference. — O'Reilly Media, 2016. — 156 p. — ISBN 978-1491941591.
3. *Zarrelli G.* Mastering Bash. — Packt Publishing, 2017. — 502 p. — ISBN 9781784396879.
4. *Таненбаум Э., Бос Х.* Современные операционные системы.. — 4-е изд. — СПб. : Питер, 2015.. — 1120 с. — (Классика Computer Science).