

ЛР №3. Файловая система

Цель работы

Ознакомиться со структурой файловой системы Линукс, типами файлов и командами управления файловой системой.

Методические указания

Организация файловой системы

Файловая система ОС Линукс (как и прочих unix-подобных систем) устроена так, что все ресурсы представлены единообразно, в виде файлов. Такой подход позволяет обеспечить универсальный интерфейс доступа к любым ресурсам: от физических устройств, до процессов, выполняющихся в системе. С точки зрения пользователя файловая система представляет логическую структуру каталогов и файлов. С другой стороны, невидимой пользователю, внутреннее устройство файловой системы реализует физические операции чтения/записи файлов на различные носители, алгоритмы доступа и многое другое.

Типы файлов

Для обеспечения единообразного доступа к файлам их прежде всего необходимо классифицировать. В Линукс это сделано следующим образом:

- *обычные (regular) файлы* - текстовые, исполняемые, графические и пр. файлы, создаваемые пользователями и прикладными программами;
- *каталоги (directories)* - именованные группы файлов и вложенных каталогов (т.е. содержимое каталога - суть файлы и другие каталоги);
- *файлы устройств (devices)* - соответствуют присутствующим в системе реальным (жесткие диски, принтеры, мыши, ЦП и т.д.) устройствам и т.н. псевдоустройствам (например, /dev/null). Файлы устройств представляют символьные (последовательного доступа) и блочные (произвольного доступа) устройства. К первым относятся, например, параллельные и последовательные порты, ко вторым - жесткие диски;
- *специальные файлы* - сокеты (sockets) и именованные каналы (named pipes), которые предназначены для обмена информацией между процессами;
- *символьные ссылки (symlinks)* - именованные указатели на физические файлы (аналог ярлыков ОС Windows), содержащие только путь к некоторому файлу. Символьные ссылки могут указывать на файлы, хранящиеся как локальных, так и в сетевых каталогах.

Символьные ссылки (или "мягкие") не нужно путать с "жесткими", которые указывают на inode файла. Inode (идентификатор узла) - это уникальный числовой идентификатор узла (файла или каталога) файловой системы, по которому и осуществляется доступ к нему. Символьное имя файла (включая полный путь) ориентировано на пользовательское восприятие. Для человека-оператора проще запомнить осмысленные имена файлов (например: report.txt, myfoto.jpg и т.п.), чем абстрактные числовые значения. Прочие отличия "жестких" и "мягких" ссылок вам предстоит выяснить в ходе выполнения этой лабораторной работы.

Каталоги Линукс

Все файлы упорядочены по каталогам. Структура и назначение каждого из каталогов, созданных на этапе установке predetermined, хотя и могут быть (что не стоит делать без крайней необходимости) изменены суперпользователем.

Файловая система имеет иерархическую структуру и начинается от корневого каталога (/). Его подкаталогами являются:

- **/bin** - исполняемые файлы общего назначения;
- **/boot** - содержит образ загружаемого ядра;
- **/dev** - файлы устройств;
- **/etc** - конфигурационные файлы общего пользования;
- **/home** - домашние каталоги пользователей, включая программы и файлы личных предпочтений;
- **/lib** - общесистемные библиотеки;
- **/mnt** - каталог монтирования внешних файловых систем;
- **/proc** - виртуальная файловая система для чтения информации о процессах;
- **/root** - домашний каталог суперпользователя;
- **/sbin** - программы системного администрирования;
- **/tmp** - каталог для хранения временной информации;
- **/usr** - каталог пользовательских прикладных программ со всеми их исполнимыми и конфигурационными файлами. Например, в подкаталог `/usr/local` инсталлируются программы, не входящие в дистрибутив Линукс, или собираемые из исходных текстов.
- **/var** - каталог для хранения часто изменяющихся файлов. Например, спулера печати, различных лог-файлов, почтовых сообщений и т.п.
- **/lost+found** - каталог для нарушенных фрагментов файлов, обнаруженных в результате проверки файловой системы после сбоя.

Такая структура типична для большинства дистрибутивов Линукс, но могут иметься и дополнительные каталоги. Например, `/opt` - для дополнительных компонентов, `/selinux` - расширение системы безопасности и т.п.

Именование файлов и каталогов

Файловая система Линукс поддерживает "длинные" имена, содержащие символы латиницы, национальных алфавитов, знаки пунктуации и спецсимволы. Абсолютно запрещенными к использованию в имени являются прямой и обратный слэши (/ и \). Максимальное количество символов в имени - 255. Понятие "расширения файла" в unix-системах отсутствует как таковое, поэтому в имени может быть несколько частей, разделенных точками. Все имена - *регистрозависимые*.

Приведенные выше правила справедливы и для каталогов.

Файлы и каталоги, названия которых начинаются с точки (т.н. dot-файлы), являются аналогами "скрытых" файлов MS-DOS. Т.е. в общем случае они не отображаются при просмотре содержимого файловой системы.

Для быстрого доступа к файлам в оболочке имеются несколько переменных окружения, хранящих соответствующие пути. Это, например, переменная `$HOME`, в которой содержится пути к домашнему каталогу текущего пользователя. Т.е. действия команд

```
[usr1@localhost var]$ cd /home/usr1
```

и

```
[usr1@localhost var]$ cd $HOME
```

приведут к одному результату - переходу в домашний каталог пользователя usr1. Более того, в оболочке определен псевдоним для домашнего каталога - символ ~ (тильда) можно использовать аналогично \$HOME. Например:

```
[usr1@localhost var]$ cd ~
```

```
[usr1@localhost ~]$ pwd
```

```
/home/usr1
```

```
[usr1@localhost var]$
```

Некоторые другие возможности оболочки будут рассмотрены в следующих лабораторных работах. Дополнительная информация доступна в справочном руководстве bash (man bash).

Команды управления файловой системой

Для управления файловой системой имеются различные команды, реализующие операции по созданию, чтению, копированию, переименованию/перемещению, изменению и удалению файлов и каталогов. Как правило, это специализированные команды, хорошо выполняющие свою задачу, однако некоторые функции могут частично дублироваться другими командами, что только добавляет гибкости управлению файлами.

Основными командами для выполнения файловых операций являются: pwd, ls, cp, mv, dir, rm, cd, rmdir, mvdir, mkdir, ln. Информацию о их назначении и параметрах доступна в формате man и info.

Несколько слов о способах создания файлов в Линукс:

- во-первых, файлы могут быть созданы как результаты работы прикладных программ и иметь определенный формат (например, графические файлы, созданные редактором GIMP);
- во-вторых, файлы могут быть созданы пользователем путем ввода информации с клавиатуры, например так:

```
• aag@stilo:~> cat > f1
```

```
• Hello, world! // нажатие Ctrl+D завершает ввод команд
```

```
• aag@stilo:~>
```

- в-третьих, файлы могут быть созданы путем перенаправления вывода команды со стандартного потока, например так:

```
• [root@localhost aag]# echo "Hello, World!" > f1
```

В первом и втором случаях символ ">" - это команда перенаправления стандартных потоков ввода/вывода, встроенная в оболочку. В первом случае она получает информацию со стандартного потока ввода (клавиатура) и, по окончании ввода (Ctrl+D), отправляет ее в файл. Во втором - принимает строку, переданную командой echo и также отправляет ее в файл.

Если файл отсутствует, то он будет создан, если имеется, то будет **перезаписан**. Для **добавления** информации в файл следует использовать команду ">>". Больше информации о командах перенаправления можно получить в справке о возможностях оболочки.

Задания к выполнению

1. Войти в систему с собственной учетной записью
2. Вывести на экран список файлов текущего каталога в краткой и расширенной форме
3. Переместиться в каталог /
4. Сохранить в файле \$HOME/filelist.lst список каталогов в каталоге /
5. Вернуться в домашний каталог и вывести рекурсивный список всех (в т.ч. и скрытых) файлов и каталогов
6. В домашнем каталоге создать подкаталоги src, dst и temp
7. В каталоге src создать текстовый файл f1 произвольного содержания
8. В каталог src скопировать файлы user.txt, root.txt и stud.txt, созданные в лабораторной работе № 2. Все ли файлы удалось скопировать?
9. В каталоге dst создать «жесткие» ссылки на все файлы из каталога src
10. В домашнем каталоге создать «мягкие» ссылки на файлы из каталога src
11. Вывести рекурсивно расширенную информацию о содержимом домашнего каталога. Обратить внимание на поле размера для физических файлов и ссылок
12. Из домашнего каталога выполнить команды:
 - o cat /src/f1
 - o cat /dst/f1
 - o cat /f1
13. Запомнить результаты выполнения
14. Переместить файл f1 из каталога src в каталог temp и повторить п.12
15. Удалить файл f1 и повторить п.12
16. Сравнить результаты выполнения пп 12, 13 и 14
17. Удалить все файлы, имеющие в названии txt из каталога dst
18. Удалить каталог dst
19. Переместить каталог temp в src
20. Рекурсивно удалить каталог src
21. Завершить сеанс