

Часть 3. Файловая система, введение

Влад 'mend0za' Шахов
Linux & Embedded Team Leader

Linux & Embedded Department



Определения

- В UNIX (и Linux) файлы организованы в виде *единой древовидной структуры* (дерева), называемой **файловой системой**.
- **Каждый файл имеет имя**, определяющее его расположение в дереве FS.
- Корнем дерева является **корневой каталог** (root directory), имеющий имя **" / "**.



Определения

- В UNIX (и Linux) файлы организованы в виде *единой древовидной структуры* (дерева), называемой **файловой системой**.
- **Каждый файл имеет имя**, определяющее его расположение в дереве FS.
- Корнем дерева является **корневой каталог** (root directory), имеющий имя **" / "**.
- **полный путь** начинается с **/** (корневого каталога), каталоги разделяются также **/**.
Пример: `/home/user/.ssh/authorized_key`
- **относительный путь** - от текущего каталога
Примеры: `../user10/.bashrc`; `./script`; `ls script`



Навигация в командной строке (shell)

Для каждой запущенной программы (в том числе и shell) в системе определен **текущий каталог**.

- **pwd**¹ - получить текущий каталог
- **cd [папка]**² - сменить каталог
- **cd -**³ - вернуться в предыдущий каталог
- **ls [папка]**⁴ - просмотр списка файлов
- **ls -l [папка]** - длинный листинг (с подробностями)

¹см также переменные окружения **PWD** и **OLDPWD**

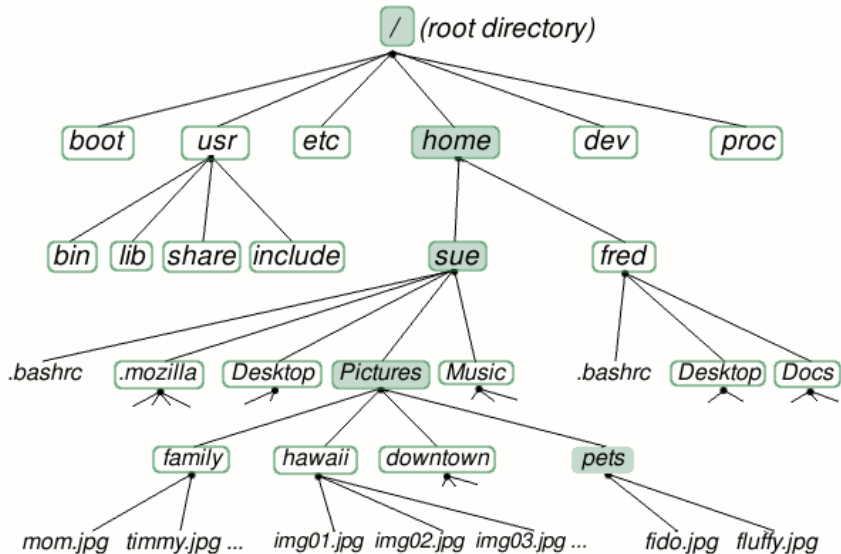
²**cd** без параметров - возврат в домашний каталог

³Используется значение переменной **OLDPWD**

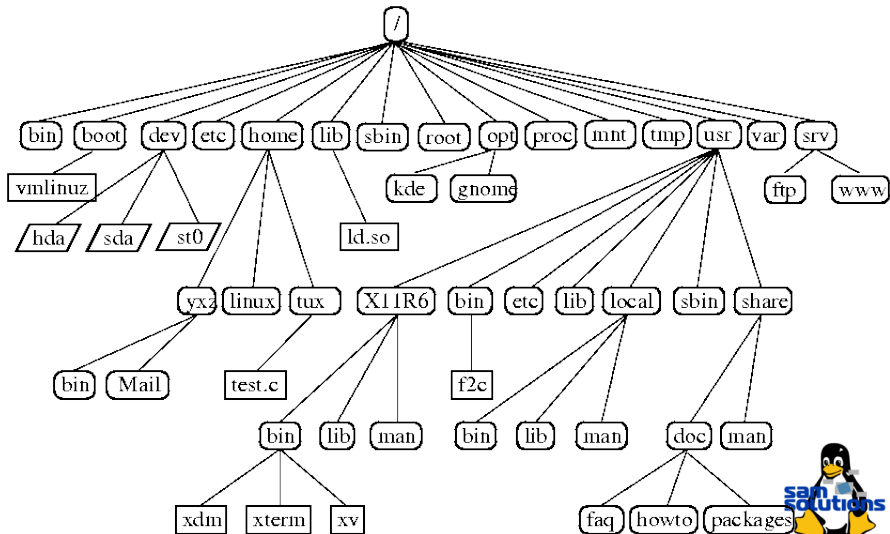
⁴**ls** без параметров - просмотр текущего каталога



Дерево файловой системы - простое

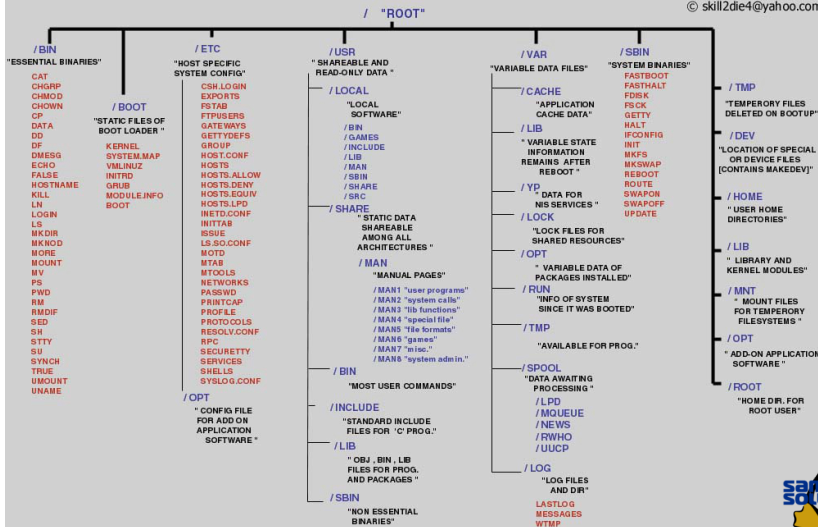


Дерево файловой системы - среднее



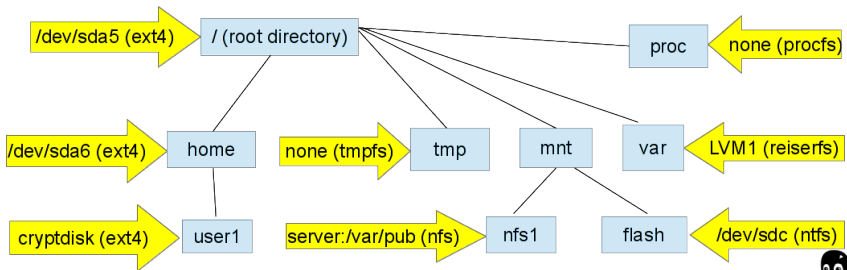
Дерево файловой системы - сложное

© skill2die4@yahoo.com



Детали реализации

- **VFS - virtual file system** - файлы и каталоги отображаются в единое дерево, независимо от их физического расположения.



Монтирование

- **Монтирование** - процесс отображения содержимого устройства в указанную папку файловой системы.
- Команды:
 - монтировать - (**mount**)
 - размонтировать (**umount**)
- **mount** без параметров - вывести список уже подключенных файловых систем



Типы файлов в Unix

Для пользователя:

- Обычные файлы (regular file): .bashrc, /bin/bash
- Каталоги (directory): /home/user1, /usr, /, /usr/local



Типы файлов в Unix

Для пользователя:

- Обычные файлы (regular file): `.bashrc`, `/bin/bash`
- Каталоги (directory): `/home/user1`, `/usr`, `/`, `/usr/local`
- Символические ссылки (symbolic links): `/bin/sh`,
`/dev/stdout`



Типы файлов в Unix

Для пользователя:

- Обычные файлы (regular file): `.bashrc`, `/bin/bash`
- Каталоги (directory): `/home/user1`, `/usr`, `/`, `/usr/local`
- Символические ссылки (symbolic links): `/bin/sh`,
`/dev/stdout`

Для администратора:

- Файлы устройств (device special file):
 - блочные: `/dev/sda5`, `/dev/loop0`, `/dev/sr0`
 - символьные: `/dev/null`, `/dev/mem`, `/dev/tty`



Типы файлов в Unix

Для пользователя:

- Обычные файлы (regular file): `.bashrc`, `/bin/bash`
- Каталоги (directory): `/home/user1`, `/usr`, `/`, `/usr/local`
- Символические ссылки (symbolic links): `/bin/sh`,
`/dev/stdout`

Для администратора:

- Файлы устройств (device special file):
 - блочные: `/dev/sda5`, `/dev/loop0`, `/dev/sr0`
 - символьные: `/dev/null`, `/dev/mem`, `/dev/tty`

Для программиста:

- FIFO (named pipe): `/dev/xconsole`
- Socket: `/dev/log`



Практика: определение типа файла

- Используем **ls -l**

```
$ ls -l
total 12
drwxr-xr-x 2 user user 4096 Jan 14 14:59 bin
-rw-r--r-- 1 user user 124 Jan 9 16:55 err
drwx----- 2 user user 4096 Jan 3 22:19 Mail
```

Первая колонка, первый символ



Практика: определение типа файла

- Используем **ls -l**

```
$ ls -l
total 12
drwxr-xr-x 2 user user 4096 Jan 14 14:59 bin
-rw-r--r-- 1 user user 124 Jan 9 16:55 err
drwx----- 2 user user 4096 Jan 3 22:19 Mail
```

Первая колонка, первый символ

- Определение типа файла: **file**

```
~$ file /dev/null
/dev/null: character special
~$ file .
.: directory
~$ file /usr/include/stdio.h
/usr/include/stdio.h: ASCII C program text
```



Просмотр обычных файлов

- Просмотр текста:
 - **cat** - вывести на stdout⁵

⁵ Для двоичных файлов: чревато порчей настроек терминала
⁶ может отсутствовать в стандартной поставке



Просмотр обычных файлов

- Просмотр текста:
 - **cat** - вывести на stdout⁵
 - **more** - вывести, разбив на страницы
 - **less**⁶ - **more** на стероидах, с прокруткой, поиском

⁵ Для двоичных файлов: чревато порчей настроек терминала

⁶ может отсутствовать в стандартной поставке



Просмотр обычных файлов

- Просмотр текста:
 - **cat** - вывести на stdout⁵
 - **more** - вывести, разбив на страницы
 - **less**⁶ - **more** на стероидах, с прокруткой, поиском
- Просмотр двоичных данных:
 - **od** - дамп файла в не-текстовых форматах

```
~$ od -c .bashrc
0000000  #      ~   /   .   b   a   s   h   r   c   :   e   x   e
0000020  c   u   t   e   d   b   y   b   a   s   h   (   1   )

~$ od -x .bashrc
0000000  2023 2f7e 622e 7361 7268 3a63 6520 6578
0000020  7563 6574 2064 7962 6220 7361 2868 2931

~$ od -f .bashrc
0000000  2.311258e-10  1.785672e+31  8.676411e-04  7.331327e+22
0000020  7.215143e+22  7.338225e+34  1.785670e+31  3.933694e-14
```

- **strings** - извлечь текстовые строки из двоичных файлов

⁵ Для двоичных файлов: чревато порчей настроек терминала

⁶ может отсутствовать в стандартной поставке



Создание (текстовых) файлов



⁷До ``конца ввода'', т.е. нажатия Ctrl+D

Создание (текстовых) файлов

- 1 Любимым редактором: **vi/vim**, **nano**, **mcedit**



⁷До ``конца ввода'', т.е. нажатия Ctrl+D

Создание (текстовых) файлов

- 1 Любимым редактором: **vi/vim**, **nano**, **mcedit**
- 2 **echo**

```
echo какойто— текст >file
```



⁷До ``конца ввода'', т.е. нажатия Ctrl+D

Создание (текстовых) файлов

- 1 Любимым редактором: **vi/vim**, **nano**, **mcedit**
- 2 **echo**

```
echo какойто— текст >file
```

- 3 **cat** с перенаправлением⁷

```
~$ cat >>file3  
ladsfa  
134097  
  
~$ ls -l file3  
-rw-r—r— 1 user user 25 Янв 23 17:13 file3
```



⁷ До ``конца ввода'', т.е. нажатия Ctrl+D

Создание (текстовых) файлов

- 1 Любимым редактором: **vi/vim**, **nano**, **mcedit**
- 2 **echo**

```
echo какойто— текст >file
```

- 3 **cat** с перенаправлением⁷

```
~$ cat >>file3  
ladsfa  
134097  
  
~$ ls -l file3  
-rw-r—r— 1 user user 25 Янв 23 17:13 file3
```

- 4 **touch** - создать пустой файл

```
~$ touch file4  
~$
```



⁷ До ``конца ввода'', т.е. нажатия Ctrl+D

Операции над каталогами (и файлами)

- **mkdir** - создать каталог

```
~$ mkdir dir1 /tmp/somedir  
~$ mkdir -p dir/and/existant/parts/in/path
```



⁸с ключом **-r**) и каталогов

Операции над каталогами (и файлами)

- **mkdir** - создать каталог

```
~$ mkdir dir1 /tmp/somedir  
~$ mkdir -p dir/and/existant/parts/in/path
```

- **rmdir** - удалить (пустой) каталог

```
~$ rmdir dir1 /tmp/somedir  
~$ rmdir -p deep/empty/dir/structure/
```



⁸с ключом **-r**) и каталогов

Операции над каталогами (и файлами)

- **mkdir** - создать каталог

```
~$ mkdir dir1 /tmp/somedir  
~$ mkdir -p dir/and/existant/parts/in/path
```

- **rmdir** - удалить (пустой) каталог

```
~$ rmdir dir1 /tmp/somedir  
~$ rmdir -p deep/empty/dir/structure/
```

- **cp** - копирование файлов⁸
- **mv** - перемещение и переименование файлов
- **rm** - удаление файлов¹⁷

```
~$ rm -rf dir1 /tmp/somedir  
~$ rm f*
```

⁸ с ключом **-r**) и каталогов



Операции над каталогами (и файлами)

- **mkdir** - создать каталог

```
~$ mkdir dir1 /tmp/somedir  
~$ mkdir -p dir/and/existant/parts/in/path
```

- **rmdir** - удалить (пустой) каталог

```
~$ rmdir dir1 /tmp/somedir  
~$ rmdir -p deep/empty/dir/structure/
```

- **cp** - копирование файлов⁸
- **mv** - перемещение и переименование файлов
- **rm** - удаление файлов¹⁷

```
~$ rm -rf dir1 /tmp/somedir  
~$ rm f*
```

⁸ с ключом **-r**) и каталогов

