

№7.

Linux.

01 1970

‘ ‘
‘ ‘

1.

2.

3.

chmod

4.

5.

man

■ Linux, , .
(),
- .

~/abc1

april

may:

```
thouston@username:~$ cd
thouston@username:~$ touch abc1
thouston@username:~$ ls
abc1      git-extended  ls      Pictures  Документы  Общедоступные
bin       git_repo     misk    project_OC Загрузки   'Рабочий стол'
Desktop   letters      nemos   work      Изображения Шаблоны
Downloads LICENSE     newdir  Видео     Музыка

thouston@username:~$ cp abc1 april
thouston@username:~$ ls
abc1      Downloads  LICENSE  newdir    Видео     Музыка
april     git-extended  ls      Pictures  Документы  Общедоступные
bin       git_repo     misk    project_OC Загрузки   'Рабочий стол'
Desktop   letters      nemos   work      Изображения Шаблоны

thouston@username:~$ cp abc1 may
thouston@username:~$ ls
abc1      git-extended  may      project_OC  Изображения
april     git_repo     misk     work        Музыка
bin       letters      nemos    Видео       Общедоступные
```

- `/usr/include/sys/io.h` `equipment.` `io.h` ,
`/usr/include/sys/` .

```
thouston@username:~1$ mv ~/ski.places/equipment ~/ski.places/equiplist
thouston@username:~$ ls ski.places/
equiplist
thouston@username:~$ touch abc1
thouston@username:~$
thouston@username:~$ cp ~/ski.places/equiplist2
cp: после '/home/thouston/ski.places/equiplist2' пропущен операнд, задающий целевой файл
По команде «cp --help» можно получить дополнительную информацию.
thouston@username:~1$ cp ~/ski.places/equiplist2
cp: после '/home/thouston/ski.places/equiplist2' пропущен операнд, задающий целевой файл
По команде «cp --help» можно получить дополнительную информацию.
thouston@username:~1$ cp abc1 ~/ski.places/equiplist2
thouston@username:~$ ls skip.places/
ls: невозможно получить доступ к 'skip.places/': Нет такого файла или каталога
thouston@username:~2$ ls ski.places/
equiplist equiplist2
thouston@username:~$
```

■ /etc/passwd.

```

MKFS(8)                                     System Administration                                     MKFS(8)

NAME
    mkfs - build a Linux filesystem

SYNOPSIS
    mkfs [options] [-t type] [fs-options] device [size]

DESCRIPTION
    This mkfs frontend is deprecated in favour of filesystem specific mkfs.<type> utils.

    mkfs is used to build a Linux filesystem on a device, usually a hard disk partition. The device argument is either the device name (e.g., /dev/hda1, /dev/sdb2), or a regular file that shall contain the filesystem. The size argument is the number of blocks to be used for the filesystem.

    The exit status returned by mkfs is 0 on success and 1 on failure.

    In actuality, mkfs is simply a front-end for the various filesystem builders (mkfs.fstype) available under Linux. The filesystem-specific builder is searched for via your PATH environment setting only. Please see the filesystem-specific builder manual pages for further details.

OPTIONS
    -t, --type type
        Specify the type of filesystem to be built. If not specified, the default filesystem type (currently ext2) is used.

    fs-options
        Filesystem-specific options to be passed to the real filesystem builder.

    -V, --verbose
        Produce verbose output, including all filesystem-specific commands that are executed. Specifying this option more than once inhibits execution of any filesystem-specific commands. This is really only useful for testing.
```

- man mount, fsck, mkfs, kill, .
.25
. 25: .25



■

Linux

.

.

1. Windows []. URL:
<https://foxford.ru/wiki/informatika/komandnaya-stroka-windows>.

