

# Основы построения файловых систем



## POSIX filesystem API

UNIX – многопользовательская ОС, поэтому требуется разделение доступа к файлам.

## POSIX filesystem API

UNIX – многопользовательская ОС, поэтому требуется разделение доступа к файлам.

Модель безопасности:

- имеется множество пользователей и групп, в которых пользователи состоят,
- каждый файл принадлежит одному пользователю и одной группе,
- файл указывает, какой доступ разрешён пользователю-владельцу, группе-владельцу и всем остальным.

# POSIX filesystem API

UNIX – многопользовательская ОС, поэтому требуется разделение доступа к файлам.

Модель безопасности:

- имеется множество пользователей и групп, в которых пользователи состоят,
- каждый файл принадлежит одному пользователю и одной группе,
- файл указывает, какой доступ разрешён пользователю-владельцу, группе-владельцу и всем остальным.

```
$ ls -lh /usr/bin
...
-rwx  r-x  --x 1 root  root    31K Feb 22  2016 afm2pl
-rwx  r-x  --x 1 root  root    38K Feb 22  2016 afm2tfm
-rwx  r-x  --x 1 root  root   485K Feb 22  2016 aleph
```

The diagram illustrates the permission string `--x` for 'other' users. Three blue arrows point from the labels below to the characters in the string: the first arrow points to the first dash, the second to the second dash, and the third to the 'x'. The labels are 'User permissions' (pointing to the first dash), 'Group permissions' (pointing to the second dash), and 'Permissions for others' (pointing to the 'x').

## POSIX filesystem API

Имеются «права доступа», которые меняют то, как запускаются программы:

```
$ ls -lh /usr/bin
-rwsrwxrwx 1 root root 134K Jan 5 2016 sudo
-rwxrwxrwx 1 root root 4 Jan 5 2016 sudoedit -> sudo
-rwxr-xr-x 1 root root 47K Jan 5 2016 sudoreplay
```

## POSIX filesystem API

Имеются «права доступа», которые меняют то, как запускаются программы:

```
$ ls -lh /usr/bin
-rwsr-xr-x 1 root root 134K Jan 5 2016 sudo
-rwxrwxrwx 1 root root 4 Jan 5 2016 sudoedit -> sudo
-rwxr-xr-x 1 root root 47K Jan 5 2016 sudoreplay
```

При запуске файла с установленным флагом set-uid (соотв., set-gid) он будет запущен от имени пользователя-владельца (соотв., группы-владельца).

## POSIX filesystem API

Права доступа к файлу и файловому дескриптору разделены:

```
int fd = open("/path/to/a/file", O_RDWR, S_IRUSR);  
write(fd, buffer, size);  
close(fd);
```

## POSIX filesystem API

Права доступа к файлу и файловому дескриптору разделены:

```
int fd = open("/path/to/a/file", O_RDWR, S_IRUSR);  
write(fd, buffer, size);  
close(fd);
```

Где применяется:

- Простая привилегированная программа проверяет права доступа и передаёт файловый дескриптор (сложной) непривилегированной программе.
- Помогает в реализации binfmt-обработчиков для файлов с правами доступа --x--x--x:  
[https://www.kernel.org/doc/Documentation/binfmt\\_misc.txt](https://www.kernel.org/doc/Documentation/binfmt_misc.txt)



## POSIX filesystem API

Есть более гибкое API для управления доступом: ACLs (Access Control Lists):

- к файлу привязывается список пользователей и действий, разрешённых тем пользователям;
- можно наследовать права доступа от родительского каталога.

См. [https://access.redhat.com/documentation/en-US/Red\\_Hat\\_Storage/2.0/html/Administration\\_Guide/ch09s05.html](https://access.redhat.com/documentation/en-US/Red_Hat_Storage/2.0/html/Administration_Guide/ch09s05.html) и [https://linux.die.net/man/5/nfs4\\_acl](https://linux.die.net/man/5/nfs4_acl) .

### Что ФС не должна делать

- Интерпретировать/изменять содержимое файлов
- Интерпретировать/изменять имена файлов

### Что ФС не должна делать

- Интерпретировать/изменять содержимое файлов
- Интерпретировать/изменять имена файлов

### И что делают Windows (VFAT, NTFS) и Mac (HFS)

Case-insensitive & case-preserving FS

### Case-insensitivity: почему так делать не надо

Необходимо сравнивать строки без учёта регистра. Для этого требуется:

- Знание всех языков из Unicode,
- Разбираться с неоднозначностью записи букв: «á» может быть одним символом, а может быть объединением символов «а» и «'»,
- Правило преобразования большой малых букв в заглавные зависит не от алфавита, а от региона в стране,
- В новых версиях Unicode правила нормализации и приведения к заглавным буквам меняются.

### Case-insensitivity: почему так делать не надо

Необходимо сравнивать строки без учёта регистра. Для этого требуется:

- Знание всех языков из Unicode,
- Разбираться с неоднозначностью записи букв: «á» может быть одним символом, а может быть объединением символов «a» и «'»,
- Правило преобразования большой малых букв в заглавные зависит не от алфавита, а от региона в стране,
- В новых версиях Unicode правила нормализации и приведения к заглавным буквам меняются.
- В Windows при создании раздела NTFS в него записываются таблицы нормализации и преобразования регистра.

### Case-insensitivity: почему так делать не надо

Необходимо сравнивать строки без учёта регистра. Для этого требуется:

- Знание всех языков из Unicode,
  - Разбираться с неоднозначностью записи букв: «á» может быть одним символом, а может быть объединением символов «а» и «'»,
  - Правило преобразования большой малых букв в заглавные зависит не от алфавита, а от региона в стране,
  - В новых версиях Unicode правила нормализации и приведения к заглавным буквам меняются.
- 
- **Правильное решение:** не вмешиваться в именование файлов – пусть пользовательские программы заботятся об этом, если им надо.