

Каталог dev

Представим накопитель нашего компьютера — жёсткий диск (HDD) или твердотельный накопитель (SSD). На нём расположены некоторые данные. Единица хранения информации — это файл, а система организации — файловая система.

Файловая система управляет тем, как информация сохраняется и считывается. Файловая система отвечает за то, что связывает имя файла (то, чем оперирует пользователь) с адресами блоков на диске (логической структурой диска, которой оперирует Linux), где фактически хранится информация, при помощи иноды. В inode также хранится информация о файле: кто может получить к нему доступ и какой, когда файл был модифицирован и так далее.

Файлы могут быть организованы в директории (директории связывают номер иноды с именем файла). В свою очередь, директории могут быть организованы в иерархическую структуру.

Когда к операционной системе подключается устройство, она пытается распознать устройство и подгрузить для него необходимый драйвер. Далее для устройства создаётся запись в **директории /dev**.

То есть для пользователя работа с простыми файлами (например, текстовыми) и с устройствами (например, с жёстким диском, оптическим диском и так далее) происходит по одному и тому же интерфейсу — файловому.

Итак, в директории /dev находятся следующие типы файлов:

- Символьные устройства.
- Блочные устройства.
- Псевдоустройства.

Блочные файлы устройств используются для передачи данных, разделённых на пакеты фиксированной длины — блоки. А символьные файлы устройств используются для небуферизованного (прямого) обмена данными. Большинство устройств способно принимать и отправлять данные либо блоками (блочные устройства), либо сплошным потоком байтов (символьные устройства), но некоторые (такие как жёсткий диск) сочетают в себе обе эти возможности.

Работа с первым типом устройств возможна либо через блочные, либо через символьные файлы, а вот для работы с последним типом подходят и те и другие.

Псевдоустройства позволяют получать доступ к различным полезным функциям операционной системы, используя файловый интерфейс. Они создаются в момент загрузки операционной системы и существуют всё время её работы.

/dev/null (пустое устройство) — специальная «чёрная дыра», запись в которую любого количества информации завершится успешно, однако прочитать оттуда будет ничего нельзя.

Посмотрим, что нам о нём расскажет утилита **ls**:

```
ls -la /dev/null
```

Перед правами доступа здесь стоит буква **s**. Буква **s** в этой позиции обозначает символическую ссылку, **d** — директорию, а **s** — символьное устройство.

Мы можем легко очистить файл, перенаправив в него содержимое **/dev/null**:

```
cat /dev/null > textfile
```

Далее файлы **/dev/random** и **/dev/urandom**. Оба этих псевдоустройства имеют схожее назначение — они отдают случайные символы.

На практике чаще всего используется **/dev/urandom** — он быстрее и удобнее для использования.

Давайте попробуем прочитать из этого устройства немного данных:

```
head /dev/urandom
```

/dev/zero — хотя этот файл очень похож названием на **/dev/null**, у него другое назначение. Он отдаёт нулевые байты.

Файлы прочих устройств

/dev/dsp — это символьное устройство, предоставляющее интерфейс для программ, которые проигрывают звук и звуковую карту (как видите, у нас такого устройства нет).

/dev/fb0 — это устройство фреймбуфера. Фреймбуфер — это прослойка между программами, которые рисуют графику (например, наш графический интерфейс), и видеопамятью.

/dev/loop0 — это очень интересное устройство, так называемый **loopback device**, или петлевое устройство. Оно позволяет нам подключать файловые системы, не расположенные на физических носителях.

/dev/psaux — устройство для подключения мыши по порту PS/2 (Этот разъём, конечно, безнадежно устарел — сейчас мышь подключается при помощи Bluetooth или USB. Однако обратная совместимость в Linux позволяет заглянуть в историю.)

/dev/sda — это первый жёсткий диск. Последующие диски будут называться **sdb**, **sdc** и так далее.

/dev/tty — таких устройств под различными номерами, как и в случае с **/dev/loop** довольно много. Это устройство отвечает за запуск терминала.

Существуют другие виды устройств. Например, устройство **/dev/js0** — джойстик. Файлы в директории **/dev** — особенные. Их нельзя создать при помощи команды **touch**.

Файлы в директории **/dev** характеризуются не только именем, но и классом устройства:

- Старшим (**major**) номером устройства.
- Младшим (**minor**) номером устройства.

Эти параметры записаны в самом файле. Класс устройства определяет символическое это устройство или блочное.

Старший номер определяет, чем является устройство — оптическим диском, жёстким диском, джойстиком и так далее.

Младший номер различает несколько устройств одного класса. Например, если у нас несколько дисков, то они будут иметь разные младшие номера.

Для создания специальных файлов устройств используют команду `mknod`. Ей нужно передать класс устройства и старший и младший номера.

Попробуем создать ещё один `/dev/null`.

`/dev/null` описывается номерами 1 и 3. Найти старшие и младшие номера для других устройств вы можете при помощи поисковых систем.

```
sudo mknod newnull c 1 3
```

Добавим право на запись:

```
sudo chmod a+w newnull
```

Давайте теперь проверим, что у нас создался не обычный файл:

```
ls -la newnull
```

И теперь попробуем что-нибудь в него записать:

```
echo "aaaaaaaa" > newnull; echo $?
```

Запись прошла успешно.

```
cat newnull
```

Но в выводе мы ничего не получаем. Строка успешно записалась в «чёрную дыру», как и в случае с устройством `/dev/null`.

Работа с подключаемыми устройствами в Linux ведётся через файловый интерфейс. Файлы, предоставляющие доступ к устройствам, находятся в специальной директории `/dev`.

Кроме «настоящих» устройств, например дисков, мыши и так далее, в директории `dev` есть псевдоустройства, например `/dev/null` и `/dev/zero`, которые помогают нам в решении распространённых задач.