

## Лабораторная работа. Получение информации об оборудовании.

### Задачи

1. Вывод информации об оборудовании
2. Исследование файловой системы /proc

### Описание

Ниже приведены команды для работы в ОС Linux в консольном режиме. Изучите приведенные команды и отработайте их в командной строке Linux. При выполнении работы проявите творчество и поэкспериментируйте с командами.

### Заметка для преподавателя

Для работы с командой fdisk потребуется путь к блочному устройству (диску) или его разделу (/dev/sdXY).

### Получение информации об оборудовании

Для определения типа установленного процессора выполните команду lscpu.

```
username@linux-pc:~$ lscpu
```

А также выведите на экран информацию из файла /proc/cpuinfo.

```
username@linux-pc:~$ cat /proc/cpuinfo
```

Для определения используемой оперативной памяти и области подкачки используйте команду free. Опции -m и -g позволят увидеть информацию в мегабайтах или гигабайтах.

Для вывода информации в человекопонятном виде, используйте опцию -h.

```
username@linux-pc:~$ free -h
```

Для вывода списка подключенных к материнской плате устройств используйте команду lspci.

```
username@linux-pc:~$ lspci
```

Вывод команды lspci может быть очень важным для идентификации устройств, которые не поддерживаются ядром Linux. Некоторые устройства, такие как видеокарты, могут предоставлять базовые функции без установки проприетарного драйвера.

Опция -k позволяет увидеть используемые модули ядра и драйвер.

```
username@linux-pc:~$ lspci -k
```

Для просмотра подключенных USB устройств применяется команда lsusb.

Однако, если USB устройств нет, то вы увидите сообщение: Unable to initialize libusb: -99

```
username@linux-pc:~$ lsusb
```

Для аппаратного обеспечения ядро Linux обычно загружает драйвер или модуль. Используйте команду lsmod для просмотра загруженных модулей.

Первый столбец - это имя модуля, а второй - объем памяти, используемый модулем. Число в столбце «Используется» указывает, сколько других модулей использует данный модуль. Имена других используемых модулей могут быть указаны в столбце «Используется», но часто являются неполными.

```
username@linux-pc:~$ lsmod
```

Системная плата многих компьютеров содержит BIOS — базовую систему ввода-вывода. System Management BIOS или SMBIOS - это стандарт, определяющий структуры данных и способы передачи информации об аппаратном обеспечении компьютера. Команда dmidecode может считывать и печатать информацию из SMBIOS многих систем. Мы не будем демонстрировать команду здесь, так как она не работает должным образом в виртуальной среде.

```
username@linux-pc:~$ dmidecode
```

Команда fdisk предназначена для получения информации о блочных устройствах (дисках) и управления ими.

Когда опция -l используется с fdisk, команда будет неинтерактивно отображать блочные устройства, включая диски и логические тома.

```
username@linux-pc:~$ fdisk -l /dev/sdX
```

Без опции -l команда fdisk вводит интерактивный режим, который обычно используется для изменения разделов на диске.

Выполните команду fdisk, чтобы отобразить дисковые устройства не интерактивно, объем в секторах и без предупреждений о совместимости с DOS.

```
username@linux-pc:~$ fdisk -cul
```

Опция -u заставляет команду fdisk отображать блоки в секторах вместо цилиндров. Параметр -c запрещает команде fdisk печатать предупреждения, которые влияют на совместимость с DOS.

## Ядро и файловая система /proc

Рассмотрим некоторые файлы, содержащиеся в каталоге /proc.

```
username@linux-pc:~$ ls /proc
```

Напомним, что каталоги, которые имеют номера для имен, представляют собой запущенные процессы в системе. Первый процесс всегда /sbin/init, поэтому каталог /proc/1 будет содержать файлы с информацией о запущенном процессе init.

Файл cmdline внутри каталога процесса (например, /proc/1/cmdline) покажет команду, которая была выполнена. Порядок запуска других процессов сильно варьируется от системы к системе. Поскольку содержимое этого файла не содержит нового символа строки, будет выполнена команда echo, чтобы заставить приглашение перейти к новой строке.

```
username@linux-pc:~$ cat /proc/1/cmdline; echo
```

```
username@linux-pc:~$ ps -p 1
```

Просмотрите файл `/proc/cmdline`, чтобы узнать, какие аргументы были переданы ядру во время загрузки.

```
username@linux-pc:~$ cat /proc/cmdline
```