

## Команды Linux. Просмотр сведений о системе.

Какая ОС установлена

```
$ cat /etc/os-release
```

```
$ cat /proc/version
```

### Команда uname

uname — это утилита командной строки, которая выводит основную информацию об имени операционной системы и аппаратном обеспечении системы. Синтаксис команды uname имеет следующий вид:

```
uname [OPTIONS]...
```

Возможные опции:

-s , ( --kernel-name ) — печатает имя ядра.

-n , ( --nodename ) — печатает имя узла системы (имя хоста). Это имя, которое система использует при обмене данными по сети. При использовании с параметром -n uname дает тот же результат, что и команда hostname .

-r , ( --kernel-release ) — печатает версию ядра.

-v , ( --kernel-version ) — --kernel-version ядра.

-m , ( --machine ) — печатает имя аппаратного имени машины.

-p , ( --processor ) — печатает архитектуру процессора.

-i , ( --hardware-platform ) — печатает аппаратную платформу.

-o , ( --operating-system ) — --operating-system имя операционной системы. В системах Linux это «GNU / Linux»

-a , ( --all ) — При использовании параметра -a uname ведет себя так же, как если бы были заданы параметры -snrvmo .

Пример:

```
$ uname -a
```

### Команда lshw

lshw Утилита или инструмент, устанавливаемый по умолчанию при установке Linux, предоставляет вам список оборудования, которое есть в вашей системе.

```
$ sudo lshw
```

```
$ sudo lshw -short
```

Запишем вывод команды в файл, затем посмотрим:

```
$ sudo lshw -html > system-info.txt
```

```
$ less system-info.txt
```

Информация об оперативной памяти в Linux. Свободная, занятая и тип памяти

Команда free

По умолчанию объем памяти выводится в килобайтах. Используя опции, можно выводить объем памяти в других форматах.

Некоторые опции:

-m — в мегабайтах

-g — в гигабайтах

-h — автоматически определить формат

Пример:

```
$ free -h
```

```
$ free -m
```

### Команда top

top — это утилита командной строки, которая используется для мониторинга процессов и используемых ресурсов компьютера.

Запуск утилиты top:

```
$ top
```

В заголовке выводится информация об использованной оперативной памяти.

Опций запуска у команды не много и использовать их активно не принято, потому что большинство действий выполняются с помощью интерактивных команд. Вот основные опции:

- v - вывести версию программы;
- b - режим только для вывода данных, программа не воспринимает интерактивных команд и выполняется пока не будет завершена вручную;
- c - отображать полный путь к исполняемым файлам команд;
- d - интервал обновления информации;
- H - включает вывод потоков процессов;
- i - не отображать процессы, которые не используют ресурсы процессора;
- n - количество циклов обновления данных, после которых надо закрыть программу;
- o - поле, по которому надо выполнять сортировку;
- O - вывести все доступные поля для сортировки;
- p - отслеживать только указанные по PID процессы, можно указать несколько PID;
- u - выводить только процессы, запущенные от имени указанного пользователя.

команды, которые вы можете выполнять во время работы программы.

- h - вывод справки по утилите;
- q или Esc - выход из top;
- A - выбор цветовой схемы;
- d или s - изменить интервал обновления информации;
- H - выводить потоки процессов;
- k - послать сигнал завершения процессу;
- W - записать текущие настройки программы в конфигурационный файл;
- Y - посмотреть дополнительные сведения о процессе, открытые файлы, порты, логи и т д;
- Z - изменить цветовую схему;
- I - скрыть или вывести информацию о средней нагрузке на систему;
- m - выключить или переключить режим отображения информации о памяти;
- x - выделять жирным колонку, по которой выполняется сортировка;
- y - выделять жирным процессы, которые выполняются в данный момент;
- z - переключение между цветным и одноцветным режимами;
- c - переключение режима вывода команды, доступен полный путь и только команда;
- F - настройка полей с информацией о процессах;
- o - фильтрация процессов по произвольному условию;
- u - фильтрация процессов по имени пользователя;
- V - отображение процессов в виде дерева;
- i - переключение режима отображения процессов, которые сейчас не используют ресурсы процессора;
- n - максимальное количество процессов, для отображения в программе;
- L - поиск по слову;
- <> - перемещение поля сортировки вправо и влево;

### Команда dmidecode

Команда dmidecode позволяет использовать утилиту с соответствующим именем для получения информации об аппаратном обеспечении компьютера. Данная утилита читает либо специальную файловую систему sysfs, либо специальный файл устройства /dev/mem, предоставляющий доступ к физической памяти компьютера. Из них утилита извлекает данные, связанные с таблицей DMI.

**sudo dmidecode -t 1**

**sudo dmidecode system**

Тип	Информация
0	BIOS
1	Система
2	Материнская плата
3	Корпус
4	Процессор

5	Контроллер памяти
6	Модуль памяти
7	Кэш
8	Коннекторы портов
9	Системные слоты
10	Интегрированные устройства
11	Строки OEM
12	Параметры системной конфигурации
13	Язык BIOS
14	Ассоциации групп
15	Журнал системных событий
16	Массив физической памяти
17	Устройство памяти
18	32-битные ошибки доступа к памяти
19	Отображенный адрес массива памяти
20	Отображенный адрес устройства памяти
21	Встроенное указывающее устройство
22	Батарея мобильного устройства
23	Устройство сброса состояния системы
24	Устройства безопасности
25	Управление питанием системы
26	Датчик напряжения
27	Устройство охлаждения
28	Датчик температуры
29	Датчик тока
30	Механизм удаленного доступа
31	Сервисы проверки целостности данных загрузки
32	Загрузочные устройства
33	64-битные ошибки доступа к памяти
34	Устройство управления
35	Компонент устройства управления
36	Граничные данные устройства управления
37	Канал памяти
38	Устройство IPMI
39	Блок питания
40	Дополнительная информация
41	Дополнительная информация об интегрированных устройствах
42	Хост-интерфейс контроллера управления
126	Деактивированная строка
127	Маркер конца таблицы

Утилита принимает не только приведенные выше идентификаторы, но и кодовые слова, являющиеся псевдонимами для одного или нескольких идентификаторов.

Кодовое слово

bios

system

baseboard

chassis

processor

memory

cache

connector

slot

### Как посмотреть список дисков в Linux

Жёсткие диски имеют особенные названия. В зависимости от интерфейса, через который подключён жёсткий диск, название может начинаться на:

sd - устройство, подключённое по SCSI;  
hd - устройство ATA;  
vd - виртуальное устройство;  
mmcblk - обозначаются флешки, подключённые через картридер;

Простой способ увидеть все подключённые диски - это посмотреть содержимое каталога /dev/ и отфильтровать устройства sd:

```
$ ls -l /dev/
```

```
$ ls -l /dev/ | grep sd
```

**Утилита df** поставляется по умолчанию во всех дистрибутивах Linux и имеет очень простой синтаксис. Фактически вы можете просто набрать df и уже получить результат, но чтобы сделать вывод более читаемым используются дополнительные опции. Вот основной синтаксис:

```
$ df [опции] [устройство]
```

Например:

```
$ df -h
```

основные опции утилиты:

- -a, --all - отобразить все файловые системы, в том числе виртуальные, псевдо и недоступные;
- -B - изменить размер одного блока перед выводом данных, например, можно использовать BM, чтобы вывести все данные в мегабайтах;
- -h - выводить размеры в читаемом виде, в мегабайтах или гигабайтах;
- -H - выводить все размеры в гигабайтах;
- -i - выводить информацию об inode;
- -k - выводить размеры в килобайтах;
- --output - использовать специальный формат вывода, если не задано, выводит все поля. Доступны такие варианты: 'source', 'fstype', 'itotal', 'iused', 'iavail', 'ipcent', 'size', 'used', 'avail', 'pcent', 'file' и 'target';
- -P - использовать формат вывода POSIX;
- --total - выводить всю информацию про использованное и доступное место;
- -t, --type - выводить информацию только про указанные файловые системы;
- -x - выводить информацию обо всех, кроме указанных файловых систем;

**Команда lsblk** позволяет получить информацию о блочных устройствах. Блочное устройство — это термин для обозначения устройства хранения данных, которое считывает или записывает данные в блоках определенного размера. Этот термин относится почти к каждому типу энергонезависимой памяти, в том числе к жестким дискам (HDD), твердотельным накопителям (SSD), флэш-памяти и так далее.

узнать ещё больше с помощью команды lsblk:

```
$ lsblk
```

Можно получить информацию об отдельном диске, если передавать его в качестве аргумента

```
$ lsblk /dev/sdb
```

Опция -f или --fs позволяет добавить в вывод информацию о файловых системах

```
$ lsblk -f
```

**fdisk** - простая утилита для разметки диска в терминале. Синтаксис утилиты ничем не отличается от других команд Linux:

\$ fdisk [опции] [устройство]

опции fdisk:

- -B, --protect-boot - не стирать первые 512 байт диска чтобы не повредить загрузочную запись;
- -L, --color - настройка цветного вывода, возможные значения auto, never или always;
- -l, --list - вывести все разделы на выбранных устройствах или если устройств не задано, то на всех устройствах;
- -o, --output - указывает какие поля данных надо показывать в выводе программы, доступные поля рассмотрим ниже;
- -u, --units - настраивает формат вывода размера разделов, доступные значения: cylinders, sectors, по умолчанию используется sectors;
- -w, --wipe - режим стирания файловой системы или RAID с диска, возможные значения auto, never или always по умолчанию используется auto;
- -W, --wipe-partition - режим стирания файловой системы или RAID из только что созданного раздела. Возможные значения аналогичны предыдущей опции;
- -h, --help - показать справку по утилите;
- -v, --version - опция выводит версию утилиты.

Опции используются для настройки работы утилиты, мы рассмотрели только самые главные, которые могут вам пригодиться, теперь перейдём к командам:

- a - включение или выключения флага boot для раздела;
- d - удалить раздел;
- F - показать свободное место;
- l - вывести список известных типов разделов;
- n - создать новый раздел;
- p - вывести таблицу разделов;
- t - изменение типа раздела;
- i - вывести информацию о разделе;
- l и O - записать или загрузить разметку в файл сценария sfdisk;
- w - записать новую таблицу разделов на диск;
- q - выйти без сохранения;
- g - создать пустую таблицу разделов GPT;
- o - создать пустую таблицу разделов MBR.

Пример. Посмотреть список доступных вам дисков и разделов:

\$ fdisk -l