



[Главная](#) >> [Команды](#) >> Команда time в Linux

Команда time в Linux

Опубликовано: 18 марта, 2022 от [zeninvlad](#) , 1 комменариев, время чтения: 10 минут

Обнаружили ошибку в тексте? Сообщите мне об этом. Выделите текст с ошибкой и нажмите Ctrl+Enter.

В некоторых случаях при работе с терминалом Linux нужно знать время выполнения тех или иных команд, например, для отслеживания проблем. Для этой задачи существует утилита time. Конкретно про нее и пойдет речь в нашей статье.

Мы расскажем, как работает команда time Linux и какой у неё синтаксис. Затем перейдем к описанию доступных опций. А в конце упомянем несколько популярных сценариев использования.

Содержание статьи

- [Синтаксис и опции time](#)
- [Опции форматирования вывода](#)
- [Как узнать время выполнения команды Linux](#)
 - [1. Вывод времени выполнения в терминал](#)
 - [2. Вывод в файл](#)
 - [3. Вывод с форматированием](#)
- [Выводы](#)

Конфиденциальность -
Условия использования

Privacy

Синтаксис и опции time

Утилита запускает заданную пользователем команду и после этого выводит информацию о времени ее выполнения. У нее достаточно удобный синтаксис. Сначала нужно указать опции для `time`, затем – выполняемую команду, а в конце – аргументы к ней:

```
$ time опции команда_для_выполнения аргументы
```

Рассмотрим список доступных опций:

- **-o, --output** – сохранять данные в выбранный файл вместо стандартного вывода в терминале. При этом старые данные в файле будут перезаписаны.
- **-a, --append** – добавлять в файл новую информацию, а не перезаписывает старую. Опция полезна только в сочетании с **-o**.
- **-f, --format** – выбрать определенный формат вывода. Подробности о форматировании описаны в следующем разделе статьи.
- **-p, --profitably** – использовать формат вывода данных для соответствия со стандартом POSIX 1003.2.
- **-v, --verbose** – выводить подробную информацию о выполнении программы.
- **-V, --version** – вывести версию утилиты `time`.

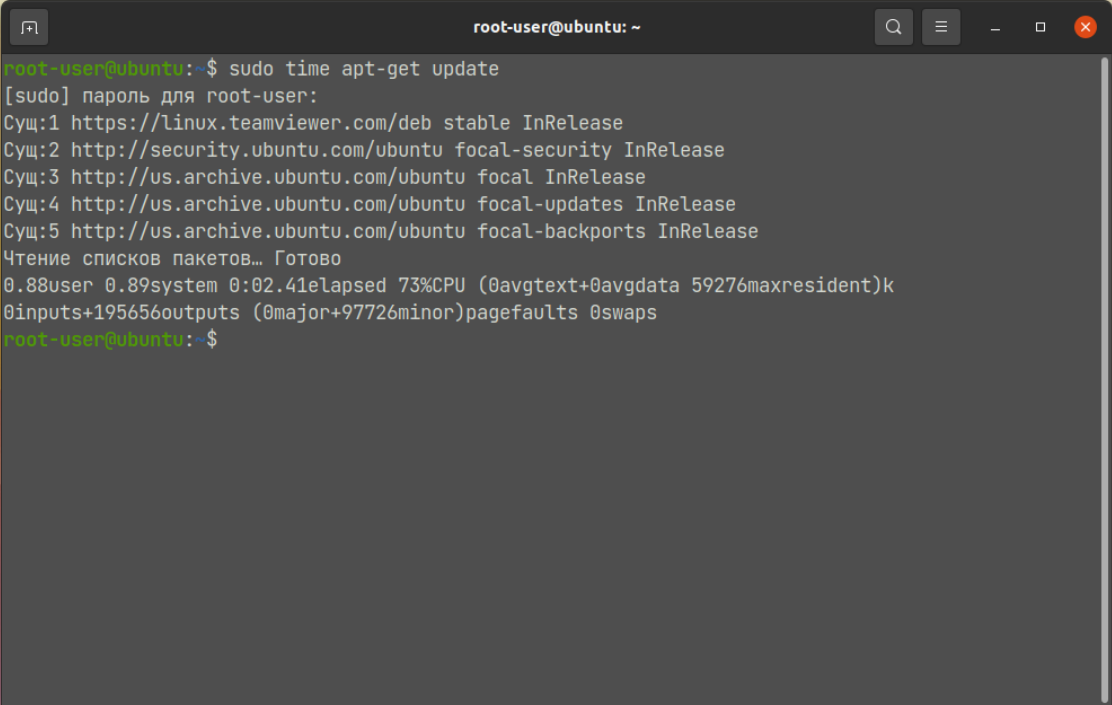
В этом списке представлены только основные опции. Получить подробную информацию можно командой:

```
$ man time
```

Опции форматирования вывода

По умолчанию `time` может выводить информацию в не комфортном для чтения виде.





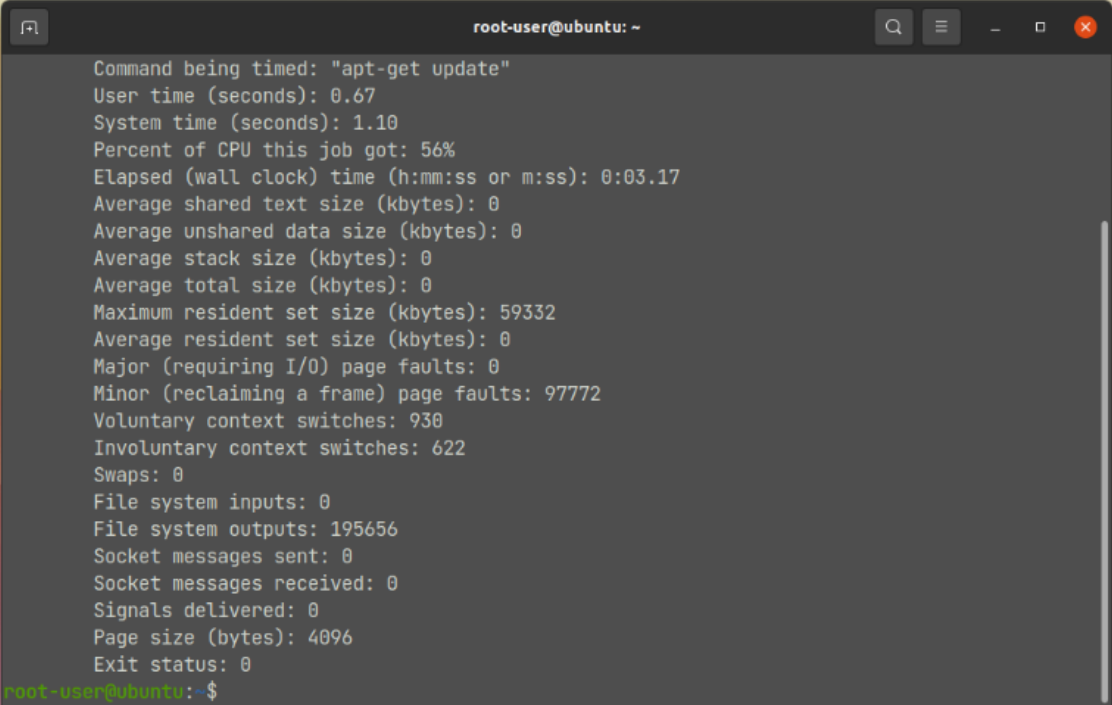
```
root-user@ubuntu: ~  
root-user@ubuntu:~$ sudo time apt-get update  
[sudo] пароль для root-user:  
Сущ:1 https://linux.teamviewer.com/deb stable InRelease  
Сущ:2 http://security.ubuntu.com/ubuntu focal-security InRelease  
Сущ:3 http://us.archive.ubuntu.com/ubuntu focal InRelease  
Сущ:4 http://us.archive.ubuntu.com/ubuntu focal-updates InRelease  
Сущ:5 http://us.archive.ubuntu.com/ubuntu focal-backports InRelease  
Чтение списков пакетов... Готово  
0.88user 0.89system 0:02.41elapsed 73%CPU (0avgtext+0avgdata 59276maxresident)k  
0inputs+195656outputs (0major+97726minor)pagefaults 0swaps  
root-user@ubuntu:~$
```

По этой причине для нее желательно задавать опции форматирования вывода, о которых и пойдет сейчас речь. Всего их три штуки. В качестве примера для их рассмотрения возьмем команду `apt-get update`.

Опция `-v` используется для вывода подробной информации:

```
$ sudo time -v apt-get update
```

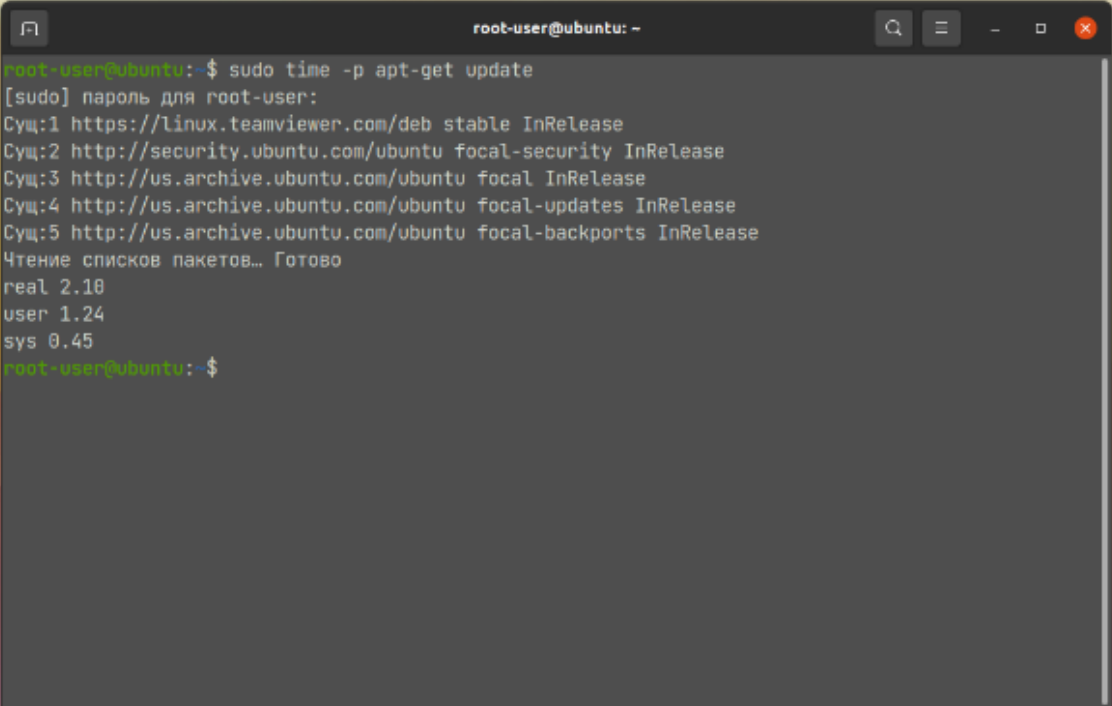




```
root-user@ubuntu: ~  
Command being timed: "apt-get update"  
User time (seconds): 0.67  
System time (seconds): 1.10  
Percent of CPU this job got: 56%  
Elapsed (wall clock) time (h:mm:ss or m:ss): 0:03.17  
Average shared text size (kbytes): 0  
Average unshared data size (kbytes): 0  
Average stack size (kbytes): 0  
Average total size (kbytes): 0  
Maximum resident set size (kbytes): 59332  
Average resident set size (kbytes): 0  
Major (requiring I/O) page faults: 0  
Minor (reclaiming a frame) page faults: 97772  
Voluntary context switches: 930  
Involuntary context switches: 622  
Swaps: 0  
File system inputs: 0  
File system outputs: 195656  
Socket messages sent: 0  
Socket messages received: 0  
Signals delivered: 0  
Page size (bytes): 4096  
Exit status: 0  
root-user@ubuntu:~$
```

Опция **-p** нужна для вывода данных в формате, соответствующем стандарту POSIX 1003.2:

```
$ sudo time -p apt-get update
```



```
root-user@ubuntu:~$ sudo time -p apt-get update  
[sudo] пароль для root-user:  
Сущ:1 https://linux.teamviewer.com/deb stable InRelease  
Сущ:2 http://security.ubuntu.com/ubuntu focal-security InRelease  
Сущ:3 http://us.archive.ubuntu.com/ubuntu focal InRelease  
Сущ:4 http://us.archive.ubuntu.com/ubuntu focal-updates InRelease  
Сущ:5 http://us.archive.ubuntu.com/ubuntu focal-backports InRelease  
Чтение списков пакетов... Готово  
real 2.10  
user 1.24  
sys 0.45  
root-user@ubuntu:~$
```

А с помощью опции **-f** или **--format** детально задается форматирование вывода. Этот момент стоит рассмотреть более детально.

Строка с данными о форматировании, как правило, включает в себя спецификаторы ресурсов и обычный текст. Знак **%** обозначает, что следующий после него символ следует воспринимать как спецификатор ресурсов.



С помощью знака **** задается разделительный символ. Есть три доступных варианта: **\t** – табуляция, **\n** – новая строка, **** – обратная косая черта. Если после **** указать любой другой символ, то в терминале появится вопросительный знак (**?**), который говорит об ошибке ввода.

Остальной текст в строке формата полностью копируется в поле вывода. При этом **time** всегда начинает вывод данных с новой строки после информации о выполнении самой команды.

Рассмотрим доступные спецификаторы ресурсов:

- **%** – литерал **%**. То есть для вывода знака процента нужно в команде указать **%%**.
- **C** – имя команды и использованные аргументы.
- **D** – средний размер неразделенной области данных. Отображается в килобайтах.
- **E** – реальное время выполнения команды в привычном часовом формате. Выводится в виде **[часы:]минуты:секунды**.
- **N** – количество мажорных ошибок или ошибок, связанных с вводом-выводом, возникших при выполнении процесса.
- **I** – количество входов в файловую систему.
- **K** – среднее значение задействованной памяти для кода (**text**), инициализированных данных (**data**) и стека (**stack**). Отображается в килобайтах.
- **M** – максимальный размер резидентного множества во время выполнения процесса в килобайтах.
- **O** – количество выходов из файловой системы.

- **P** – процент загруженности CPU (центрального процессора).
- **R** – количество минорных ошибок.
- **S** – время в секундах, в течении которого CPU использовался системой от имени процесса в режиме супервизора (kernel mode).
- **U** – время в секундах, в течении которого CPU использовался процессом напрямую в пользовательском режиме (user mode).
- **W** – сколько раз процесс был выгружен из оперативной памяти.
- **X** – тут не понял.
- **Z** – размер системной страницы. Это значение – константа, но она различается между системами.
- **c** – количество невольных переключений контекста при выполнении процесса.
- **e** – реальное время выполнения команды в привычном часовом формате. Выводится в секундах.
- **k** – количество сигналов, дошедших до процесса.
- **p** – средний размер неразделенного стека процесса, в килобайтах.
- **r** – количество полученных сокетных сообщений.
- **s** – количество отправленных сокетных сообщений.
- **t** – средний размер резидентного множества процесса, в килобайтах.
- **w** – количество добровольных переключений контекста при выполнении процесса.
- **x** – код возврата для команды.

Это были все спецификаторы ресурсов, использующиеся при выборе форматирования для time. Теперь перейдем к сценариям использования команды.

Как узнать время выполнения команды Linux

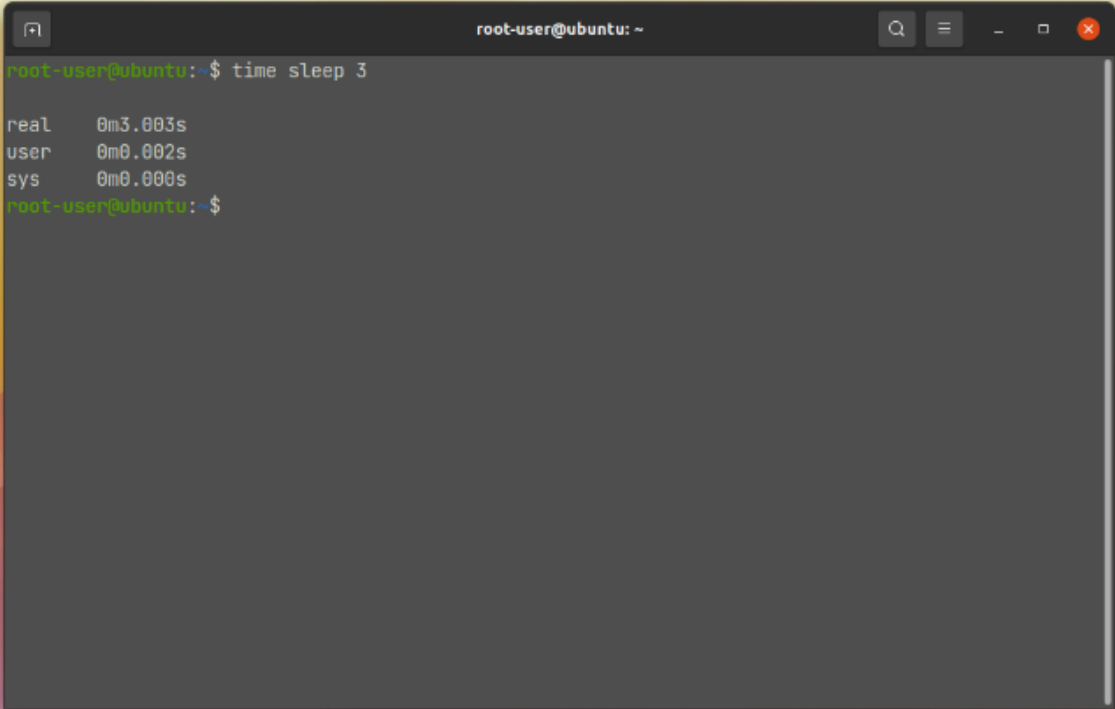
Мы рассмотрим три основных примера, которые довольно часто используются: вывод данных в терминал, вывод в отдельный файл и вывод с форматированием.

1. Вывод времени выполнения в терминал

В качестве примера мы возьмем команду sleep, которая делает паузу на указанное время. Это будет очень наглядный пример, ведь время паузы будет совпадать с временем выполнения команды в time:

```
$ time sleep 3
```





```
root-user@ubuntu: ~  
root-user@ubuntu:~$ time sleep 3  
real    0m3.003s  
user    0m0.002s  
sys     0m0.000s  
root-user@ubuntu:~$
```

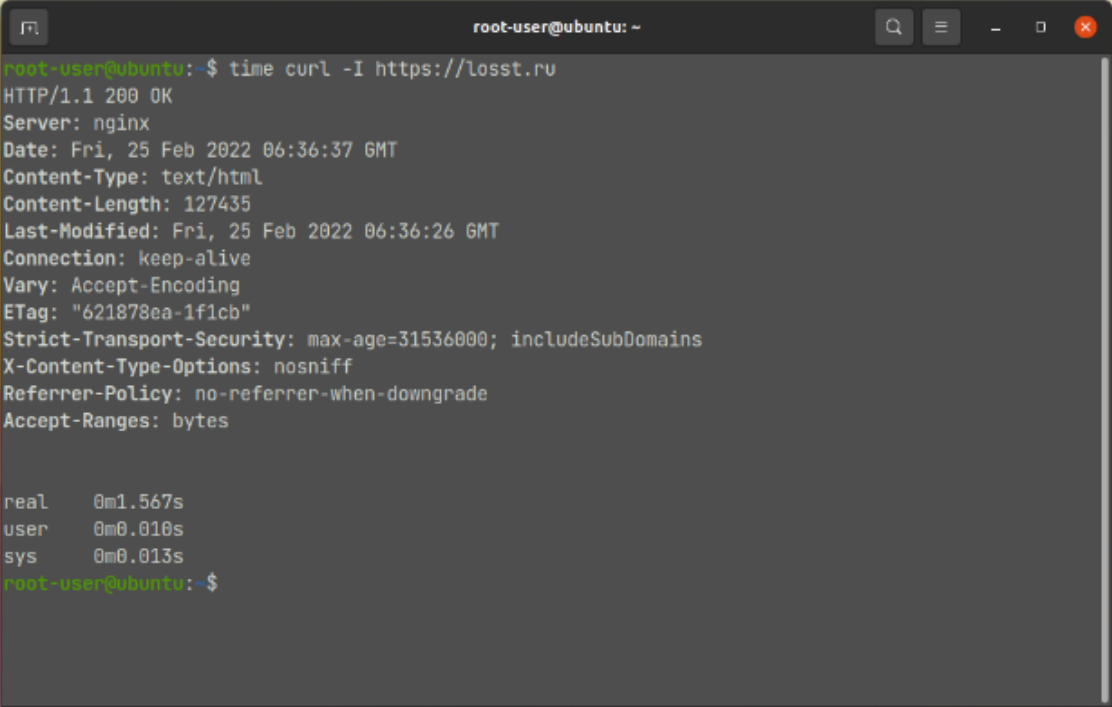
Обратите внимание, что время записано сразу же в трех графах. Пройдемся по каждой из них:

- **real** – общее время от начала выполнения процесса и до его завершения.
- **user** – время, в течение которого процесс был задействован в режиме пользователя (user mode).
- **sys** – время, в течение которого процесс был задействован в режиме супервизора (kernel mode).

Следующий полезный сценарий – вывод времени, в течение которого загрузится заголовок веб-страницы через утилиту `curl`. В качестве примера возьмем наш сайт:

```
$ time curl -I https://losst.pro
```



A terminal window titled 'root-user@ubuntu: ~' is shown against a background of a sunset over mountains. The terminal displays the output of the command 'time curl -I https://losst.ru'. The output includes HTTP headers and timing statistics.

```
root-user@ubuntu:~$ time curl -I https://losst.ru
HTTP/1.1 200 OK
Server: nginx
Date: Fri, 25 Feb 2022 06:36:37 GMT
Content-Type: text/html
Content-Length: 127435
Last-Modified: Fri, 25 Feb 2022 06:36:26 GMT
Connection: keep-alive
Vary: Accept-Encoding
ETag: "621878ea-1f1cb"
Strict-Transport-Security: max-age=31536000; includeSubDomains
X-Content-Type-Options: nosniff
Referrer-Policy: no-referrer-when-downgrade
Accept-Ranges: bytes

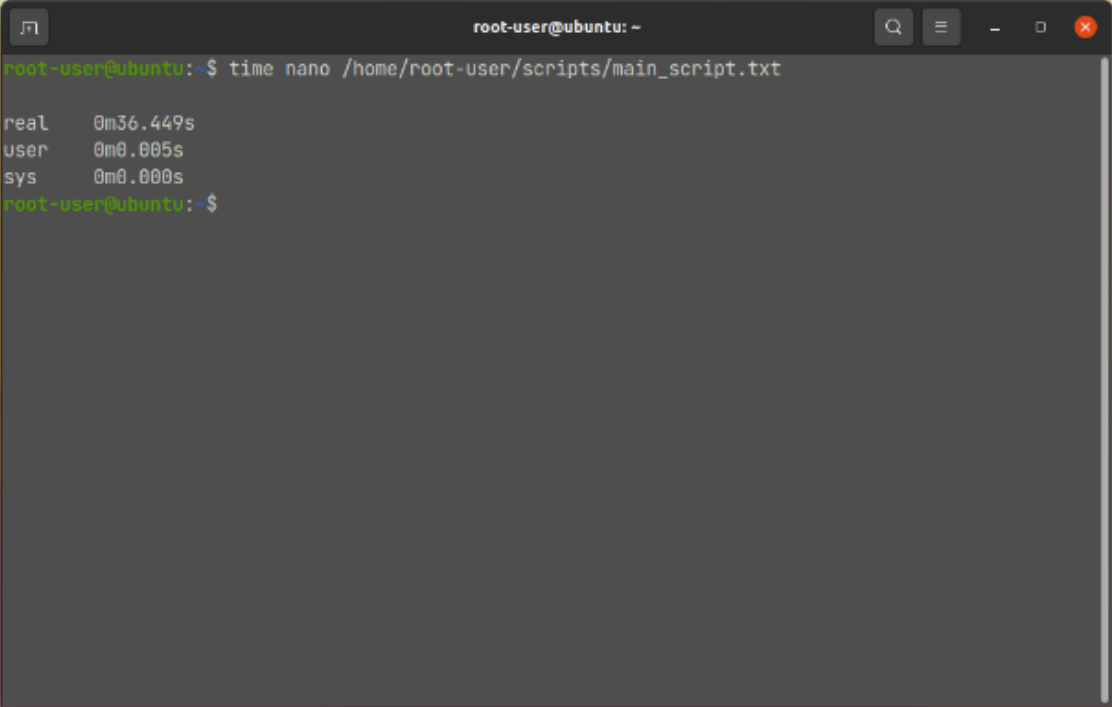
real    0m1.567s
user    0m0.010s
sys     0m0.013s
root-user@ubuntu:~$
```

Еще хотелось бы упомянуть ситуацию, когда выполнение команды переводит окно терминала в другой режим, например, при запуске редактора nano для изменения файла `/home/root-user/script.txt`:

```
$ time nano /home/root-user/scripts/main_script.txt
```

После завершения работы редактора вы увидите общее время, в течение которого вы работали с файлом.





```
root-user@ubuntu: ~  
root-user@ubuntu:~$ time nano /home/root-user/scripts/main_script.txt  
real    0m36.449s  
user    0m0.005s  
sys     0m0.000s  
root-user@ubuntu:~$
```

Теперь вы знаете как посмотреть время выполнения команды Linux.

2. Вывод в файл

Информацию о результатах работы команды `time` можно сохранить в отдельный файл с помощью опции `-o`. В таком случае в окне терминала они выводиться не будут. Возьмем для примера извлечение содержимого архива `~/data/data.tar.gz` с помощью утилиты `tar`. Информацию о времени выполнения сохраним в новом файле `~/data/data_time.txt`. Еще добавим к команде опцию `-v`, чтобы получить подробные сведения:

```
$ sudo time -v -o ~/data/data_time.txt tar -xvf ~/data/data.tar.gz -C ~/data
```



```
root-user@ubuntu: ~  
root-user@ubuntu:~$ sudo time -v -o ~/data/data_time.txt tar -xvf ~/data/data.tar.gz -C ~/data  
[sudo] пароль для root-user:  
data/  
data/8.  
data/7  
data/5  
data/teamviewer_15.27.3_i386.deb  
data/4  
data/debs/  
data/debs/2.deb  
data/debs/z.deb  
data/debs/b.deb  
data/debs/5.deb  
data/debs/g.deb  
data/debs/aa.deb  
data/debs/ab.deb  
data/debs/p.deb  
data/debs/l.deb  
data/debs/1.deb  
data/debs/c.deb  
data/debs/d.deb  
data/debs/14.deb  
data/debs/a.deb  
data/debs/j.deb
```

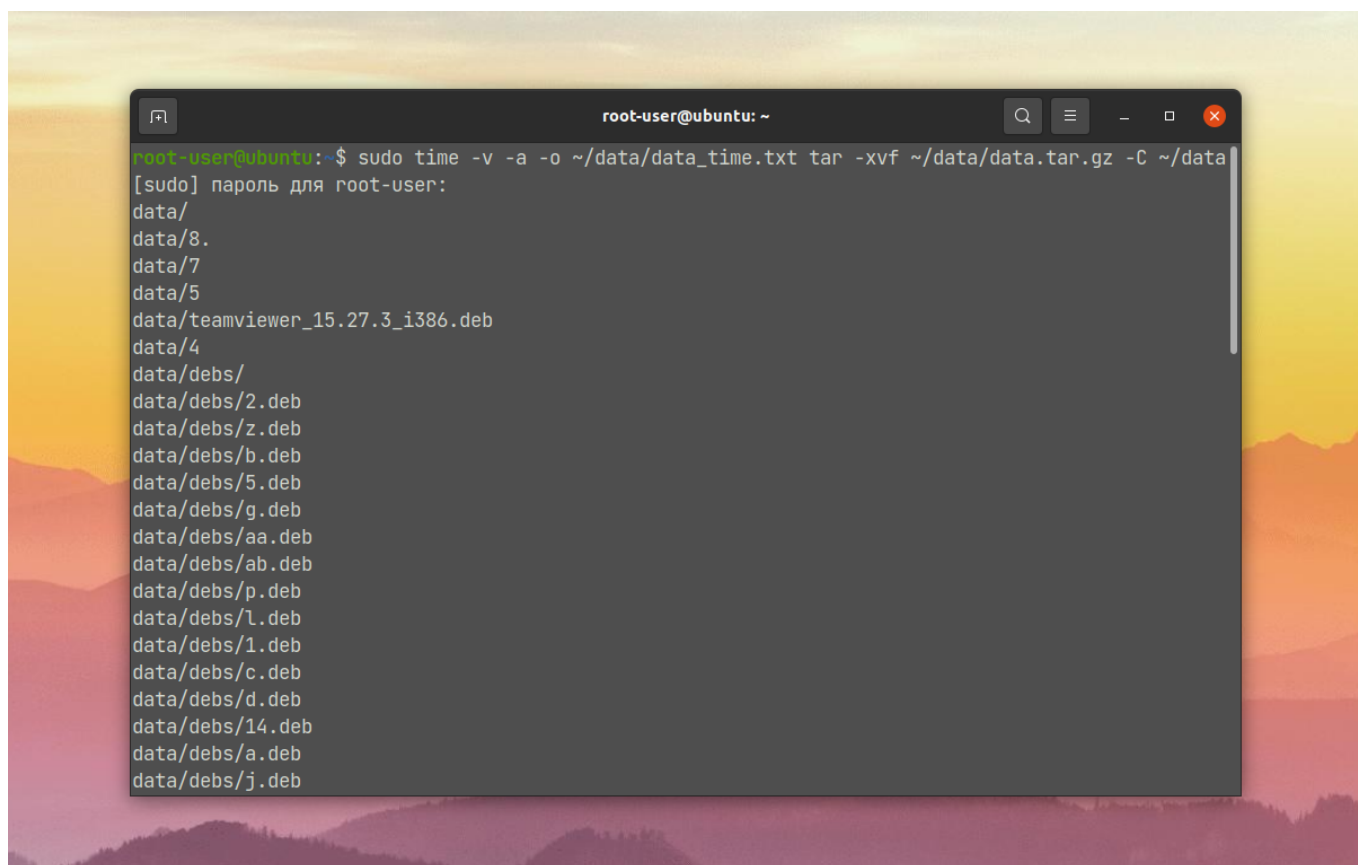
Файл получится следующего содержания.

```
data_time.txt [Только для чтения]  
1 Command being timed: "tar -xvf /home/root-user/data/data.tar.gz -C /home/root-user/data"  
2 User time (seconds): 0.65  
3 System time (seconds): 1.11  
4 Percent of CPU this job got: 112%  
5 Elapsed (wall clock) time (h:mm:ss or m:ss): 0:01.56  
6 Average shared text size (kbytes): 0  
7 Average unshared data size (kbytes): 0  
8 Average stack size (kbytes): 0  
9 Average total size (kbytes): 0  
10 Maximum resident set size (kbytes): 3136  
11 Average resident set size (kbytes): 0  
12 Major (requiring I/O) page faults: 0  
13 Minor (reclaiming a frame) page faults: 361  
14 Voluntary context switches: 8382  
15 Involuntary context switches: 3496  
16 Swaps: 0  
17 File system inputs: 0  
18 File system outputs: 537800  
19 Socket messages sent: 0  
20 Socket messages received: 0  
21 Signals delivered: 0  
22 Page size (bytes): 4096  
23 Exit status: 0  
Текст Ширина табуляции: 8 Стр 23, Стлб 23 ВСТ
```

При использовании опции **-o** следует помнить, что она перезаписывает старую информацию в выходном файле на новую. Это применимо при создании новых файлов, но для веде

логов не подходит. С помощью дополнительной опции **-a** перезапись содержимого заменяется на добавление новой информации:

```
$ sudo time -v -a -o ~/data/data_time.txt tar -xvf ~/data/data.tar.gz -C ~/data
```



3. Вывод с форматированием

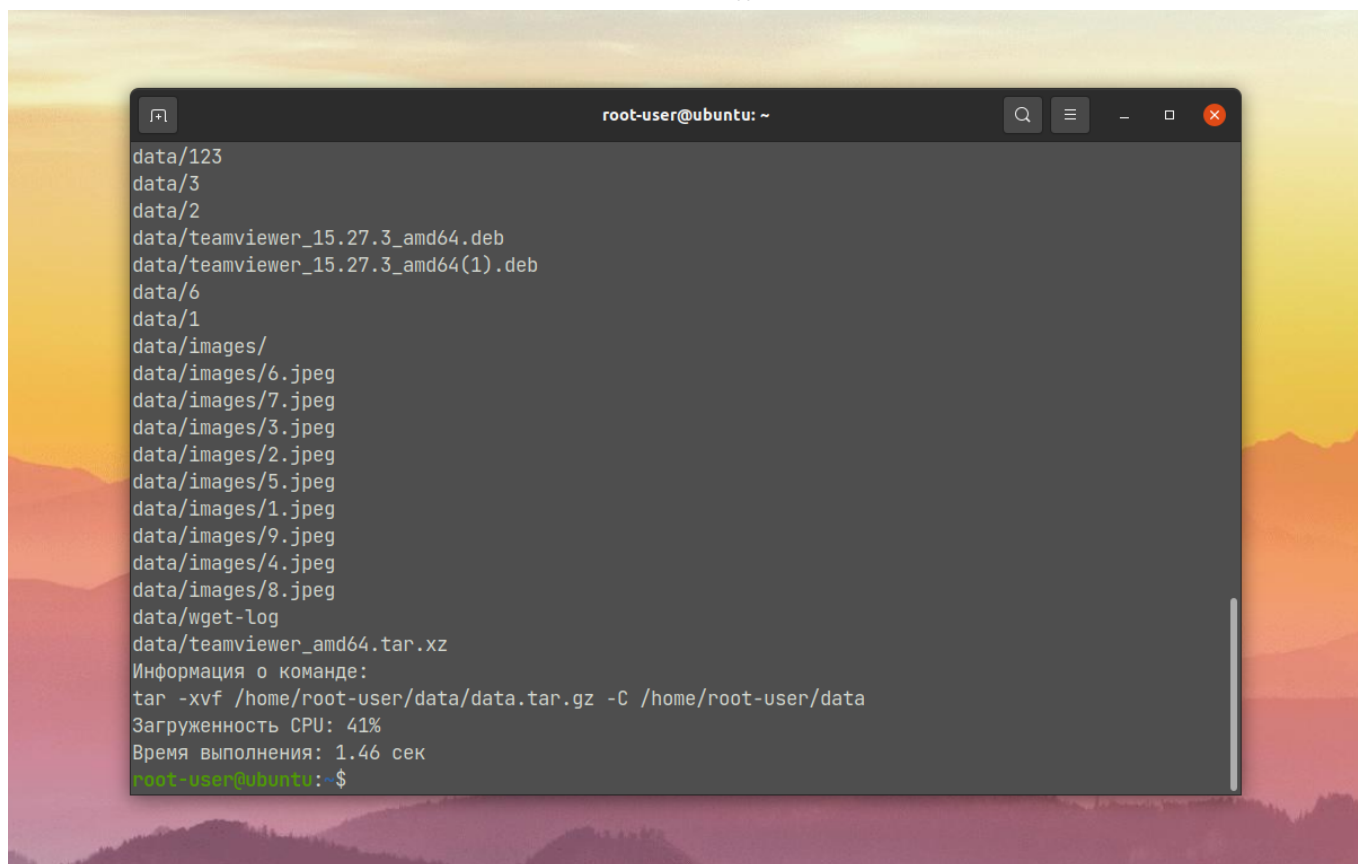
Исходный вариант форматирования данных подходит далеко не во всех случаях. В качестве примера мы возьмем ранее использованную команду для извлечения архива, но не станем сохранять данные в файл. При ее выполнении нам интересно:

- Информация о самой команде и заданных опциях – это спецификатор **%C**.
- Загруженность CPU – спецификатор **%P**.
- Общее время выполнения в секундах – спецификатор **%e**.

Для комфортного восприятия каждый пункт будет выводиться в отдельной строке с помощью разделителя **\n**. При этом они все будут подписаны понятным образом.

Вот как выглядит итоговый вариант команды:

```
$ sudo time -f "Информация о команде:\n%C\nЗагруженность CPU: %P\nВремя выполнения: %e сек" tar -xvf ~/data/data.tar.gz -C ~/data
```



```
root-user@ubuntu: ~  
data/123  
data/3  
data/2  
data/teamviewer_15.27.3_amd64.deb  
data/teamviewer_15.27.3_amd64(1).deb  
data/6  
data/1  
data/images/  
data/images/6.jpeg  
data/images/7.jpeg  
data/images/3.jpeg  
data/images/2.jpeg  
data/images/5.jpeg  
data/images/1.jpeg  
data/images/9.jpeg  
data/images/4.jpeg  
data/images/8.jpeg  
data/wget-log  
data/teamviewer_amd64.tar.xz  
Информация о команде:  
tar -xvf /home/root-user/data/data.tar.gz -C /home/root-user/data  
Загруженность CPU: 41%  
Время выполнения: 1.46 сек  
root-user@ubuntu:~$
```

Обратите внимание, что весь текст, кроме символов после % и \, полностью отображается в терминале. Это удобный способ разметки

Выводы

В рамках данной статьи вы узнали как работает команда `time` Linux, которая следит за выполнением заданной команды и выдает подробную информацию о времени выполнения и т. д. Основная сложность при ее использовании – понимание опций для форматирования вывода. Но если в них разобраться, то больше никаких проблем не возникнет. Для вашего удобства мы упомянули конкретные примеры использования.

Была ли эта информация полезной для вас?

Да

Нет

X

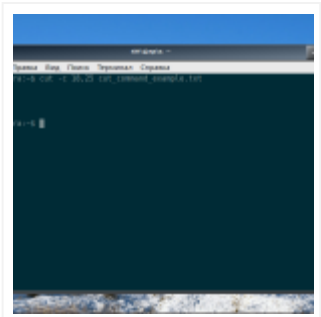
Похожие записи



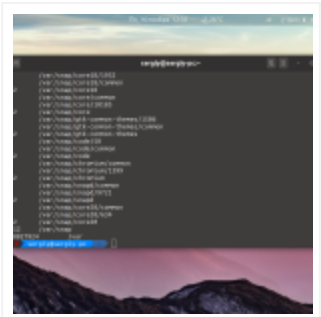
Privacy



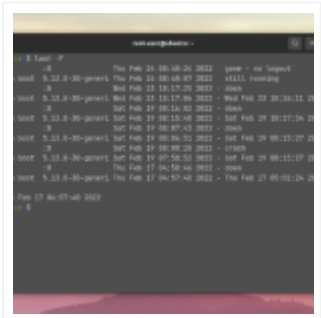
Команда wc в Linux



Команда cut Linux



Команда du в Linux



Команда last в Linux

Оцените статью

★★★★★ (5 оценок, среднее: 5,00 из 5)



Статья распространяется под лицензией Creative Commons ShareAlike 4.0 при копировании материала ссылка на источник обязательна .

 [Команды](#)

Об авторе



ZENINVLAD



1 комментарий к “Команда time в Linux”



Вадим DevOps

9 декабря, 2022 в 4:53 пп

Мне показалось, или это ошибочно? При вводе `time -o` и далее, я не мог вывести ничего в файл, `time` просто не видит такие опции, нужно вводить `/bin/time:`
`/bin/time -o ~/file.txt [команда]`
Либо же `/usr/bin/time -o ~/fileNEW.txt [команда]`

[Ответить](#)

Оставьте комментарий

Имя *

Email

☐ Я прочитал и принимаю политику конфиденциальности. Подробнее [Политика конфиденциальности](#) *

Privacy

Комментировать

Русский

Поиск

ПОИСК ПО КОМАНДАМ

Начните вводить команду

Поиск



Начните изучать
Linux прямо сейчас!

Карта сайта



Как пользоваться редактором Vim

Полезно

Лучшие

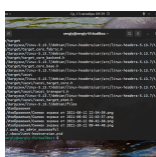
Свежие

Теги



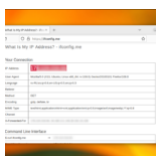
Команда chmod Linux

2020-04-13



Команда find в Linux

2021-10-17



Как узнать IP-адрес Linux

2023-04-14



Настройка Cron

2021-10-01



Права доступа к файлам в Linux

2020-10-09



Privacy

РАССЫЛКА

Ваш E-Mail адрес

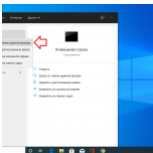
☐ Я прочитал(а) и принимаю политику конфиденциальности

Sign up



Windows

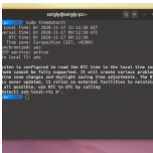
Списки



Восстановление Grub после установки Windows 10
2020-08-15



Установка Linux рядом с Windows 10 или 11
2023-02-08

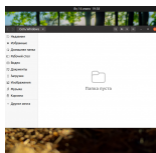


Сбивается время в Ubuntu и Windows
2023-02-18

Ошибка Ubuntu не видит сеть Windows



Privacy



2023-02-18

[Смотреть ещё](#)

МЕТА

[Регистрация](#)[Войти](#)[Лента записей](#)[Лента комментариев](#)

СЛЕДИТЕ ЗА НАМИ В СОЦИАЛЬНЫХ СЕТЯХ

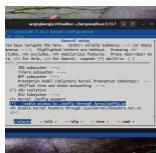


Интересное

[Что такое демоны в Linux](#)

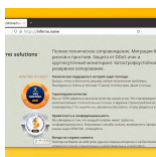
2021-06-01

[Privacy](#)



Сборка ядра Linux

2021-08-14



Лучшие VPS сервера для VPN

2022-09-04



Линус Торвальдс – человек, создавший Linux

2021-01-28

©Losst 2024 CC-BY-SA [Политика конфиденциальности](#)

