



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

Команда last в Linux

Опубликовано: 18 апреля, 2022 от [zeninvlad](#) , 0 комменариев, время чтения: 10 минут

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

Иногда при администрировании сервера под управлением Linux нужно посмотреть историю входов в систему и выходов для пользователей. Это удобно делать с помощью команды last. Хотя она пригодится и в других ситуациях.

В данной статье мы подробно расскажем, как работает last в Linux, какой у нее синтаксис и доступные опции. А затем перейдем к конкретным примерам использования и рассмотрим несколько сценариев.

Содержание статьи

- [Синтаксис и опции last](#)
- [Примеры использования last](#)
 - [1. Просмотр всей истории](#)
 - [2. История определённого пользователя](#)
 - [3. Ограничение количества строк](#)
 - [3. История за определённый день](#)
 - [4. История за определённый период](#)
 - [5. Вывод хоста и имени пользователя \(Заголовок 3\) \(-i/-R\)](#)

Конфиденциальность -
Условия использования

Privacy

- [6. Вывод полного времени](#)
- [8. История перезагрузок](#)
- [9. История неудачных входов](#)
- [Выводы](#)

Синтаксис и опции last

Данная утилита по умолчанию берет данные из файла `/var/log/wtmp` и выводит полный список пользователей, вошедших в систему и вышедших. Рассмотрим ее синтаксис. В терминале нужно задать опции, имя одного или нескольких пользователей и `tty`:

\$ last **опции** **имя_пользователя** **название_tty**

Вместо **last** можно использовать **lastb**. В таком случае по умолчанию будет использоваться файл `/var/log/btmp`, в котором сохранены все неудачные попытки входа. Теперь пройдемся по доступным опциям:

- **-a, --hostlast** – выводить информацию об имени хоста (hostname) в последнем столбце. Изначально она отображается в третьем столбце.
- **-d, --dns** – выводить IP-адрес пользователя в случае удаленного подключения. Удобно использовать вместе с предыдущей опцией.
- **-f, --file** – выбрать свой файл для загрузки данных вместо используемого по умолчанию `/var/log/wtmp`. Опцию можно использовать несколько раз, выбрав два и более файла.
- **-F, --fulltimes** – выводить полные дату и время входов и выходов из системы.
- **-i, --ip** – выводить IP-адрес пользователя при удаленном подключении вместо имени хоста. Это несколько похоже на упомянутую ранее опцию **--dns**.
- **-число, -n, --limit** – указать, сколько строк информации будет выведено при выполнении команды.
- **-p, --present** – выводить информацию о сессиях в конкретное время.
- **-R, --nohostname** – не отображать графу с именем хоста.
- **-s, --since** – выводить информацию начиная с указанного момента времени.
- **-t, --until** – выводить информацию до указанного времени.
- **--time-format** – выбрать один из 4 доступных способов отображения времени: **notime** – вообще не выводить информацию, **short** – стандартный вариант, использующийся по умолчанию, **full** – полная информация, как в случае с опцией **--fulltimes**, **iso** – использовать формат ISO-8601, включающий в себя часовой пояс.
- **-w, --fullname** – выводить полные имена пользователей и доменов.
- **-x, --system** – отображать записи о выключении системы и изменении уровня выполнения (run level).

Детальную информацию о команде `last`, синтаксис и все опции можно получить в терминале:

```
$ man last
```

А теперь перейдем к примерам ее использования.

Примеры использования last

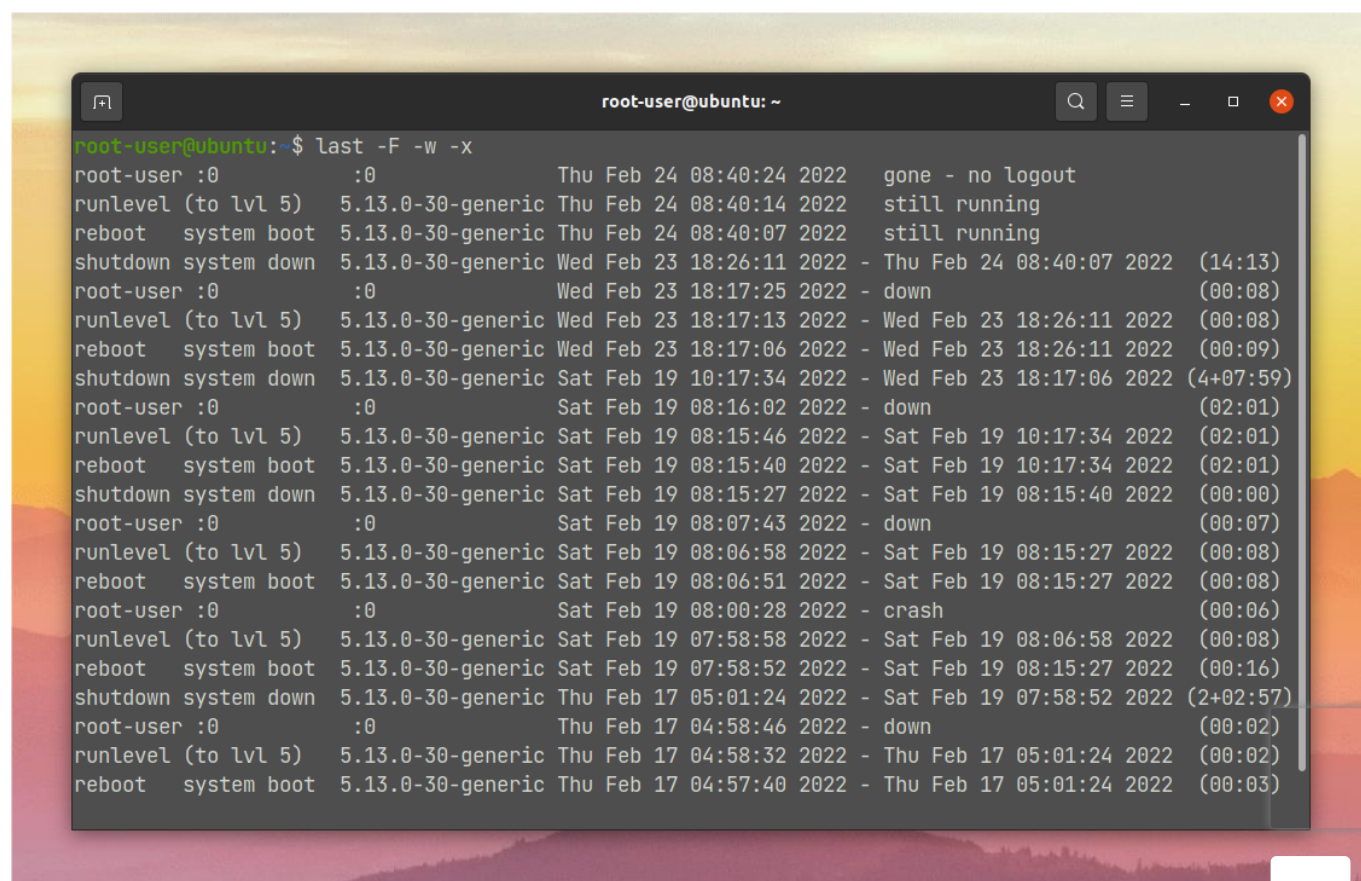
Сначала рассмотрим способ просмотра всей истории входов и выходов, объяснив детально каждое поле. Затем перейдем к изучению данных для конкретного пользователя. Это основные сценарии, которые могут пригодиться.

Еще мы разберем несколько полезных вариантов. Ориентируйтесь по оглавлению, чтобы перейти к нужному.

1. Просмотр всей истории

Главная задача – получить полную информацию для всех пользователей. Для этих целей команде last нужно передать опции **-F**, **-w** и **-x**:

```
$ last -F -w -x
```



```
root-user@ubuntu: ~  
root-user@ubuntu:~$ last -F -w -x  
root-user :0 :0 Thu Feb 24 08:40:24 2022 gone - no logout  
runlevel (to lvl 5) 5.13.0-30-generic Thu Feb 24 08:40:14 2022 still running  
reboot system boot 5.13.0-30-generic Thu Feb 24 08:40:07 2022 still running  
shutdown system down 5.13.0-30-generic Wed Feb 23 18:26:11 2022 - Thu Feb 24 08:40:07 2022 (14:13)  
root-user :0 :0 Wed Feb 23 18:17:25 2022 - down (00:08)  
runlevel (to lvl 5) 5.13.0-30-generic Wed Feb 23 18:17:13 2022 - Wed Feb 23 18:26:11 2022 (00:08)  
reboot system boot 5.13.0-30-generic Wed Feb 23 18:17:06 2022 - Wed Feb 23 18:26:11 2022 (00:09)  
shutdown system down 5.13.0-30-generic Sat Feb 19 10:17:34 2022 - Wed Feb 23 18:17:06 2022 (4+07:59)  
root-user :0 :0 Sat Feb 19 08:16:02 2022 - down (02:01)  
runlevel (to lvl 5) 5.13.0-30-generic Sat Feb 19 08:15:46 2022 - Sat Feb 19 10:17:34 2022 (02:01)  
reboot system boot 5.13.0-30-generic Sat Feb 19 08:15:40 2022 - Sat Feb 19 10:17:34 2022 (02:01)  
shutdown system down 5.13.0-30-generic Sat Feb 19 08:15:27 2022 - Sat Feb 19 08:15:40 2022 (00:00)  
root-user :0 :0 Sat Feb 19 08:07:43 2022 - down (00:07)  
runlevel (to lvl 5) 5.13.0-30-generic Sat Feb 19 08:06:58 2022 - Sat Feb 19 08:15:27 2022 (00:08)  
reboot system boot 5.13.0-30-generic Sat Feb 19 08:06:51 2022 - Sat Feb 19 08:15:27 2022 (00:08)  
root-user :0 :0 Sat Feb 19 08:00:28 2022 - crash (00:06)  
runlevel (to lvl 5) 5.13.0-30-generic Sat Feb 19 07:58:58 2022 - Sat Feb 19 08:06:58 2022 (00:08)  
reboot system boot 5.13.0-30-generic Sat Feb 19 07:58:52 2022 - Sat Feb 19 08:15:27 2022 (00:16)  
shutdown system down 5.13.0-30-generic Thu Feb 17 05:01:24 2022 - Sat Feb 19 07:58:52 2022 (2+02:57)  
root-user :0 :0 Thu Feb 17 04:58:46 2022 - down (00:02)  
runlevel (to lvl 5) 5.13.0-30-generic Thu Feb 17 04:58:32 2022 - Thu Feb 17 05:01:24 2022 (00:02)  
reboot system boot 5.13.0-30-generic Thu Feb 17 04:57:40 2022 - Thu Feb 17 05:01:24 2022 (00:03)
```

Остановимся на каждом столбце более подробно:

Privacy

- Сначала идет имя пользователя. Обратите внимание, что **reboot** – это отдельный пользователь, появляющийся во время перезагрузки, **shutdown** – при выключении, **runlevel** – при изменении уровня выполнения.
- Информация о tty сессии.
- Имя хоста.
- Время входа.
- Время выхода.
- Продолжительность сессии.

Обратите внимание, что в последней строке **wtmp begins** указано, с какого момента существует файл **/var/log/btmp**.

```

root-user@ubuntu: ~
runlevel (to lvl 5) 5.13.0-30-generic Thu Feb 24 08:40:14 2022 still running
reboot system boot 5.13.0-30-generic Thu Feb 24 08:40:07 2022 still running
shutdown system down 5.13.0-30-generic Wed Feb 23 18:26:11 2022 - Thu Feb 24 08:40:07 2022 (14:13)
root-user :0 :0 Wed Feb 23 18:17:25 2022 - down (00:08)
runlevel (to lvl 5) 5.13.0-30-generic Wed Feb 23 18:17:13 2022 - Wed Feb 23 18:26:11 2022 (00:08)
reboot system boot 5.13.0-30-generic Wed Feb 23 18:17:06 2022 - Wed Feb 23 18:26:11 2022 (00:09)
shutdown system down 5.13.0-30-generic Sat Feb 19 10:17:34 2022 - Wed Feb 23 18:17:06 2022 (4+07:59)
root-user :0 :0 Sat Feb 19 08:16:02 2022 - down (02:01)
runlevel (to lvl 5) 5.13.0-30-generic Sat Feb 19 08:15:46 2022 - Sat Feb 19 10:17:34 2022 (02:01)
reboot system boot 5.13.0-30-generic Sat Feb 19 08:15:40 2022 - Sat Feb 19 10:17:34 2022 (02:01)
shutdown system down 5.13.0-30-generic Sat Feb 19 08:15:27 2022 - Sat Feb 19 08:15:40 2022 (00:00)
root-user :0 :0 Sat Feb 19 08:07:43 2022 - down (00:07)
runlevel (to lvl 5) 5.13.0-30-generic Sat Feb 19 08:06:58 2022 - Sat Feb 19 08:15:27 2022 (00:08)
reboot system boot 5.13.0-30-generic Sat Feb 19 08:06:51 2022 - Sat Feb 19 08:15:27 2022 (00:08)
root-user :0 :0 Sat Feb 19 08:00:28 2022 - crash (00:06)
runlevel (to lvl 5) 5.13.0-30-generic Sat Feb 19 07:58:58 2022 - Sat Feb 19 08:06:58 2022 (00:08)
reboot system boot 5.13.0-30-generic Sat Feb 19 07:58:52 2022 - Sat Feb 19 08:15:27 2022 (00:16)
shutdown system down 5.13.0-30-generic Thu Feb 17 05:01:24 2022 - Sat Feb 19 07:58:52 2022 (2+02:57)
root-user :0 :0 Thu Feb 17 04:58:46 2022 - down (00:02)
runlevel (to lvl 5) 5.13.0-30-generic Thu Feb 17 04:58:32 2022 - Thu Feb 17 05:01:24 2022 (00:02)
reboot system boot 5.13.0-30-generic Thu Feb 17 04:57:40 2022 - Thu Feb 17 05:01:24 2022 (00:03)

wtmp begins Thu Feb 17 04:57:40 2022
root-user@ubuntu:~$

```

Если вам нужно получить подробную информацию об удаленных подключениях, то к команде добавится опция **-d**:

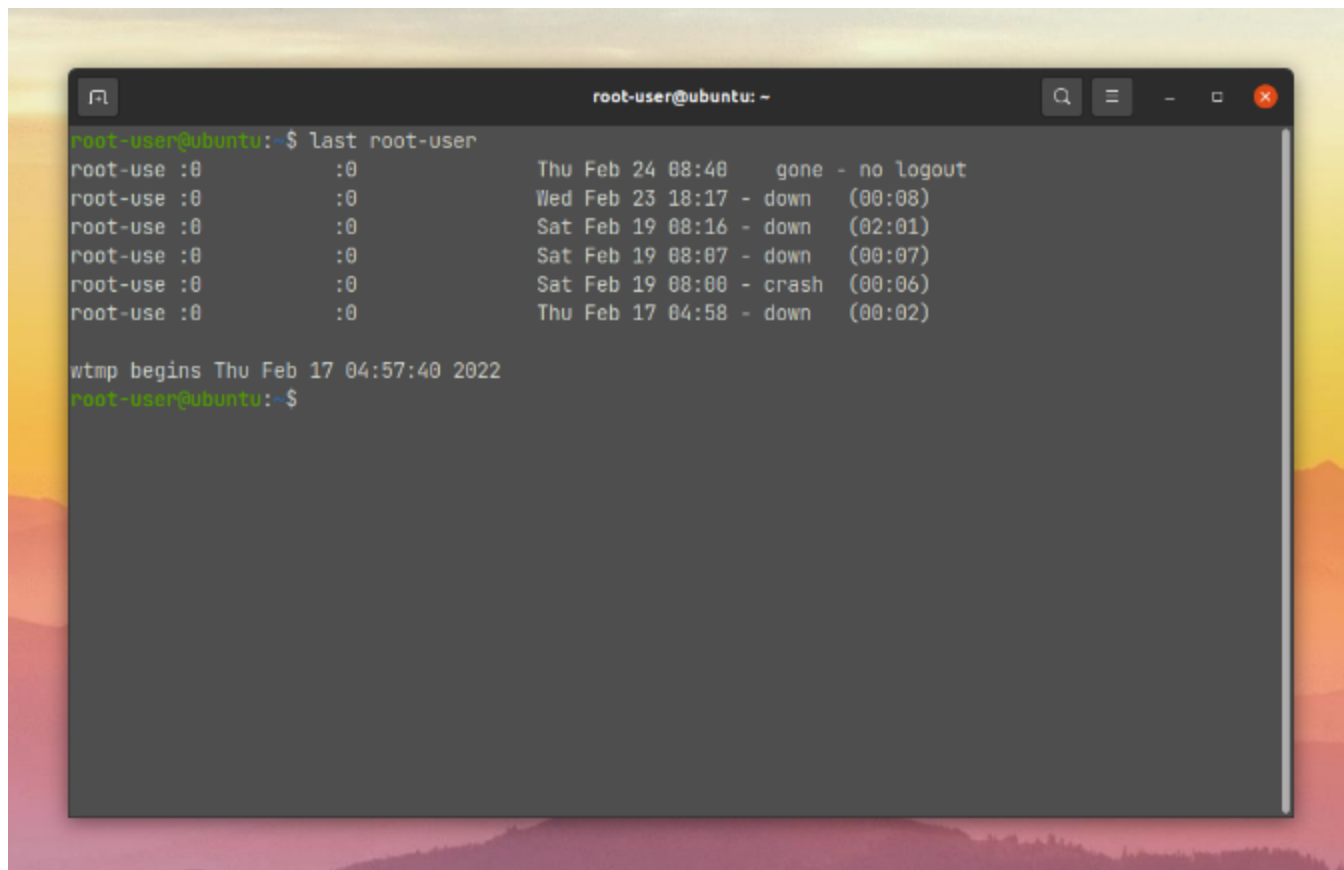
```
$ last -d -F -w -x
```

В результате в отдельной графе будет записан IP-адрес подключения.

2. История определённого пользователя

Когда компьютером пользуется сразу же несколько пользователей, информацию можно получить только по одному из них. Для этого утилите `last` нужно указать его имя. Рассмотрим все на примере пользователя `root-user`:

```
$ last root-user
```



```
root-user@ubuntu:~$ last root-user
root-use :0      :0      Thu Feb 24 08:48   gone - no logout
root-use :0      :0      Wed Feb 23 18:17   - down (00:08)
root-use :0      :0      Sat Feb 19 08:16   - down (02:01)
root-use :0      :0      Sat Feb 19 08:07   - down (00:07)
root-use :0      :0      Sat Feb 19 08:00   - crash (00:06)
root-use :0      :0      Thu Feb 17 04:58   - down (00:02)

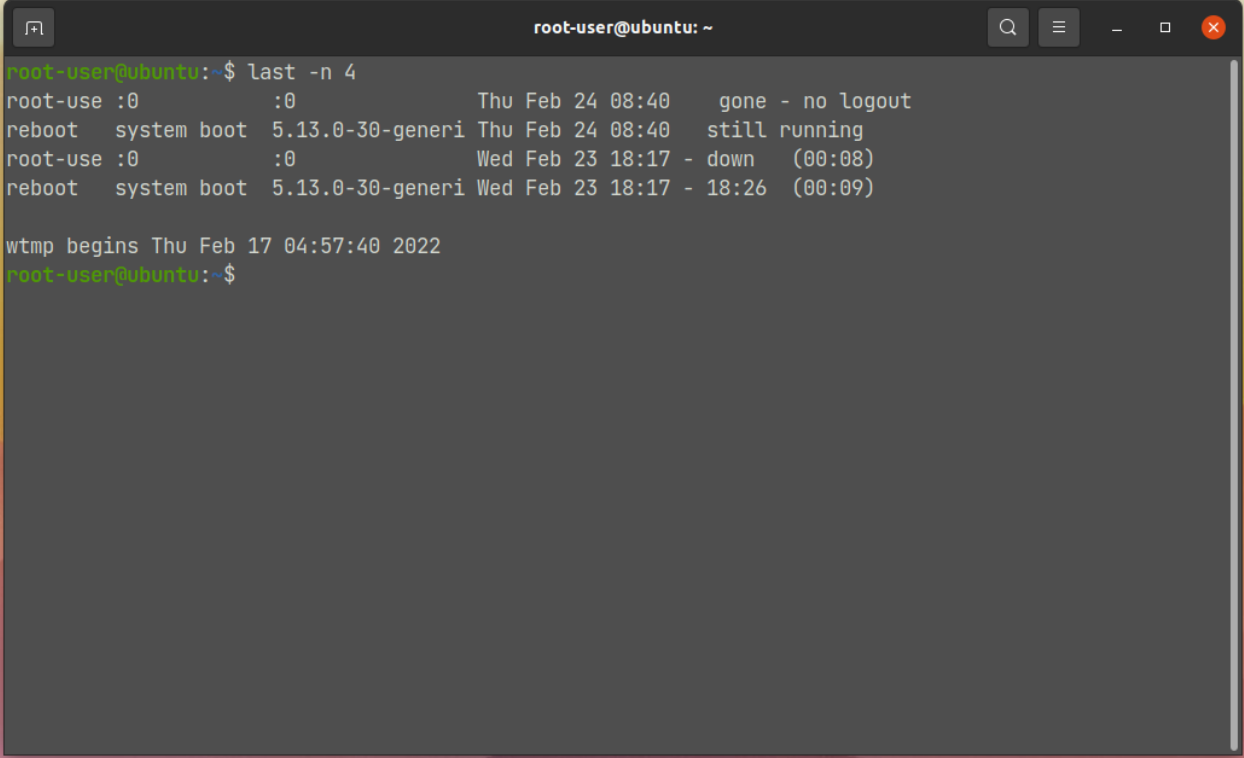
wtmp begins Thu Feb 17 04:57:40 2022
root-user@ubuntu:~$
```

3. Ограничение количества строк

Если команда `last` выводит слишком большое количество информации, неудобное для изучения, то можно сократить количество строк. За это отвечает опция `-n`. Вот как будет выглядеть команда с ограничением на 4 строки:

```
$ last -n 4
```





```
root-user@ubuntu:~$ last -n 4
root-use :0      :0      Thu Feb 24 08:40   gone - no logout
reboot  system boot  5.13.0-30-generi Thu Feb 24 08:40   still running
root-use :0      :0      Wed Feb 23 18:17 - down   (00:08)
reboot  system boot  5.13.0-30-generi Wed Feb 23 18:17 - 18:26 (00:09)

wtmp begins Thu Feb 17 04:57:40 2022
root-user@ubuntu:~$
```

Существует еще несколько вариантов ввода опции:

```
$ last -4
```

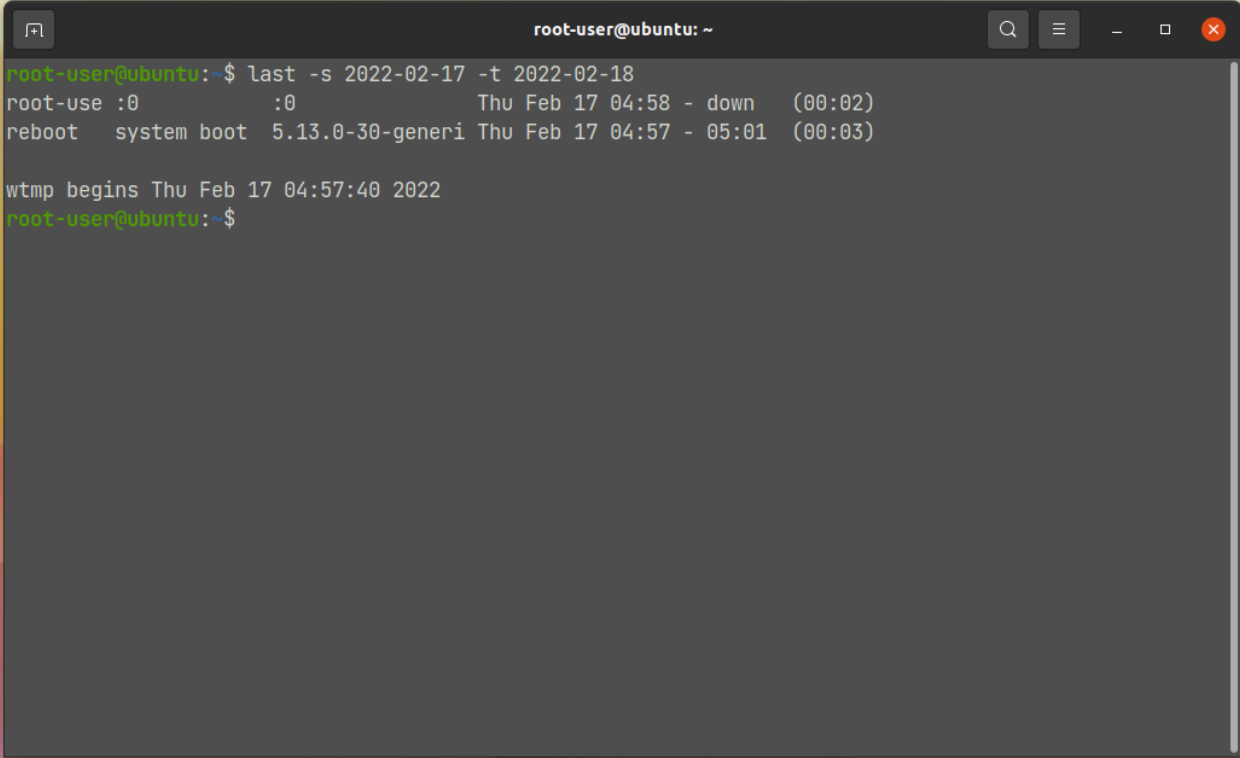
```
$ last --limit 4
```

3. История за определённый день

С помощью опции **-p** добиться желаемого результата не выйдет, ведь она отображает информацию о сессиях только в конкретное время. Поэтому мы будем использовать опции **-s** и **-t**. Они задают время начала и конца сбора информации. В качестве примера возьмем 17 февраля 2022 года. Эта дата задается для **-s**. А для **-t** указывается следующий день:

```
$ last -s 2022-02-17 -t 2022-02-18
```





```
root-user@ubuntu: ~  
root-user@ubuntu:~$ last -s 2022-02-17 -t 2022-02-18  
root-use :0      :0      Thu Feb 17 04:58 - down    (00:02)  
reboot  system boot  5.13.0-30-generi Thu Feb 17 04:57 - 05:01  (00:03)  
  
wtmp begins Thu Feb 17 04:57:40 2022  
root-user@ubuntu:~$
```

[Своя электроэнергия для майнинга 1,5 руб
кВт.ч!](#)

Самая низкая цена генератора! 3,5 млн.р за
200кВт.

gaz-electrogenerator.ru

[Своя электроэнергия для майнинга 1,5 руб
кВт.ч!](#)

Самая низкая цена генератора! 3,5 млн.р за
200кВт.

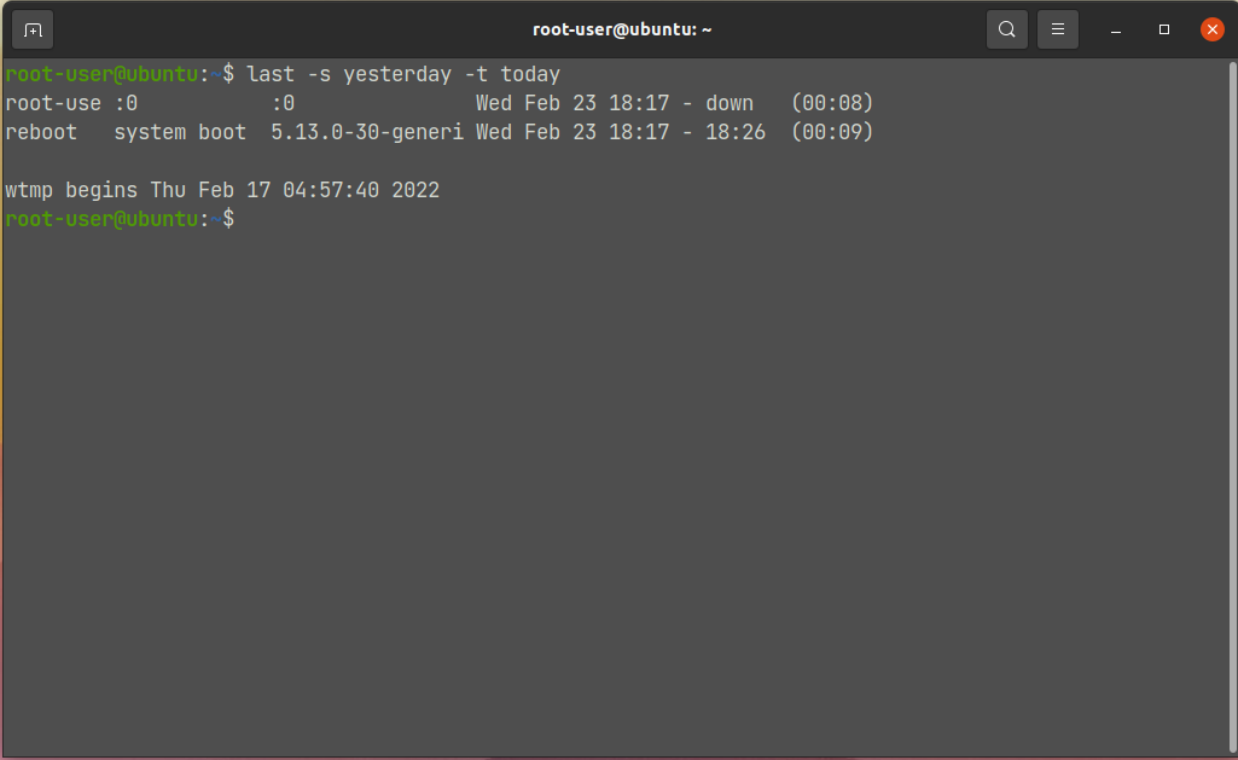
gaz-electrogenerator.ru

[Автоматизируй свою торговлю с нашими
ботами](#)

Veles поможет тебе понять криптотрейдинг
veles.finance

Обратите внимание на формат ввода даты. Если речь идет о вчерашнем дне, то вместо первого числа можно записать **yesterday**, а следующий день – **today**:

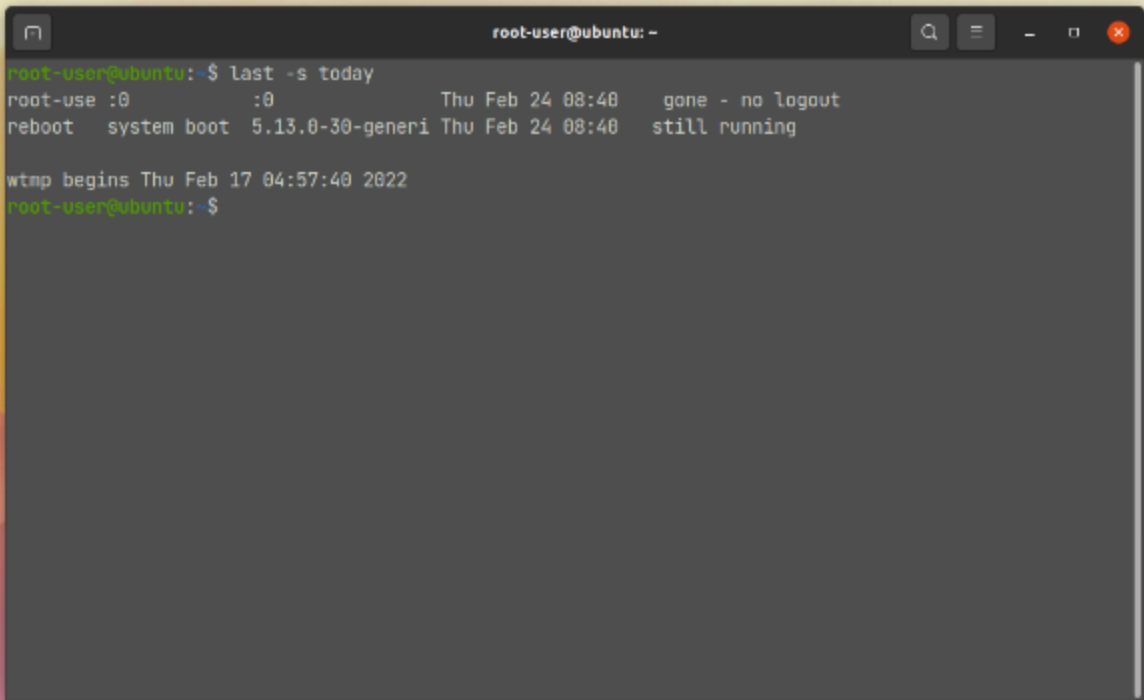
```
$ last -s yesterday -t today
```



```
root-user@ubuntu: ~  
root-user@ubuntu:~$ last -s yesterday -t today  
root-use :0      :0      Wed Feb 23 18:17 - down (00:08)  
reboot  system boot  5.13.0-30-generi Wed Feb 23 18:17 - 18:26 (00:09)  
  
wtmp begins Thu Feb 17 04:57:40 2022  
root-user@ubuntu:~$
```

Для вывода информации за сегодняшний день для **-s** следует задать значение **today**, а **-t** просто не использовать:

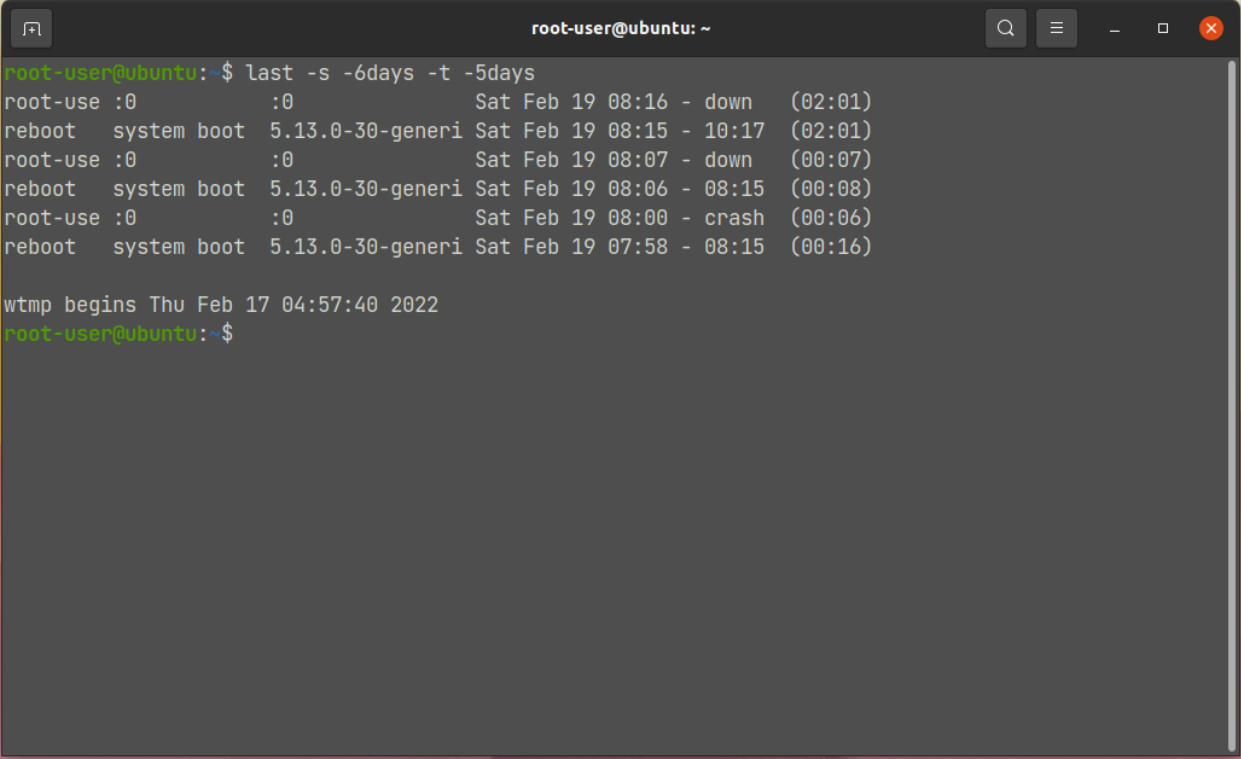
```
$ last -s today
```



```
root-user@ubuntu: ~  
root-user@ubuntu:~$ last -s today  
root-use :0      :0      Thu Feb 24 08:40  gone - no logout  
reboot  system boot  5.13.0-30-generi Thu Feb 24 08:40  still running  
  
wtmp begins Thu Feb 17 04:57:40 2022  
root-user@ubuntu:~$
```


Ну и еще вместо ввода даты можно указать **-ndays**, вместо **n** введя число, насколько дней назад откатиться. Вот как будет выглядеть команда для просмотра данных, сохраненных 6 дней назад:

```
$ last -s -6days -t -5days
```



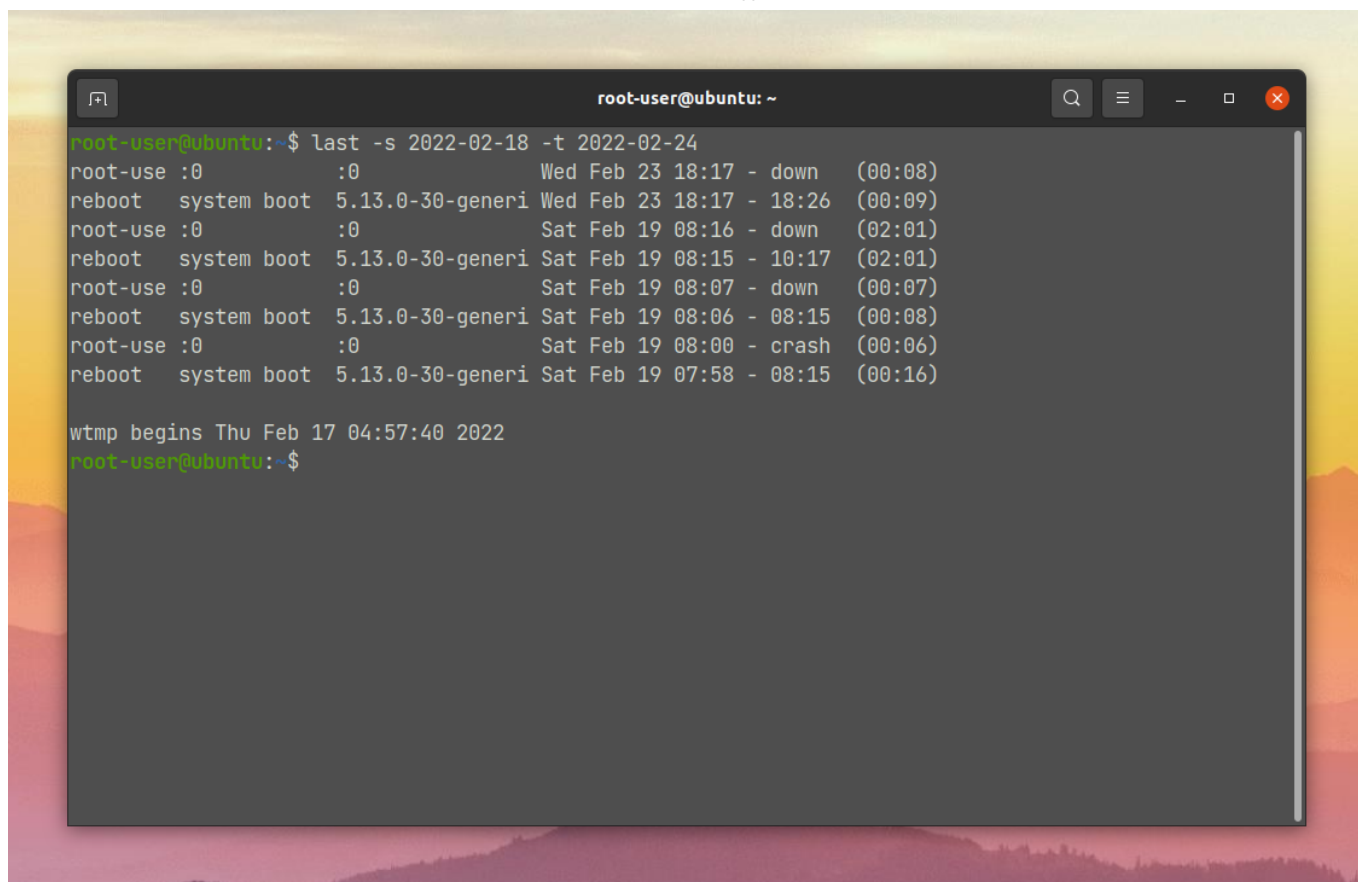
```
root-user@ubuntu: ~  
root-user@ubuntu:~$ last -s -6days -t -5days  
root-use :0      :0      Sat Feb 19 08:16 - down (02:01)  
reboot   system boot 5.13.0-30-generi Sat Feb 19 08:15 - 10:17 (02:01)  
root-use :0      :0      Sat Feb 19 08:07 - down (00:07)  
reboot   system boot 5.13.0-30-generi Sat Feb 19 08:06 - 08:15 (00:08)  
root-use :0      :0      Sat Feb 19 08:00 - crash (00:06)  
reboot   system boot 5.13.0-30-generi Sat Feb 19 07:58 - 08:15 (00:16)  
  
wtmp begins Thu Feb 17 04:57:40 2022  
root-user@ubuntu:~$
```

4. История за определённый период

Как и в предыдущем случае, будут использоваться опции **-s** и **-t**. Но на этот раз можно указать любой временной промежуток. Форматы ввода данных те же самые, что и до этого. В качестве примера возьмем время с 2022-02-18 по 2022-02-23:

```
$ last -s 2022-02-18 -t 2022-02-24
```





```
root-user@ubuntu:~$ last -s 2022-02-18 -t 2022-02-24
root-use :0 :0 Wed Feb 23 18:17 - down (00:08)
reboot system boot 5.13.0-30-generi Wed Feb 23 18:17 - 18:26 (00:09)
root-use :0 :0 Sat Feb 19 08:16 - down (02:01)
reboot system boot 5.13.0-30-generi Sat Feb 19 08:15 - 10:17 (02:01)
root-use :0 :0 Sat Feb 19 08:07 - down (00:07)
reboot system boot 5.13.0-30-generi Sat Feb 19 08:06 - 08:15 (00:08)
root-use :0 :0 Sat Feb 19 08:00 - crash (00:06)
reboot system boot 5.13.0-30-generi Sat Feb 19 07:58 - 08:15 (00:16)

wtmp begins Thu Feb 17 04:57:40 2022
root-user@ubuntu:~$
```

Для `-t` мы указываем 24 число, потому что отчет данных берется с самого начала суток (00:00 на часах). Также вы можете указать точное время для начальной и конечной точки. В таком случае дату и время нужно вводить в несколько ином формате:

- YYYYMMDDhhmmss
- YYYY-MM-DD hh:mm:ss
- YYYY-MM-DD hh:mm

Вот пример команды в случае с временным промежутком от 17 февраля 04:57 до 19 февраля 08:15:

```
$ last -s '2022-02-17 04:57' -t '2022-02-19 08:15'
```



```
root-user@ubuntu: ~  
root-user@ubuntu:~$ last -s '2022-02-17 04:57' -t '2022-02-19 08:15'  
root-use :0      :0      Sat Feb 19 08:07   gone - no logout  
reboot   system boot 5.13.0-30-generi Sat Feb 19 08:06   still running  
root-use :0      :0      Sat Feb 19 08:00 - crash (00:06)  
reboot   system boot 5.13.0-30-generi Sat Feb 19 07:58   still running  
root-use :0      :0      Thu Feb 17 04:58 - down (00:02)  
reboot   system boot 5.13.0-30-generi Thu Feb 17 04:57 - 05:01 (00:03)  
  
wtmp begins Thu Feb 17 04:57:40 2022  
root-user@ubuntu:~$
```

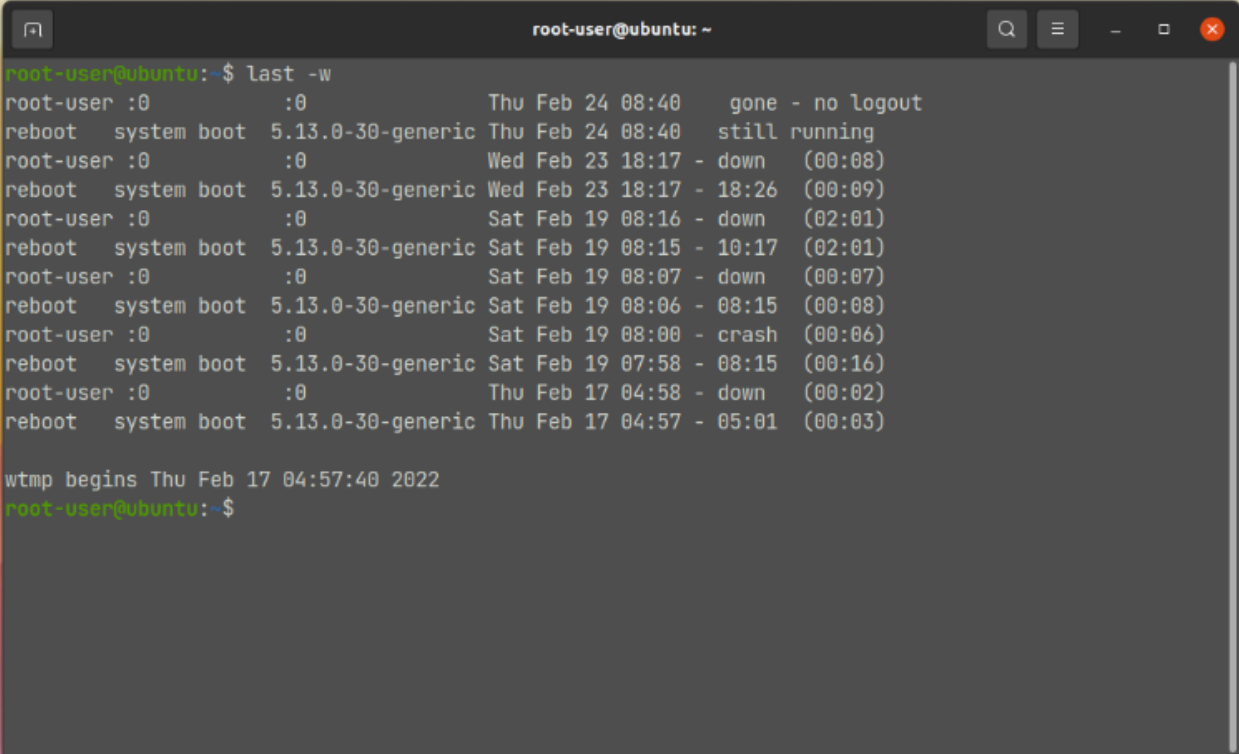
5. Вывод хоста и имени пользователя (Заголовок 3) (-i/-R)

По умолчанию имя пользователя отображается в первом столбце, а имя хоста – в третьем.

```
root-user@ubuntu: ~  
root-user@ubuntu:~$ last  
root-use :0      :0      Thu Feb 24 08:40   gone - no logout  
reboot   system boot 5.13.0-30-generi Thu Feb 24 08:40   still running  
root-use :0      :0      Wed Feb 23 18:17 - down (00:08)  
reboot   system boot 5.13.0-30-generi Wed Feb 23 18:17 - 18:26 (00:09)  
root-use :0      :0      Sat Feb 19 08:16 - down (02:01)  
reboot   system boot 5.13.0-30-generi Sat Feb 19 08:15 - 10:17 (02:01)  
root-use :0      :0      Sat Feb 19 08:07 - down (00:07)  
reboot   system boot 5.13.0-30-generi Sat Feb 19 08:06 - 08:15 (00:08)  
root-use :0      :0      Sat Feb 19 08:00 - crash (00:06)  
reboot   system boot 5.13.0-30-generi Sat Feb 19 07:58 - 08:15 (00:16)  
root-use :0      :0      Thu Feb 17 04:58 - down (00:02)  
reboot   system boot 5.13.0-30-generi Thu Feb 17 04:57 - 05:01 (00:03)  
  
wtmp begins Thu Feb 17 04:57:40 2022  
root-user@ubuntu:~$
```

Но при этом они могут быть записаны не в полном виде. Чтобы это исправить, используется опция **-w**:

```
$ last -w
```



```
root-user@ubuntu:~$ last -w
root-user :0          :0          Thu Feb 24 08:40      gone - no logout
reboot   system boot  5.13.0-30-generic Thu Feb 24 08:40      still running
root-user :0          :0          Wed Feb 23 18:17 -   down   (00:08)
reboot   system boot  5.13.0-30-generic Wed Feb 23 18:17 -   18:26 (00:09)
root-user :0          :0          Sat Feb 19 08:16 -   down   (02:01)
reboot   system boot  5.13.0-30-generic Sat Feb 19 08:15 -   10:17 (02:01)
root-user :0          :0          Sat Feb 19 08:07 -   down   (00:07)
reboot   system boot  5.13.0-30-generic Sat Feb 19 08:06 -   08:15 (00:08)
root-user :0          :0          Sat Feb 19 08:00 -   crash  (00:06)
reboot   system boot  5.13.0-30-generic Sat Feb 19 07:58 -   08:15 (00:16)
root-user :0          :0          Thu Feb 17 04:58 -   down   (00:02)
reboot   system boot  5.13.0-30-generic Thu Feb 17 04:57 -   05:01 (00:03)

wtmp begins Thu Feb 17 04:57:40 2022
root-user@ubuntu:~$
```

Ввод опции **-a** перемещает имя хоста в самый конец:

```
$ last -a
```



При просмотре списка сеансов удаленного доступа будет полезной опция **-i**. Она заменяет имя хоста на IP-адрес пользователя:

```
$ last -i
```

Ну и еще упомянем опцию **-R**. При ее вводе команда `last` не станет выводить столбец с именем хоста:

```
$ last -R
```

6. Вывод полного времени

Изначально время начала и конца сеанса записывается в кратком формате. Получить подробную информацию можно через опцию **-F**:

```
$ last -F
```

Еще хотелось бы упомянуть опцию **--time-format**. С ее помощью можно полностью отключить показ даты и времени, за исключением продолжительности самой сессии:

```
$ last --time-format notime
```



А при просмотре сессий удаленного подключения полезным окажется следующий формат вывода:

```
$ last --time-format iso
```



В нем время и дата записываются по стандарту ISO-8601. При этом отдельно выводится часовой пояс подключившегося пользователя.

8. История перезагрузок

Все перезагрузки в Linux отмечаются как действие пользователя **reboot** в команде `last`. А для получения подробных сведений обо всех завершениях сеансов подойдет опция `-x`:

```
$ last -x
```



Мы уже описывали в начале статьи, за что отвечают пользователи **shutdown** и **runlevel**. Так что останавливаться на них не будем.

9. История неудачных входов

История неудачных входов хранится в файле `/var/log/btmp`. Для быстрого получения доступа к ней используется команда **lastb**. Но выполнять ее следует с правами супер-пользователя:

```
$ sudo lastb
```



В остальном у нее такие же опции и синтаксис.

Выводы

В рамках данной статьи мы разобрались с особенностями и нюансами использования команды `last` Linux. С ее помощью получится посмотреть историю входов из системы и выходов для всех пользователей. В том числе и при удаленном подключении к ПК.

Была ли эта информация полезной для вас?

Да

Нет

X

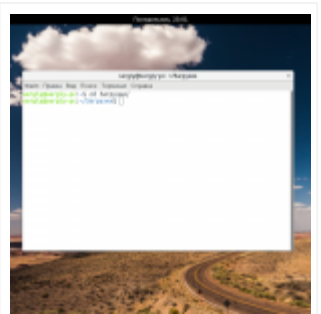
Похожие записи



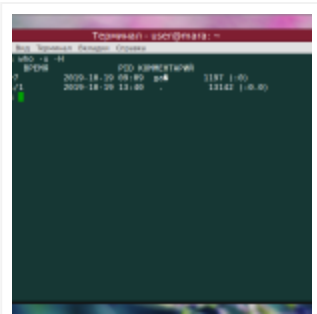
Privacy



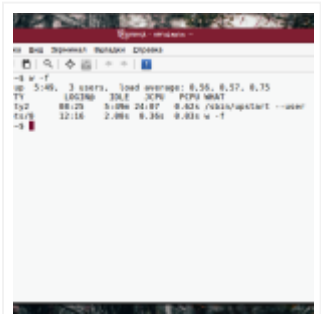
Команда df Linux



Команда cd Linux



Команда who Linux



Команда w 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



Оставьте комментарий

Имя *

Email

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

Комментировать

Русский





ПОИСК ПО КОМАНДАМ



Начните изучать
Linux прямо сейчас!

Карта сайта



Как пользоваться
редактором Vim

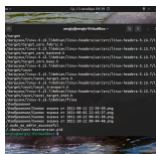
Полезно

[Лучшие](#)[Свежие](#)[Теги](#)[Privacy](#)



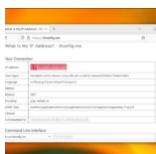
Команда chmod Linux

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

РАССЫЛКА

Ваш E-Mail адрес

☐ Я прочитал(а) и принимаю политику конфиденциальности

Sign up

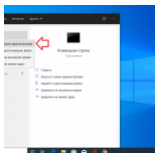


[Privacy](#)



Windows

Списки



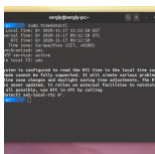
Восстановление Grub после установки Windows 10

2020-08-15



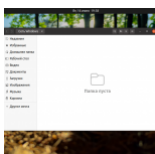
Установка Linux рядом с Windows 10 или 11

2023-02-08



Сбивается время в Ubuntu и Windows

2023-02-18



Ошибка Ubuntu не видит сеть Windows

2023-02-18

[Смотреть ещё](#)

META

Privacy

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

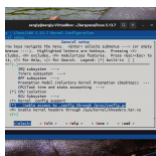
СЛЕДИТЕ ЗА НАМИ В СОЦИАЛЬНЫХ СЕТЯХ



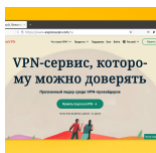
Интересное

[Полезные утилиты для Linux](#)

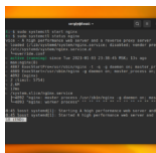
2021-10-12

[Сборка ядра Linux](#)

2021-08-14

[Лучшие VPN сервисы для Linux](#)

2022-10-10

[Управление службами Linux](#)

2023-01-20

[Privacy](#)

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

