



[Главная](#) >> [Команды](#) >> Использование awk в Linux

Использование awk в Linux

Обновлено: 20 апреля 2020 Опубликовано: 23 июня, 2015 от [admin](#), 6 комментариев, время чтения: 8 минут

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

Текст это сердце Unix. Философия "все есть файл" полностью пронизывает всю систему и разработанные для нее инструменты. Вот почему работа с текстом является одним из обязательных навыков не только системного администратора, но и обычного пользователя Linux, который хочет поглубже разобраться в этой операционной системе.

Команда awk - один из самых мощных инструментов для обработки и фильтрации текста, доступный даже для людей никак не связных с программированием. Это не просто утилита, а целый язык разработанный для обработки и извлечения данных. В этой статье мы разберемся как пользоваться awk.

Синтаксис команды awk

Сначала надо понять как работает утилита. Она читает документ по одной строке и выполняет указанные вами действия и выводит результат на стандартный вывод. Одна из самых частых задач, для которых используется awk - это выборка одной из колонок. Результатом параметры awk находятся в кавычках, а действие, которое надо выполнить - в фигурных скобках. Вот основной её синтаксис:

Конфиденциальность - Условия использования

Privacy

\$ awk опции 'условие {действие}'

\$ awk опции 'условие {действие} условие {действие}'

С помощью действия можно выполнять преобразования с обрабатываемой строкой. Об этом мы поговорим позже, а сейчас давайте рассмотрим **опции** утилиты:

- **-F, --field-separator** - разделитель полей, используется для разбиения текста на колонки;
- **-f, --file** - прочитать данные не из стандартного вывода, а из файла;
- **-v, --assign** - присвоить значение переменной, например `foo=bar`;
- **-b, --characters-as-bytes** - считать все символы однобайтовыми;
- **-d, --dump-variables** - вывести значения всех переменных awk по умолчанию;
- **-D, --debug** - режим отладки, позволяет вводить команды интерактивно с клавиатуры;
- **-e, --source** - выполнить указанный код на языке awk;
- **-o, --pretty-print** - вывести результат работы программы в файл;
- **-V, --version** - вывести версию утилиты.

Это далеко не все опции awk, однако их вам будет достаточно на первое время. Теперь перечислим несколько **функций-действий**, которые вы можете использовать:

- **print(строка)** - вывод чего либо в стандартный поток вывода;
- **printf(строка)** - форматированный вывод в стандартный поток вывода;
- **system(команда)** - выполняет команду в системе;
- **length(строка)** - возвращает длину строки;
- **substr(строка, старт, количество)** - обрезает строку и возвращает результат;
- **tolower(строка)** - переводит строку в нижний регистр;
- **toupper(строка)** - переводить строку в верхний регистр.

Функций намного больше, но чтобы не загромождать статью я привел только те, которые мы будем использовать сегодня, а также ещё несколько для чтобы вы могли оценить масштаб возможностей утилиты.

В функциях-действиях можно использовать различные переменные и операторы, вот несколько из них:

- **FNR** - номер обрабатываемой строки в файле;
- **FS** - разделитель полей;
- **NF** - количество колонок в данной строке;
- **NR** - общее количество строк в обрабатываемом тексте;
- **RS** - разделитель строк, по умолчанию символ новой строки;
- **\$** - ссылка на колонку по номеру.


 Privacy

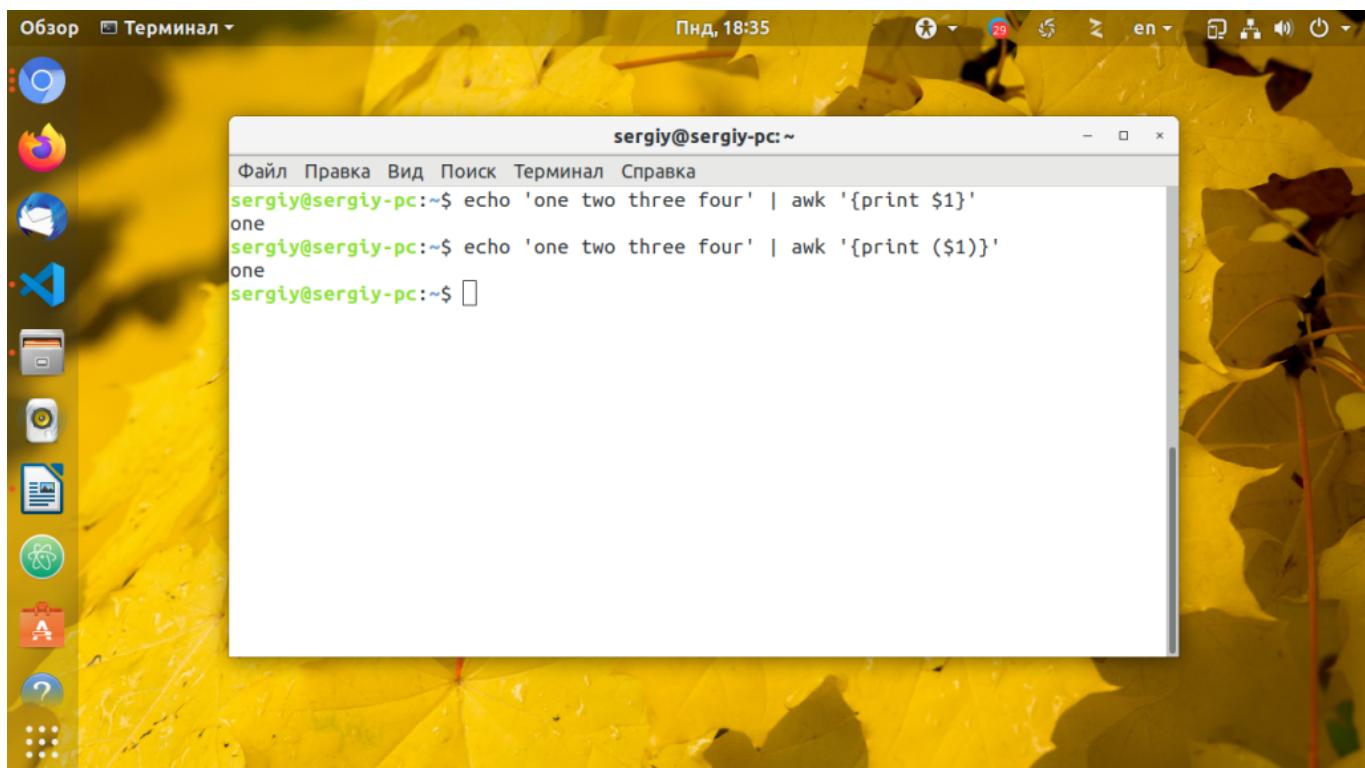
Кроме этих переменных, есть и другие, а также можно объявлять свои.

Условие позволяет обрабатывать только те строки, в которых содержатся нужные нам данные, его можно использовать в качестве фильтра, как grep. А ещё условие позволяет выполнять определенные блоки кода awk для начала и конца файла, для этого вместо регулярного выражения используйте директивы **BEGIN** (начало) и **END** (конец). Там ещё есть очень много всего, но на сегодня пожалуй достаточно. Теперь давайте перейдем к примерам.

Использование awk в Linux

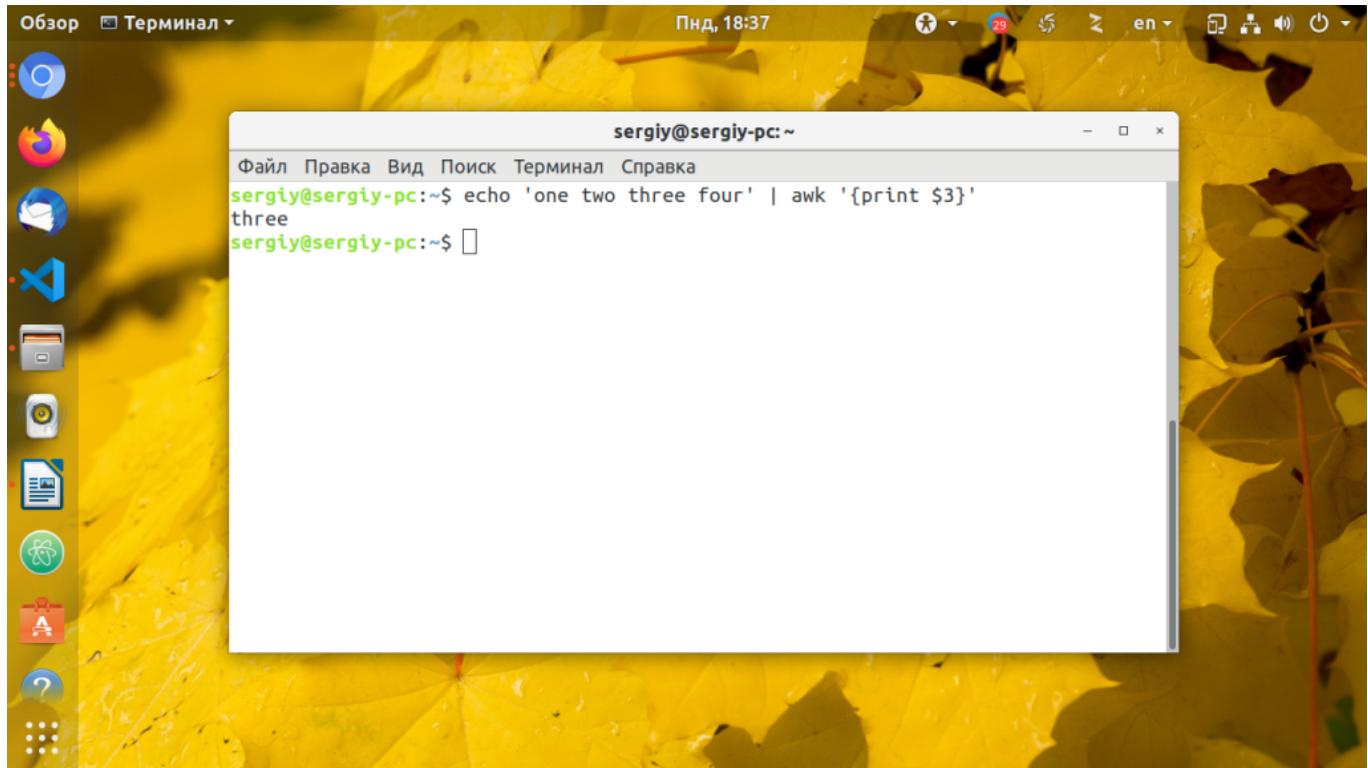
Простейшая и часто востребованная задача - выборка полей из стандартного вывода. Вы не найдете более подходящего инструмента для решения этой задачи, чем awk. По умолчанию awk разделяет поля пробелами. Если вы хотите напечатать первое поле, вам нужно просто использовать функцию `print` и передать ей параметр `$1`, если функция одна, то скобки можно опустить:

```
$ echo 'one two three four' | awk '{print $1}'
```



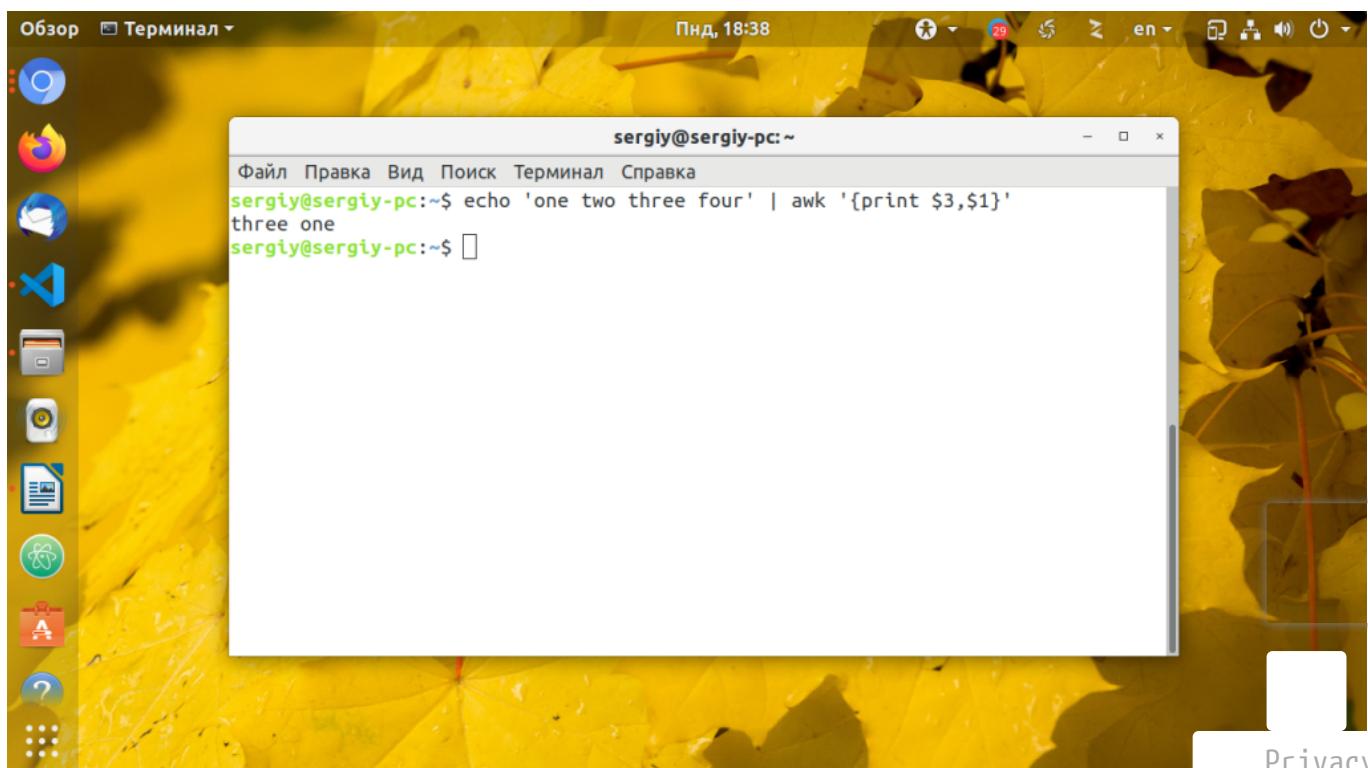
Да, использование фигурных скобок немного непривычно, но это только в первое время. Вы уже догадались как напечатать второе, третье, четвертое, или другие поля? Правильно это `$2`, `$3`, `$4` соответственно.

```
$ echo 'one two three four' | awk '{print $3}'
```



Иногда необходимо представить данные в определенном формате, например, выбрать несколько слов. AWK легко справляется с группировкой нескольких полей и даже позволяет включать статические данные:

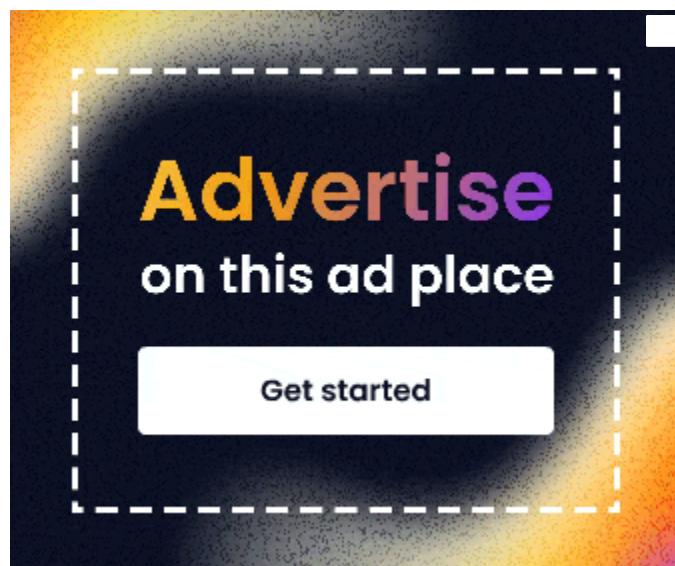
```
$ echo 'one two three four' | awk '{print $3,$1}'
```



```
$ echo 'one two three four' | awk '{print "foo:",$3,"| bar:",$1}'
```



Если поля разделены не пробелами, а другим разделителем, просто укажите в параметре -F нужный разделитель в кавычках, например ":" :

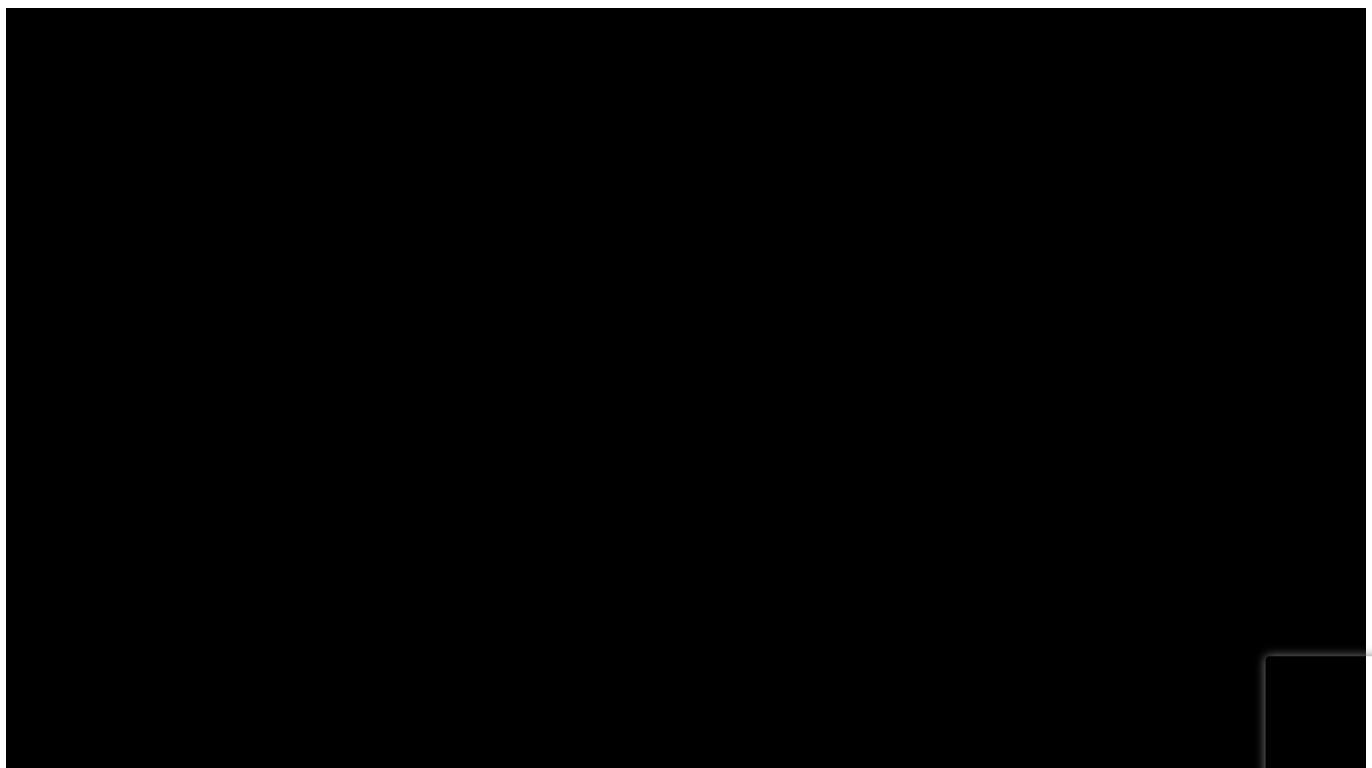


```
$ echo 'one mississippi:two mississippi:three mississippi:four mississippi' | awk -F":" '{print $4}'
```



Но разделитель не обязательно заключать в кавычки. Следующий вывод аналогичен предыдущему:

```
$ echo 'one mississippi:two mississippi:three mississippi:four mississippi' |  
awk -F: '{print $4}'
```



Иногда нужно обработать данные с неизвестным количеством полей. Если вам нужно выбрать последнее поле можно воспользоваться переменной \$NF. Вот так вы мо

Privacy

вывести последнее поле:

```
$ echo 'one two three four' | awk '{print $NF}'
```

Также вы можете использовать переменную \$NF для получения предпоследнего поля:

```
$ echo 'one two three four' | awk '{print $(NF-1)}'
```

Или поля с середины:

```
$ echo 'one two three four' | awk '{print $((NF/2)+1)}'
```

```
$ echo 'one two three four five' | awk '{print $((NF/2)+1)}'
```

Все это можно сделать с помощью таких утилит как sed, cut и grep но это будет намного сложнее.

Как я рассказывал выше, awk обрабатывает одну строку за раз, вот этому подтверждение:

```
$ echo -e 'one 1\n two 2' | awk '{print $1}'
```



Privacy

А вот пример фильтрации с помощью условия, выведем только строку, в которой содержится текст one:

```
$ echo -e 'one 1\n two 2' | awk '/one/ {print $1}'
```

А вот пример использования операций с переменными:

Privacy

```
$ echo -e 'one 1\n two 2' | awk '{sum+=$2} END {print sum}'
```

Это означает что мы должны выполнять следующий блок кода для каждой строки. Это можно использовать, например, для подсчета количества переданных данных по запросам из журнала веб-сервера.

Представьте себе, у нас есть журнал доступа, который выглядит так:

Мы можем подсчитать, что количество переданных байт, это десятое поле. Дальше идёт User-Agent пользователя и он нам не интересен:

```
$ cat /var/log/apache2/access.log | awk '{print $10}'
```

Вот так можно подсчитать количество байт:

```
$ < requests.log awk '{totalBytes+=$NF} END {print totalBytes}'
```

Это только несколько примеров показывающих использование awk в Linux , освоив awk один раз в получите очень мощный и полезный инструмент на всю жизнь.

Была ли эта информация полезной для вас?

Да

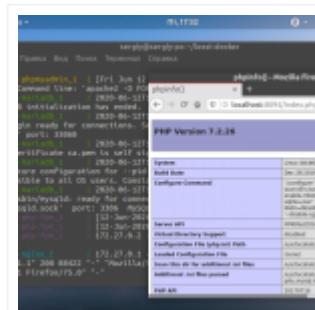
Нет

X

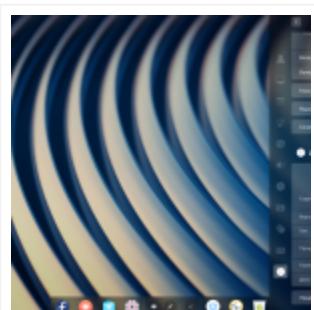
Похожие записи



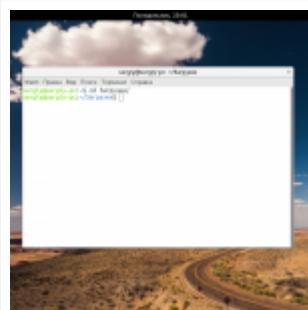
Использование Elasticsearch



Использование Docker для



Установка и
использование



Команда cd Linux



Privacy

Оцените статью

 (15 оценок, среднее: 4,33 из 5)

 [Команды](#)

Об авторе



ADMIN

Основатель и администратор сайта losst.ru, увлекаюсь открытым программным обеспечением и операционной системой Linux. В качестве основной ОС сейчас использую Ubuntu. Кроме Linux, интересуюсь всем, что связано с информационными технологиями и современной наукой.

6 комментариев к “Использование awk в Linux”



Acestor

[11 июля, 2016 в 7:01пп](#)

День добрый!

У Вас в статье описка: \$MF вместо \$NF

[Ответить](#)



admin

[11 июля, 2016 в 7:21пп](#)

Подправил, спасибо!

[Privacy](#)

[Ответить](#)**Nick_Kray**29 марта, 2020 в 8:30 дп

Очень качество подано, информативно и доступно. Благодарю

[Ответить](#)**DIGITAL**28 августа, 2020 в 12:52 пп

читаю из файла с доменами и хочу оставить только домены первых порядков
(например, uslugi.mos.ru -> mos.ru или m.rau.vk.com ->vk.com)
делаю так: awk -F. '{print \$(NF-1) "." \$NF}'
но столкнулся с проблемой если запись с номером (FN-1) отсутствует (например
localhost или undefined не содержат точек), выдаёт ошибку.
чтобы её исключить пришлось добавлять в начале каждой записи подстроку
"по_domain." (любое слово с точкой), а потом исключать записи содержащие
такое слово из выдачи.
может посоветуете более пряморукий способ?

[Ответить](#)**Дмитрий**12 февраля, 2021 в 10:18 дп

Здравствуйте, а почему вы в примере пишете:
< requests.log awk '{totalBytes+=\$NF} END {print totalBytes}'
а на скриншоте:
totalBytes+=\$10
???

[Ответить](#)[Privacy](#)



Илья

26 мая, 2021 в 7:50 дп

Потому что автор \$10 (десятое поле) берет сразу из файла /var/log/apache2/access.log

По всей видимости предыдущий пример задумывался как:

```
cat /var/log/apache2/access.log | awk '{print $10}' > requests.log
```

Но что-то пошло не так.

[Ответить](#)

Оставьте комментарий

Имя *

Email

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

Комментировать

Privacy

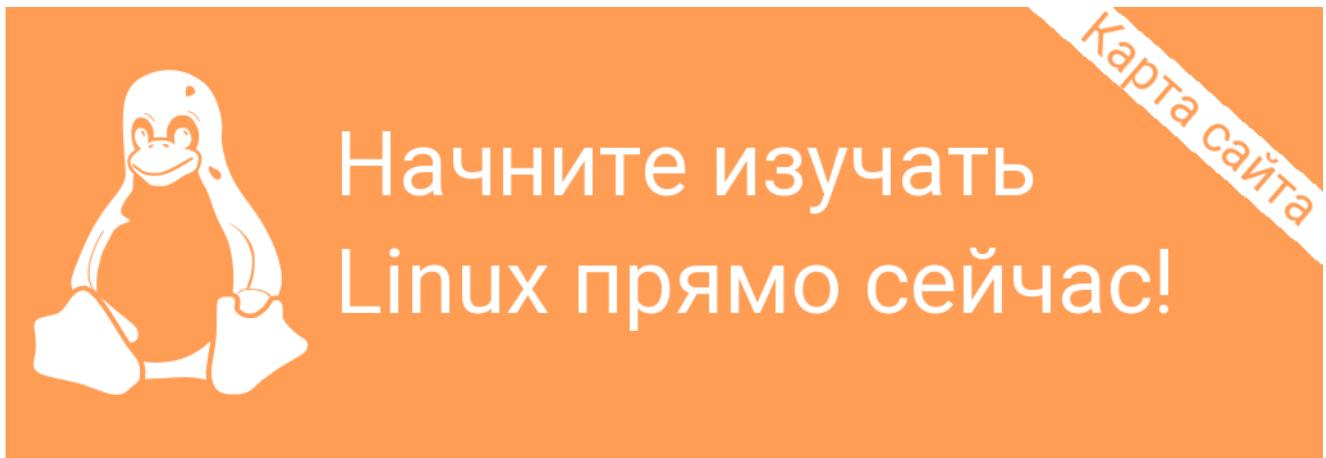
Русский

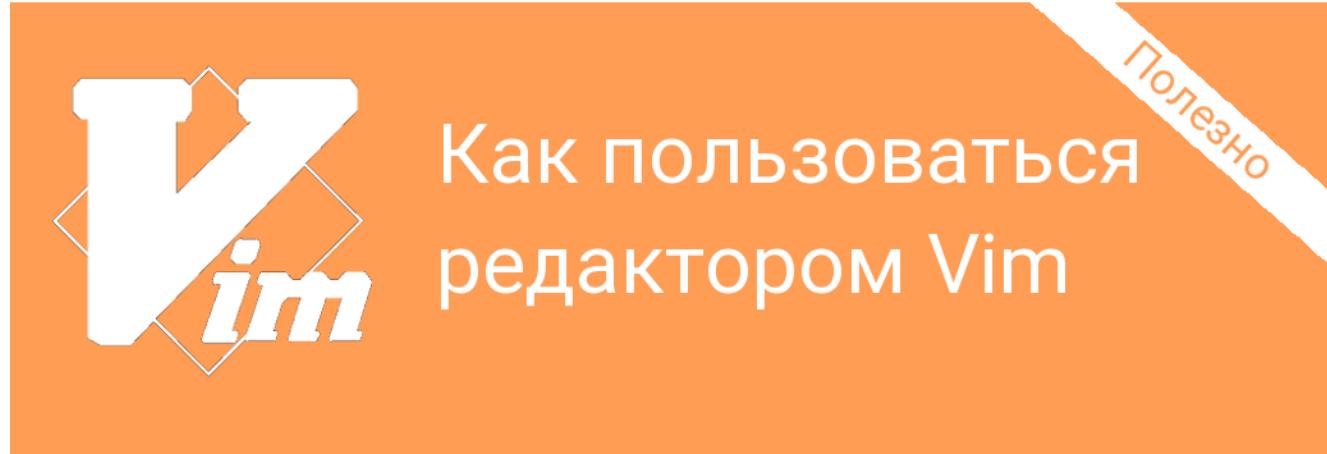
Поиск

ПОИСК ПО КОМАНДАМ

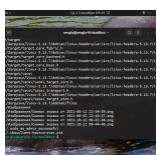
Начните вводить команду

Поиск



[Лучшие](#)[Свежие](#)[Теги](#)[Команда chmod в Linux](#)

2020-04-13

[Команда find в Linux](#)

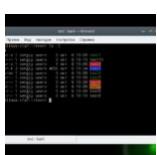
2021-10-17

[Как узнать IP-адрес в Linux](#)

2023-04-14

[Настройка Сроп](#)

2021-10-01

[Права доступа к файлам в Linux](#)

2020-10-09

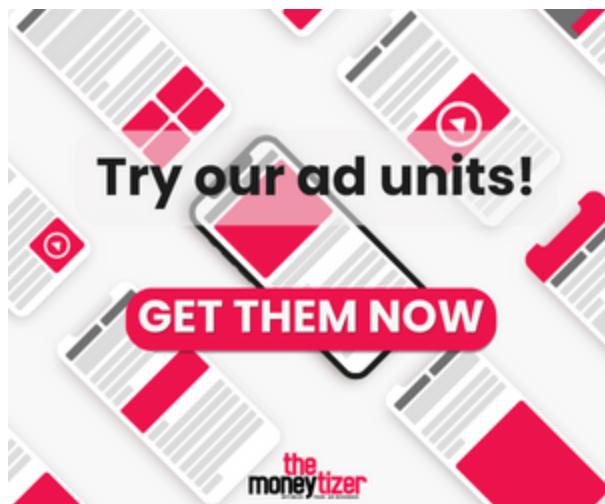
[Privacy](#)

РАССЫЛКА

Ваш E-Mail адрес

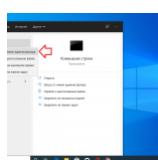
Я прочитал(а) и принимаю политику конфиденциальности

Sign up



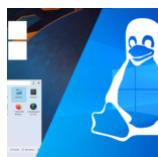
Windows

Списки



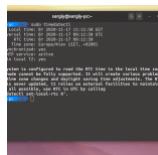
Восстановление Grub после установки Windows 10

2020-08-15



Установка Linux рядом с Windows 10 или 11

2023-02-08

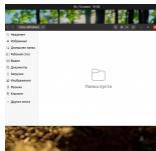


Сбивается время в Ubuntu и Windows

2023-02-18

Ошибка Ubuntu не видит сеть Windows

Privacy



2023-02-18

[Смотреть ещё](#)

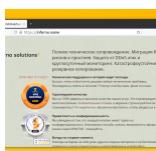
МЕТА

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

СЛЕДИТЕ ЗА НАМИ В СОЦИАЛЬНЫХ СЕТЯХ



Интересное



Лучшие VPS сервера для VPN

2022-09-04

[Privacy](#)



Лучшие темы курсоров Linux

2020-12-18

Classic

Команды Linux для работы с файлами

2020-04-14



Управление службами Linux

2023-01-20

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