



3033.39

Рейтинг

RUVDS.com

VDS/VPS-хостинг. Скидка 15% по коду **HABR15**



Bright_Translate 13 мая 2022 в 15:00

Команда `awk` – примеры использования в Linux и Unix



5 мин



109K

Блог компании RUVDS.com, Настройка Linux*, Системное администрирование*, *nix*

Тutorial

Перевод

Автор оригинала: Dionysia Lemonaki





В этом базовом руководстве вы узнаете самые основы команды `awk`, а также увидите некоторые способы её использования при работе с текстом, включая вывод содержимого файла, а также его конкретных столбцов, строк и слов по указанным критериям. Приступим!

Что это за команда `awk`?

AWK – это скриптовый язык, который полезен при работе в командной строке и широко применяется для обработки текста.

При использовании `awk` вы можете выбирать данные – один или более отдельных фрагментов текста – на основе заданного критерия. Например, с помощью `awk` можно выполнять поиск конкретного слова или шаблона во фрагменте текста, а также выбирать определённую строку/столбец в файле.

Базовый синтаксис `awk`

Простейшая форма команды `awk` подразумевает описание основного действия в одинарных кавычках и фигурных скобках с указанием после него целевого файла.

Выглядеть она может так:

```
awk '{action}' your_file_name.txt
```

Когда вам нужно найти текст, соответствующий конкретному шаблону, или же конкретное слово в

тексте, команда принимает следующий вид:

```
awk '/regex pattern/{action}' your_file_name.txt
```

Создание образца файла

Для создания файла в командной строке используется команда `touch` . Например: `touch filename.txt` , где `filename` – это произвольное имя файла.

Затем можно с помощью команды `open (open filename.txt)` запустить обработчик текста вроде TextEdit, который позволит внести в файл нужное содержимое.

Предположим, у вас есть текстовый файл *information.txt*, содержащий данные, разделённые по столбцам.

Выглядеть этот файл может так:

firstName	lastName	age	city	ID
Thomas	Shelby	30	Rio	400
Omega	Night	45	Ontario	600
Wood	Tinker	54	Lisbon	N/A
Giorgos	Georgiou	35	London	300
Timmy	Turner	32	Berlin	N/A

В приведённом примере мы видим по одному столбцу для `firstName` , `lastName` , `age` , `city` и `ID` .

В любой момент можно просмотреть вывод содержимого вашего файла, выполнив `cat text_file` , где `text_file` представляет имя файла.

Вывод всего содержимого файла

Для вывода всего содержимого файла в качестве действия в фигурных скобках нужно указать `print $0` .

Сработает эта команда аналогично ранее упомянутой `cat` .

```
awk '{print $0}' information.txt
```

Вывод:

firstName	lastName	age	city	ID
Thomas	Shelby	30	Rio	400
Omega	Night	45	Ontario	600
Wood	Tinker	54	Lisbon	N/A
Giorgos	Georgiou	35	London	300
Timmy	Turner	32	Berlin	N/A

Если захотите добавить нумерацию строк, то нужно будет дополнить действие переменной NR :

```
awk '{print NR,$0}' information.txt
```

1	firstName	lastName	age	city	ID
2					
3	Thomas	Shelby	30	Rio	400
4	Omega	Night	45	Ontario	600
5	Wood	Tinker	54	Lisbon	N/A
6	Giorgos	Georgiou	35	London	300
7	Timmy	Turner	32	Berlin	N/A

Вывод конкретных столбцов

При использовании `awk` можно указывать для вывода конкретные столбцы.

Вывод первого производится следующей командой:

```
awk '{print $1}' information.txt
```

Вывод:

```
Thomas
Omega
Wood
Giorgos
Timmy
```

Здесь `$1` означает первое поле, то есть в данном случае первый столбец.

Для вывода второго столбца используется `$2` :

```
awk '{print $2}' information.txt
```

Вывод:

```
lastName
```

```
Shelby
```

```
Night
```

```
Tinker
```

```
Georgiou
```

```
Turner
```

По умолчанию начало и конец каждого столбца `awk` определяет по пробелу.

Для вывода большего числа столбцов, например, первого и четвёртого, нужно выполнить:

```
awk '{print $1, $4}' information.txt
```

Вывод:

```
firstName city
```

```
Thomas      Rio
```

```
Omega       Ontario
```

```
Wood        Lisbon
```

```
Giorgos     London
```

```
Timmy       Berlin
```

Здесь `$1` представляет первое поле ввода (первый столбец), а `$4` четвёртое. При этом они отделяются запятой, чтобы вывод разделялся пробелом и был более читаемым.

Для вывода последнего поля (последнего столбца) также можно использовать команду `$NF` , представляющую последнее поле записи:

```
awk '{print $NF}' information.txt
```

Вывод:

```
ID
400
600
N/A
300
N/A
```

Вывод конкретных строк столбца

Также можно указывать для вывода строку определённого столбца:

```
awk '{print $1}' information.txt | head -1
```

Вывод:

```
FirstName
```

Разделим эту команду на две части. Сначала `awk '{print $1}' information.txt` выводит первый столбец. Затем её результат (который мы видели выше) с помощью символа `|` передаётся на обработку команде `head`, где аргумент `-1` указывает на выбор первой строки столбца.

Для вывода двух строк команда будет такой:

```
awk '{print $1}' information.txt | head -2
```

Вывод:

```
FirstName
Dionysia
```

Вывод строк с заданным шаблоном

Вы можете выводить строку, начинающуюся с заданной буквы. Например:

```
awk '/^0/' information.txt
```

Вывод:

Omega	Night	45	Ontario	600
-------	-------	----	---------	-----

Эта команда выбирает все строки с текстом, начинающимся на 0 .

Действие команды начинается с символа ^ , который указывает на начало строки. После этого прописывается буква, с которой нужная вам строка должна начинаться.

По аналогичному принципу можно выводить строку, завершающуюся конкретным шаблоном:

```
awk '/0$/' information.txt
```

Вывод:

Thomas	Shelby	30	Rio	400
Omega	Night	45	Ontario	600
Giorgos	Georgiou	35	London	300

Эта команда выводит строки, оканчивающиеся на 0 – здесь с помощью символа \$ мы указываем, как должна заканчиваться нужная строка.

При этом её можно несколько изменить:

```
awk '! /0$/' information.txt
```

Символ ! используется в качестве приставки «НЕ», а значит, в этом случае будут выбраны строки, которые не оканчиваются на 0 .

firstName	lastName	age	city	ID
Wood	Tinker	54	Lisbon	N/A
Timmy	Turner	32	Berlin	N/A

Использование регулярных выражений

Для вывода слов, содержащих определённые буквы, а также слов, соответствующих указанному шаблону, мы снова используем прямые слэши.

К примеру, если нас интересуют слова, содержащие `io` , мы пишем:

```
awk ' /io/{print $0}' information.txt
```

Вывод:

Thomas	Shelby	30	Rio	400
Omega	Night	45	Ontario	600
Giorgos	Georgiou	35	London	300

Мы получили строки, в которых содержатся слова, содержащие `io` .

Теперь предположим, что в файле есть дополнительный столбец `department` :

firstName	lastName	age	city	ID	department
Thomas	Shelby	30	Rio	400	IT
Omega	Night	45	Ontario	600	Design
Wood	Tinker	54	Lisbon	N/A	IT
Giorgos	Georgiou	35	London	300	Data
Timmy	Turner	32	Berlin	N/A	Engineering

Для поиска всей информации о людях, работающих в `IT` , нужно указать искомую строку между `//` :

```
awk '/IT/' information.txt
```

Вывод:

Thomas	Shelby	30	Rio	400	IT
Wood	Tinker	54	Lisbon	N/A	IT

А что, если мы хотим увидеть только имена и фамилии сотрудников из `IT` ?

Тогда можно указать столбец так:

```
awk '/IT/{print $1, $2}' information.txt
```

Вывод:

```
Thomas Shelby  
Wood Tinker
```

В этом случае отобразятся только первый и второй столбцы строк, содержащих IT .

При поиске слов, содержащих конкретный шаблон, бывают случаи, когда требуется использовать экранирующий символ:

```
awk '/N\A$/' information.txt
```

Вывод:

```
Wood Tinker 54 Lisbon N/A
```

Моя Все
лента потоки

Разработка Администрирование Дизайн Менеджмент Маркетинг Научпоп



Войти

Я хотела найти строки, оканчивающиеся на N/A . Поэтому при указании критериев поиска в ' // ' , как это показывалось выше, мне пришлось использовать между N/A символ перехода \ . В противном случае возникла бы ошибка.

Использование операторов сравнения

Если вы, предположим, захотите найти всю информацию о сотрудниках в возрасте до 40 лет, то нужно будет использовать оператор сравнения < так:

```
awk '$3 < 40 { print $0 }' information.txt
```

Вывод:

```
Thomas Shelby 30 Rio 400  
Giorgos Georgiou 35 London 300  
Timmy Turner 32 Berlin N/A
```

В выводе представлена информация о людях моложе 40.

Заключение

Вот и всё. Теперь у вас есть необходимая основа для начала работы с `awk` и управления текстовыми данными.

Благодарю за чтение и успехов вам в обучении!

Habrahabr10

Промокод для скидки 10% на виртуальные серверы RUVDS

Теги: `ruvds_перевод`, `awk`, `linux`

Хабы: Блог компании RUVDS.com, Настройка Linux, Системное администрирование, *nix

↑ +47 ↓ 181 ↗ 16 +16

Редакторский дайджест

Присылаем лучшие статьи раз в месяц

Электронная почта



RUVDS.com

VDS/VPS-хостинг. Скидка 15% по коду **HABR15**

[Telegram](#) [ВКонтакте](#) [Twitter](#)



331

219

Карма

Рейтинг

Дмитрий Брайт @Bright_Translate

Переводчик

Комментарии 16



dlinyj 13 мая 2022 в 15:56

Моя производственная практика показала, что эти древние команды BASH используются очень часто. И умение эффективно использовать их выводит на новый уровень. И, как по мне, хоть уже тысячи раз говорили про всевозможные `bash` команды, освежить в памяти всегда полезно. Тем более что часто возвращаюсь к различным публикациям на хабре. Огромное спасибо за перевод!

↑  ↓ Ответить  

o  **lrrr11** 13 мая 2022 в 16:23 ^

строго говоря bash - это bash, а awk - это awk)

↑  ↓ Ответить  

o  **dlinyj** 13 мая 2022 в 17:53 ^

Согласен. Я в том ключе, что базовые команды.

↑  ↓ Ответить  

o  **randomsimpnumber** 13 мая 2022 в 21:42

<https://likegeeks.com/awk-command/>

↑  ↓ Ответить  

o  **A1EF** 14 мая 2022 в 04:04 ^

Тоже добавлю в копилочку неплохую шпаргалку: <https://learnxinyminutes.com/docs/awk/>

↑  ↓ Ответить  

o  **pfemidi** 14 мая 2022 в 00:54

Дочитав до слов «символ перехода» я на некоторое время завис. Но продолжив чтение я понял что имеется в виду «escape character». Посмотрел оригинал и понял что я правильно предположил.

↑  ↓ Ответить  

o  **Bright_Translate** 14 мая 2022 в 09:36 ^

Да, этот термин я неверно истолковал. Исправил.

↑  ↓ Ответить  

o  **shovdmi** 14 мая 2022 в 12:42 

Для создания файла в командной строке используется команда `touch`

Формально говоря, утилита `touch` не для создания файла, а обновления его timestamp. То что она создает файл это её дополнительная фишка. Ее может и не быть в системе, или могут отсутствовать права на ее запуск.

```
man touch
touch - change file timestamps
```

создать можно перенаправлением вывода

```
> newfile.txt
```

↑  ↓ Ответить  ...

o  **orl1an1k** 14 мая 2022 в 23:36

Добавлю из своей практики полезные кейсы.

Ключом -F можно указывать разделитель столбцов:

```
awk -F '*' '{print $1}'
```

Также полезно применять арифметические операторы к переменным. Вывод предпоследнего столбца:

```
awk '{print $NF-1}'
```

А также пару крутых однострочников которые показывают всю прелесть awk с другими командами.

Вывод потребления памяти с сортировкой из ps:

```
ps axo rss,comm,pid | awk '{ proc_list[procinprocist 1/=1024;printf "%.0fMB\t",$1}{print $2}'
```

Потребление SWAP:

```
for file in /proc/*/status ; do awk '/VmSwap|Name/{printf $2 " " $3}END{ print ""}' $file; done | sort -k 2 -n -r | head
```

↑  ↓ Ответить  ...

o  **maslyaeв** 15 мая 2022 в 02:02

Таблички в текстовых файликах по колоночкам? Серьёзно? Почему хотя бы не csv или тот же jsonl?

↑  ↓ Ответить  ...

o НЛО прилетело и опубликовало эту надпись здесь

o  **Gutt** 17 мая 2022 в 20:20 ^

то ни о каких FreeBSD, Linux и прочих Unix-like системах мы бы сейчас и не слышали.

Спасибо, что у нас есть jq, и этот ужас мы теперь тоже парсим прямо в командной строке.

↑  ↓ Ответить  ...

randomsimpnumber 15 мая 2022 в 23:13 ^

Почему хотя бы не csv или тот же jsonl?

Выхлоп почти любой команды достаточно просто преобразовать в вид, понятный awk. А вот преобразовать произвольный текст в json не всегда просто.

↑ ↓ Ответить

Murtagy 14 июл 2022 в 13:46 ^

Часто команды в терминале выдают табличный вывод - ls, lsof, ps и так далее

↑ ↓ Ответить

ner0 19 мая 2022 в 17:57

Как расшифровывается AWK?

↑ ↓ Ответить

Bright_Translate 19 мая 2022 в 19:26 ^

Название AWK складывается из инициалов разработчиков языка: Alfred V. **A**ho, Peter J. **W**einberger и Brian W. **K**ernighan.

↑ ↓ Ответить

Только полноправные пользователи могут оставлять комментарии. Войдите, пожалуйста.

Публикации

ЛУЧШИЕ ЗА СУТКИ ПОХОЖИЕ

ru_vds 23 часа назад

Вам не нужен для этого JavaScript

Средний 11 мин 11K

Мнение

Перевод

+57 150 20 +20

ntsaplin 5 часов назад

Сложный был год

 Простой  12 мин  3.3K

Кейс

 +31  14  3 +3



Holmogorov 6 часов назад

Две цивилизации. Почему пользователи технологий из мегаполисов и небольших городов живут на разных планетах

 Простой  9 мин  4.4K

Мнение

 +23  19  44 +44



ann_vakhromova 5 часов назад

Простой софт, чтобы водители не дрались в очереди на выгрузку металлолома

 8 мин  1.9K

 +19  9  6 +6



MaxRokatansky 22 часа назад

Как я ускорил QEMU в 10 раз с помощью небольшого фокуса

 9 мин  8.7K

Перевод

 +19  46  1 +1



Doctor_IT 23 часа назад

Что такое LUN-СХД, или как мы виртуализировали хранилище

 5 мин  2.1K

Обзор

 +18  21  4 +4



CyberBot 4 часа назад

Домофон, который знает, как вас зовут — своими руками

 Средний  12 мин  1.2K

Кейс

+16

11

6 +6

foxtrotkilomike 7 часов назад

Кастомизируем VS Code для веб-разработки

Простой

10 мин

3К

Мнение

+16

58

2 +2

AntonPolyakov 22 часа назад

Книги, которые помогут прокачать писательские скилы

4 мин

1.9К

+16

56

21 +21

ко_ya 23 часа назад

Даешь магнетрон: микроволновка для советского общепита

Простой

6 мин

3.4К

Ретроспектива

+16

17

9 +9

Показать еще

Ваш аккаунт

- Войти
- Регистрация

Разделы

- Статьи
- Новости
- Хабы
- Компании
- Авторы
- Песочница

Информация

- Устройство сайта
- Для авторов
- Для компаний
- Документы
- Соглашение
- Конфиденциальность

Услуги

- Корпоративный блог
- Медийная реклама
- Нативные проекты
- Образовательные программы
- Стартапам



[Настройка языка](#)

[Техническая поддержка](#)

© 2006–2023, Habr

ИНФОРМАЦИЯ

Сайт	ruvds.com
Дата регистрации	18 марта 2016
Дата основания	27 июля 2015
Численность	11–30 человек
Местоположение	Россия
Представитель	ruvds

ССЫЛКИ

VPS / VDS сервер от 130 рублей в месяц.
ruvds.com

Дата-центры RUVDS в Москве, Санкт-Петербурге, Казани, Екатеринбурге, Новосибирске, Лондоне, Франкфурте, Цюрихе, Амстердаме
ruvds.com

Помощь и вопросы
ruvds.com

Партнерская программа RUVDS
ruvds.com

VPS (CPU 1x2ГГц, RAM 512Mb, SSD 10 Gb) — 190 рублей в месяц
ruvds.com

VPS Windows от 523 рублей в месяц. Бесплатный тестовый период 3 дня.
ruvds.com

VDS в Цюрихе. Дата-центр TIER III — швейцарское качество по низкой цене.
ruvds.com

Антивирусная защита виртуального сервера. Легкий агент для VPS.
ruvds.com

VPS в Лондоне. Дата-центр TIER III — английская точность за рубли.
ruvds.com

VPS с видеокартой на мощных серверах 3,4ГГц
ruvds.com

ПРИЛОЖЕНИЯ



RUVDS Client

Приложение для мониторинга и управления виртуальными серверами RUVDS с мобильных устройств.

Android iOS

ВИДЖЕТ

**КОСМИЧЕСКИЕ
ТЕХНОЛОГИИ,
ЗЕМНЫЕ ЦЕНЫ**



СКИДКА 30%
при оплате за год

Flipper Zero, смарт-часы и наушники
в подарок каждую неделю


Macbook Pro 14
суперприз в конце декабря

Реклама: ООО «МТ ФИНАНС», erid: LjN8KDmrk

БЛОГ НА ХАБРЕ

5 часов назад

Сложный был год

 3.3K  3 +3

23 часа назад

Вам не нужен для этого JavaScript

 11K  20 +20

вчера в 15:00

Графические оболочки FFmpeg

 12K  29 +29

10 дек в 23:10

Фантастический OpenRISC и где он обитает, или недетский разбор детской камеры

 11K  25 +25

10 дек в 20:00

Что нас бесит в интерфейсах игр

 7.7K  70 +70