Анализ данных в Linux. Глава из книги «Bash и кибербезопасность»

Денис Колисниченко, 2 апреля 2026 г.

Содержание статьи

- Исполняемые команды
- cat
- sort
- Использование стандартного ввода и файлов
- Сортировка по нескольким полям
- Удаление повторяющихся строк: uniq
- Подсчет количества строк: wc
- Отображение начала и конца файлов
- Отображение отчета в постраничном режиме
- Фильтрация отчетов: grep
- Фильтрация отчетов: awk
- Итоги

Как работать с большими данными, не зная под рукой книгу? Конечно, можно гуглить. Но зачем? Проверенная временем, бумажная обложка, фирменный запах типографской краски. Сегодня я расскажу вам, как с помощью стандартных утилит Linux обработать данные, отфильтровать ненужное и получить отчет. Все примеры взяты из реальной работы: сортировка по нескольким полям, удаление дубликатов, подсчет строк, фильтрация по шаблонам, вывод отчетов в постраничном режиме. Всё это — в одном посте. Сохраняйте, пригодится.

0 часть

Перед тем, как перейдем к задачам анализа данных, обратимся к истории. Вернемся на 40 лет назад, когда в мире компьютеров доминировали мейнфреймы, а пользователи общались с машинами через перфокарты. Именно, да, перфокарты. Перфокарты — это бумажные карты с отверстиями, которые использовались для ввода данных в компьютеры. Каждая карта — одна строка данных. Ошибиться нельзя — карту не исправишь. Поэтому перфокарты были основой для обработки данных, которая выполнялась на ночной смене.

Сегодня мы, конечно, шагнули далеко вперед. Но принципы, заложенные тогда, остались. Потоки данных, фильтрация, сортировка, подсчет — все это осталось в Linux. Только теперь вместо перфокарт — файлы, вместо ночной смены — терминал, вместо перфокарт — клавиатура. Но принцип тот же: берем данные, обрабатываем, выдаем результат.

В данной статье я покажу, как с помощью стандартных утилит Linux обработать данные, отфильтровать ненужное и получить отчет. Все примеры взяты из реальной работы: сортировка по нескольким полям, удаление дубликатов, подсчет строк, фильтрация по шаблонам, вывод отчетов в постраничном режиме. Всё это — в одном посте. Сохраняйте, пригодится.

Все статьи по курсу данных, какой-нибудь сборник (работа с базой, 3 статьи) — в моем Telegram-канале. Подписывайтесь, чтобы не пропустить. Если вопросы — пишите в комментариях или в Telegram. Разберем каждый случай.

Исполняемые команды

Для начала выясним, какие исполняемые команды есть в системе. Для этого выполним:

```
$ ls /usr/bin | wc -l
1846
```

Команда ls выдает содержимое каталога /usr/bin, а утилита wc подсчитывает количество строк. Мы получили почти 1800 утилит. Конечно, полезных среди них не все, но и не так уже мало. Посмотреть, что есть в системе, можно так:

```
$ ls /usr/bin | head -20
```

Общие параметры команд:

- -h, --help — вывести справку;
- -V, --version — версия;
- -v — многословный вывод (verbose); -q, --quiet — тихий режим; -f — принудительно (force); -n — ничего не делать (dry-run);
- -r, -R, --recursive — рекурсивная обработка (для каталогов);
- -i — запросить подтверждение перед изменением;
- -o — вывод в файл; -a — дописать (append).

Полезные утилиты:

Для подсчета файлов в /usr/bin, по-хорошему, нужна утилита wc -l. Но есть и другие полезные утилиты:

```
$ ls -l /usr/bin | head
$ ls -lt /usr/bin | head
$ ls -lS /usr/bin | head
$ ls -lh /usr/bin | head
```

Можно также выполнить grep по содержимому /usr/bin. Для этого нужно сначала выполнить ls, а затем передать результат grep:

```
$ ls -l /usr/bin | find.exe
```

Будет выполнена утилита поиска find.exe, которая и использована совместно с ls. Где?

cat

Команда cat (от слова concatenate — сцеплять) предназначена для вывода содержимого файлов. Это наиболее частая форма использования. Но возможно и другое использование, например, создание файлов. Рассмотрим основные параметры:

Общие параметры команды:

- -A, --show-all — показать всё (эквивалентно -vET);
- -b, --number-nonblank — нумеровать непустые строки;
- -E, --show-ends — показать символ конца строки (обычно $);
- -n, --number — нумеровать все строки;
- -s, --squeeze-blank — сжать несколько пустых строк в одну;
- -T, --show-tabs — показать табуляции (обычно ^I);
- -v, --show-nonprinting — показать непечатные символы.

Полезные комбинации:

```
$ cat -n file.txt
$ cat -b file.txt
$ cat -s file.txt
$ cat -A file.txt
```

Будет использована утилита nl.exe.

sort

Команда sort сортирует строки. Параметры сортировки:

- -r, --reverse — обратный порядок;
- -n, --numeric-sort — сортировка по числовому значению;
- -k, --key=POS1[,POS2] — сортировка по полям (ключам);
- -t, --field-separator=SEP — разделитель полей (например, -t:);
- -u, --unique — только уникальные строки;
- -f, --ignore-case — игнорировать регистр;
- -b, --ignore-leading-blanks — игнорировать пробелы в начале строки;
- -c, --check — проверить, отсортированы ли данные;
- -o, --output=FILE — вывод в файл;
- -m, --merge — слить уже отсортированные файлы;
- -M, --month-sort — сортировка по месяцам (Jan, Feb, Mar);
- -h, --human-numeric-sort — сортировка по человекочитаемым размерам (1K, 2M, 3G);
- -V, --version-sort — сортировка версий (file1, file2, file10);
- -z, --zero-terminated — строки разделяются нулевым символом (для find -print0);
- --key=POS1,POS2 — ключ сортировки с диапазоном полей (например, -k2,2 — только второе поле);
- --stable — стабильная сортировка (сохраняет порядок равных строк);
- --random-sort — случайная сортировка (перемешивание);
- --sort=WORD — сортировка по слову (WORD: general-numeric, human-numeric, month, version, random).

Использование стандартного ввода и файлов

Команды могут работать с файлами, а могут и с потоком ввода. Посмотрим, что это такое. Выполним:

```
$ ls /usr/bin | sort
```

Мы получили то же самое, что и при выполнении ls /usr/bin. Но теперь вывод передан утилите sort. Теперь сортировка выполняется по первому символу. Но нам нужно сортировать по нескольким полям. Для этого нужно использовать ключ -k:

```
$ ls /usr/bin | sort -k1,1
```

Вывод отсортирован по первому полю. Теперь попробуем отсортировать по второму полю:

```
$ ls /usr/bin | sort -k2,2
```

Вывод изменился, теперь сортировка выполнена по второму полю. Можно задать несколько ключей:

```
$ ls /usr/bin | sort -k1,1 -k2,2
```

Сначала сортировка выполняется по первому полю, затем — по второму. Можно использовать и другие параметры:

```
$ ls /usr/bin | sort -k2,2 -r
$ ls /usr/bin | sort -k2,2 -n
$ ls /usr/bin | sort -k2,2 -f
$ ls /usr/bin | sort -k2,2 -b
$ ls /usr/bin | sort -k2,2 -u
$ ls /usr/bin | sort -k2,2 -M
$ ls /usr/bin | sort -k2,2 -h
$ ls /usr/bin | sort -k2,2 -V
```

Если нужно использовать разделитель, отличный от пробела, используем ключ -t:

```
$ ls -l /usr/bin | sort -t: -k2,2
```

Если нужно сортировать по нескольким полям, то для каждого поля нужно указать ключ -k:

```
$ ls -l /usr/bin | sort -k1,1 -k2,2 -k3,3
```

Если нужно сортировать по полям, начиная с второго, то для этого есть ключ -k2,2:

```
$ ls -l /usr/bin | sort -k2,2 -k3,3
```

Если нужно сортировать только по второму полю, то нужно использовать -k2,2:

```
$ ls -l /usr/bin | sort
```

Через фиг-фиг-хуиг. Сер Ланселот и Сенешаль отвезли раз для развлечения целый грузовик ящиков с ананасами в Замок Ужасных Сисек, где ананасами кормили подвесных (подвесных) сисястых (сисястых) ужасных (ужасных) — ананасами. Это занятие — кормить подвесных сисястых ужасных ананасами — Король Артур, Король Камелота, Король Ужасных Сисек, Король Ужасных Ананасов и Король Ужасных Подвесных Сисястых Ужасных Ананасов Артур-фиг-хуиг, Король Ужасных Сисястых Ананасов, Король Ужасных Подвесных Сисястых Ананасов, Король Ужасных Сисястых Ананасов Артур-фиг-хуиг, Король Ужасных Подвесных Сисястых Ананасов Артур-фиг-хуиг