

Текстовая расшифровка видео:

ЗАПРОСЫ В API ИЗ ТЕРМИНАЛА

План:

- Web API;
- HTTPie.

Web API

В случае с Linux есть две утилиты – **cURL** и **Wget** (конкурируют между собой).

- **Wget** чаще всего используется для скачивания файлов;
- **cURL** используется для общения с API.

Как это выглядит:

Например, мы можем задать через «-X», какой метод хотим использовать:

- -XGET/ -XPOST/ -XPUT/ -XHEAD

Обратите внимание на строчку:

```
~$ curl -s http://url/some_file | tr '[:upper:]' '[:lower:]' | grep -oE '\w+' | sort | uniq
```

Мы скачиваем файл, выводим содержимое напрямую из какого-то файла в консоль. Можем заметить, что у tr есть забавный синтаксис. Можно догадаться, что мы конвертируем регистр для всех символов в файле (из верхнего в нижний). Далее, мы разбиваем файл по словам: по регулярному выражению матчатся все последовательности непробельных символов.

«-oE» сообщает о необходимости использования расширенного синтаксиса регулярных выражений. Также это означает, что мы будем выводить строчки не целиком, а только каждый последующий шаблон, заматченный под регулярное выражение.



Далее, делаем «**sort/ uniq -c**» и еще один sort. Мы делаем sort с командой «-n» (использовать не лексикографическую сортировку, а числовую), далее задаем диапазон колонок, по которым нужно все просмотреть. Так, в примере нас интересует первая колонка, диапазон, соответственно, от первой до первой. Команда «-r» означает сортировку в обратном порядке.

Часто лучше просто сохранять файл, так как в реальных API вас могут забанить.

Рассмотрим на примере:

Посмотрим, какие слова в «Гекльберри Финне» встречаются чаще всего (топ-100):


```
meow-nofer@pikachu ~/Experiments/slurm$ sleep 3 | echo "Hello"
Hello
meow-nofer@pikachu ~/Experiments/slurm$ cat 76-0.txt | tr '[:upper:]' '[:lower:]' | grep -oE '\w+' | sort -nr -k1,1 | head -n 100
6439 and
5077 the
3666 i
3258 a
3022 to
2567 it
2086 t
2044 was
1847 he
1777 of
1555 you
1514 in
1231 that
1129 s
1040 but
965 we
964 so
899 on
```

HTTPIe

cURL и **Wget** – низкоуровневые, которые используют обертку, частично написанную на Python. Данная обертка называется «HTTPIe».

Ее вывод выглядит следующим образом:

```
[~] http PUT httpbin.org/put hello=world
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Origin: *
Connection: keep-alive
Content-Length: 452
Content-Type: application/json
Date: Thu, 06 Dec 2018 15:43:38 GMT
Server: gunicorn/19.9.0
Via: 1.1 vegur
```



Напомним: в консоли cURL и Wget есть в стандартной поставке, а HTTPIe скорее всего не будет (нужно будет ставить дополнительно).

Как вам урок?



Далее >

