

Практическая работа №2

Нагрузочное тестирование

Утилита `ab` поставляется в комплекте с `Apache`.

Пример проведения нагрузочного тестирования:

Для начала давайте организуем нагрузку нашему серверу в одну тысячу последовательных запросов. Для указания числа запросов используется опция `'-n'`. Номер порта можно не указывать, если он не отличен от 80-го:

```
$ ab -n 1000 http://localhost/test.html
```

Итак, что у нас получилось. В процессе тестирования утилита будет сообщать вам о ходе работы:

```
Completed 100 requests
Completed 200 requests
Completed 300 requests
Completed 400 requests
Completed 500 requests
Completed 600 requests
Completed 700 requests
Completed 800 requests
Completed 900 requests
Completed 1000 requests
Finished 1000 requests
```

Далее вы увидите информацию о версии ПО сервера, его имени, какой документ загружался и каков его размер:

Server Software:	Apache/2.2.14
Server Hostname:	localhost
Server Port:	80
Document Path:	/test.html
Document Length:	177 bytes

И далее, собственно, результаты:

```
Concurrency Level:      1
Time taken for tests:    1.500 seconds
Complete requests:      1000
Failed requests:         0
Write errors:            0
Total transferred:      453000 bytes
HTML transferred:       177000 bytes
Requests per second:    666.58 [#/sec] (mean)
Time per request:       1.500 [ms] (mean)
Time per request:       1.500 [ms] (mean, across all concurrent requests)
Transfer rate:          294.88 [Kbytes/sec] received
```

Как видно:

- `Concurrency Level`: количество одновременно-отправляемых запросов — 1;
- `Time taken for tests`: тысяча запросов к серверу заняла 1,5 секунды;

- Complete requests: успешно получен ответ на всю тысячу запросов;
- Failed requests: неудавшихся запросов — ноль;
- Write errors: ошибок записи — ноль;
- Total transferred: общий объём переданных данных: 453000 байт;
- HTML transferred: из них «полезного» HTML — 177000 байт;
- Requests per second: среднее количество запросов в секунду составило 666.58;
- Time per request: среднее время на один запрос 1,5 миллисекунды;
- Transfer rate: скорость обмена данными с сервером составила 294.88 килобайта в секунду.

Далее в выводе идёт информация о времени, затраченном на сетевые подключения:

Connection Times (ms)				
	min	mean[+/-sd]	median	max
Connect:	0	0 0.1	0	2
Processing:	1	1 0.7	1	13
Waiting:	1	1 0.5	1	8
Total:	1	1 0.7	1	14

И на обслуживание запросов сервером:

Percentage of the requests served within a certain time (ms)	
50%	1
66%	1
75%	1
80%	1
90%	2
95%	3
98%	4
99%	5
100%	14 (longest request)

Как видим, сервер успешно справился с тысячей последовательных загрузок статического файла небольшого размера. Давайте теперь посмотрим, как сервер поведёт себя, если вся эта тысяча запросов будет направлена к нему одновременно, указав это при помощи опции '-c':

```
$ ab -n 1000 -c 1000 http://localhost:80/test.html

Benchmarking localhost (be patient)
Completed 100 requests
Completed 200 requests
Completed 300 requests
Completed 400 requests
Completed 500 requests
Completed 600 requests
Completed 700 requests
Completed 800 requests
apr_socket_recv: Connection reset by peer (104)
Total of 804 requests completed
```

Здесь тест не смог быть завершён по причине того, что после одновременной отправки 804 запросов сервер перестал принимать входящие соединения. Опытным путём, снижая количество одновременных запросов, было установлено что безболезненно Apache в текущей конфигурации может обрабатывать примерно 300 одновременных не Keep-Alive запросов.

```
$ ab -n 1000 -c 300 http://localhost:80/test.html

Server Software:      Apache/2.2.14
Server Hostname:      localhost
Server Port:          80

Document Path:        /test.html
Document Length:      55716 bytes

Concurrency Level:     300
Time taken for tests:  13.658 seconds
Complete requests:     1000
Failed requests:        0
Write errors:           0
Total transferred:     55998000 bytes
HTML transferred:      55716000 bytes
Requests per second:   73.22 [#/sec] (mean)
Time per request:      4097.409 [ms] (mean)
Time per request:      13.658 [ms] (mean, across all concurrent requests)
Transfer rate:          4003.91 [Kbytes/sec] received

Connection Times (ms)
  min  mean[+/-sd] median  max
Connect:    0   17 190.0      6   3015
Processing: 224 1659 2376.3    644 13644
Waiting:    212 1628 2360.6    621 13636
Total:      230 1677 2379.8    648 13650

Percentage of the requests served within a certain time (ms)
 50%    648
 66%    654
 75%    668
 80%    785
 90%   7003
 95%   7243
 98%   7384
 99%   7425
100%  13650 (longest request)
```

Естественно, с Keep-Alive запросами дело будет обстоять ещё хуже, поскольку занимаемые Apache'м ресурсы сервера освобождаются не так быстро. Для выполнения теста с Keep-Alive-соединениями просто добавьте опцию '-k':

```
$ ab -k -n 1000 -c 1000 http://localhost:80/test.html
```

Задание: Провести нагрузочное тестирование, аналогичное примеру, при следующих конфигурациях:

1. При подключенном nginx в качестве веб-сервера.
2. При подключенном nginx в качестве веб-сервера и загрузке статики (картинок).