# Nginx、swoole、golang web性能测试

目录

[Nginx、swoole、golang web性能测试 1](#_Toc520994926)

[机器配置： 1](#_Toc520994927)

[swoole 不使用mysql 连接池 2](#_Toc520994928)

[swoole 使用mysql 连接池 3](#_Toc520994929)

[nginx + fastCGI + php 5](#_Toc520994930)

## 机器配置：

**Cpu配置: 单核cpu**

$ lscpu

Architecture: x86\_64

CPU op-mode(s): 32-bit, 64-bit

Byte Order: Little Endian

CPU(s): 1

On-line CPU(s) list: 0

Thread(s) per core: 1

Core(s) per socket: 1

Socket(s): 1

NUMA node(s): 1

Vendor ID: GenuineIntel

CPU family: 6

Model: 69

Stepping: 1

CPU MHz: 2494.258

BogoMIPS: 4988.51

Hypervisor vendor: VMware

Virtualization type: full

L1d cache: 32K

L1i cache: 32K

L2 cache: 256K

L3 cache: 3072K

NUMA node0 CPU(s): 0

**内存配置：1.5G内存**

$ free -mh

total used free shared buffers cached

Mem: 1.5G 1.2G 229M 13M 241M 442M

-/+ buffers/cache: 572M 912M

Swap: 1.9G 0B 1.9G

## swoole 不使用mysql 连接池（2w并发没问题）

$ ab -c 20000 -n 20000 http://192.168.132.128:9502/

This is ApacheBench, Version 2.3 <$Revision: 655654 $>

Copyright 1996 Adam Twiss, Zeus Technology Ltd, http://www.zeustech.net/

Licensed to The Apache Software Foundation, http://www.apache.org/

Benchmarking 192.168.132.128 (be patient)

Completed 2000 requests

Completed 4000 requests

Completed 6000 requests

Completed 8000 requests

Completed 10000 requests

Completed 12000 requests

Completed 14000 requests

Completed 16000 requests

Completed 18000 requests

Completed 20000 requests

Finished 20000 requests

Server Software: swoole-http-server

Server Hostname: 192.168.132.128

Server Port: 9502

Document Path: /

Document Length: 31836 bytes

Concurrency Level: 20000

Time taken for tests: 45.572 seconds

Complete requests: 20000

Failed requests: 0

Write errors: 0

Total transferred: 639740000 bytes

HTML transferred: 636720000 bytes

Requests per second: 438.87 [#/sec] (mean)

Time per request: 45571.932 [ms] (mean)

Time per request: 2.279 [ms] (mean, across all concurrent requests)

Transfer rate: 13709.01 [Kbytes/sec] received

Connection Times (ms)

min mean[+/-sd] median max

Connect: 0 1662 514.6 1668 2563

Processing: 2835 21167 11983.8 20941 42300

Waiting: 7 21144 12013.7 20939 42300

Total: 3427 22828 12497.5 22611 44862

Percentage of the requests served within a certain time (ms)

50% 22611

66% 29666

75% 33691

80% 35881

90% 40360

95% 42625

98% 43948

99% 44416

100% 44862 (longest request)

## swoole 使用mysql 连接池

ab -c 10000 -n 20000 http://192.168.132.128:9501/

This is ApacheBench, Version 2.3 <$Revision: 655654 $>

Copyright 1996 Adam Twiss, Zeus Technology Ltd, http://www.zeustech.net/

Licensed to The Apache Software Foundation, http://www.apache.org/

Benchmarking 192.168.132.128 (be patient)

Completed 2000 requests

Completed 4000 requests

Completed 6000 requests

Completed 8000 requests

Completed 10000 requests

Completed 12000 requests

Completed 14000 requests

Completed 16000 requests

Completed 18000 requests

Completed 20000 requests

Finished 20000 requests

Server Software: swoole-http-server

Server Hostname: 192.168.132.128

Server Port: 9501

Document Path: /

Document Length: 31836 bytes

Concurrency Level: 10000

Time taken for tests: 48.486 seconds

Complete requests: 20000

Failed requests: 0

Write errors: 0

Total transferred: 639740000 bytes

HTML transferred: 636720000 bytes

Requests per second: 412.49 [#/sec] (mean)

Time per request: 24242.771 [ms] (mean)

Time per request: 2.424 [ms] (mean, across all concurrent requests)

Transfer rate: 12885.20 [Kbytes/sec] received

Connection Times (ms)

min mean[+/-sd] median max

Connect: 0 407 324.4 439 1509

Processing: 932 18545 7960.5 20965 29541

Waiting: 0 18267 7741.9 20851 28127

Total: 1237 18952 7870.7 21506 29704

Percentage of the requests served within a certain time (ms)

50% 21506

66% 23813

75% 24843

80% 25355

90% 27193

95% 28207

98% 28983

99% 29242

100% 29704 (longest request)

## nginx + fastCGI + php

nginx 开启4 个worker 进程

fastCGI 初始化200 个进程，最大进程数1024

每一个fastCGI worker 进程，内存大约2.7M

$ ab -c 350 -n 1000 http://192.168.132.128:9508/pool/sync-mysql.php

This is ApacheBench, Version 2.3 <$Revision: 655654 $>

Copyright 1996 Adam Twiss, Zeus Technology Ltd, http://www.zeustech.net/

Licensed to The Apache Software Foundation, http://www.apache.org/

Benchmarking 192.168.132.128 (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

Completed 900 requests

Completed 1000 requests

Finished 1000 requests

Server Software: nginx

Server Hostname: 192.168.132.128

Server Port: 9508

Document Path: /pool/sync-mysql.php

Document Length: 31836 bytes

Concurrency Level: 350

Time taken for tests: 2.561 seconds

Complete requests: 1000

Failed requests: 0

Write errors: 0

Total transferred: 32526978 bytes

HTML transferred: 32345952 bytes

Requests per second: 390.48 [#/sec] (mean)

Time per request: 896.326 [ms] (mean)

Time per request: 2.561 [ms] (mean, across all concurrent requests)

Transfer rate: 12403.54 [Kbytes/sec] received

Connection Times (ms)

min mean[+/-sd] median max

Connect: 1 89 50.9 73 213

Processing: 42 730 618.5 515 2108

Waiting: 0 532 568.3 267 1657

Total: 65 819 642.5 600 2247

Percentage of the requests served within a certain time (ms)

50% 600

66% 659

75% 1229

80% 1543

90% 1983

95% 2210

98% 2221

99% 2246

100% 2247 (longest request)

## 结论

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 并发用户数 | QPS | TPS | 服务器平均处理时间 |
| Swoole+mysql池 | 1w | 412 | 24242.771ms | 2.279ms |
| Swoole + 同步mysql | 2w | 438 | 45571.932ms | 2.424ms |
| Nginx+cgi+php | 350 | 390 | 896.326ms | 2.561ms |

1. **Swoole 比cgi 方式 相比，并发数明显提升多倍**
2. **Swoole mysql 数据库连接池会降低并发，但用户平均等待时长降低大约1倍**
3. **CGI 方式TPS 最低，能支持并发数也最少**

## 性能指标

**并发连接数:**

某个时刻服务器所接受的请求数目，简单的讲，就是一个会话

**并发用户数:**

一个用户可能同时会产生多个会话

**吞吐率QPS：**

指的是某个并发用户数下单位时间内处理的请求数。计算公式：总请求数  / 处理完成这些请求数所花费的时间

**用户平均请求等待时间TPS:**

计算公式：处理完成所有请求数所花费的时间/ （总请求数 / 并发用户数）

**服务器平均请求等待时间（Time per request: across all concurrent requests）**

计算公式：处理完成所有请求数所花费的时间 / 总请求数, 是吞吐率的倒数