

# 高并发 Web 系统的设计与应用

吴锐

(安徽省教育招生考试院 网络信息中心, 安徽 合肥 230022)

**摘要:** 设计了高并发 Web 系统的软硬件框架, 提出了 HTTP 并发数的测量和监控方法, 给出了软硬件的配置和优化方案, 最后以高考网上查分系统为例进行了应用。

**关键词:** 网上查分; 并发; nginx; PHP

**中图分类号:** TP393 **文献标识码:** A **文章编号:** 1009-3044(2013)13-3049-04

## High Concurrent Web Application Design and Application

WU Rui

(Internet Information Center, Anhui Education Examinations Authority, Hefei 230022, China)

**Abstract:** The high concurrency application framework of hardware and software is designed, methods of HTTP concurrent connections number measurement and monitor are proposed, the programs of hardware and software configuration and optimization are advanced. An application example of the college entrance examination scores query is presented in this paper.

**Key words:** online scores query; concurrent; nginx; PHP

通过网络对外发布信息已被政府部门普遍采用, 但由于社会对一些热点信息(如高考分数)的极度关注, 用户每秒数千次访问造成的高并发, 会导致 Web 系统运行缓慢。因此需采用先进的软硬件架构设计 Web 系统, 确定软硬件的最优参数, 充分发挥软硬件的计算能力, 合理决定软硬件的使用数量, 确保系统的正常运行。

## 1 高并发系统

Web 系统的并发一般指的是单位时间内系统与用户之间所有 HTTP 请求与响应的总和<sup>[1]</sup>, 随着用户每秒 HTTP 请求数逐渐增多, 系统运行负载逐渐加重, 系统每秒可完成的 HTTP 连接数逐渐减少。

### 1.1 软硬件架构

基于 x86 硬件平台的操作系统有 Windows 和 Linux 两大类, 业界普遍公认动态网页并发数最高的软件组合是 Linux/Nginx/PHP<sup>[2]</sup>, 高并发 Web 系统由互联网宽带、防火墙、负载均衡器、www 服务器、数据库服务器等组成, 如图 1 所示。

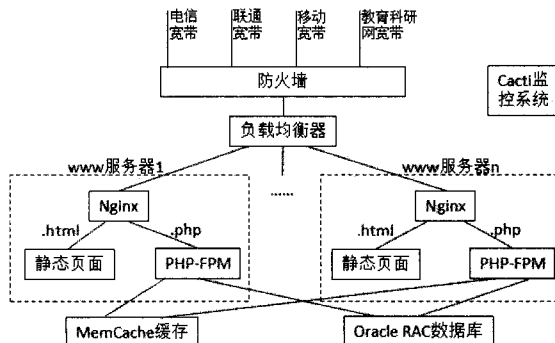


图1 基于Linux/Nginx/PHP高并发系统的架构

防火墙用于保证系统安全, 负载均衡器实现用户 HTTP 请求被分发到多台 www 服务器。

www 服务器的 php 系统采用 FastCGI 结构, Nginx 将 php 动态网页请求转发给驻留内存的 php-fpm 进程, php-fpm 进程运行代码、连接 Oracle 数据库、返回结果给 Nginx。php 对缓存、数据库采用持久化(persistent)连接提高并发性能。

Cacti<sup>[3]</sup>软件每隔一分钟采集一次 HTTP 并发数、CPU 使用率、互联网带宽等数据并绘制图形, 实现对系统负载的定时监控。

收稿日期: 2013-03-25

作者简介: 吴锐(1975-), 男, 工程师, 工学学士。

## 1.2 HTTP并发数的测量和监控

### 1.2.1 HTTP最大并发数的测量

系统HTTP最大并发数的测量是高并发Web系统建设不可缺少的环节,Apache自带的ab软件可用于测量HTTP并发性能<sup>[4]</sup>,命令格式一般为:ab -c pc1 -n tn1 http://ip/url,pc1为并行请求数,tn1为请求总数,测试结果中“Requests per second”数值rs1就是被测系统每秒在pc1个并行请求时可完成的HTTP请求数。随着pc1的增大,rs1变小,当pc1=rs1时,系统处于动态吞吐平衡状态,HTTP并发数达到最大;pc1若再加大,则HTTP请求数超过了系统可处理的请求数,系统对用户的HTTP响应变慢、甚至超时。

### 1.2.2 HTTP并发数的监控

基于Nginx软件的系统,可从NginxStatus状态监控获取服务器与用户的HTTP连接数,NginxStatus状态监控输出的“Active connections”数值a1为当前活动连接数、“Reading”数值r1为读取到客户端Header信息的连接数、“Writing”数值w1为返回给客户端Header信息的连接数、“Waiting”数值wa为已经处理完正在等候下一次请求指令的驻留连接数,HTTP开启Keep-Alive(长连接)时,wa1等于a1-(r1+w1),关闭Keep-Alive时,wa1为0。

## 2 系统设置与优化

### 2.1 系统设置

Linux、Nginx、PHP等软件的具体安装步骤、配置过程可见参考文献<sup>[5-6]</sup>,epoll机制<sup>[7]</sup>是Nginx、PHP等软件实现高并发的关键技术,有关软件的参数配置范围并无现成公式可参考,数值大小跟服务器CPU的计算能力相关。最优参数的寻找与HTTP最大并发数的测量是个互动过程,假设服务器的HTTP最大并发数为M,该文提出有关参数经验数值如下:

- 1)Linux中涉及高并发进程的用户对系统资源的使用上限:nproc为M,nofile为M,stack为M×4K。
- 2)Linux中TCP的syn、listen队列长度最少为M,TIME-WAIT状态连接数最大为M。
- 3)Nginx进程总数为CPU内核数的1或2倍,
- 4)Nginx每个进程的最大连接数为M×2
- 5)PHP-FPM的listen队列长度为M,静态驻留内存的进程数为32至64。

### 2.2 系统优化

系统优化目的是在相同硬件条件下,尽可能地提高系统HTTP最大并发数,实现用户对Web系统的快速访问。

#### 2.2.1 页面长度与HTTP压缩传输

所有页面长度都应尽可能的短,在计算复杂度确定的前提下,网页的HTTP最大并发数与页面长度直接相关。

页面可采用HTTP压缩传输,减少对互联网带宽的需求。ab软件测试表明,启用Nginx静态网页压缩、PHP-FPM动态网页压缩可使长度大于5K字节的纯文字网页传输长度减小为原来的1/3左右,页面越大压缩比越高,但因压缩消耗了部分CPU,页面HTTP最大并发数下降1/4左右。

#### 2.2.2 宽带和DNS

不同宽带运营商之间的网络互连较慢,系统可使用多个运营商的宽带以保证所有用户的快速访问,主要运营商的宽带可能需要多条。

域名可采用DNS视图、DNS轮询<sup>[8]</sup>实现:1、实现各宽带只对该运营商IP地址范围的用户提供流量,2、同运营商多条宽带之间流量均衡。

#### 2.2.3 防火墙

将允许用户访问Web系统HTTP端口的规则设置为第一条,减少防火墙CPU的无效计算,因为防火墙采用最先匹配原则,根据数据包满足条件的第一条过滤规则决定放行还是阻塞。

关闭防火墙的入侵检测、防网络攻击等功能,测试表明,当HTTP并发数达到数百时,开启这些功能将导致防火墙CPU使用率为100%。Linux软件自带的iptables防火墙也要关闭。

#### 2.2.4 负载均衡器

负载均衡器将用户HTTP请求分发给后台的多个www服务器,分发策略有最少连接数(Least-Connection)、源地址等方式:

- 1)基于最少连接数的策略可实现HTTP请求对后台服务器的平均分配,但需应用系统解决不同服务器之间的会话共享问题。
- 2)基于源地址的策略可以实现同一IP地址的请求被分发到同一台服务器上,应用系统不存在服务器之间的会话共享问题,缺点就是不同服务器的负载可能相差较大,因为NAT技术使互联网每个IP地址的实际用户数可能相差较大。

#### 2.2.5 Nginx

关闭HTTP的长连接(Keep-Alive),大多数用户对Web系统的访问是一次性的,打开长连接将导致高并发系统短时间内被迫保持大量tcp连接不关闭,反而加大了系统负载、降低了并发性能。

2.2.6 PHP

缓存、持久化(persistent)连接等技术可大幅提高 PHP 动态网页的并发数：

1)PHP 的 session 数据以哈希表的方式保存在 memcache 缓存,读写速度不会因 session 数据的增多而下降,PHP 的配置如下：

```
session.save_handler=memcache
session.save_path="tcp://ip?persistent=1&weight=1&timeout=1&retry_interval=15"
```

2)持久化连接用于 php 与缓存、数据库之间的数据交换。

2.2.7 Oracle RAC 数据库

Oracle RAC 数据库可实现数据库业务的负载均衡、提高系统可靠性,数据库的索引、SGA 大小对访问速度也有较大影响<sup>[10]</sup>。

3 高考网上查分系统

安徽省教育招生考试院从 2007 年开始使用基于 JBoss/Oracle 软件的高考网上查分系统,投入的查分服务器有十余台,多次发生在并发高峰时查分速度变慢问题。

2012 年高考网上查分系统使用了本文前述的软硬件配置和优化方案,所有页面长度控制在 5K 字节以内,采用 HTTP 压缩传输。负载均衡器采用基于源地址的均衡策略,全省 50 万考生的成绩数据库大小约为 130M 字节,Oracle 数据库的 SGA 设置为 512M 字节即可,数据库中作为查询条件的座位号、身份证号创建索引。宽带、查分服务器、数据库服务器的数量由各设备 HTTP 最大并发数的测量和计算决定。

3.1 查分系统

3.1.1 功能需求

系统正式开放前,使用静态页面 index.html 发布提示信息给考生,查分系统正式开放后,由信息输入 index.php、分数输出 out.php 两个动态页面组成,考生在信息输入动态页面提供座位号、身份证号、验证码,分数输出动态页面根据考生输入的信息从数据库中查出成绩返回给考生。

3.1.2 查询代码

php 采用持久化技术连接 Oracle 数据库的代码如下：

```
function connectDB($oracle)
{
    $dsn = "oci:dbname=//[{$oracle->host}]:{$oracle->port}/{{$oracle->service}}";
    $dbh = new PDO($dsn, $oracle->user, $oracle->password, array(PDO::ATTR_PERSISTENT => $oracle->persistent));
    $dbh->setAttribute(PDO::ATTR_CASE, PDO::CASE_LOWER);
    $dbh->setAttribute(PDO::ATTR_DEFAULT_FETCH_MODE, PDO::FETCH_OBJ);
    $dbh->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
    return $dbh;
}
```

3.2 HTTP 最大并发数与软硬件数量

ab 软件测量 HTTP 压缩传输最大并发数的命令为:ab -c pc1 -n tn1 -H 'Accept-Encoding: gzip' http://ip/url。

分数查询页面 out.php 需修改代码以 GET 方式从 ab 软件命令行获取座位号、身份证号、验证码:ab -c pc1 -n tn1 -H 'Accept-Encoding: gzip' http://ip/out.php?zwh=z0&sfzh=s0&code=c1。因 ab 软件运行时不调用被测网页的嵌套资源,所以信息输入页面 index.php 要分解为静态文字页面 index1.php 和验证码动态图片页面 code.php,分别测量再合成计算,查分服务器的最大并发数也由信息输入和分数查询 2 个页面合成计算,计算方法如下:假设页面 A 最大并发数为 M1、页面 B 最大并发数为 M2,合成并发数为 M,页面 A、页面 B 并发数都为 M/2,根据 CPU 使用率列出方程： $\frac{M/2}{M1} \times 100\% + \frac{M/2}{M2} \times 100\% = 100\%$ ,解得  $M=2/(\frac{1}{M1} + \frac{1}{M2})$ ,测量过程和结果如表 1 所示。

表 1 页面 HTTP 压缩传输最大并发数

测量编号	名称	最大并发数
1	静态首页 index.html	8710
2	信息输入文字提示页面 index1.php	4290
3	信息输入验证码图片页面 code.php	4450
4	分数查询页面 out.php	2130
5	信息输入页面 index.php (页面 2+页面 3,计算)	4368
6	服务器(页面 5+页面 4,计算)	2863
7	防火墙	1.02 万
8	负载均衡器	1.36 万

### 3.3 软硬件数量

防火墙的HTTP最大并发数决定了查分系统的HTTP最大并发数为1.02万。

所有查分服务器的最大并发数之和不低于1.02万即可,所以静态网页服务器2台( $8710 \times 2 = 17420$ ),动态网页服务器4台( $2863 \times 4 = 11452$ ),网页服务器合计4台。

在对单台服务器进行HTTP最大并发数测量时,数据库查询引起的Oracle数据库使用率最大约为3%,4台查分服务器同时运行引起的数据库服务器使用率为12%,使用1台Oracle数据库服务器即可,不需要使用Oracle RAC数据库。

查分系统以增加查分服务器数量为代价使用HTTP压缩传输,将互联网带宽的需求减小为未压缩前的1/3左右,HTTP最大并发数时所需的带宽约为 $1.02 \text{万} \times 5 \text{K字节} \times 8 \div 3 = 136 \text{Mbits}$ ,所以系统需使用2条电信百兆宽带,移动、联通、教育科研网宽带各1条,教育科研网、移动、联通的宽带用户少,预计各宽带的带宽可满足实际需要。

### 3.4 运行效果

安徽省2012年高考网上查分系统可承担的HTTP最大并发数为1.02万,信息输入和分数查询并发数各为5100( $1.02 \text{万} / 2$ ),预计全省50万考生可在98秒之内完成查分。实际运行时,Cacti监控显示HTTP并发数最大约为7000,电信宽带的最大流量约为90Mbits/s,移动、联通、教育科研网3个宽带各自最大流量不超过10Mbits/s,每个考生都可在输入信息后1秒之内获得分数结果,社会反响良好。新查分系统在服务器数量只有往年1/3的情况下,运行获得成功。

### 4 结束语

本文设计的高并发Web系统架构,主要基于Linux/Nginx/PHP等自由软件,可根据硬件配置灵活调整运行参数,HTTP并发数的测量和监控方法简单可行,HTTP最大并发数使系统软硬件的配置和优化具有明确方向,可供各类高并发Web系统建设时参考。

#### 参考文献:

- [1] 李军.高并发Web系统的设计与优化[D].北京:北京交通大学,2009.
- [2] 张宴.实战Nginx:取代Apache的高性能Web服务器[M].北京:电子工业出版社,2010.
- [3] 何斌斌,周恩浩,张波,等.基于Cacti的校园网络监控[J].科技信息,2009(23):60-61.
- [4] 赵佳,赵铭,李昌华.分级网格服务的Apache ab测试分析[J].电子设计工程,2009,17(3):22-24.
- [5] 陶利军.决战Nginx系统卷——高性能Web服务器详解与运维[M].北京:清华大学出版社,2012.
- [6] Nemeth E, Snyder G, Hein T R. Linux系统管理技术手册[M].2版.北京:人民邮电出版社,2008.
- [7] 梁明刚,陈西曲. Linux下基于epoll+线程池高并发服务器实现研究[J].武汉工业学院学报,2012,31(3):54-59.
- [8] Albitz P, Liu C. DNS与BIND[M].4版.北京:中国电力出版社,2002:309-314.
- [9] php.net. PHP Data Objects[EB/OL].(2012)[2012-05-08].<http://php.net/manual/en/book.pdo.php>.
- [10] 潘伟. ORACLE SGA的性能优化[J].中国水运,2009,9(6):117-118.

---

(上接第3036页)

设置支付工程中通知功能的实现方法,与设置获取支付成功后返回信息的实现方法基本相同,需要分别调用GetMD5()、BubbleSort()和Get\_Http()三个方法。

### 4 结束语

在线支付模块是购物网站中的关键模块。该文在分析在线支付流程的基础上,给出了利用支付宝接口程序进行在线支付的实现方法。在实现过程中利用了MD5加密算法对客户信息进行加密,保证了在线支付的安全性和可靠性。

#### 参考文献:

- [1] 高宏,李俊民.ASP.NET典型模块与项目实战大全[M].北京:清华大学出版社,2012.
- [2] 尚维来,周键.在线支付在商务资源库系统中的应用研究[J].科技信息,2012(36).
- [3] 奚嘉敏.基于J2EE的网上购物系统的设计与实现[D].上海:复旦大学,2012.
- [4] 李毅兵,张哲.基于B/S模式的在线支付系统的设计与实现[J].电脑开发与应用,2012(8).
- [5] 吴代文,郭军军.电子商务网站在线支付模块的集成研究[J].信息技术,2012(7).
- [6] 李旭东.浅谈在线支付对电子商务的影响[J].中国外资,2012(17).