

Nginx|Tengine

讲师：小陈老师

- Nginx
- Nginx ("engine x") 是一个高性能的 HTTP 和 反向代理 服务器，也是一个 IMAP/POP3/SMTP 代理服务器。
- 第一个公开版本0.1.0发布于2004年10月4日。
- 其将源代码以类BSD许可证的形式发布，因它的稳定性、丰富的功能集、示例配置文件和低系统资源的消耗而闻名
- 官方测试nginx能够支撑5万并发链接，并且cpu、内存等资源消耗却非常低，运行非常稳定

- 2011年6月1日，nginx 1.0.4发布。
- Nginx是一款轻量级的Web 服务器/反向代理服务器及电子邮件（IMAP/POP3）代理服务器，
- 并在一个BSD-like 协议下发行。由俄罗斯的程序设计师Igor Sysoev所开发，
- 其特点是占有内存少，并发能力强，事实上nginx的并发能力确实在同类型的网页服务器中表现较好，中国大陆使用nginx网站用户有：新浪、网易、腾讯等。
- 功能：
- web服务器
- web reverse proxy
- smtp reverse proxy

Nginx和Tengine



- Nginx和apache的优缺点

- 1、nginx相对于apache的优点：
轻量级，同样起web 服务，比apache 占用更少的内存及资源
抗并发，nginx 处理请求是异步非阻塞的，而apache 则是阻塞型的，在高并发下nginx
能保持低资源低消耗高性能
高度模块化的设计，编写模块相对简单
社区活跃，各种高性能模块出品迅速啊
- 2.apache 相对于nginx 的优点：
rewrite ，比nginx 的rewrite 强大
模块超多，基本想到的都可以找到
少bug ，nginx 的bug 相对较多
- 3、Nginx 配置简洁, Apache 复杂
- 4、最核心的区别在于apache是同步多进程模型，一个连接对应一个进程；nginx是异步的，多个连接（万级别）可以对应一个进程

- Tengine 是nginx的加强版，封装版，淘宝开源
 - [官网http://tengine.taobao.org/](http://tengine.taobao.org/)
 - [动态模块加载 \(DSO \)](#)支持。加入一个模块不再需要重新编译整个Tengine；
 - [支持SO_REUSEPORT选项](#)，建连性能提升为[官方nginx的三倍](#)；
 - 支持[SPDY v3协议](#)，自动检测同一端口的SPDY请求和HTTP请求；
 - [流式上传](#)到HTTP后端服务器或FastCGI服务器，大量减少机器的I/O压力；
 - 更加强大的负载均衡能力，包括[一致性hash模块](#)、[会话保持模块](#)，[还可以对后端的服务器进行主动健康检查](#)，根据服务器状态自动上线下线，以及[动态解析upstream中出现的域名](#)；
 - [输入过滤器机制](#)支持。通过使用这种机制Web应用防火墙的编写更为方便；
 - 支持设置proxy、memcached、fastcgi、scgi、uwsgi[在后端失败时的重试次数](#)
 - [动态脚本语言Lua](#)支持。扩展功能非常高效简单；
 - [支持管道 \(pipe \) 和syslog \(本地和远端 \) 形式的日志以及日志抽样](#)；
 - 支持按指定关键字(域名，url等)[收集Tengine运行状态](#)；
 - [组合多个CSS、JavaScript文件的访问请求变成一个请求](#)；
 - [自动去除空白字符和注释](#)从而减小页面的体积
 -

- 什么是高并发和负载均衡？
- 如何解决高并发和负载均衡
- 什么是高可用？
- 如何解决高可用问题
- 负载均衡的session一致性问题

- 单个tomcat支持最高并发
- 怎么解决高并发问题，解决单个服务器过载问题

1可以看到在达到600用户同时在线的时候，系统响应时间为6秒钟

100人-----响应时间0.8秒 完美

150人-----响应时间1秒 完美

200人-----响应时间1.5秒 响应时间有微小波动 比较完美

250人-----响应时间1.8秒 比较完美（此时是理想情况下最大的并发用户数量）

280人-----开始出现连接丢失问题，连接开始不稳定

300人-----响应时间3秒 响应时间有较大波动峰值为6秒 较差

350人-----响应时间3秒 开始大量出现连接丢失问题 连接很不稳定

400人-----响应时间3.8秒 连接丢失数量达到3000次以上

450人-----响应时间4秒 连接丢失数量达到6000次以上

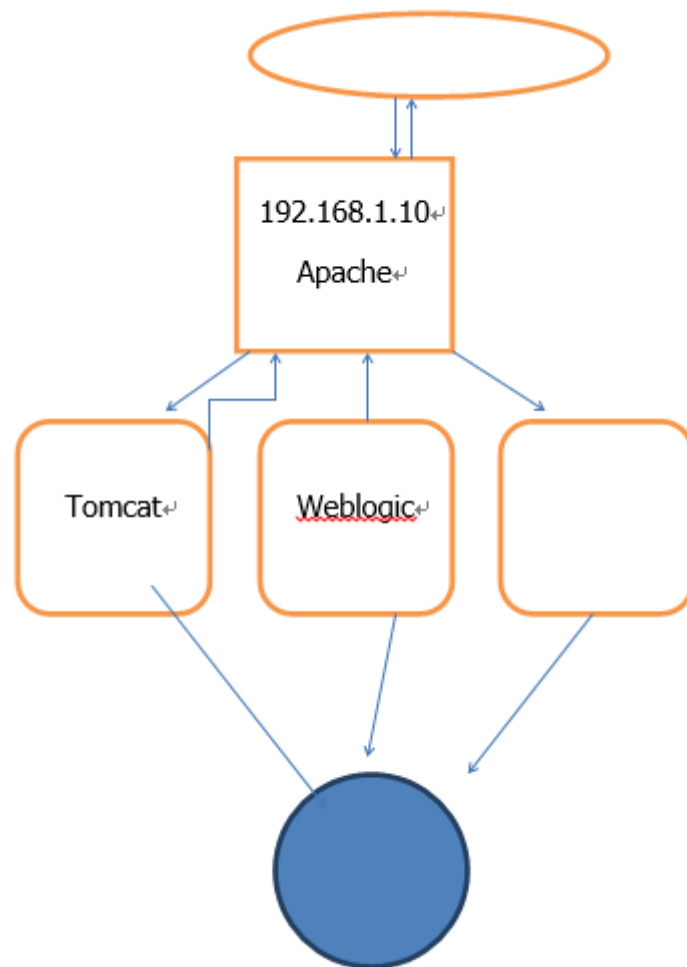
500人-----响应时间4秒 连接丢失数量达到11000次以上

550人-----响应时间6秒 连接丢失数量达到21000次以上

600人-----响应时间6秒 连接丢失数量达到25000次以上

600人开始系统出现异常情况，因此停止测试。测试数据到此为止。

- 前端和后端架构
- Tomcat,一个Servlet和JSP容器
- 前端服务器处理静态页面



- 安装之前准备
- 1、依赖 gcc openssl-devel pcre-devel zlib-devel
- 安装：yum install gcc openssl-devel pcre-devel zlib-devel
- 2、创建用户和用户组。为了方便nginx运行而不影响linux安全
- 创建组：groupadd -r nginx
- 创建用户：useradd -r -g nginx -M nginx
- -M 表示不创建用户的家目录。
- 简洁方式：
 - ./configure \
 - --prefix=/usr/tengine
 - make && make install

- 安装

- ./configure \
- --prefix=/usr \
- --sbin-path=/usr/sbin/nginx \
- --conf-path=/etc/nginx/nginx.conf \
- --error-log-path=/var/log/nginx/error.log \
- --http-log-path=/var/log/nginx/access.log \
- --pid-path=/var/run/nginx/nginx.pid \
- --lock-path=/var/lock/nginx.lock \
- --user=nginx \
- --group=nginx \
- --with-http_ssl_module \
- --with-http_flv_module \
- --with-http_stub_status_module \
- --with-http_gzip_static_module \
- --http-client-body-temp-path=/var/tmp/nginx/client/ \
- --http-proxy-temp-path=/var/tmp/nginx/proxy/ \
- --http-fastcgi-temp-path=/var/tmp/nginx/fcgi/ \
- --http-uwsgi-temp-path=/var/tmp/nginx/uwsgi \
- --http-scgi-temp-path=/var/tmp/nginx/scgi \
- --with-pcre
- make && make install

- 安装
 - ./configure \
 - --prefix=/opt/sxt/soft/tengine-2.1.0/ \
 - --error-log-path=/var/log/nginx/error.log \
 - --http-log-path=/var/log/nginx/access.log \
 - --pid-path=/var/run/nginx/nginx.pid \
 - --lock-path=/var/lock/nginx.lock \
 - --with-http_ssl_module \
 - --with-http_flv_module \
 - --with-http_stub_status_module \
 - --with-http_gzip_static_module \
 - --http-client-body-temp-path=/var/tmp/nginx/client/ \
 - --http-proxy-temp-path=/var/tmp/nginx/proxy/ \
 - --http-fastcgi-temp-path=/var/tmp/nginx/fcgi/ \
 - --http-uwsgi-temp-path=/var/tmp/nginx/uwsgi \
 - --http-scgi-temp-path=/var/tmp/nginx/scgi \
 - --with-pcre
 - make && make install
 - 启动和配置路径用默认的，用户和用户组限制，都去掉
 - 其中/var/tmp/nginx/client/目录需要手动创建

- 添加安装的tengine到注册表，具体内容见附件nginx

```
8 start() {
9     [ -x $nginx ] || exit 5
10    [ -f $NGINX_CONF_FILE ] || exit 6
11    # make_dirs
12    echo -n "Starting $prog: "
13    daemon $nginx -c $NGINX_CONF_FILE
14    retval=$?
15    echo
16    [ $retval -eq 0 ] && touch $lockfile
17    return $retval
18 }
19
20 stop() {
21    echo -n "Stopping $prog: "
22    killproc $prog -QUIT
23    retval=$?
24    echo
25    [ $retval -eq 0 ] && rm -f $lockfile
26    return $retval
27 }
28
```

- 注意修改路径，**而且必须是在/etc/init.d、下面touch或者vi来新建**
- **不能用xftp传进去，否则文件不被识别**

```
#!/bin/bash
#
# chkconfig: - 85 15
# description: nginx is a World Wide Web server. It is used to serve
# Source function library.
. /etc/rc.d/init.d/functions

# Source networking configuration.
. /etc/sysconfig/network

# Check that networking is up.
[ "$NETWORKING" = "no" ] && exit 0

nginx="/usr/tengine-2.1/sbin/nginx"
prog=$(basename $nginx)

NGINX_CONF_FILE="/usr/tengine-2.1/conf/nginx.conf"

#[ -f /etc/sysconfig/nginx ] && . /etc/sysconfig/nginx

lockfile=/var/lock/subsys/nginx
```

- 1、修改nginx文件的执行权限
 - `chmod +x nginx`
- 2、添加该文件到系统服务中去
 - `chkconfig --add nginx`
 - 查看是否添加成功
 - `chkconfig --list nginx`
- 启动，停止，重新装载
- `service nginx start|stop|reload`