

Nginx 配置文件详解

```
user www www;
```

#程序运行用户和组

```
worker_processes auto;
```

#启动进程，指定 nginx 启动的工作进程数量，建议按照 cpu 数目来指定，一般等于 cpu 核心数目

```
error_log /home/wwwlogs/nginx_error.log crit;
```

#全局错误日志

```
pid /usr/local/nginx/logs/nginx.pid;
```

#主进程 PID 保存文件

```
worker_rlimit_nofile 51200;
```

#文件描述符数量

```
events
```

```
{
```

```
use epoll;
```

#使用 epoll 模型，对于 2.6 以上的内核，建议使用 epoll 模型以提高性能

```
worker_connections 51200;
```

#工作进程的最大连接数量，根据硬件调整，和前面工作进程配合起来用，尽量大，但是别把 cpu 跑到 100% 就行每个进程允许的最多连接数，理论上每台 nginx 服务器的最大连接数为 $worker_processes * worker_connections$ ，具体还要看服务器的硬件、带宽等。

```
}
```

```
http
```

#整体环境配置--网站配置

```
{
```

```
include mime.types;
```

```
default_type application/octet-stream;
```

#设定 mime 类型,文件传送类型由 mime.type 文件定义

```
server_names_hash_bucket_size 128;
```

#保存服务器名字的 hash 表大小

```
client_header_buffer_size 32k;
```

#客户端请求头部缓冲区大小

```
large_client_header_buffers 4 32k;
```

#最大客户端头缓冲大小

```
client_max_body_size 50m;
```

```
#客户端最大上传文件大小（M）
```

```
sendfile on;
```

#sendfile 指令指定 nginx 是否调用 sendfile 函数来输出文件，对于普通应用，必须设为 on。如果用来进行下载等应用磁盘 IO 重负载应用，可设置为 off，以平衡磁盘与网络 I/O 处理速度，降低系统的 uptime。

```
tcp_nopush on;
```

#这个是默认的，结果就是数据包不会马上传送出去，等到数据包最大时，一次性的传输出去，这样有助于解决网络堵塞。（只在 sendfile on 时有效）

```
keepalive_timeout 60;
```

```
#连接超时时间
```

```
tcp_nodelay on;
```

```
#禁用 nagle 算法，也即不缓存数据。有效解决网络阻塞
```

```
fastcgi_connect_timeout 300;
```

```
fastcgi_send_timeout 300;
```

```
fastcgi_read_timeout 300;
```

```
fastcgi_buffer_size 64k;
```

```
fastcgi_buffers 4 64k;
```

```
fastcgi_busy_buffers_size 128k;
```

```
fastcgi_temp_file_write_size 256k;
```

```
#fastcgi 设置
```

```
gzip on;
```

```
gzip_min_length 1k;
```

```
gzip_buffers 4 16k;
```

```
gzip_http_version 1.1;
```

```
gzip_comp_level 2;
```

```
gzip_types text/plain application/javascript application/x-javascript text/javascript  
text/css application/xml application/xml+rss;
```

```
gzip_vary on;
```

```
gzip_proxied expired no-cache no-store private auth;
```

```
gzip_disable "MSIE [1-6]\\.";
```

```
#limit_conn_zone $binary_remote_addr zone=perip:10m;
```

```
##If enable limit_conn_zone,add "limit_conn perip 10;" to server section.
```

```
server_tokens off;
```

```
#隐藏 nginx 版本号（curl -I 192.168.4.154 可以查看，更加安全）
```

```
#log format
```

```
log_format access '$remote_addr - $remote_user [$time_local] "$request" '
'$status $body_bytes_sent "$http_referer" '
'"$http_user_agent" $http_x_forwarded_for';
#定义日志格式
```

server

```
{
    listen 80 default_server;
    #listen [::]:80 default_server ipv6only=on;
    #监听 80 端口，WEB 服务的监听设置，可以采用"IP 地址:端口"形式
    server_name www.lnmp.org lnmp.org;
    #服务器名，可以写多个域名，用空格分隔
    index index.html index.htm index.php;
    #默认网页文件
    root /home/wwwroot/default;
    #网页主目录

    #error_page 404 /404.html;
    include enable-php.conf;

    location /nginx_status
    {
        stub_status on;
        access_log off;
    }
    #开启 status 状态监测
    location ~ .*\. (gif|jpg|jpeg|png|bmp|swf)$
    {
        expires 30d;
    }
    #静态文件处理，保存期 30 天
    location ~ .*\. (js|css)?$
    {
        expires 12h;
    }
    #js 和 css 文件处理，保存期 12 小时
    location ~ /\.
    {
        deny all;
    }

    access_log /home/wwwlogs/access.log access;
```

```
#正确访问日志
```

```
    }  
include vhost/*.conf;  
#vhost/下子配置文件生效  
}
```

尚硅谷