链滴

Nginx配置文件详细说明(转)

作者：dwei
原文链接：https://ld246.com/article/1451484559958
来源网站：链滴
许可协议：署名-相同方式共享 4.0 国际 (CC BY-SA 4.0)

在此记录下Nginx服务器nginx.conf的配置文件说明, 部分注释收集与网络.
<blockquote>
#运行用户
user www-data;
#启动进程,通常设置和cpu的数量相等
worker_processes 1;
#全局错误日志及PID文件
error_log /var/log/nginx/error.log;
pid   /var/run/nginx.pid;
#工作模式及连接数上限
events {
    use   epoll   #epol是多路复用IO(I/O Multiplexing)中的一种方式,但是仅用于linux2.6以上内核,可以大大提高nginx的能
    worker_connections 1024;#单个后台worker process进的最大并发链接数
    # multi_accept on;
}
#设定http服务器，利用它的反向代理功能提供负载均衡支持
http {
     #设定mime类型,类型由mime.type文件定义
    include nbsp;   /etc/nginx/mime.types;
    default_type application/octet-stream;
    #设定日志格式
    access_log   /var/log/nginx/access.log;
    #sendfile 指令指定 nginx 是否调用 sendfile 函数（zero copy 方式）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.
 &bsp; sendfile   on;
 &nbp; #tcp_nopush   on;
    #连接超时时间
    #keepalive_timeout&nbp; 0;
    keepalive_timeout 65;
    tc_nodelay   on;
    nbsp;
    #开启gzip压缩
    gzip on; br />   gzip_disable "MSIE [1-6]\.(?!.*SV1)";
    #设定请求缓冲
    client_header_buffer_siz   1k;
    large_client_header_buffers 4 4k
    include /etc/nginx/conf.d/*.conf;
    include /etc/nginx/sites-enabled/*;
    #设定负载均衡的服务器列表
     upstream mysvr {
    #weigth参数表示权值，权值越高被分配到的几率越大<r />   #本机上的Squid开启3128端口
    server 192168.8.1:3128 weight=5;
    server 192.168.8.2:80 weight=1;<b />   server 192.168.8.3:80 weight=6;

   server {
    #侦听80端口
 &nbp;   listen   80;<b />   #定义使用www.xx.com访问
        server_nam www.xx.com;
        #设定本虚拟主机的访问日志
&nbp;   access_log logs/www.xx.com.access.log nbsp; main;
    #默认请求
    location / {
 &nsp;   root   /root;   #定义服务器的默认网站根目录位置
    &bsp;   index index.php index.html index.htm;   #定义页索引文件的名称
          fastcgi_pass ww.xx.com;
         fastcgi_param&nbp; SCRIPT_FILENAME $document_root/$fastcgi_script_name;
 &nbp;   include /etc/nginx/fastcgi_params;

        }
    # 定义错误提示页面
    error_page
nbsp; 500 502 503 504 /50x.html;
    &
bsp; location = /50x.html {
       r
ot   /root;
    }
    #静态文件，nginx自己处理
    location ~ ^
(images|javascript|js|css|flash|media|static)/ {
  root /var/www/virtual/htdocs;
     &nbs
; #过期30天，静态文件不怎么更新，过期可以设大一点，如果频繁更新，则可以设置得小一点。
       expires 30d;

    #PHP 脚本请求全部转发到 FastCGI处理. 使用FastCGI默认配置.
    location ~ \.php$ {
nbsp; root /root;
       fastcgi_pass 127.0.0
1:9000;
       fastcgi_index index.php;
       fastcgi_param SCRIPT_FILENAME /home
www/www$fastcgi_script_name;
       incl
de fastcgi_params;
    }
    #设定查看Ngin
状态的地址
    location /NginxStatus {
   &
bsp;   stub_status   &
nbp;   on;
       access_l
g   &nb
p; on;
       auth_basic &nb
p;   "NginxStatus";
        auth_basic_user_file conf/htpass
d;
    }
    #禁止访问 .htxxx 文件
&nb
p;   location ~ /\.ht {
        d
ny all;
    }

nbsp;   }
}
</blockquote>
以上是一些基本的配置,使用Nginx最大的好处就是负载均衡
如果要使用负载均衡的话,可以修改配置http节点如下：
<blockquote>
#设定http服务器，利用它的反向代理功能提供负载均衡支持
http {
   #设定mime类型,类型由mime.type文件定义
    include
nbsp;   /etc/nginx/mime.types;
default_type application/octet-stream;
    #设定日志格式
    access_log   /var/log/nginx/access.log;
    #省略上文有的一些配置节点
    #。。。。。。。。。。。
    #设定负载均衡的服务器列表
     upst
eam mysvr {
    #weigth参数表示权值，权值越高被分配到的几率越大<
r />   server 192.168.8.1x:3128 weight=5;#本机上的Squid开启3128端口
    server 192.168.8.2x:80 weight=1;
erver 192.168.8.3x:80 weight=6;
    }
   upstream mysvr2 {
    #weigth参数表示权值，权
越高被分配到的几率越大
    server 192.168.8.x:80 weight=1;
server 192.168.8.x:80 weight=6;
    }
   #第一个虚拟服务器
   server {
  &nbs
; #侦听192.168.8.x的80端口
       listen&n
sp;   80;
nbsp; server_name 192.168.8.x;
      #对aspx后缀的进行负载均衡请求
    locati
n ~ .*\.aspx$ {

         root   /root;   #定义服器的默认网站根目录位置
        &nbs; index index.php index.html index.htm;   #定义首页索引文件的名称
          proxy_pass http:/mysvr ;#请求转向mysvr 定义的服务器列表
          #以下是一些反向代理的置可删除.
          proxy_redirect off;
          #后端的Web服务器可通过X-Forwarded-For获取用户真实IP
          proxy_set_header Host $host;
          proxy_set_header X-Real-IP $remote_addr;
          proxy_set_header X-Forwarded-For $proxy_add_xforwarded_for;
          client max_body_size 10m;   #允许客户端请求的最大单文件字节数
 &bsp;   client_body_buffer_size 128k; 缓冲区代理缓冲用户端请求的最大字节数，
      &nsp;   proxy_connect_timeout 90; #nginx跟后端服务器连接超时时间(代理连超时)
          proxy_send_timout 90;   #后端服务器数据回传时间(代理发送时)
          proxy_read_timeot 90;   #连接成功后，后端服务器响应间(代理接收超时)
          prox_buffer_size 4k;   #设置代理服务器（nginx）保存用户头信息的缓冲区大小
   &nbp;   proxy_buffers 4 32k;   &nbs;   #proxy_buffers缓冲区，网平均在32k以下的话，这样设置
          proxy_busy_buffers_size 64k;   #高负荷下缓冲大小（proxy_buffers*）
          proxy_temp_file_write_size 64k; #设定缓存文件夹大小，大于这个值，将从upstream服务器传
       }
     }
}
</blockquote>