**Nginx**是一款轻量级的Web服务器，也是一款反向代理服务器。

能干什么？

1. 直接支持Rails和PHP的程序
2. **可作为HTTP反向代理服务器**
3. **作为负载均衡服务器**
4. 作为邮件代理服务器
5. 帮助**实现前端动静分离**

特点：

**高稳定、高性能、资源占用少**、功能丰富、模块化结构、支持热部署

安装步骤：

1. 安装gcc yum install gcc
2. 安装pcre yum install pcre-devel
3. 安装zlib yum install zlib zlib-devel
4. 安装openssl yum install openssl openssl-devel
5. 下载源码包 wget <http://nginx.org/download/nginx-1.10.2.tar.gz>，解压缩
6. 安装，进入nginx目录后，执行**./configure；执行make；执行make install**

常用命令：

测试配置文件 **nginx/sbin/nginx -t**

启动命令 nginx/sbin/**nginx**

停止命令 nginx/sbin/nginx -s stop或者 nginx -s quit

重启命令 nginx/sbin/nginx -s reload

查看进程 ps -ef|grep nginx

**平滑重启** **kill HUP Nginx主进程号**

增加防火墙访问权限，添加如下配置

-A INPUT -p tcp -m state --state NEW

-m tcp --dport **80** -j ACCEPT

配置

nginx/conf/**nginx.conf**文件增加 **include vhost/\*.conf;** 保存退出

在conf文件夹下新建目录vhost，写好配置文件后启动即可访问

1. 配置域名的转发（只展示核心配置，所有配置见文件）

listen 80; 监听的端口

server\_name [www.xxxxxx.com](http://www.xxxxxx.com); 需要转发的域名

location / {

proxy\_pass <http://127.0.0.1:8080>; 转发到哪里

…

}

如果想在Header中保留客户端发起的host，配置

proxy\_set\_header Host $host;

1. 静态资源转发到文件服务器上

location / {

root /ftpfile/;

}

本地模拟需要配置hosts文件，文件的位置

Linux **/etc/hosts**

Windows C:/Windows/System32/drivers/etc/hosts

查看端口号占用情况

netstat -atpn

**Nginx缓存**的作用：对于更通用的功能，在第一个用户请求之后，在代理服务器缓存，其他用户请求时，可以直接从代理服务器获取。

配置方式：

proxy\_cache\_path /var/cache levels=1:2 keys\_zone=my\_cache:10m;

在server的location中配置

proxy\_cache my\_cache;

在Http Header中可以配置代理服务器的缓存超时时间，如下：

‘Cache-Control’:’max-age=5, **s-maxage**=20’

如果不想让代理服务器缓存，设置为private

‘Cache-Control’:’max-age=5, s=maxage=20, private’

如果要添加其他的Header认证，设置如下：

‘Vary’:’User-Agent’ 请求header中的’User-Agent’必须相同才能从缓存获取，区分开不同的浏览器和系统。

Nginx缓存默认写在磁盘上，可以通过配置缓存到内存服务器上增加效率。

**如何配置一个HTTPS服务**

将监听的端口号从80该为443

listen 443;

开启ssl认证

ssl on;

ssl\_certificate\_key 私钥地址

ssl\_certificate 公钥

**如何配置http自动跳转到https**

server {

listen 80 default\_server;

listen [::]:80 default\_server;

server\_name xxx.com;

return 302 https://$server\_name$request\_uri;

}

**如何配置HTTP2**

server {

listen 443 http2;

server\_name xxx.com

http2\_push\_preload on; 配置服务器可推送资源

}