**Nginx文档**

# Nginx是什么

Nginx是一个高性能的HTTP和反向代理服务器，同时也是一个IMAP/POP3/SMTP代理服务器。

## Nginx的优点

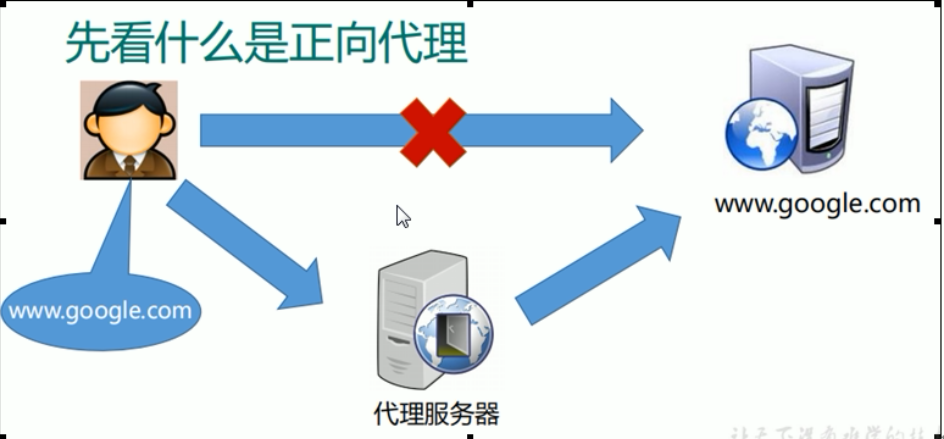
1. Nginx专门为性能而开发,能经受高负载的考验,有报告表名能支持高达50000个并发连接数.
2. Nginx具有很高的稳定性。Nginx官方表示保持10000个没有活动的连接，它只占用2.5M内存.
3. Nginx支持热部署。它的启动特别容易,并且几乎可以做到7\*24小时不间断服务

## Nginx作为服务器的特性

1. Nginx处理静态文件，索引文件以及自动索引;打开文件描述符缓冲.
2. 无缓存的反向代理加速,简单的负载均衡和容错.
3. 支持SSL和TSSNI

# Nginx相关概念

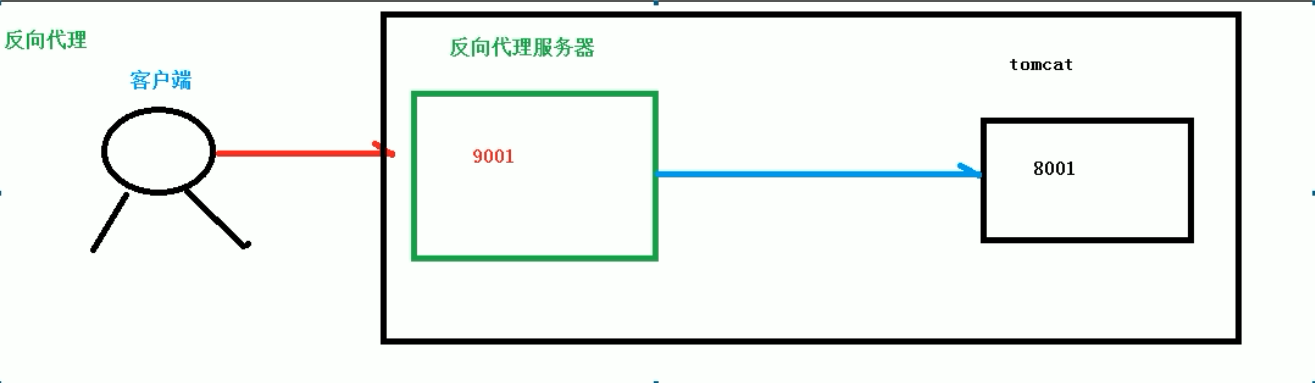
## Nginx正向代理



**在客户端（浏览器）配置代理服务器，通过代理服务器进行服务器访问**

## Nginx反向代理

**反向代理，其实客户端对代理是无感知的，因为客户端不需要任何配置就可以访问，我们只需要将请求发送到反向代理服务器，由反向代理服务器去选择目标服务器获取数据后，在返回给客户端，此时反向代理服务器和目标服务器就是一个服务器，暴露的是代理服务器地址，隐藏了真实服务器IP地址。**



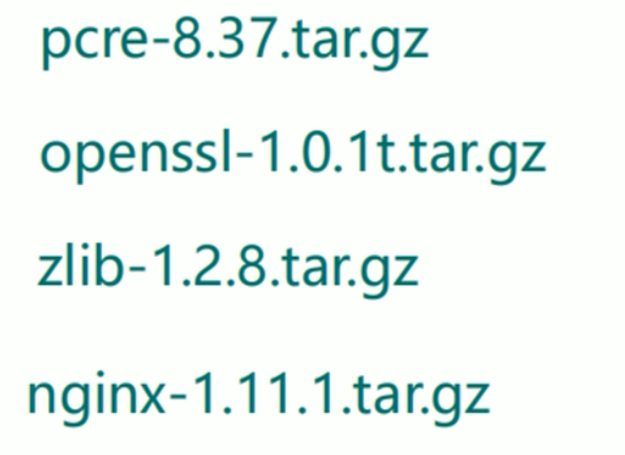
## Nginx负载均衡

**单个服务器解决不了，我们增加服务器的数量，然后将请求分发到各个服务器上，将原先请求集中到单个服务器上的情况改将请求分发到多个服务器上，将负载分发到不同的服务器，也就是我们所说的负载均衡.**

## Nginx动静分离

**为了加载网站的解析速度，可以把动态页面和静态页面由不同的服务器来解析，加快解析速度.降低原来的当服务器的压力.**

# Nginx在Linux中安装



**步骤1. 一键安装其他依赖**

**yum -y install make zlib zlib-devel gcc-c++ libtool openssl openssl-devel**

**步骤2. 安装pcre依赖**

**下载pcre安装包:**

**wget http://downloads.sourceforge.net/project/pcre/pcre/8.37/pcre-8.37.tar.gz**

**或**

<https://ftp.pcre.org/pub/pcre/> **直接下载**

(1). 将安装包放入到linux系统中/usr/目录下

(2). 解压缩安装包  **tar zxvf pcre-8.37.tar.gz**

(3). 进入解压之后的目录，执行./configure命令

(4). 使用make && make install

(5). 安装好之后，查看版本号：**pcre-config --version**

**步骤3. 安装nginx**

**下载安装包**

http://nginx.org/

(1). 将安装包放入到linux系统中/usr/nginx目录下

(2). 解压缩安装包  **tar zxvf nginx-1.12.2.tar.gz**

(3). 进入解压之后的目录，执行./configure命令

(4). 使用make && make install

(5). 到/usr/local/nignx/sbin 启动nginx  **./nginx**

**防火墙设置**

**查看防火墙：**

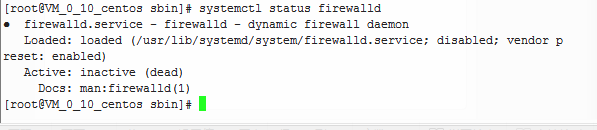
**firewall-cmd --list-all**

**设置开放的端口号:**

**firewall-cmd --add-service=http -permanent**

**firewall-cmd --add-port=80/tcp --permanent**

1.查看firewalld状态:systemctl status firewalld，如果是dead状态，即防火墙未开启。



1. 开启防火墙systemctl start firewalld
2. 确认firewalld状态:systemctl status firewalld
3. 关闭防火墙：systemctl stop firewalld

# Nginx常用命令

首先要进入Nginx命令目录 **cd /usr/local/nginx/sbin**

## 查看版本号

**./nginx -v**



## **启动Nginx**

**./nginx**

## **关闭Nginx**

**./nginx -s stop**

## **重新加载Nginx**

**./nginx -s reload**

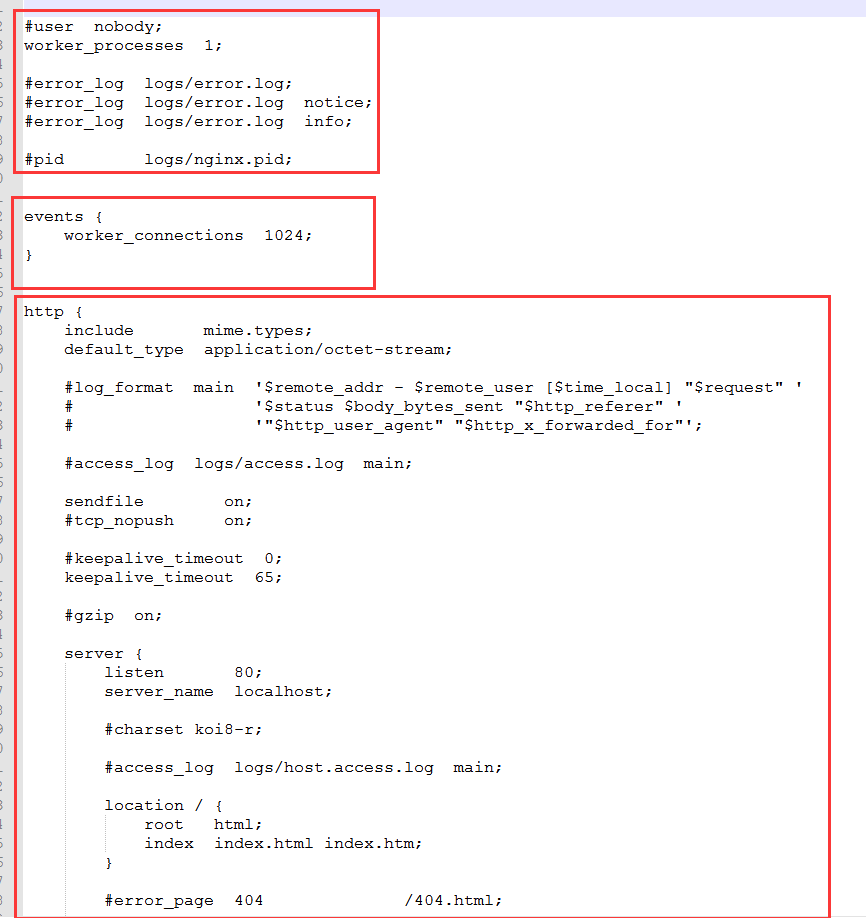
# Nginx的配置文件

**Nginx的配置文件位置：**

**cd /usr/local/nginx/conf**



Nginx文件主要分为三块 **全局块、events块、http块**



## 全局块

从配置文件开始到events块之间的内容，主要会设置一些影响nginx服务器整体运行的配置指令，主要包括配置运行Nginx服务器的用户（组）、允许生成的worker process数，进程PID存放路径、日志存放路径和类型以及配置文件的引入等。

比如：

worker\_process 1;

这是Nginx服务器并发处理服务的关键配置，worker\_process值越大，可以支持的并发处理数量也越多，但是会受到硬件、软件等设备的制约。

## events块

events块涉及的指令主要影响Nginx服务器与用户的网络连接，常用的设置包括是否开启对work process下的网络连接进行序列化，是否运行同事接受多个网络连接，选取哪种事件驱动模型来处理连接请求，每个word process可以同时支持的最大连接数。

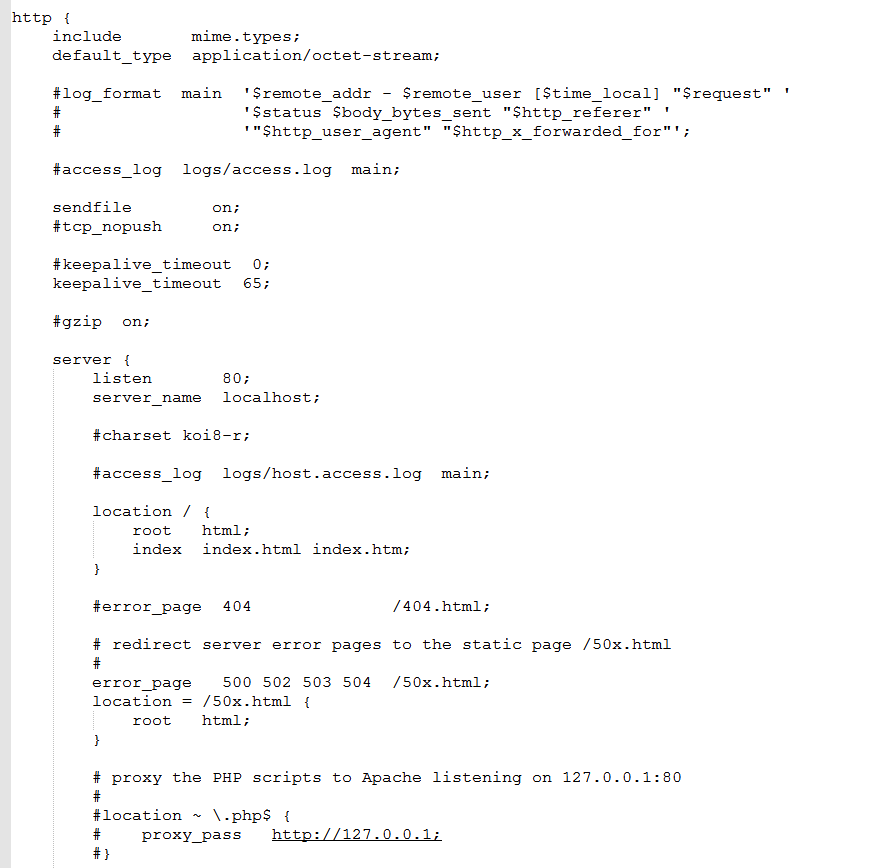
events {

Worker\_connections 1024; ## 最大连接数为1024

}

这部分的配置对Nginx的性能影响较大，在实际中应该灵活配置

## http块



这是Nginx服务器配置中最频繁的部分，代理、缓存和日志定义等绝大多数功能和第三方模块的配置都在这里。

需要注意的是：http块也包括http全局快、server块

### http全局块

Http全局块配置的指令包括：

1. 文件引入
2. MIME-TYPE定义
3. 日志自定义
4. 连接超时时间
5. 单链接请求数上限等

### server块

这块和虚拟主机有密切关系，虚拟主机从用户角度看，和一台独立的硬件主机是完全一样的，该技术的产生是为了节省互联网服务器硬件成本。

每个http块可以包括多个server块，而每个server块就相当于一个虚拟主机

每个server块也分为全局server块，以及可以同时包括多个location块

1. **全局server块**

最常见的配置是本虚拟主机的监听配置和本虚拟主机的名称或IP配置

1. **location块**

**一个server块可以配置多个localtion块。**

**这块主要作用是基于nginx服务器接受到的请求字符串(例如：server\_name/uri-string)对虚拟主机名称(IP)之外的字符串(例如 前面的/uri-string)进行匹配，对特定的请求进行处理。地址定向、数据缓存、应答控制等功能。还有许多第三方模块也在这里配置。**

# Nginx反向代理配置

## 实例一、

进入nginx配置文件