技能目标

- 学会Nginx网站服务的基本构建
- 学会构建Nginx虚拟主机
- 学会部署LNMP网站服务平台

一、关于Nginx

1.1.什么是nginx

> Nginx是一款高性能、轻量级Web服务软件/反向代理服务器及电子邮件（IMAP/POP3）代理服务器。
>
> 由俄罗斯的程序设计师Igor Sysoev所开发，官方测试nginx能够支支撑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。

1.2.应用场景

> 1、http服务器：Nginx是一个http服务可以独立提供http服务。可以做网页静态服务器。
>
> 2、虚拟web主机：可以实现在一台服务器虚拟出多个网站。例如个人网站使用的虚拟主机。
>
> 3、反向代理，负载均衡。当网站的访问量达到一定程度后，单台服务器不能满足用户的请求时，需要用多台服务器集群可以使用nginx做反向代理。并且多台服务器可以平均分担负载，不会因为某台服务器负载高宕机而某台服务器闲置的情况。

1.3.Nginx特点

> 高可靠：稳定性master进程管理调度请求分发到worker;worker进程响应请求 ;一master多worker
>
> 热部署：1.支持长时间运行不重启平滑升级 2.可以快速重载配置
>
> 高并发：单台物理服务器可支持30 000 ~ 50 000个并发请求，(epoll模型内核级)
>
> 响应快：尤其在处理静态文件上，响应速度很快
>
> 低消耗：cpu和内存1w个请求内存平平均使用在2-3MB
>
> 分布式支持：反向代理 七层负载均衡

1.4.Nginx下载

> 官方网址：http://nginx.org/

版本

> Mainline version：Mainline 是 Nginx 目前主力在做的版本，可以说是开发版
>
> Stable version：最新稳定版，生产环境上建议使用的版本
>
> Legacy versions：历史的老版本的稳定版

二、编译安装Nginx

2.1.准备环境

查看系统版本

```
[root@jianfeng ~]# cat /etc/centos-release
CentOS Linux release 7.1.1503
```

禁用防火墙firewalld和selinux

```
[root@jianfeng ~]# systemctl mask firewalld 禁用防火墙
[root@jianfeng ~]# systemctl stop firewalld 停止防火墙
[root@jianfeng ~]# vim /etc/selinux/config
SELINUX=disabled
```

安装编译工具及依赖

```
[root@jianfeng ~]# yum -y install gcc gcc-c++* openssl openssl-devel zlib zlib-
devel pcre pcre-devel

PCRE(Perl Compatible Regular Expressions)是一个Perl库，包括 perl 兼容的正则表达式库。
zlib库提供了很多种压缩和解压缩的方式，nginx使用zlib对http包的内容进行gzip，所以需要在linux上
安装zlib库。
OpenSSL 是一个强大的安全套接字层密码库，包括主要的密码算法、常用的密钥和证书封装管理功能及SSL协
议，并提供丰富的应用程序供测试或其它目的使用。
ncurses(new curses)是一套编程库，它提供了一系列的函数以便使用者调用它们去生成基于文本的用户界
面。

编译时，如出现缺少依赖，一般需要安装以下模块，安装完再次编译：
yum -y install libxml2 libxml2-dev libxslt-devel
yum -y install gd-devel
yum -y install perl-devel perl-ExtUtils-Embed
yum -y install GeoIP GeoIP-devel GeoIP-data
ncurses ncurses-devel
libtool-ltdl-devel libjpeg-devel libpng-devel libtiff-devel fontconfig-devel
freetype-devel libXpm-devel gettext-devel
```

2.2.安装nginx

```
1、创建nginx程序用户
[root@jianfeng ~]# useradd -M -s /sbin/nologin nginx
[root@jianfeng ~]# tar -xzvf nginx-1.20.1.tar.gz -C /usr/src
[root@jianfeng ~]# cd /usr/src/nginx-1.20.1/
2、配置编译安装
[root@jianfeng nginx-1.20.1]# ./configure --prefix=/usr/local/nginx --user=nginx
--group=nginx --with-http_stub_status_module --with-http_ssl_module

--with-http_stub_status_module  //开启状态监控模块
--with-http_ssl_module          //ssl模块支持https等功能

3、编译安装
[root@jianfeng nginx-1.20.1]# make && make install

4、安装目录介绍
[root@jianfeng nginx-1.20.1]# ls /usr/local/nginx/
conf  html  logs  sbin
```

```
其中:
conf -----配置文件
html -----默认网站根文件
logs -----日志文件
sbin -----主要二进制程序nginx主程序
```

2.3.nginx设置及启动

```
1、路径优化（目的是在任意位置执行nginx）
[root@jianfeng ~]# ln -s /usr/local/nginx/sbin/nginx /usr/local/sbin/

2、配置文件语法检查（nginx.conf）
[root@jianfeng ~]# nginx -t

3、启动nginx服务
[root@jianfeng ~]# nginx
[root@jianfeng ~]# netstat -anpt | grep :80
tcp        0      0 0.0.0.0:80                  0.0.0.0:*
LISTEN      39026/nginx

4、重载nginx配置文件
[root@jianfeng ~]# killall -s HUP nginx
[root@jianfeng ~]# nginx -s stop

5、关闭nginx服务
[root@jianfeng ~]# killall -s QUIT nginx
[root@jianfeng ~]# nginx -s reload
```

2.4.nginx服务脚本

```
1、service服务脚本centos6、7通用
[root@jianfeng ~]# vim /etc/init.d/nginx
#!/bin/sh
#chkconfig: - 33 33
#description: Nginx is a high-performance web and proxy server.

PROG="/usr/local/nginx/sbin/nginx"
PIDF="/usr/local/nginx/logs/nginx.pid"
case "$1" in
start)
    $PROG
;;
stop)
    kill -s QUIT $(cat $PIDF) &> /dev/null
;;
restart)
    $0 stop &> /dev/null
    $0 start
;;
reload)
    kill -s HUP $(cat $PIDF)
;;
*)
    echo "Usage: $0 {start|stop|restart|reload}"
    exit 1
```

```
esac
exit 0

-------------或使用下面的内容------------------

#!/bin/bash
# chkconfig: - 85 15
# description: Nginx is a high-performance web and proxy server.
case $1 in
'stop')
        /usr/local/nginx/sbin/nginx -s stop
;;
'start')
        /usr/local/nginx/sbin/nginx
;;
'restart')
        /usr/local/nginx/sbin/nginx -s stop
        /usr/local/nginx/sbin/nginx
;;
'reload')
        /usr/local/nginx/sbin/nginx -s reload
;;
*)
        echo 'Usage:service nginx stop|restart|start|reload'
esac
```

添加执行权限

```
[root@jianfeng ~]# chmod +x /usr/lib/systemd/system/nginx.service
[root@localhost nginx-1.6.0]# chkconfig --add nginx
[root@localhost nginx-1.6.0]# chkconfig nginx on
[root@localhost nginx-1.6.0]# service nginx start
```

2、systemctl服务脚本centos7适用

```
[root@jianfeng ~]# vim /usr/lib/systemd/system/nginx.service
[Unit]
Description=nginx - high performance web server
Documentation=http://nginx.org/en/docs/
After=network.target remote-fs.target nss-lookup.target

[Service]
Type=forking
PIDFile=/usr/local/nginx/logs/nginx.pid
ExecStartPre=/usr/local/nginx/sbin/nginx -t -c /usr/local/nginx/conf/nginx.conf
ExecStart=/usr/local/nginx/sbin/nginx -c /usr/local/nginx/conf/nginx.conf
ExecReload=/bin/kill -s HUP $MAINPID
ExecStop=/bin/kill -s QUIT $MAINPID
PrivateTmp=true

[Install]
WantedBy=multi-user.target
```

添加执行权限

```
[root@jianfeng ~]# chmod +x /usr/lib/systemd/system/nginx.service
[root@jianfeng ~]# systemctl daemon-reload
[root@jianfeng ~]# systemctl start nginx
[root@jianfeng ~]# systemctl enable nginx
```

测试

```
http://localhost
```

2.5.nginx配置文件详解

> 配置文件nginx.conf主要分为三部分
>
> - 全局配置
> - I/O事件配置
> - HTTP配置

```
[root@jianfeng ~]# vim /usr/local/nginx/conf/nginx.conf
#全局配置
user nobody;    #运行用户
worker_processes  1;    #启动进程,通常设置成和cpu的数量相等

#全局错误日志及PID文件，日志输出级别有debug、info、notice、warn、error、crit可供选择
#error_log  logs/error.log;
#error_log  logs/error.log  notice;
#error_log  logs/error.log  info;

#pid        logs/nginx.pid;

#I/O事件配置，工作模式及连接数上限
events {
    #epoll是多路复用IO(I/O Multiplexing)中的一种方式,仅用于linux2.6以上内核,可以大大提高
nginx的性能
    use   epoll;

    #单个后台worker process进程的最大并发链接数
    worker_connections  1024;

    # 并发总数是 worker_processes 和 worker_connections 的乘积
    # 即 max_clients = worker_processes * worker_connections
    # 在设置了反向代理的情况下，max_clients = worker_processes * worker_connections /
4
    # 为什么上面反向代理要除以4，应该说是一个经验值
    # 根据以上条件，正常情况下的Nginx Server可以应付的最大连接数为：4 * 8000 = 32000
    # worker_connections 值的设置跟物理内存大小有关
    # 因为并发受IO约束，max_clients的值须小于系统可以打开的最大文件数
    # 而系统可以打开的最大文件数和内存大小成正比，一般1GB内存的机器上可以打开的文件数大约是10万
左右
    # 我们来看看360M内存的VPS可以打开的文件句柄数是多少：
    # $ cat /proc/sys/fs/file-max
    # 输出 34336
    # 32000 < 34336，即并发连接总数小于系统可以打开的文件句柄总数，这样就在操作系统可以承受的
范围之内
    # 所以，worker_connections 的值需根据 worker_processes 进程数目和系统可以打开的最大
文件总数进行适当地进行设置
    # 使得并发总数小于操作系统可以打开的最大文件数目
    # 其实质也就是根据主机的物理CPU和内存进行配置
    # 当然，理论上的并发总数可能会和实际有所偏差，因为主机还有其他的工作进程需要消耗系统资源。
    # ulimit -SHn 65535
}
#HTTP配置
```

```
http {
    #设定mime类型,类型由mime.type文件定义
    include    mime.types;
    default_type  application/octet-stream;

    #设定日志格式
    log_format  main  '$remote_addr - $remote_user [$time_local] "$request" '
              '$status $body_bytes_sent "$http_referer" '
              '"$http_user_agent" "$http_x_forwarded_for"';

    access_log  logs/access.log  main;

    #sendfile 指令指定 nginx 是否调用 sendfile 函数（zero copy 方式）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.
    sendfile     on;
    #tcp_nopush     on;

    #连接超时时间
    keepalive_timeout  65;
    tcp_nodelay     on;

    #开启gzip压缩
    gzip  on;
    gzip_disable "MSIE [1-6].";

    #设定请求缓冲
    client_header_buffer_size    128k;
    large_client_header_buffers  4 128k;

    #设定虚拟主机配置(一个server模块就是一个web虚拟主机)
    server {
        listen       80;        #侦听80端口
        server_name  localhost; #定义访问域名
        access_log  logs/access.log  main;  #设定本虚拟主机的访问日志


        location / {
                root   html;    #定义服务器的默认网站根目录位置
                index  index.html index.htm;    #定义首页索引文件的名称
        }
    }

    server {
        ......
    }
    ......
}
```

2.6.基于域名的虚拟web主机

```
1、使用hosts文件设置域名
[root@jianfeng ~]# vim /etc/hosts
127.0.0.1   localhost localhost.localdomain localhost4 localhost4.localdomain4
::1         localhost localhost.localdomain localhost6 localhost6.localdomain6
192.168.65.128  www.qingmei.com
```

```
2、修改配置文件，创建基于域名的虚拟web主机
[root@jianfeng ~]# vim /usr/local/nginx/conf/nginx.conf
#更改，无需添加
server {
    listen  80;
    server_name  www.qingmei.com;
    charset utf8;
    #charset koi8-r;

    #access_log  logs/host.access.log  main;
    location / {
        root   /var/www/html;
        index  index.html index.htm;
    }
}

[root@jianfeng ~]# service nginx restart
```

测试

```
http://www.qingmei.com
```

2.7.Nginx的访问状态统计

```
1、修改nginx.conf配置文件
[root@jianfeng ~]# vim /usr/local/nginx/conf/nginx.conf
server {
        listen       80;
        server_name  www.qingmei.com;
        charset utf8;

        #access_log  logs/host.access.log  main;
        location / {
            root   /var/www/html;
            index  index.html index.htm;
        }
        #添加虚拟目录
        location /status{
            stub_status on;     //启用HTTP_STUB_STATUS状态统计模块
            access_log off;     //关闭该目录访问日志
        }
}

浏览器访问http://www.qingmei.com/status
Active connections: 2
server accepts handled requests
2 2 30
Reading: 0 Writing: 1 Waiting: 1    //writing 写入;  waiting 等待
```

小结思考：

- Nginx与Apache相比，具有哪些优势?
- 如何启动、关闭Nginx服务程序?
- 如何修改Nginx服务器的监听地址、网站根目录?

三、lnmp环境搭建

3.1.LNMP

LNMP平台的组成

- Linux、Nginx、MySQL、PHP

工作原理

- nginx的FastCGI模块将Http Server和动态脚本语言分离开
- Nginx专门处理静态请求，转发动态请求
- PHP/PHP_FPM专门解析PHP动态请求

构建步骤

- 安装MySQL数据库
- 安装PHP解析环境
- 启用FPM（FastCGI Process Manager，FastCGI进程管理器）模块
- 配置Nginx支持PHP环境

3.1.二进制方法安装mysql

```
1、安装gcc-c++、make等编译工具
yum -y install  gcc gcc-c++ cmake make

2、安装编译所需要的依赖软件包
[root@jianfeng ~]# yum -y install \
libtool \
zlib zlib-devel \
pcre pcre-devel \
openssl openssl-devel \
ncurses  ncurses-devel \
bison

3、部署mysql
[root@jianfeng ~]# tar -xzvf mysql-5.7.32-el7-x86_64.tar.gz -C /usr/src
[root@jianfeng ~]# mv /usr/src/mysql-5.7.32-el7-x86_64 /usr/local/mysql

4、创建数据库存储目录,mysql用户与用户组，赋予权限
[root@jianfeng ~]# mkdir -p /data
[root@jianfeng ~]# useradd -M -s /sbin/nologin mysql
[root@jianfeng ~]# chown -R mysql:mysql /data
```

手动创建my.cnf文件

```
[root@jianfeng ~]# vim /etc/my.cnf
[client]
socket = /data/mysql.sock
[mysql]
socket = /data/mysql.sock
# 设置mysql客户端默认字符集
default-character-set=utf8
[mysqld]
socket = /data/mysql.sock
#跳过权限表校验
#skip-grant-tables
skip-name-resolve
#设置3306端口
port = 3306
```

```
# 设置mysql的安装目录
basedir=/usr/local/mysql
# 设置mysql数据库的数据的存放目录
datadir=/data/mysql
# 允许最大连接数
max_connections=200
# 服务端使用的字符集默认为8比特编码的latin1字符集
character-set-server=utf8
# 创建新表时将使用的默认存储引擎
default-storage-engine=INNODB
lower_case_table_names=1
max_allowed_packet=16M
```

初始化mysql

```
1、进入安装目录
[root@jianfeng ~]# cd /usr/local/mysql/bin

2、初始化数据库
[root@jianfeng bin]# ./mysqld --initialize-insecure --user=mysql --
basedir=/usr/local/mysql --datadir=/data/mysql

3、生成MySQL秘钥用于ssl连接
[root@jianfeng bin]# ./mysql_ssl_rsa_setup  --datadir=/data/mysql
```

创建mysql服务脚本并启动

```
1、复制mysql服务脚本
[root@jianfeng support-files]# cp /usr/local/mysql/support-files/mysql.server
/etc/init.d/mysqld

2、编辑脚本控制程序：
[root@jianfeng ~]# vim /etc/init.d/mysqld
#修改以下内容：指定程序运行命令和数据库存储目录。
basedir = /usr/local/mysql
datadir = /data/mysql

3、修改控制脚本执行权限
[root@jianfeng ~]#  chmod +x /etc/init.d/mysqld
[root@jianfeng ~]#  chkconfig --add mysqld
[root@jianfeng ~]#  chkconfig mysqld on
[root@jianfeng ~]#  service mysqld start
```

设置root密码

```
1、设置密码
[root@jianfeng ~]# cd /usr/local/mysql/bin
[root@jianfeng bin]# ./mysqladmin -uroot password "123.com"

2、登入数据库创建用户并创建数据库
[root@jianfeng bin]# mysql -uroot -p123.com
mysql> create database wordpress;
mysql> grant all on *.* to "root"@"127.0.0.1" identified by '123.com';
mysql> quit;

3、重启数据库
[root@jianfeng ~]#  service mysqld restart
```

3.2.编译安装php

```
1、安装编工具及依赖
[root@jianfeng ~]#  yum -y install \
libjpeg \
libjpeg-devel \
libpng libpng-devel \
freetype freetype-devel \
libxml2 \
libxml2-devel \
zlib zlib-devel \
curl curl-devel \
openssl openssl-devel

2、安装php
[root@jianfeng ~]# tar -xzvf php-7.3.9.tar.gz -C /usr/src
[root@jianfeng ~]# cd /usr/src/php-7.3.9/
[root@jianfeng php-7.3.9]# ./configure \
--prefix=/usr/local/php \
--with-mysql-sock=/data/mysql.sock \
--with-mysqli \
--with-zlib \
--with-curl \
--with-gd \
--with-jpeg-dir \
--with-png-dir \
--with-freetype-dir \
--with-openssl \
--enable-fpm \
--enable-mbstring \
--enable-xml \
--enable-session \
--enable-ftp \
--enable-pdo \
--enable-tokenizer


--enable-fpm    #选项必须添加

[root@jianfeng php-7.3.9]# make && make install
```

配置php-fpm

> 使用php-fpm进行管理php服务，有两个配置文件：
>
> ①php.ini　　#默认php配置文件
>
> ②php-fpm.conf　　#php-fpm相关的配置

启用php-fpm进程

> **默认监听本机的9000端口**

```
1、复制配置文件
[root@jianfeng ~]# cp /usr/local/php/etc/php-fpm.conf.default
/usr/local/php/etc/php-fpm.conf
[root@jianfeng ~]# cp /usr/local/php/etc/php-fpm.d/www.conf.default
/usr/local/php/etc/php-fpm.d/www.conf
[root@jianfeng ~]# cp /usr/src/php-7.3.9/php.ini-development
/usr/local/php/php.ini

2、添加启动服务
[root@jianfeng ~]# cp /usr/src/php-7.3.9/sapi/fpm/init.d.php-fpm /etc/init.d/php-
fpm
[root@jianfeng ~]# chmod +x /etc/init.d/php-fpm
[root@jianfeng ~]# chkconfig --add php-fpm

3、添加环境变量（方便php、phpize、phpconfig查找使用）
[root@jianfeng ~]# echo 'PATH=/usr/local/php/bin:$PATH' >> /etc/profile
[root@jianfeng ~]# source /etc/profile

4、配置php.ini文件
[root@jianfeng ~]# vim /usr/local/php/php.ini
date.timezone = PRC    //设置时区
mysqli.default_socket = /data/mysql.sock        //将php与mysql关联

5、启动php-fpm
[root@jianfeng ~]# service php-fpm start

[root@localhost etc]# netstat -anpt | grep php-fpm      #端口为9000
tcp        0      0 127.0.0.1:9000              0.0.0.0:*
LISTEN      46733/php-fpm
```

3.3.Nginx+php-fpm配置

```
[root@jianfeng ~]# vim /usr/local/nginx/conf/nginx.conf
server {
    listen       80;
    server_name  www.qingmei.com;
    root /var/www/abc;      #把root变量站点根目录定义提升上层
    location / {
        #root   /var/www/qingmei;       #注释掉，默认使用上层的root变量
        index  index.php ndex.html ;    #添加默认索引页
    }
    #添加以下定义如何处理php文件
    location ~ \.php$ {
        fastcgi_pass 127.0.0.1:9000;    #Nginx通过本机的9000端口将PHP请求转发给PHP-
FPM进行处理。
        fastcgi_index  index.php;       #添加,php首页文件
        include fastcgi.conf;           #Nginx调用fastcgi接口处理PHP请求。
    }
```

```
}

[root@jianfeng ~]# service nginx restart
```

测试

```
1、测试nginx+php
[root@jianfeng ~]# vim /var/www/html/test.php
<?php
phpinfo();
?>

访问：http://www.qingmei.com/test.php


2、测试nginx+php+mysql
[root@jianfeng ~]# vim /var/www/html/testdb.php
<?php
$link=mysqli_connect('127.0.0.1','root','123.com');
if($link) echo "<h1>Success!!</h1>";
else echo "Fail!!";
?>

访问：http://www.qingmei.com/testdb.php
```

3.4.安装wordpress

```
1、部署wordpress文件到站点目录
[root@jianfeng ~]# unzip wordpress-5.6-zh_CN.zip
[root@jianfeng ~]# mv ./wordpress/* /var/www/html/

2、将wp-config-sample.php文件，重命名为：wp-config.php。打开文件，填进相关数据库信息。
[root@jianfeng ~]# cd /var/www/html
[root@jianfeng html]# cp wp-config-sample.php wp-config.php
[root@jianfeng html]# vim wp-config.php
.......
/** WordPress数据库的名称 */
define( 'DB_NAME', 'wordpress' );

/** MySQL数据库用户名 */
define( 'DB_USER', 'root' );

/** MySQL数据库密码 */
define( 'DB_PASSWORD', '123.com' );

/** MySQL主机 */
define( 'DB_HOST', '127.0.0.1' );
........
```

打开浏览器，输入域名，即可弹出安装界面。填上相关网站信息，点击”安装Wordpress”按钮即可完成。





小结思考

- 如何在编译时使得Nginx支持状态统计?
- 如何加大Nginx的工作进程数量?
- 使得Nginx支持PHP解析的方法有哪些?

实验需求描述

- 构建Nginx+MySQL+PHP平台
- 在本机配置named服务，提供DNS解析环境
- 实现5个虚拟Web主机：www、mail、ftp、blog、vod
- 为其中的vod站点部署“wordpress博客系统”

实现思路

- 准备DNS解析环境
- 依次安装MySQL、PHP、Nginx
- 配置虚拟Web主机服务，并部署PHP应用系统