1:安装工具包 wget、vim和gcc

yum install -y wget

yum install -y vim-enhanced

yum install -y make cmake gcc gcc-c++

1

2

3

2:下载nginx安装包

wget http://nginx.org/download/nginx-1.6.2.tar.gz

1

3:安装依赖包

yum install -y pcre pcre-devel

yum install -y zlib zlib-devel

yum install -y openssl openssl-devel

1

2

3

4:解压nginx-1.6.2.tar.gz到/usr/local/目录下

tar -zxvf nginx-1.6.2.tar.gz -C /usr/local/

1

5:进行configure配置

进入nginx-1.6.2目录然后在执行./configure命令

[root@MiWiFi-R3-srv nginx-1.6.2]# ./configure --prefix=/usr/local/nginx

1

6:编译安装

[root@MiWiFi-R3-srv nginx-1.6.2]# make && make install

1

7:启动Nginx，启动完之后检查nginx是否已经正常启动，看到如下信息说明正常启动

[root@MiWiFi-R3-srv nginx-1.6.2]# /usr/local/nginx/sbin/nginx

[root@MiWiFi-R3-srv nginx-1.6.2]# ps -ef | grep nginx

root 24956 1 0 19:41 ? 00:00:00 nginx: master process /usr/local/nginx/sbin/nginx

nobody 24957 24956 0 19:41 ? 00:00:00 nginx: worker process

root 24959 10533 0 19:41 pts/0 00:00:00 grep --color=auto nginx

[root@MiWiFi-R3-srv nginx-1.6.2]#

1

2

3

4

5

6

如果要关闭nginx，我们可以使用如下命令：

[root@MiWiFi-R3-srv nginx-1.6.2]# /usr/local/nginx/sbin/nginx -s stop

1

如果想要重新热启动nginx，则使用如下命令：

[root@MiWiFi-R3-srv nginx-1.6.2]# /usr/local/nginx/sbin/nginx -s reload

1

8:配置防火墙,nginx默认的端口是80

[root@MiWiFi-R3-srv nginx-1.6.2]# firewall-cmd --zone=public --add-port=80/tcp --permanent

success

[root@MiWiFi-R3-srv nginx-1.6.2]# firewall-cmd --reload

success

[root@MiWiFi-R3-srv nginx-1.6.2]#

1

2

3

4

5

9:测试nginx

通过浏览器访问nginx欢迎页，在地址栏输入：http://192.168.31.241/（80端口可以不用输）或http://192.168.156.11:80/，如下图所示。

这里写图片描述

10:学习nginx配置

在nginx目录下进入conf目录，该目录下有个nginx.conf文件，这是nginx最重要的配置文件

[root@MiWiFi-R3-srv conf]# vim /usr/local/nginx/conf/nginx.conf

1

nginx.conf文件的全部内容如下(有注释版):

#user nobody;

#开启进程数 <=CPU数

worker\_processes 1;

#错误日志保存位置

#error\_log logs/error.log;

#error\_log logs/error.log notice;

#error\_log logs/error.log info;

#进程号保存文件

#pid logs/nginx.pid;

#每个进程最大连接数（最大连接=连接数x进程数）每个worker允许同时产生多少个链接，默认1024

events {

worker\_connections 1024;

}

http {

#文件扩展名与文件类型映射表

include mime.types;

#默认文件类型

default\_type application/octet-stream;

#日志文件输出格式 这个位置相于全局设置

#log\_format main '$remote\_addr - $remote\_user [$time\_local] "$request" '

# '$status $body\_bytes\_sent "$http\_referer" '

# '"$http\_user\_agent" "$http\_x\_forwarded\_for"';

#请求日志保存位置

#access\_log logs/access.log main;

#打开发送文件

sendfile on;

#tcp\_nopush on;

#keepalive\_timeout 0;

#连接超时时间

keepalive\_timeout 65;

#打开gzip压缩

#gzip on;

server {

#监听端口，默认是80端口

listen 80;

#监听域名

server\_name localhost;

#charset koi8-r;

#nginx访问日志放在logs/host.access.log下，并且使用main格式（还可以自定义格式）

#access\_log logs/host.access.log main;

#如果没有location更明确的匹配访问路径的话，访问请求都会被该location处理。

location / {

#root指定nginx的根目录为/usr/local/nginx/html

root html;

#默认访问文件，欢迎页先去html目录下找index.html，如果找不到再去找index.htm

index index.html index.htm;

}

#error\_page 404 /404.html;

# redirect server error pages to the static page /50x.html

#

#错误页面及其返回地址，错误码为500、502、503、504都会返回50.html错误页面。

error\_page 500 502 503 504 /50x.html;

#location后面是"="的话，说明是精确匹配

location = /50x.html {

root html;

}

# proxy the PHP scripts to Apache listening on 127.0.0.1:80

#

#location ~ \.php$ {

# proxy\_pass http://127.0.0.1;

#}

# pass the PHP scripts to FastCGI server listening on 127.0.0.1:9000

#

#location ~ \.php$ {

# root html;

# fastcgi\_pass 127.0.0.1:9000;

# fastcgi\_index index.php;

# fastcgi\_param SCRIPT\_FILENAME /scripts$fastcgi\_script\_name;

# include fastcgi\_params;

#}

# deny access to .htaccess files, if Apache's document root

# concurs with nginx's one

#

#location ~ /\.ht {

# deny all;

#}

}

# another virtual host using mix of IP-, name-, and port-based configuration

#

#server {

# listen 8000;

# listen somename:8080;

# server\_name somename alias another.alias;

# location / {

# root html;

# index index.html index.htm;

# }

#}

# HTTPS server

#

#server {

# listen 443 ssl;

# server\_name localhost;

# ssl\_certificate cert.pem;

# ssl\_certificate\_key cert.key;

# ssl\_session\_cache shared:SSL:1m;

# ssl\_session\_timeout 5m;

# ssl\_ciphers HIGH:!aNULL:!MD5;

# ssl\_prefer\_server\_ciphers on;

# location / {

# root html;

# index index.html index.htm;

# }

#}

}

---------------------

作者：PengTDY

来源：CSDN

原文：https://blog.csdn.net/yougoule/article/details/78186138

版权声明：本文为博主原创文章，转载请附上博文链接！