1. 安装 gcc 环境

$ sudo yum -y install gcc gcc-c++

2. 安装 pcre

$ sudo yum -y install pcre pcre-devel *# 让nginx支持重写功能*

3.安装 zlib

1. *# zlib库提供了很多压缩和解压缩的方式，nginx使用zlib对http包内容进行gzip压缩*
2. $ sudo yum -y install zlib zlib-devel

4. 安装 openssl

1. *# 安全套接字层密码库，用于通信加密*
2. $ sudo yum -y install openssl openssl-devel

以上安装完成后，进行nginx安装。

**nginx 源码包安装**

将准备好的 nginx-1.17.8.tar.gz包，拷贝至/usr/local/nginx目录下（一般习惯在此目录下进行安装）进行解压缩。  
源码包下载地址：<https://nginx.org/en/download.html>

$ sudo tar -zxvf nginx-1.17.8.tar.gz *# 解压缩*

在完成解压缩后，进入nginx-1.17.8目录进行源码编译安装。

1. $ cd nginx-1.17.8
2. $ ./configure --prefix=/usr/local/nginx *# 检查平台安装环境*
3. --prefix=/usr/local/nginx 是nginx编译安装的目录（推荐），安装完后会在此目录下生成相关文件

#如果将来需要ssl证书安装选择下面的命令

./configure --prefix=/usr/local/nginx --with-http\_ssl\_module

如果前面的依赖库都安装成功后，执行./configure --prefix=/usr/local/nginx命令会显示一些环境信息。如果出现错误，一般是依赖库没有安装完成，可按照错误提示信息进行所缺的依赖库安装。  
进行源码编译并安装nginx

1. $ make *# 编译*
2. $ make install *# 安装*

源码包安装与yum安装的nginx服务操作命令也不同。

* 启动服务

$ /usr/local/nginx/sbin/nginx

* 重新加载服务

$ /usr/local/nginx/sbin/nginx -s reload

* 停止服务

$ /usr/local/nginx/sbin/nginx -s stop

# [设置nginx开机自启动](https://www.cnblogs.com/bkwxx/p/10717891.html)

**第一步：进入到/lib/systemd/system/目录**

[root@iz2z init.d]# cd /lib/systemd/system/

**第二步：创建nginx.service文件，并编辑**

# vim nginx.service

[Unit]

Description=nginx service

After=network.target

[Service]

Type=forking

ExecStart=/usr/local/nginx/sbin/nginx

ExecReload=/usr/local/nginx/sbin/nginx -s reload

ExecStop=/usr/local/nginx/sbin/nginx -s quit

PrivateTmp=true

[Install]

WantedBy=multi-user.target

systemctl enable nginx.service

设置/usr/local/nginx/nginx.conf

参考配置

user root root;

worker\_processes 4;

error\_log /data/log/nginx/error.log warn;

pid /run/nginx.pid;

events {

worker\_connections 1024;

}

http {

include mime.types;

default\_type application/octet-stream;

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 /data/log/nginx/access.log main;

server\_tokens off;

sendfile on;

#tcp\_nopush on;

keepalive\_timeout 65;

client\_max\_body\_size 2000m;

server\_names\_hash\_bucket\_size 2560;

fastcgi\_connect\_timeout 300s;

fastcgi\_send\_timeout 300s;

fastcgi\_read\_timeout 300s;

fastcgi\_buffer\_size 128k;

fastcgi\_buffers 8 256k;

#tcp\_nopush on;

#keepalive\_timeout 0;

send\_timeout 120;

proxy\_connect\_timeout 360;

proxy\_send\_timeout 360;

proxy\_read\_timeout 360;

#gzip on;

#add\_header X-Frame-Options SAMEORIGIN

gzip on;

gzip\_min\_length 1k;

gzip\_buffers 4 16k;

#gzip\_http\_version 1.0;

gzip\_comp\_level 2;

gzip\_types text/plain application/x-javascript application/javascript text/css application/xml application/json text/javascript application/x-httpd-php image/jpeg image/gif image/png image/jpg;

gzip\_vary off;

gzip\_disable "MSIE [1-6]\.";

server {

listen 80 default\_server;

server\_name \_;

return 404;

}

upstream tomcatups {

server 192.168.3.10:10001 weight=1;

server 192.168.3.11:10001 weight=1;

server 192.168.3.12:10001 weight=1;

}

include /usr/local/nginx/conf/vhost/\*.conf;

}

#cd vhost

#vi web.conf web程序服务

创建代理server

server {

listen 8010;

server\_name 39.130.181.3;

location / {

index index.html index.jhtml;

proxy\_pass http://127.0.0.1:10001;

proxy\_set\_header X-Real-IP $remote\_addr;

proxy\_set\_header X-Forwarded-For $proxy\_add\_x\_forwarded\_for;

proxy\_set\_header Host $http\_host;

}

}

192.168.3.11 配置资源服务

（add\_header Access-Control-Allow-Origin \*;）加上这个防止遇到跨域问题

#vi web.conf

server {

listen 8010;

server\_name 39.130.181.4;

add\_header Access-Control-Allow-Origin \*;

location / {

root /home/upload/;

index index.html index.htm;

}

}

以上红色标注不可删除，否则站群访问的时候访问站点可能错误，找到对应的模板是错误的

参考示例 （结合静态分离配置和缓存配置）

server {

listen 80;

server\_name demo.jeecms.com gov.jeecms.com;

#静态分离 可以在tomcat挂掉的时候前台html页面能提供访问

location ~/(u|r|thirdparty|h5center|resource/form)/{

#配置静态资源目录，如果集群下u p1 m1独立文件夹则分开配置

root /data/tomcat/webapps/ROOT;

}

location = / {

root /data/tomcat/webapps/ROOT/p1;

}

location ~/p1/{

root /data/tomcat/webapps/ROOT/;

}

#重定向手机端静态页到pc静态页 手机端模板没有做的情况

location /m1/ {

rewrite "^/m1/(.\*)$" /p1/$1 last;

}

location ~^(.\*\.html)$ {

add\_header Access-Control-Allow-Origin \*;

root /data/tomcat/webapps/ROOT/;

}

location / {

index index.html index.jhtml;

proxy\_pass http://127.0.0.1:10015;

proxy\_set\_header X-Real-IP $remote\_addr;

proxy\_set\_header X-Forwarded-For $proxy\_add\_x\_forwarded\_for;

proxy\_set\_header Host $http\_host;

#静态资源缓存配置

if ($request\_filename ~\* .\*\.(?:js|css)$)

{

expires 30d;

}

if ($request\_filename ~\* .\*\.(?:jpg|jpeg|gif|png|ico|cur|gz|svg|svgz|mp4|ogg|ogv|webm)$)

{

expires 30d;

}

proxy\_read\_timeout 30; # 秒

}

access\_log /data/log/nginx/access\_demo.log main;

}

示例配置文件，目录和端口文件需要调整自己的



* 常见问题 大文件上传报504

#修改Nginx配置：

fastcgi\_connect\_timeout 1200s;#原设置为300s

fastcgi\_send\_timeout 1200s;#原设置为300s

fastcgi\_read\_timeout 1200s;#原设置为300s

1024000 worker\_connections exceed open file resource limit: 65535

解决办法

<https://www.jianshu.com/p/d870ce724dc1>

Nginx 杀掉全部

killall -9 nginx