1. gcc 安装

|  |
| --- |
| yum install gcc-c++ |

1. PCRE pcre-devel 安装

|  |
| --- |
| yum install -y pcre pcre-devel |

1. zlib 安装

|  |
| --- |
| yum install -y zlib zlib-devel |

1. openssl 安装

|  |
| --- |
| yum install -y openssl openssl-devel |

1. 安装nginx， 官网地址：<https://nginx.org/en/download.html>

可以手动去官网下载对应的版本，然后拷贝到linux系统中

也可以使用 wget 下载

|  |
| --- |
| wget -c https://nginx.org/download/nginx-1.14.0.tar.gz |

1. 下载完成后，进行解压：

|  |
| --- |
| tar -zxvf nginx-1.13.7.tar.gz |

1. 进入解压后的目录

|  |
| --- |
| cd nginx-1.13.7 |

1. 应用默认配置，执行

|  |
| --- |
| ./configure |

1. 执行编译：

|  |
| --- |
| make |

1. 执行安装：

|  |
| --- |
| make install |

1. 查看安装路径：

|  |
| --- |
| whereis nginx |

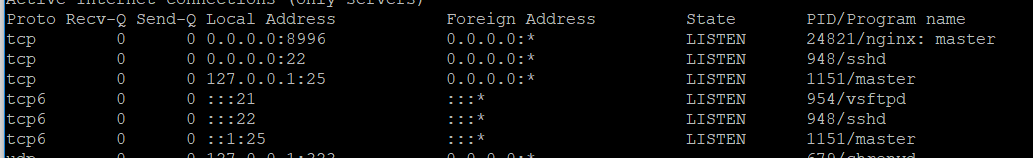
路径默认在： /usr/local/nginx

1. 常用命令：

|  |
| --- |
| # 首先进行nginx安装目录下的 sbin 目录  cd /usr/local/nginx/sbin  # 启动nginx  ./nginx  # 停止nginx：  # 此方式停止步骤是待nginx进程处理任务完毕进行停止，推荐这个  ./nginx -s quit  # 此方式相当于先查出nginx进程id再使用kill命令强制杀掉进程  ./nginx -s stop  # 查看nginx进程  ps aux|grep nginx  # 修改了nginx的配置文件，如果需要重新加载配置文件，使用：  ./nginx -s reload |

1. 启动nginx后，可以查看端口监听情况

|  |
| --- |
| netstat -nlp |



可以得知， nginx 监听的是 8996

1. 关闭linux防火墙或者开放8996端口，在浏览器中访问，即可查看到对应的欢迎页面



1. 设置nginx为开机启动

|  |
| --- |
| 1、编辑文件：  vi /etc/rc.local  2、增加一行  /usr/local/nginx/sbin/nginx  3、设置权限  chmod 755 /etc/rc.local |

1. nginx配置

配置文件位置：在 /usr/local/nginx/conf/nginx.conf

|  |
| --- |
| ########### 每个指令必须有分号结束。#################  #user administrator administrators; #配置用户或者组，默认为nobody nobody。  #worker\_processes 2; #允许生成的进程数，默认为1  #pid /nginx/pid/nginx.pid; #指定nginx进程运行文件存放地址  error\_log log/error.log debug; #制定日志路径，级别。这个设置可以放入全局块，http块，server块，级别以此为：debug|info|notice|warn|error|crit|alert|emerg  events {  accept\_mutex on; #设置网路连接序列化，防止惊群现象发生，默认为on  multi\_accept on; #设置一个进程是否同时接受多个网络连接，默认为off  #use epoll; #事件驱动模型，select|poll|kqueue|epoll|resig|/dev/poll|eventport  worker\_connections 1024; #最大连接数，默认为512  }  http {  include mime.types; #文件扩展名与文件类型映射表  default\_type application/octet-stream; #默认文件类型，默认为text/plain  #access\_log off; #取消服务日志  log\_format myFormat '$remote\_addr–$remote\_user [$time\_local] $request $status $body\_bytes\_sent $http\_referer $http\_user\_agent $http\_x\_forwarded\_for'; #自定义格式  access\_log log/access.log myFormat; #combined为日志格式的默认值  sendfile on; #允许sendfile方式传输文件，默认为off，可以在http块，server块，location块。  sendfile\_max\_chunk 100k; #每个进程每次调用传输数量不能大于设定的值，默认为0，即不设上限。  keepalive\_timeout 65; #连接超时时间，默认为75s，可以在http，server，location块设置。  server {  listen 8996; # 端口  server\_name localhost; # 虚拟主机名，发布到互联网上写 域名，如: \*.aura.com  charset utf-8; # 默认编码  location / {  include uwsgi\_params; # 包含 uwsgi配置，这个是相对路径，也可以使用绝对路径  uwsgi\_pass 127.0.0.1:8997; # uwsgi的端口  uwsgi\_send\_timeout 1060; # 指定连接到后端uWSGI的超时时间>。  uwsgi\_connect\_timeout 1060; # 指定向uWSGI传送请求的超时时间，>完成握手后向uWSGI传送请求的超时时间。  uwsgi\_read\_timeout 1060; # 指定接收uWSGI应答的超时时间，>完成握手后接收uWSGI应答的超时时间。  }  location /static/ {  alias /var/www/childrenshop/static/;  }  location /media/ {  alias /var/www/childrenshop/media/;  }  #error\_page 404 /404.html;  # redirect server error pages to the static page /50x.html  #  error\_page 500 502 503 504 /50x.html;  location = /50x.html {  root html;  }  }  } |