根据操作系统的不同，Nginx可以安装也不太相同，安装Nginx有以下几种不同的方式。

**在Linux上安装**

对于Linux安装首先要确认是否安装 gcc g++开发库，默认已经装好。

可以在ububtu平台编译环境下可以使用以下指令

apt-get install build-essential

apt-get install libtool

centos平台编译环境使用如下指令

安装make

Yum -y install gcc outomake outoconf libtool make

安装g++:

Yum install gcc gcc-c++

下面正式开始

一般我们都需要先安装pcre,zlib,前者为了重写rewrite,后者为了gzip压缩。

1. 选定源码目录

可以是任何目录，本文选定的是/usr/local/src

cd /usr/local/src

1. 安装PCRE库

<ftp://ftp.csx.cam.ac.uk/pub/software/programming/pcre/>下载最新的PCRE源码包，使用下面命令下载编译和安装PCRE包：

cd /usr/local/src

wget <ftp://ftp.csx.com.oc.uk/pub/software/propgramming/pcre/pcre-8.37.tar.gz>

tar -zxvf pcre-8.37.tar.gz

cd pcre-8.34

./configure

make

make install

1. 安装zlib库

<http://zlib.net/zlib-1.2.8.tar.gz> 下载最新的zlib源码包，使用下面命令下载编译和安装zlib包：

cd /usr/local/src

wget <http://zlib.net/zlibe-1.2.8.tar.gz>

cd zlib-1.2.8

./configure

make

make install

1. 安装ssl（某些vps默认没装ssl）

cd /usr/local/src

wget <https://www.openssl.org/source/openssl-1.0.1t.tar.gz>

tar -zxvf openssl-1.0.1t.tar.gz

1. 安装nginx

Nginx一般有两个版本，分别是稳定版和开发版，您可以根据您的目的来选择这两个版本的其中一个，下面是把Nginx安装/usr/local/nginx目录下的详细步骤：

cd /usr/local/src

wget <http://nginx.org/download/nginx-1.4.2.tar.gz>

tar -zxvf nginx-1.4.2.tar.gz

cd nginx-1.4.2

./configure –sbin-path=/usr/local/nginx/niginx \

--conf-path=/usr/local/nginx/nginx.conf \

--pid-path=/usr/local/nginx/nginx.pid \

--with-http\_ssl\_module \

--with-pcre=/opt/app/openet/oetall/chenhe/pcre-8.37 \

--with-zlib=/opt/app/openet/oetall/chenhe/zlib-1.2.8 \

--with-openssl=/opt/app/openet/oetall/chenhe/openssl-1.0.1t

make

make install

--with-pcre=/usr/src/pcre-8.34 指的是pcre-8.34 的源码路径。

--with-zlib=/usr/src/zlib-1.2.7 指的zlib-1.2.7 的源码路径。

安装成功后/usr/local/nginx 目录下如下

fastcgi.conf koi-win nginx.conf.default

fastcgi.conf.default logs scgi\_params

fastcgi\_params mime.types scgi\_params.default

fastcgi\_params.default mime.types.default uwsgi\_params

html nginx uwsgi\_params.default

koi\_utf nginx.conf win-utf

1. 启动

确保系统的80端口没被其他程序占用，运行/usr/local/nginx/nginx命令来启动Nginx.

netstat -ano|grep 80

如果查不到结果执行，有结果则忽略此步骤(ubuntu下必须用sudo启动)，不然只能在前台运行）

sudo /usr/local//nginx/nginx

打开浏览器访问此机器的IP，如果浏览器出现Welcome to nginx!则表示Nginx已经安装并运行成功。

到这里nginx就安装完成了，如果只是处理静态html就不用继续安装了。

1. nginx重启、关闭、启动

启动

启动代码格式：nginx安装目录地址 -c nginx配置文件地址

例如：

[root@LinuxServer sbin] # /usr/local/nginx/sbin/nginx -c /usr/local/nginx/conf/nginx.conf

停止

Nginx的停止有三种方式：

从容停止

1.[root@LinuxServer ~ ]# ps -ef | grep nginx

杀死进程

[root@LinuxServer ~ ] # kill -QUIT 2072

快速停止

查看进行号

[root@LinuxServer ~ ] # ps -ef | grep nginx

杀死进程

[root@LinuxServer ~ ] # kill – TERM 2132

或

[root@LinuxServer ~ ] # kill -INT 2132

强制停止

[root@LinuxServer ~ ] # pkill -9 nginx

重启

1. 验证nginx配置文件是否正确

方法一：进入nginx安装目录sbin下，输入命令。/nginx-t

看到如下显示nginx.conf syntax is ok

nginx.conf test is successful

说明配置文件正确！

1. 重启Nginx服务

方法一：进入nginx可执行目录sbin下，输入命令。/nginx -s reload 即可

方法二：查找当前nginx进程号，然后输入命令：kill-HUP 进程号 实现重启nginx服务

9、项目存放文件

Nginx/html

安装排错

* 1. 编译pcre错误

libtool: compile: unrecognized option ‘-DHAVE\_CONFIG\_H’

libtool: compile: Try ‘libtool – help’ for more information.

make[1]: \*\*\* [pcrecpp.lo] Error 1

make[1]: Leaving directory ‘/usr/local/src/pcre-8.34’

make: \*\*\* [all] Error 2

解决办法：安装g++,别忘了重新configure

opt-get install g++

opt-get install build-essential

make clean

./configure

Make

* 1. make出错

make: \*\*\* No rule to make target ‘build’, needed by ‘default’. Stop.

./configure: error: SSL modules require the OpenSSL library.

You can either do not enable the modules, or install the Open SSL library

into the system, or build the OpenSSL library statically from the source

with nginx by using –with-openssl=<path> option.

按照第4步的安装方法或

ubuntu下

opt-get install openssl

opt-get install libssl-dev

centos下

yum -y install openssl openssl-devel

1. nginx编译选项

make是用来编译的，它从Makefile中读取指令，然后编译。

make install是用来安装，它也从Makefile中读取指令，安装到指定的位置

configure命令是用来检测你的安装平台的目标特征的。它定义了系统的各个方面，包括nginx的被允许使用的连接处理的方法，比如它会检测你是不是有CC或GCC，并不是需要CC或GCC，它是个shell脚本，执行结束时，它会创建一个Makefile文件，nginx的configure命令支持以下参数。

--prefix=path 定义一个目录，存放服务器上的文件，也就是nginx的安装目录。默认使用/usr/local/nginx 。

--sbin-path=path 设置nginx的可执行文件的路径，默认为 prefix/sbin/nginx 。

--config-path=path 设置在nginx.conf配置文件的路径。Nginx允许使用不同的配置文件启动，通过命令行中的-c选项。默认为prefix/conf/nginx.conf。

--pid-path=path 设置nginx.pid文件，将存储的主进程的进程号。安装完成后，可以随时改变的文件名，在nginx.conf配置文件中使用PID指令。默认情况下，文件名为 prefix/logs/nginx.pid。

--error-log-path=path设置主错误，警告，和诊断文件的名称。安装完成后，可以随时改变的文件名，在nginx.conf配置文件中使用的error\_log指令。默认情况下，文件名为prefix/logs/error.log。

--http-log-path=path 设置主请求的HTTP服务器的日志文件的名称。安装完成后，可以随时改变的文件名，在nginx.conf配置文件中使用的access\_log指令。默认情况下，文件名为prefix/logs/access.log。

--user=name 设置nginx工作进程的用户。安装完成后，可以随时更改的名称nginx.conf配置文件中使用的user指令。默认的用户名是nobody。

--group=name设置nginx工作进程的用户组。安装完成后，可以随时更改的名称在nginx.conf配置文件中使用的user指令。默认的为非特权用户。

--with-select\_module –without-select\_module 启用或禁用构建一个模块来允许服务器使用select()方法。该模块将自动建立，如果平台不支持的kqueue，epoll，rtsig或/dev/poll。

--without-http\_gaip\_module - 不编译压缩的HTTP服务器的响应模块。编译并运行此模块需要zlib库。

--without-http\_rewrite\_module 不编译重写模块。编译并运行此模块需要PCRE库支持。

--without-http\_proxy\_module – 不编译http\_proxy模块。

--with-http\_ssl\_module -使用https协议模块。默认情况下，该模块没有被构建。建立并运行此模块的OpenSSL库是必需的。

--with-pcre=path – 设置PCRE库的源码路径。PCRE库的源码（版本4.4 – 8.30）需要从PCRE网站下载并解压。其余的工作是Nginx的./ configure和make来完成。正则表达式使用在location指令的ngx\_http\_rewrite\_module模块中。

--with-pcre-jit -编译PCRE包含“just-In-time compilation“（1.1.12中，pcre\_jit指令）。

--with-zlib=path -设置的zlib库的源码路径。要下载从zlib（版本1.1.3-1.2.5）的并解压。其余的工作是Nginx的./configure和make完成。ngx\_http\_gzip\_module模块需要使用zlib.。

--with-ld-opt=paramsters – 设置附加的参数，将用于在链接期间。例如，当在FreeBSD下使用该系统的PCRE库，应指定：--with-opt=”-L /usr/local/lib”.

典型实例（下面为了展示需要写在多行，执行时内容需要在同一行）

./configure

--sbin-path=/usr/local/nginx/nginx

--conf-path=/usr/local/nginx/nginx.conf

--pid-path=/usr/local/nginx/nginx.pid

--with-pcre=../pcre-4.4

--with-zlib=../zlib-1.1.3