# Nginx编译安装说明

## 需要使用的工具及其安装：

gcc gcc-c++ make flex bison autoconf automake ncurses-devel vim-enhanced apr-devel apr-util-devel curl curl-devel

### 安装方法：

yum –y install gcc gcc-c++ make flex bison autoconf automake ncurses-devel vim-enhanced apr-devel apr-util-devel curl curl-devel

## 依赖的包及其安装：

yum –y install bzip2-devel zlib-devel ncurses-devel libjpeg-devel libpng-devel libtiff-devel freetype-devel pam-devel openssl-devel libxml2-devel gettext-devel pcre-devel ntp libc-client-devel gmp-devel pspell-devel libmcrypt-devel freetype libevent\* libtool gd-devel

**模块依赖性Nginx需要依赖下面3个包[最小依赖]**

**1. gzip 模块需要zlib库 (**[**http://www.zlib.net/**](http://www.zlib.net/)**)**

**2. rewrite 模块需要pcre库 (**[**http://www.pcre.org/**](http://www.pcre.org/)**)**

**3. ssl 功能需要openssl库 (**[**http://www.openssl.org**](http://www.openssl.org/)**)**

## 获取nginx

<http://nginx.org/en/download.html> (下载稳定版本1.2.8)

## 为nginx建组和用户（nginx以独立的账户和组运行可以方便管理）

groupadd nginx

useradd nginx –M –g nginx

## 安装选项说明：

1）目录属性

**--prefix=**<path> - Nginx安装路径。如果没有指定，默认为 /usr/local/nginx。

**--sbin-path=**<path> - Nginx可执行文件安装路径。只能安装时指定，如果没有指定，默认为<prefix>/sbin/nginx。

**--conf-path=**<path> - 在没有给定-c选项下默认的nginx.conf的路径。如果没有指定，默认为<prefix>/conf/nginx.conf。

**--pid-path=**<path> - 在nginx.conf中没有指定pid指令的情况下，默认的nginx.pid的路径。如果没有指定，默认为 <prefix>/logs/nginx.pid。

**--lock-path=**<path> - nginx.lock文件的路径，默认为<prefix>/logs/nginx.lock

**--error-log-path=**<path> - 在nginx.conf中没有指定error\_log指令的情况下，默认的错误日志的路径。如果没有指定，默认为 <prefix>/logs/error.log。

**--http-log-path=**<path> - 在nginx.conf中没有指定access\_log指令的情况下，默认的访问日志的路径。如果没有指定，默认为 <prefix>/logs/access.log。

**--user=**<user> - 在nginx.conf中没有指定user指令的情况下，默认的nginx使用的用户。如果没有指定，默认为 nobody。

**--group=**<group> - 在nginx.conf中没有指定user指令的情况下，默认的nginx使用的组。如果没有指定，默认为 nobody。

**--builddir=DIR** - 指定编译的目录

2）模块

**--with-rtsig\_module** - 启用 rtsig 模块

**--with-select\_module** **--without-select\_module** -允许或不允许开启SELECT模式，如果 configure 没有找到更合适的模式，比如：kqueue(sun os),epoll (linux kenel 2.6+), rtsig(实时信号)或者/dev/poll(一种类似select的模式，底层实现与SELECT基本相 同，都是采用轮训方法) SELECT模式将是默认安装模式

**--with-poll\_module** **--without-poll\_module** - Whether or not to enable the poll module. This module is enabled by default if a more suitable method such as kqueue, epoll, rtsig or /dev/poll is not discovered by configure.

**--with-http\_ssl\_module** -开启HTTP SSL模块，使NGINX可以支持HTTPS请求。这个模块需要已经安装了OPENSSL，在DEBIAN上是libssl

**--with-http\_realip\_module** - 启用 ngx\_http\_realip\_module

**--with-http\_addition\_module** - 启用 ngx\_http\_addition\_module

**--with-http\_sub\_module** - 启用 ngx\_http\_sub\_module

**--with-http\_dav\_module** - 启用 ngx\_http\_dav\_module

**--with-http\_flv\_module** - 启用 ngx\_http\_flv\_module

**--with-http\_stub\_status\_module** - 启用 "server status" 页

**--without-http\_charset\_module** - 禁用 ngx\_http\_charset\_module

**--without-http\_gzip\_module** - 禁用 ngx\_http\_gzip\_module. 如果启用，需要 zlib 。

**--without-http\_ssi\_module** - 禁用 ngx\_http\_ssi\_module

**--without-http\_userid\_module** - 禁用 ngx\_http\_userid\_module

**--without-http\_access\_module** - 禁用 ngx\_http\_access\_module

**--without-http\_auth\_basic\_module** - 禁用 ngx\_http\_auth\_basic\_module

**--without-http\_autoindex\_module** - 禁用 ngx\_http\_autoindex\_module

**--without-http\_geo\_module** - 禁用 ngx\_http\_geo\_module

**--without-http\_map\_module** - 禁用 ngx\_http\_map\_module

**--without-http\_referer\_module** - 禁用 ngx\_http\_referer\_module

**--without-http\_rewrite\_module** - 禁用 ngx\_http\_rewrite\_module. 如果启用需要 PCRE 。

**--without-http\_proxy\_module** - 禁用 ngx\_http\_proxy\_module

**--without-http\_fastcgi\_module** - 禁用 ngx\_http\_fastcgi\_module

**--without-http\_memcached\_module** - 禁用 ngx\_http\_memcached\_module

**--without-http\_limit\_zone\_module** - 禁用 ngx\_http\_limit\_zone\_module

**--without-http\_empty\_gif\_module** - 禁用 ngx\_http\_empty\_gif\_module

**--without-http\_browser\_module** - 禁用 ngx\_http\_browser\_module

**--without-http\_upstream\_ip\_hash\_module** - 禁用 ngx\_http\_upstream\_ip\_hash\_module

**--with-http\_perl\_module** - 启用 ngx\_http\_perl\_module

**--with-perl\_modules\_path=PATH** - 指定 perl 模块的路径

**--with-perl=PATH** - 指定 perl 执行文件的路径

**--http-log-path=PATH** - Set path to the http access log

**--http-client-body-temp-path=PATH** - Set path to the http client request body temporary files

**--http-proxy-temp-path=PATH** - Set path to the http proxy temporary files

**--http-fastcgi-temp-path=PATH** - Set path to the http fastcgi temporary files

**--without-http** - 禁用 HTTP server

**--with-mail** - 启用 IMAP4/POP3/SMTP 代理模块

**--with-mail\_ssl\_module** - 启用 ngx\_mail\_ssl\_module

**--with-cc=PATH** - 指定 C 编译器的路径

**--with-cpp=PATH** - 指定 C 预处理器的路径

**--with-cc-opt=OPTIONS** - Additional parameters which will be added to the variable CFLAGS. With the use of the system library PCRE in FreeBSD, it is necessary to indicate --with-cc-opt="-I /usr/local/include". If we are using select() and it is necessary to increase the number of file descriptors, then this also can be assigned here: --with-cc-opt="-D FD\_SETSIZE=2048".

**--with-ld-opt=OPTIONS** - Additional parameters passed to the linker. With the use of the system library PCRE in FreeBSD, it is necessary to indicate --with-ld-opt="-L /usr/local/lib".

**--with-cpu-opt=CPU** - 为特定的 CPU 编译，有效的值包括：pentium, pentiumpro, pentium3, pentium4, athlon, opteron, amd64, sparc32, sparc64, ppc64

**--without-pcre** - 禁止 PCRE 库的使用。同时也会禁止 HTTP rewrite 模块。在 "location" 配置指令中的正则表达式也需要 PCRE 。

**--with-pcre=DIR** - 指定 PCRE 库的源代码的路径。

**--with-pcre-opt=OPTIONS** - Set additional options for PCRE building.

**--with-md5=DIR** - Set path to md5 library sources.

**--with-md5-opt=OPTIONS** - Set additional options for md5 building.

**--with-md5-asm** - Use md5 assembler sources.

**--with-sha1=DIR** - Set path to sha1 library sources.

**--with-sha1-opt=OPTIONS** - Set additional options for sha1 building.

**--with-sha1-asm** - Use sha1 assembler sources.

**--with-zlib=DIR** - Set path to zlib library sources.

**--with-zlib-opt=OPTIONS** - Set additional options for zlib building.

**--with-zlib-asm=CPU** - Use zlib assembler sources optimized for specified CPU, valid values are: pentium, pentiumpro

**--with-openssl=DIR** - Set path to OpenSSL library sources

**--with-openssl-opt=OPTIONS** - Set additional options for OpenSSL building

**--with-debug** - 启用调试日志

**--add-module=PATH** - Add in a third-party module found in directory PATH

## 配置文件