**nginx参数：**

--prefix= 指向安装目录

--sbin-path 指向（执行）程序文件（nginx）

--conf-path= 指向配置文件（nginx.conf）

--error-log-path= 指向错误日志目录

--pid-path= 指向pid文件（nginx.pid）

--lock-path= 指向lock文件（nginx.lock）（安装文件锁定，防止安装文件被别人利用，或自己误操作。）

--user= 指定程序运行时的非特权用户

--group= 指定程序运行时的非特权用户组

--builddir= 指向编译目录

--with-rtsig\_module 启用rtsig模块支持（实时信号）

--with-select\_module 启用select模块支持（一种轮询模式,不推荐在高载环境下使用）禁用：--without-select\_module

--with-poll\_module 启用poll模块支持（功能与select相同，与select特性相同，为一种轮询模式,不推荐在高载环境下使用）

--with-file-aio 启用file aio支持（一种APL文件传输格式）

--with-ipv6 启用ipv6支持

--with-http\_ssl\_module 启用ngx\_http\_ssl\_module支持（使支持https请求，需已安装openssl）

--with-http\_realip\_module 启用ngx\_http\_realip\_module支持（这个模块允许从请求标头更改客户端的IP地址值，默认为关）

--with-http\_addition\_module 启用ngx\_http\_addition\_module支持（作为一个输出过滤器，支持不完全缓冲，分部分响应请求）

--with-http\_xslt\_module 启用ngx\_http\_xslt\_module支持（过滤转换XML请求）

--with-http\_image\_filter\_module 启用ngx\_http\_image\_filter\_module支持（传输JPEG/GIF/PNG 图片的一个过滤器）（默认为不启用。gd库要用到）

--with-http\_geoip\_module 启用ngx\_http\_geoip\_module支持（该模块创建基于与MaxMind GeoIP二进制文件相配的客户端IP地址的ngx\_http\_geoip\_module变量）

--with-http\_sub\_module 启用ngx\_http\_sub\_module支持（允许用一些其他文本替换nginx响应中的一些文本）

--with-http\_dav\_module 启用ngx\_http\_dav\_module支持（增加PUT,DELETE,MKCOL：创建集合,COPY和MOVE方法）默认情况下为关闭，需编译开启

--with-http\_flv\_module 启用ngx\_http\_flv\_module支持（提供寻求内存使用基于时间的偏移量文件）

--with-http\_gzip\_static\_module 启用ngx\_http\_gzip\_static\_module支持（在线实时压缩输出数据流）

--with-http\_random\_index\_module 启用ngx\_http\_random\_index\_module支持（从目录中随机挑选一个目录索引）

--with-http\_secure\_link\_module 启用ngx\_http\_secure\_link\_module支持（计算和检查要求所需的安全链接网址）

--with-http\_degradation\_module   启用ngx\_http\_degradation\_module支持（允许在内存不足的情况下返回204或444码）

--with-http\_stub\_status\_module 启用ngx\_http\_stub\_status\_module支持（获取nginx自上次启动以来的工作状态）

--without-http\_charset\_module 禁用ngx\_http\_charset\_module支持（重新编码web页面，但只能是一个方向--服务器端到客户端，并且只有一个字节的编码可以被重新编码）

--without-http\_gzip\_module 禁用ngx\_http\_gzip\_module支持（该模块同-with-http\_gzip\_static\_module功能一样）

--without-http\_ssi\_module 禁用ngx\_http\_ssi\_module支持（该模块提供了一个在输入端处理处理服务器包含文件（SSI）的过滤器，目前支持SSI命令的列表是不完整的）

--without-http\_userid\_module 禁用ngx\_http\_userid\_module支持（该模块用来处理用来确定客户端后续请求的cookies）

--without-http\_access\_module 禁用ngx\_http\_access\_module支持（该模块提供了一个简单的基于主机的访问控制。允许/拒绝基于ip地址）

--without-http\_auth\_basic\_module 禁用ngx\_http\_auth\_basic\_module（该模块是可以使用用户名和密码基于http基本认证方法来保护你的站点或其部分内容）

--without-http\_autoindex\_module 禁用disable ngx\_http\_autoindex\_module 支持（该模块用于自动生成目录列表，只在ngx\_http\_index\_module模块未找到索引文件时发出请求。）

--without-http\_geo\_module 禁用ngx\_http\_geo\_module支持（创建一些变量，其值依赖于客户端的IP地址）

--without-http\_map\_module 禁用ngx\_http\_map\_module支持（使用任意的键/值对设置配置变量）

--without-http\_split\_clients\_module 禁用ngx\_http\_split\_clients\_module支持（该模块用来基于某些条件划分用户。条件如：ip地址、报头、cookies等等）

--without-http\_referer\_module 禁用disable ngx\_http\_referer\_module支持（该模块用来过滤请求，拒绝报头中Referer值不正确的请求）

--without-http\_rewrite\_module 禁用ngx\_http\_rewrite\_module支持（该模块允许使用正则表达式改变URI，并且根据变量来转向以及选择配置。如果在server级别设置该选项，那么他们将在 location之前生效。如果在location还有更进一步的重写规则，location部分的规则依然会被执行。如果这个URI重写是因为location部分的规则造成的，那么 location部分会再次被执行作为新的URI。 这个循环会执行10次，然后Nginx会返回一个500错误。）

--without-http\_proxy\_module 禁用ngx\_http\_proxy\_module支持（有关代理服务器）

--without-http\_fastcgi\_module 禁用ngx\_http\_fastcgi\_module支持（该模块允许Nginx 与FastCGI 进程交互，并通过传递参数来控制FastCGI 进程工作。 ）FastCGI一个常驻型的公共网关接口。

--without-http\_uwsgi\_module 禁用ngx\_http\_uwsgi\_module支持（该模块用来医用uwsgi协议，uWSGI服务器相关）

--without-http\_scgi\_module 禁用ngx\_http\_scgi\_module支持（该模块用来启用SCGI协议支持，SCGI协议是CGI协议的替代。它是一种应用程序与HTTP服务接口标准。它有些像FastCGI但他的设计 更容易实现。）

--without-http\_[memcache](http://www.ttlsa.com/nosql/memcache/)d\_module 禁用ngx\_http\_memcached\_module支持（该模块用来提供简单的缓存，以提高系统效率）

-without-http\_limit\_zone\_module 禁用ngx\_http\_limit\_zone\_module支持（该模块可以针对条件，进行会话的并发连接数控制）

--without-http\_limit\_req\_module 禁用ngx\_http\_limit\_req\_module支持（该模块允许你对于一个地址进行请求数量的限制用一个给定的session或一个特定的事件）

--without-http\_empty\_gif\_module 禁用ngx\_http\_empty\_gif\_module支持（该模块在内存中常驻了一个1\*1的透明GIF图像，可以被非常快速的调用）

--without-http\_browser\_module 禁用ngx\_http\_browser\_module支持（该模块用来创建依赖于请求报头的值。如果浏览器为modern ，则$modern\_browser等于modern\_browser\_value指令分配的值；如 果浏览器为old，则$ancient\_browser等于 ancient\_browser\_value指令分配的值；如果浏览器为 MSIE中的任意版本，则 $msie等于1）

--without-http\_upstream\_ip\_hash\_module 禁用ngx\_http\_upstream\_ip\_hash\_module支持（该模块用于简单的负载均衡）

--with-http\_[perl](http://www.ttlsa.com/perl/)\_module 启用ngx\_http\_perl\_module支持（该模块使nginx可以直接使用perl或通过ssi调用perl）

--with-perl\_modules\_path= 设定perl模块路径

--with-perl= 设定perl库文件路径

--http-log-path= 设定access log路径

--http-client-body-temp-path= 设定http客户端请求临时文件路径

--http-proxy-temp-path= 设定http代理临时文件路径

--http-fastcgi-temp-path= 设定http fastcgi临时文件路径

--http-uwsgi-temp-path= 设定http uwsgi临时文件路径

--http-scgi-temp-path= 设定http scgi临时文件路径

-without-http 禁用http server功能

--without-http-cache 禁用http cache功能

--with-mail 启用POP3/IMAP4/SMTP代理模块支持

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

--without-mail\_pop3\_module 禁用pop3协议（POP3即邮局协议的第3个版本,它是规定个人计算机如何连接到互联网上的邮件服务器进行收发邮件的协议。是因特网电子邮件的第一个离线协议标 准,POP3协议允许用户从服务器上把邮件存储到本地主机上,同时根据客户端的操作删除或保存在邮件服务器上的邮件。POP3协议是TCP/IP协议族中的一员，主要用于 支持使用客户端远程管理在服务器上的电子邮件）

--without-mail\_imap\_module 禁用imap协议（一种邮件获取协议。它的主要作用是邮件客户端可以通过这种协议从邮件服务器上获取邮件的信息，下载邮件等。IMAP协议运行在TCP/IP协议之上， 使用的端口是143。它与POP3协议的主要区别是用户可以不用把所有的邮件全部下载，可以通过客户端直接对服务器上的邮件进行操作。）

--without-mail\_smtp\_module 禁用smtp协议（SMTP即简单邮件传输协议,它是一组用于由源地址到目的地址传送邮件的规则，由它来控制信件的中转方式。SMTP协议属于TCP/IP协议族，它帮助每台计算机在发送或中转信件时找到下一个目的地。）

--with-google\_perftools\_module 启用ngx\_google\_perftools\_module支持（调试用，剖析程序性能瓶颈）

--with-cpp\_test\_module 启用ngx\_cpp\_test\_module支持

--add-module= 启用外部模块支持

--with-cc= 指向C编译器路径

--with-cpp= 指向C预处理路径

--with-cc-opt= 设置C编译器参数（PCRE库，需要指定–with-cc-opt=”-I /usr/local/include”，如果使用select()函数则需要同时增加文件描述符数量，可以通过–with-cc- opt=”-D FD\_SETSIZE=2048”指定。）

--with-ld-opt= 设置连接文件参数。（PCRE库，需要指定–with-ld-opt=”-L /usr/local/lib”。）

--with-cpu-opt= 指定编译的CPU，可用的值为: pentium, pentiumpro, pentium3, pentium4, athlon, opteron, amd64, sparc32, sparc64, ppc64

--without-pcre 禁用pcre库

--with-pcre 启用pcre库

--with-pcre= 指向pcre库文件目录

--with-pcre-opt= 在编译时为pcre库设置附加参数

--with-md5= 指向md5库文件目录（消息摘要算法第五版，用以提供消息的完整性保护）

--with-md5-opt= 在编译时为md5库设置附加参数

--with-md5-asm 使用md5汇编源

--with-sha1= 指向sha1库目录（数字签名算法，主要用于数字签名）

--with-sha1-opt= 在编译时为sha1库设置附加参数

--with-sha1-asm 使用sha1汇编源

--with-zlib= 指向zlib库目录

--with-zlib-opt= 在编译时为zlib设置附加参数

--with-zlib-asm= 为指定的CPU使用zlib汇编源进行优化，CPU类型为pentium, pentiumpro

--with-libatomic 为原子内存的更新操作的实现提供一个架构

--with-libatomic= 指向libatomic\_ops安装目录

--with-openssl= 指向openssl安装目录

--with-openssl-opt 在编译时为openssl设置附加参数

--with-debug 启用debug日志