

# Nginx 安装手册

## 1 nginx 安装环境

nginx 是 C 语言开发，建议在 linux 上运行，本教程使用 Centos6.5 作为安装环境。

### ■ gcc

安装 nginx 需要先将官网下载的源码进行编译，编译依赖 gcc 环境，如果没有 gcc 环境，需要安装 gcc: `yum install gcc-c++`

### ■ PCRE

PCRE(Perl Compatible Regular Expressions)是一个 Perl 库，包括 perl 兼容的正则表达式库。nginx 的 http 模块使用 pcre 来解析正则表达式，所以需要在 linux 上安装 pcre 库。

**`yum install -y pcre pcre-devel`**

注：pcre-devel 是使用 pcre 开发的一个二次开发库。nginx 也需要此库。

### ■ zlib

zlib 库提供了很多种压缩和解压缩的方式，nginx 使用 zlib 对 http 包的内容进行 gzip，所以需要在 linux 上安装 zlib 库。

**`yum install -y zlib zlib-devel`**

### ■ openssl

OpenSSL 是一个强大的安全套接字层密码库，囊括主要的密码算法、常用的密钥和证书封装管理功能及 SSL 协议，并提供丰富的应用程序供测试或其它目的使用。

nginx 不仅支持 http 协议，还支持 https（即在 ssl 协议上传输 http），所以需要在 linux 安装 openssl 库。

**`yum install -y openssl openssl-devel`**

## 2 编译安装

将 nginx-1.8.0.tar.gz 拷贝至 linux 服务器。

解压：

```
tar -zxvf nginx-1.8.0.tar.gz
cd nginx-1.8.0
```

### 1、configure

`./configure --help` 查询详细参数（参考本教程附录部分：nginx 编译参数）

**注意：上边将临时文件目录指定为/var/temp/nginx，需要在/var下创建temp及nginx目录**

参数设置如下：

```
./configure \
--prefix=/usr/local/nginx \
--pid-path=/var/run/nginx/nginx.pid \
--lock-path=/var/lock/nginx.lock \
--error-log-path=/var/log/nginx/error.log \
--http-log-path=/var/log/nginx/access.log \
--with-http_gzip_static_module \
--http-client-body-temp-path=/var/temp/nginx/client \
--http-proxy-temp-path=/var/temp/nginx/proxy \
--http-fastcgi-temp-path=/var/temp/nginx/fastcgi \
--http-uwsgi-temp-path=/var/temp/nginx/uwsgi \
--http-scgi-temp-path=/var/temp/nginx/scgi
```

**#注意：**上边将临时文件目录指定为/var/temp/nginx，需要在/var 下创建 temp 及 nginx 目录

## 2、编译安装

```
make
make install
```

安装成功查看安装目录：

```
[root@server01 nginx-1.8.0]# cd ../nginx
[root@server01 nginx]# ll
总用量 12
drwxr-xr-x. 2 root root 4096 5月 10 19:54 conf
drwxr-xr-x. 2 root root 4096 5月 10 19:54 html
drwxr-xr-x. 2 root root 4096 5月 10 19:54 sbin
```

## 3 启动 nginx

```
cd /usr/local/nginx/sbin/
./nginx
```

查询 nginx 进程：

```
[root@server01 sbin]# ps aux|grep nginx
root      15098  0.0  0.0  3568  528 ?        Ss   20:15   0:00 nginx: master process ./nginx
nobody    15099  0.0  0.0  3768  884 ?        S    20:15   0:00 nginx: worker process
```

15098 是 nginx 主进程的进程 id，15099 是 nginx 工作进程的进程 id

**注意：**执行./nginx 启动 nginx，这里可以-c 指定加载的 nginx 配置文件，如下：

```
./nginx -c /usr/local/nginx/conf/nginx.conf
```

如果不指定-c，nginx 在启动时默认加载 conf/nginx.conf 文件，此文件的地址也可以在编译安装 nginx 时指定./configure 的参数（--conf-path= 指向配置文件（nginx.conf））

## 4 停止 nginx

方式 1，快速停止：

```
cd /usr/local/nginx/sbin
```

```
./nginx -s stop
```

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。

方式 2，完整停止(建议使用)：

```
cd /usr/local/nginx/sbin
```

```
./nginx -s quit
```

此方式停止步骤是待 nginx 进程处理任务完毕进行停止。

## 5 重启 nginx

方式 1，先停止再启动（建议使用）：

对 nginx 进行重启相当于先停止 nginx 再启动 nginx，即先执行停止命令再执行启动命令。

如下：

```
./nginx -s quit
```

```
./nginx
```

方式 2，重新加载配置文件：

当 nginx 的配置文件 nginx.conf 修改后，要想让配置生效需要重启 nginx，使用-s reload 不用先停止 nginx 再启动 nginx 即可将配置信息在 nginx 中生效，如下：

```
./nginx -s reload
```

## 6 测试

nginx 安装成功，启动 nginx，即可访问虚拟机上的 nginx：



## Welcome to nginx!

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to [nginx.org](http://nginx.org).  
Commercial support is available at [nginx.com](http://nginx.com).

*Thank you for using nginx.*

到这说明 nginx 上安装成功。

## 7 开机自启动 nginx

### 7.1 编写 shell 脚本

这里使用的是编写 shell 脚本的方式来处理

vi /etc/init.d/nginx (输入下面的代码)

```
#!/bin/bash
# nginx Startup script for the Nginx HTTP Server
# it is v.0.0.2 version.
# chkconfig: - 85 15
# description: Nginx is a high-performance web and proxy server.
#               It has a lot of features, but it's not for everyone.
# processname: nginx
# pidfile: /var/run/nginx.pid
# config: /usr/local/nginx/conf/nginx.conf
nginxd=/usr/local/nginx/sbin/nginx
nginx_config=/usr/local/nginx/conf/nginx.conf
nginx_pid=/var/run/nginx.pid
RETVAL=0
prog="nginx"
# Source function library.
. /etc/rc.d/init.d/functions
# Source networking configuration.
. /etc/sysconfig/network
# Check that networking is up.
```

```

[ ${NETWORKING} = "no" ] && exit 0
[ -x $nginxd ] || exit 0
# Start nginx daemons functions.
start() {
if [ -e $nginx_pid ];then
    echo "nginx already running...."
    exit 1
fi
    echo -n $"Starting $prog: "
    daemon $nginxd -c ${nginx_config}
    RETVAL=$?
    echo
    [ $RETVAL = 0 ] && touch /var/lock/subsys/nginx
    return $RETVAL
}
# Stop nginx daemons functions.
stop() {
    echo -n $"Stopping $prog: "
    killproc $nginxd
    RETVAL=$?
    echo
    [ $RETVAL = 0 ] && rm -f /var/lock/subsys/nginx /var/run/nginx.pid
}
# reload nginx service functions.
reload() {
    echo -n $"Reloading $prog: "
    #kill -HUP `cat ${nginx_pid}`
    killproc $nginxd -HUP
    RETVAL=$?
    echo
}
# See how we were called.
case "$1" in
start)
    start
    ;;
stop)
    stop
    ;;
reload)
    reload
    ;;
restart)
    stop

```

```
        start
        ;;
status)
        status $prog
        RETVAL=$?
        ;;
*)
        echo $"Usage: $prog {start|stop|restart|reload|status|help}"
        exit 1
esac
exit $RETVAL
```

:wq 保存并退出

## 7.2 设置文件的访问权限

chmod a+x /etc/init.d/nginx (a+x ==> all user can execute 所有用户可执行)

这样在控制台就很容易的操作 nginx 了：查看 Nginx 当前状态、启动 Nginx、停止 Nginx、重启 Nginx...

```
[root@DevelopServer nginx-0.8.54]# /etc/init.d/nginx status
nginx is stopped
[root@DevelopServer nginx-0.8.54]# /etc/init.d/nginx start
Starting nginx: [ OK ]
[root@DevelopServer nginx-0.8.54]# /etc/init.d/nginx stop
Stopping nginx: [ OK ]
[root@DevelopServer nginx-0.8.54]# /etc/init.d/nginx restart
Stopping nginx: [FAILED]
Starting nginx: [ OK ]
[root@DevelopServer nginx-0.8.54]# /etc/init.d/nginx stop
Stopping nginx: [ OK ]
[root@DevelopServer nginx-0.8.54]# /etc/init.d/nginx restart
Stopping nginx: [FAILED]
Starting nginx: [ OK ]
[root@DevelopServer nginx-0.8.54]#
```

如果修改了 nginx 的配置文件 nginx.conf，也可以使用上面的命令重新加载新的配置文件并运行，可以将此命令加入到 rc.local 文件中，这样开机的时候 nginx 就默认启动了

## 7.3 加入到 rc.local 文件中

`vi /etc/rc.local`

加入一行 `/etc/init.d/nginx start` 保存并退出，下次重启会生效。