# 在Ubuntu 14.04下编译安装 Nginx

## 本机安装（apt-get在线方式）

### 1、先更新一下安装包：

apt-get update

### 2、安装nginx服务器：

apt-get install nginx

### 3、nginx的目录存放位置：

**配置文件目录 /etc/nginx**

**网站资源文件存放位置/usr/share/nginx**

1. root@coder-671T-M:~# cd /etc/nginx/
2. root@coder-671T-M:/etc/nginx# ll
3. 总用量 64
4. drwxr-xr-x   5 root root  4096 2011-11-29 11:41 ./
5. drwxr-xr-x 172 root root 12288 2011-11-29 11:41 ../
6. drwxr-xr-x   2 root root  4096 2011-07-27 23:37 conf.d/
7. -rw-r--r--   1 root root   867 2011-03-23 12:20 fastcgi\_params
8. -rw-r--r--   1 root root  2258 2010-12-02 14:19 koi-utf
9. -rw-r--r--   1 root root  1805 2010-12-02 14:19 koi-win
10. -rw-r--r--   1 root root  2059 2011-02-22 15:49 mime.types
11. -rw-r--r--   1 root root  1340 2010-12-02 14:19 nginx.conf
12. -rw-r--r--   1 root root   131 2011-05-02 13:25 proxy\_params
13. -rw-r--r--   1 root root   465 2010-12-02 14:19 scgi\_params
14. drwxr-xr-x   2 root root  4096 2011-11-29 11:41 sites-available/
15. drwxr-xr-x   2 root root  4096 2011-11-29 11:41 sites-enabled/
16. -rw-r--r--   1 root root   497 2010-12-02 14:19 uwsgi\_params
17. -rw-r--r--   1 root root  3071 2010-12-02 14:19 win-utf
18. root@coder-671T-M:/etc/nginx# tree
19. .
20. ├── conf.d
21. ├── fastcgi\_params
22. ├── koi-utf
23. ├── koi-win
24. ├── mime.types
25. ├── nginx.conf
26. ├── proxy\_params
27. ├── scgi\_params
28. ├── sites-available
29. │   └── default
30. ├── sites-enabled
31. │   └── default -> /etc/nginx/sites-available/default
32. ├── uwsgi\_params
33. └── win-utf
35. 3 directories, 11 files

1. root@coder-671T-M:/usr/share/nginx# tree
2. .
3. └── www
4. ├── 50x.html
5. └── index.html
7. 1 directory, 2 files
8. root@coder-671T-M:/usr/share/nginx#

### 3、配置conf文件

配置文件默认在etc文件夹下。

cd /etc/nginx

vi nginx.conf

在http下加入server配置：

1. server{
2. listen 81;
3. server\_name localhost;
4. location / {
5. root html;
6. index index.html index.htm;
7. }
8. location /image/ {
9. root image;
10. index index.html;
11. }
12. }

### 5、nginx修改端口

默认的是80，修改成81

ubuntu的配置文件分成三部分：

/etc/nginx根目录存放公共的配置

在sites\_available和sites\_enabled配置站点相关的信息例如域名端口等等。所以修改默认的sites\_available／default文件如下

1. # You may add here your
2. # server {
3. #   ...
4. # }
5. # statements for each of your virtual hosts to this file
7. ##
8. # You should look at the following URL's in order to grasp a solid understanding
9. # of Nginx configuration files in order to fully unleash the power of Nginx.
10. # http://wiki.nginx.org/Pitfalls
11. # http://wiki.nginx.org/QuickStart
12. # http://wiki.nginx.org/Configuration
13. #
14. # Generally, you will want to move this file somewhere, and start with a clean
15. # file but keep this around for reference. Or just disable in sites-enabled.
16. #
17. # Please see /usr/share/doc/nginx-doc/examples/ for more detailed examples.
18. ##
20. server {
21. <strong>listen   7888; </strong>## listen for ipv4; this line is default and implied
22. #listen   [::]:80 default ipv6only=on; ## listen for ipv6
24. root /usr/share/nginx/www;
25. index index.html index.htm;
27. # Make site accessible from http://localhost/
28. server\_name localhost;
30. location / {
31. # First attempt to serve request as file, then
32. # as directory, then fall back to index.html
33. try\_files $uri $uri/ /index.html;
34. }
36. location /doc {
37. root /usr/share;
38. autoindex on;
39. allow 127.0.0.1;
40. deny all;
41. }
43. location /images {
44. root /usr/share;
45. autoindex off;
46. }
48. #error\_page 404 /404.html;
50. # redirect server error pages to the static page /50x.html
51. #
52. #error\_page 500 502 503 504 /50x.html;
53. #location = /50x.html {
54. #   root /usr/share/nginx/www;
55. #}
57. # proxy the PHP scripts to Apache listening on 127.0.0.1:80
58. #
59. #location ~ \.php$ {
60. #   proxy\_pass http://127.0.0.1;
61. #}
63. # pass the PHP scripts to FastCGI server listening on 127.0.0.1:9000
64. #
65. #location ~ \.php$ {
66. #   fastcgi\_pass 127.0.0.1:9000;
67. #   fastcgi\_index index.php;
68. #   include fastcgi\_params;
69. #}
71. # deny access to .htaccess files, if Apache's document root
72. # concurs with nginx's one
73. #
74. #location ~ /\.ht {
75. #   deny all;
76. #}
77. }

80. # another virtual host using mix of IP-, name-, and port-based configuration
81. #
82. #server {
83. #   listen 8000;
84. #   listen somename:8080;
85. #   server\_name somename alias another.alias;
86. #   root html;
87. #   index index.html index.htm;
88. #
89. #   location / {
90. #       try\_files $uri $uri/ /index.html;
91. #   }
92. #}

95. # HTTPS server
96. #
97. #server {
98. #   listen 443;
99. #   server\_name localhost;
100. #
101. #   root html;
102. #   index index.html index.htm;
103. #
104. #   ssl on;
105. #   ssl\_certificate cert.pem;
106. #   ssl\_certificate\_key cert.key;
107. #
108. #   ssl\_session\_timeout 5m;
109. #
110. #   ssl\_protocols SSLv3 TLSv1;
111. #   ssl\_ciphers ALL:!ADH:!EXPORT56:RC4+RSA:+HIGH:+MEDIUM:+LOW:+SSLv3:+EXP;
112. #   ssl\_prefer\_server\_ciphers on;
113. #
114. #   location / {
115. #       try\_files $uri $uri/ /index.html;
116. #   }
117. #}

这样默认的端口就修改成7888了

### 6、nginx的服务器启动，重启，停止

本机使用的是

1. service nginx start
2. service nginx restart
3. service nginx reload
4. more userage：
5. Usage: nginx {start|stop|restart|reload|force-reload|status|configtest|rotate|upgrade}

其他用法：

1. root@coder-671T-M:/etc/nginx/sites-available# /etc/init.d/nginx start
2. Starting nginx: nginx.
3. root@coder-671T-M:/etc/nginx/sites-available# /etc/init.d/nginx start
4. Starting nginx: nginx.
5. root@coder-671T-M:/etc/nginx/sites-available# /etc/init.d/nginx stop
6. Stopping nginx: nginx.
7. root@coder-671T-M:/etc/nginx/sites-available# /etc/init.d/nginx restart
8. Restarting nginx: nginx.

### 7、查看端口情况

1. root@coder-671T-M:/etc/nginx/sites-available# lsof -i:7888
2. COMMAND  PID     USER   FD   TYPE DEVICE SIZE/OFF NODE NAME
3. nginx   6029     root    6u  IPv4  89046      0t0  TCP \*:7888 (LISTEN)
4. nginx   6030 www-data    6u  IPv4  89046      0t0  TCP \*:7888 (LISTEN)
5. nginx   6031 www-data    6u  IPv4  89046      0t0  TCP \*:7888 (LISTEN)
6. nginx   6032 www-data    6u  IPv4  89046      0t0  TCP \*:7888 (LISTEN)
7. nginx   6033 www-data    6u  IPv4  89046      0t0  TCP \*:7888 (LISTEN)
8. root@coder-671T-M:/etc/nginx/sites-available# netstat -anpl | grep 7888
9. tcp        0      0 0.0.0.0:7888            0.0.0.0:\*               LISTEN      6029/nginx
10. root@coder-671T-M:/etc/nginx/sites-available#

## 启动nginx出现的错误 ：nginx: [emerg] bind() to 0.0.0.0:80 failed (98: Address already in use)

使用命令关闭占用80端口的程序

sudo fuser -k 80/tcp

##建立软链接sudo ln -s /usr/local/nginx/sbin/nginx /etc/init.d/##启动nginx/etc/init.d/nginx

之后就打开浏览器 输入http://localhost ;

测试一下吧！

## 给nginx 设定支持PHP功能

首先编辑

1. sudo vi /usr/local/nginx/conf/nginx.conf

直接修改

1. location / {
2. root /usr/local/nginx/html/; //网站的根路径 是 /usr/local/nginx/html/下
3. index index.html index.htm index.php; //主页文件的类型
4. }
5. location ~ \.php$ {
6. root /usr/local/nginx/html; // /网站的根路径 是 /usr/local/nginx/html/下
7. fastcgi\_pass 127.0.0.1:9000;
8. fastcgi\_index index.php;
9. fastcgi\_param SCRIPT\_FILENAME /usr/local/nginx/html$fastcgi\_script\_name; //支持php的路径 意思就是 在/usr/local/nginx

include fastcgi\_params; /html 的目录下 支持php的文件

}

修改这两处即可让nginx 支持 php 功能

然后 sudo /etc/init.d/nginx -s reload //重新载入nginx 的配置文件

## 配置文件详解

## Nginx常用命令

重新装载：service nginx reload

其他的用法还有：

Usage: nginx {start|stop|restart|reload|force-reload|status|configtest|rotate|upgrade}

## Ubuntu下Nginx启动、停止等常用命令

本文详细介绍Ubuntu下Nginx启动、停止等常用命令。在开发过程中，我们会经常的修改Nginx的配置文件，每次修改配置文件都可以先测试下本次修改的配置文件是否正确，可以利用以下命令：

service nginx -t -c /alidata/server/nginx/conf/nginx.conf

http://www.linuxidc.com/upload/2015_04/15040421237672.jpg

当然执行命令前得换到您机器自己的nginx路径。如果打印如下结果，则证明本次修改配置是正确的，否则是错误的。

当然，这只是说你的配置文件语法等没有错误，具体的业务您还得自行测试。

想让该次配置生效，无需关闭ngnix服务，只需要重启服务即可，执行以下命令：

service nginx -s reload

### 1. 运行nginx

可以参考Ubuntu 14.04 安装最新稳定版Nginx 1.6.0 http://www.linuxidc.com/Linux/2014-08/105422.htm

运行nginx。

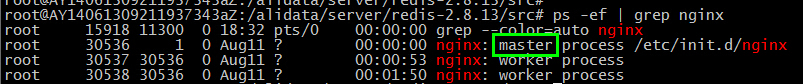
/etc/init.d/nginx

可以运行该命令(前提是按照安装nginx文章中的做法，把运行的脚本复制到/etc/init.d目录下)

### 2. 查询nginx进程

ps -ef | grep nginx

图中绿色框内的master就代表该进程是nginx的主进程



### 3. 停止nginx

停止Nginx：

kill -QUIT 主进程号

本例中就是 ： kill -QUIT 30536

当然还有通用的停止进程的命令，那就是暴力停止了

kill -9 进程号

kill一个进程，在查询一下进程是否还存在，一个个的kill，全部kill完就ok了。