

# Nginx 编译安装

Version:nginx-1.16.1

**说明：在国产银河麒麟操作系统上默认使用的是这个版本的Nginx，以此版本为例说明安装过程。**

**安装需求：支持cookie、支持ssl、http request header转发**

步骤：

## 1. 下载相关源码

1. Nginx源码：[下载连接](#)
2. Sticky：[下载链接](#)

## 2. 依赖

1. zlib-devel
2. openssl-devel
3. pcre-devel

## 3. 解压

1. tar -xvf nginx-1.16.1.tar.gz
2. tar -xvf master.tar.gz 如果解压后的文件名称过长，则重名称文件夹。

## 4. configure

1. nginx-sticky-module 移动到/usr/local/nginx-sticky-module
2. 将目录切换到nginx源码目录。
3. 执行./configure --prefix=/opt/nginx  
--sbin-path=/opt/nginx  
--conf-path=/opt/nginx.conf  
--pid-path=/opt/nginx.pid  
--with-http\_ssl\_module  
--add-module=/usr/local/nginx-sticky-module
  - 特别提醒：一开始安装的nginx-sticky-module的时候一直在nginx-sticky-module目录下面找configure文件。导致一致找不到，实际上nginx-sticky-module是要nginx源码一起编译安装的。

## 5. 编译安装：在nginx源码目录下执行make && make install

## 安装命令汇总

```
1 | ~# tar -xvf nginx-1.16.1.tar.gz
2 | ~# tar -xvf master.tar.gz
3 | ~# mv nginx-goodies-nginx-sticky-module-ng-08a395c66e42 /usr/local/nginx-sticky-module
4 | ~# dnf install zlib-devel openssl-devel pcre-devel
5 | ~# cd nginx-1.16.1
6 | ~# ./configure --prefix=/opt/nginx --add-module=/usr/local/nginx-sticky-module --with-http
7 | ~# make && make install
8 | ~# cd /opt/nginx
9 | ~# vim conf/nginx.conf
```

## 安装问题解决办法：

**sticky问题，ngx\_http\_sticky\_misc.c:176:15: 错误：‘SHA\_DIGEST\_LENGTH’未声明(在此函数内第一次使用)**

- 处理方式：  
安装openssl-devel即可。如果openssl-devel安装后问题还是存在，则修改如下sticky源码文件：

```
1 | ~# vim /usr/local/nginx-sticky-module/ngx_http_sticky_misc.c
```

```
1 | #include <nginx.h>
2 | #include <ngx_config.h>
3 | #include <ngx_core.h>
4 | #include <ngx_http.h>
5 | #include <ngx_md5.h>
6 | #include <ngx_sha1.h>
7 | #include <openssl/sha.h> //添加头文件1
8 | #include <openssl/md5.h> //添加头文件2
9 |
10 | #include "ngx_http_sticky_misc.h"
```

**Nginx启动服务器时出现 [emerg] https protocol requires SSL support in XXXXXX**

- 处理方式：  
重新配置，编译安装。添加--with-http\_ssl\_module

```
1 | ~# ./configure --prefix=/opt/nginx --add-module=/usr/local/nginx-sticky-module --with-http.
```

**解决nginx反向代理proxy不能转发header报头**

使用nginx做负载均衡或http代理时，碰到http header不转发的问题。

配置里只有转发设置原始ip和host的

```
proxy_set_header Host $host; proxy_set_header X-Real-IP $remote_addr;  
proxy_set_header X-Forwarded-Server $host;
```

## 自定义header不能转发问题

恰好我自定义的header中都是用的下划线。

处理办法：

1. 配置中http部分 增加underscores\_in\_headers on; 配置
2. 用减号-替代下划线符号\_，避免这种变态问题。nginx默认忽略掉下划线可能有些原因。

可以加到http或者server中

语法：underscores\_in\_headers on|off

默认值： off

使用字段： http, server

是否允许在header的字段中带下划线。

## 完整配置文件示例

```

1  user root;
2  worker_processes 1;
3
4  #error_log logs/error.log;
5  #error_log logs/error.log notice;
6  #error_log logs/error.log info;
7
8  #pid logs/nginx.pid;
9
10
11 events {
12     worker_connections 1024;
13 }
14
15
16 http {
17     include mime.types;
18     default_type application/octet-stream;
19     underscores_in_headers on;
20     #log_format main '$remote_addr - $remote_user [$time_local] "$request" '
21     #                 '$status $body_bytes_sent "$http_referer" '
22     #                 '"$http_user_agent" "$http_x_forwarded_for"';
23
24     #access_log logs/access.log main;
25
26     sendfile on;
27     #tcp_nopush on;
28
29     #keepalive_timeout 0;
30     keepalive_timeout 65;
31
32     #gzip on;
33     upstream cl_test {
34         # 转发cookie
35         sticky expires=1d secure path=/;
36         server 10.8.8.14:8228;
37     }
38
39     server {
40         listen 80;
41         server_name _;
42         # 转发http请求头
43         proxy_set_header Host $host;
44         proxy_set_header X-Real-IP $remote_addr;
45         proxy_set_header X-Forwarded-for $proxy_add_x_forwarded_for;
46         #charset koi8-r;
47
48         #access_log logs/host.access.log main;
49         #
50         location /sign {

```

```

51         proxy_pass http://10.8.8.14:9999/sign/;
52     }
53     location / {
54         # 转发http请求头 这里比较关键
55         proxy_set_header Host cl_test;
56         proxy_pass https://cl_test;
57         #    root    html;
58         index    default.htm;
59     }
60
61     #error_page 404                /404.html;
62
63     # redirect server error pages to the static page /50x.html
64     #
65     error_page 500 502 503 504 /50x.html;
66     location = /50x.html {
67         root    html;
68     }
69
70     # proxy the PHP scripts to Apache listening on 127.0.0.1:80
71     #
72     #location ~ /\.php$ {
73     #    proxy_pass    http://127.0.0.1;
74     #}
75
76     # pass the PHP scripts to FastCGI server listening on 127.0.0.1:9000
77     #
78     #location ~ /\.php$ {
79     #    root            html;
80     #    fastcgi_pass    127.0.0.1:9000;
81     #    fastcgi_index   index.php;
82     #    fastcgi_param   SCRIPT_FILENAME /scripts$fastcgi_script_name;
83     #    include          fastcgi_params;
84     #}
85
86     # deny access to .htaccess files, if Apache's document root
87     # concurs with nginx's one
88     #
89     #location ~ /\.ht {
90     #    deny    all;
91     #}
92 }
93
94
95 # another virtual host using mix of IP-, name-, and port-based configuration
96 #
97 #server {
98 #    listen    8000;
99 #    listen    somename:8080;
100 #    server_name somename alias another.alias;

```

```
102 # location / {
103 #     root    html;
104 #     index   index.html index.htm;
105 # }
106 #}
107
108
109 # HTTPS server
110 #
111 #server {
112 #     listen      443 ssl;
113 #     server_name localhost;
114
115 #     ssl_certificate      cert.pem;
116 #     ssl_certificate_key  cert.key;
117
118 #     ssl_session_cache    shared:SSL:1m;
119 #     ssl_session_timeout  5m;
120
121 #     ssl_ciphers  HIGH:!aNULL:!MD5;
122 #     ssl_prefer_server_ciphers  on;
123
124 #     location / {
125 #         root    html;
126 #         index   index.html index.htm;
127 #     }
128 #}
129
130 }
```