



链滴

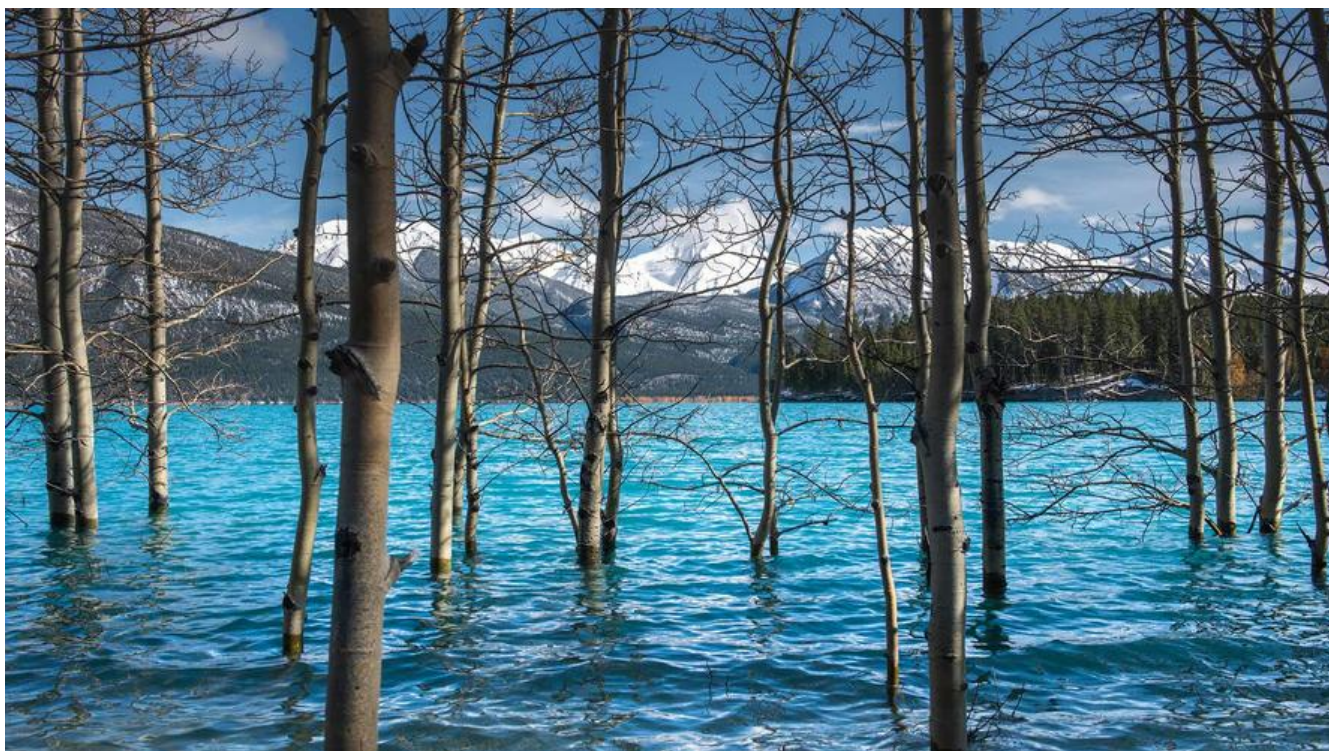
Centos 编译安装 Nginx

作者: [alei](#)

原文链接: <https://ld246.com/article/1636469065557>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)



Centos 7 编译安装Nginx

原文信息：

- 博客传送门： <https://aleivip.com>
- 原文传送门： <https://www.aleivip.com/articles/2021/11/09/1636469062336.html>

附上Nginx官方地址：

- 官网传送门： <http://nginx.org>
 - 官网下载传送门： <http://nginx.org/en/download.html>
-

1 更新yum源

`yum -y update`

2 安装Nginx编译所需所有依赖项

`yum -y install gcc gcc-c++ make zlib-devel pcre-devel openssl-devel`

3 编译我们需要下载源代码。

1. 官网下载

- 下载地址: <http://nginx.org/en/download.html>

```
wget http://nginx.org/download/nginx-1.20.1.tar.gz
```

2. github下载

- 下载地址: <https://github.com/nginx/nginx>

```
git clone git@github.com:nginx/nginx.git
```

4 下载完毕后我们需要解压他们

```
tar -zxvf nginx-1.20.1.tar.gz
```

5 进入Nginx文件夹

```
cd nginx-1.20.1
```

6 选择安装的模块（可选择编译）

```
./configure \
--user=nginx \
--group=nginx \
--prefix=/etc/nginx \
--sbin-path=/usr/sbin/nginx \
--conf-path=/etc/nginx/nginx.conf \
--pid-path=/var/run/nginx.pid \
--lock-path=/var/run/nginx.lock \
--error-log-path=/var/log/nginx/error.log \
--http-log-path=/var/log/nginx/access.log \
--with-http_gzip_static_module \
--with-http_stub_status_module \
--with-http_ssl_module \
--with-pcre \
--with-file-aio \
--with-http_realip_module \
--without-http_scgi_module \
--without-http_uwsgi_module \
--without-http_fastcgi_module
```

7 执行编译安装

```
make & make install
```

8 我们需要在init.d文件夹中创建nginx启动文件，这样每次服务器重新启动init进程都会自动启动我们的Web服务器

```
cd /etc/init.d
```

9 创建nginx执行配置文件

```
touch /etc/init.d/nginx
```

10 编辑nginx 执行配置文件

```
vim nginx
```

- 文件内容

```
#!/bin/sh
#
# nginx - this script starts and stops the nginx daemin
#
# chkconfig:   - 85 15
# description: Nginx is an HTTP(S) server, HTTP(S) reverse \
#               proxy and IMAP/POP3 proxy server
# processname: nginx
# config:      /etc/nginx/nginx.conf
# pidfile:     /var/run/nginx.pid
# user:        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

nginx="/usr/sbin/nginx"
prog=$(basename $nginx)

NGINX_CONF_FILE="/etc/nginx/nginx.conf"

lockfile=/var/run/nginx.lock

start() {
    [ -x $nginx ] || exit 5
    [ -f $NGINX_CONF_FILE ] || exit 6
    echo -n $"Starting $prog: "
    daemon $nginx -c $NGINX_CONF_FILE
    retval=$?
    echo
    [ $retval -eq 0 ] && touch $lockfile
    return $retval
}

stop() {
    echo -n $"Stopping $prog: "
    killproc $prog -QUIT
    retval=$?
    echo
    [ $retval -eq 0 ] && rm -f $lockfile
    return $retval
}
```

```

}

restart() {
    configtest || return $?
    stop
    start
}

reload() {
    configtest || return $?
    echo -n $"Reloading $prog: "
    killproc $nginx -HUP
    RETVAL=$?
    echo
}

force_reload() {
    restart
}

configtest() {
    $nginx -t -c $NGINX_CONF_FILE
}

rh_status() {
    status $prog
}

rh_status_q() {
    rh_status >/dev/null 2>&1
}

case "$1" in
    start)
        rh_status_q && exit 0
        $1
        ;;
    stop)
        rh_status_q || exit 0
        $1
        ;;
    restart|configtest)
        $1
        ;;
    reload)
        rh_status_q || exit 7
        $1
        ;;
    force-reload)
        force_reload
        ;;
    status)
        rh_status
        ;;

```

```

condrestart|try-restart)
    rh_status_q || exit 0
    ;;
*)
    echo $"Usage: $0 {start|stop|status|restart|condrestart|try-restart|reload|force-reload|conf
gtest}"
    exit 2
esac

```

添加执行权限

```
chmod +x /etc/init.d/nginx
```

11 创建nginx用户

```
useradd -r nginx & chkconfig --add nginx & chkconfig --level 345 nginx on
```

12 修改nginx配置文件

1. 打开nginx配置文件

```
vim /etc/nginx/nginx.conf
```

2. 修改内容如下

```
worker_processes 1;
```

```
events {
    worker_connections 1024;
}
```

```
http {
    include mime.types;
    default_type application/octet-stream;

    # ===== 以下两行为添加内容 =====
    types_hash_bucket_size 64;
    server_names_hash_bucket_size 128;
    # ===== 以上两行为添加内容 =====
}
```

```

sendfile    on;
keepalive_timeout 65;
server {
    listen    80;
    server_name localhost;
    location / {
        root html;
        index index.html index.htm;
    }
    error_page 500 502 503 504 /50x.html;
    location = /50x.html {

```

```

        root html;
    }
}
# ===== https ssl证书配置模板 【开始】 =====
#server {
#    #监听443端口
#    listen 443;
#    #你的域名
#    server_name aleivip.com,www.aleivip.com;
#    #SSL开启
#    ssl on;
#    #ssl证书的pem文件路径
#    ssl_certificate /etc/nginx/cert/www.aleivip.com/5751730_aleivip.com.pem;
#    #ssl证书的key文件路径
#    ssl_certificate_key /etc/nginx/cert/www.aleivip.com/5751730_aleivip.com.key;
#    location / {
#        proxy_pass http://公网地址:项目端口号;
#    }
#}
# ===== https ssl证书配置模板 【结束】 =====
}

```

13 启动nginx

```
service nginx start
```

14 验证启动

- 在浏览器输入请求地址

```
http://{你的IP}/
```

13 相关指令

1. 启动nginx

```
service nginx start
```

2. 停止nginx

```
systemctl stop nginx
```

--- END ---

本文转载请携带原文地址

如有问题欢迎指出