



链滴

# Mac 下用 brew 搭建 LNMP 和 LAMP 开发环境

作者: [JinFengYi](#)

原文链接: <https://ld246.com/article/1545384970662>

来源网站: [链滴](#)

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

# Mac下用brew搭建LNMP和LAMP开发环境

Mac下搭建lamp开发环境很容易，有xampp和mamp现成的集成环境。但是集成环境对于经常需要定义一些配置的开发者来说会非常麻烦，而且Mac本身自带apache和php，在brew的帮助下非常容易手动搭建，可控性很高

## Brew

brew对于mac，就像apt-get对于ubuntu，安装软件的好帮手，不能方便更多...

brew的安装方式如下：

```
ruby -e "$(curl -fsSL https://raw.githubusercontent.com/mxcl/homebrew/go/install)"
```

brew常用选项

```
brew install xxx
brew uninstall xxx
brew list
brew update xxx
```

## Apache 或 Nginx

两者只需要安装一个

### Apache

Apache的话使用mac自带的基本就够了，我的系统是10.9，可以使用以下命令控制Apache

```
sudo apachectl start
sudo apachectl restart
sudo apachectl stop
```

唯一要改的是主目录，mac默认在home下有个Sites（站点）目录，访问路径是

```
http://localhost/~user_name
```

这样不太容易访问，修改/etc/apache2/httpd.conf内容

```
DocumentRoot /Users/username/Sites
<Directory /Users/username/Sites>
  Options Indexes MultiViews
  # apache 2.2
  # AllowOverride All
  # Order allow,deny
  # Allow from 127.0.0.1

  # apache 2.4
```

```
Require local
</Directory>
```

这样Sites目录就是网站根目录了，通过<http://localhost>就可以访问了。

## 常用命令

```
sudo apachectl start # 启动apache
sudo apachectl stop # 关闭apache
sudo apachectl restart # 重启apache
```

## 设置vhost

设置vhost可以很方便的进行站点调试，尤其是对于需要在根目录运行的站点。推荐 [VirtualHostX](#)。

设置方法是：

```
sudo vi /etc/apache2/extra/httpd-vhosts.conf
```

然后输入：

```
<Directory "/path/to/site">
  Allow From All
  AllowOverride All
  Options +Indexes
</Directory>
<VirtualHost *:80>
  ServerName "site.test"
  DocumentRoot "/path/to/site"
</VirtualHost>
```

这么设置会有个问题，默认的<http://localhost>会被转到这儿来，所以专门再设置一下localhost的host就好了。

```
<VirtualHost *:80>
  ServerName localhost
  DocumentRoot /path/to/root
</VirtualHost>
```

然后把host文件改一下，把刚刚设置的ServerName改到本地就好了。

```
sudo vi /etc/hosts
```

写入：

```
127.0.0.1 site.test
```

# Nginx

要使用Nginx也比较方便，首先安装

```
brew install nginx
```

启动关闭Nginx的命令如下（如果想要监听80端口，必须以管理员身份运行）

```
#打开 nginx
sudo nginx
#重新加载配置|重启|停止|退出 nginx
nginx -s reload|reopen|stop|quit
#测试配置是否有语法错误
nginx -t
```

## 配置Nginx

```
cd /usr/local/etc/nginx/
mkdir conf.d
```

修改Nginx配置文件

```
vim nginx.conf
```

主要修改位置是最后的include

```
worker_processes 1;

error_log /usr/local/var/log/nginx/error.log warn;

pid /usr/local/var/run/nginx.pid;

events {
    worker_connections 256;
}

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

    log_format main '$remote_addr - $remote_user [$time_local] "$request" '
        '$status $body_bytes_sent "$http_referer" '
        '"$http_user_agent" "$http_x_forwarded_for"';

    access_log /usr/local/var/log/nginx/access.log main;
    port_in_redirect off;
    sendfile on;
    keepalive_timeout 65;

    include /usr/local/etc/nginx/conf.d/*.conf;
```

```
}
```

修改自定义文件

```
vim ./conf.d/default.conf
```

增加一个监听端口

```
server {
    listen    80;
    server_name localhost;

    root /Users/username/Sites/; # 该项要修改为你准备存放相关网页的路径

    location / {
        index index.php;
        autoindex on;
    }

    #proxy the php scripts to php-fpm
    location ~ \.php$ {
        include /usr/local/etc/nginx/fastcgi.conf;
        fastcgi_intercept_errors on;
        fastcgi_pass 127.0.0.1:9000;
    }
}
```

这个时候还不能访问php站点，因为还没有开启php-fpm。

虽然mac 10.9自带了php-fpm，但是由于我们使用了最新的PHP，PHP中自带php-fpm，所以使用P中的php-fpm可以保证版本的一致。

这里的命令在安装完下一步的php后再执行

```
sudo nginx
sudo php-fpm -D
```

## PHP

PHP在mac下默认安装了，但是不好控制版本，利用brew可以再mac下安装最新版本，甚至是多个版本，我装了php5.5

```
brew update
brew tap homebrew/dupes
brew tap josegonzalez/homebrew-php
# brew install php55 --with-fpm #Nginx
brew install php55 #Apache
```

然后修改php的cli路径和apache使用的php模块。在.bashrc或.zshrc里头加以下内容

```
#export PATH="$(brew --prefix josegonzalez/php/php55)/bin:$PATH"  
export PATH="/usr/local/bin:/usr/local/sbin:$PATH"
```

就用刚刚安装的php代替了系统默认cli的php版本。然后在/etc/apache2/httpd.conf下增加

```
LoadModule php5_module /usr/local/Cellar/php55/5.5.8/libexec/apache2/libphp5.so
```

这样就对apache使用的php版本也进行了修改。

后面会用到mongo和mysql，所以可以直接利用下面命令安装php模块，其他模块也类似

```
#brew install php55-mysql #这个已经不需要再安装了  
brew install php55-mongo
```

## MySQL

mac不自带mysql，这里需要重新安装，方法依然很简单

```
brew install mysql  
unset TMPDIR  
mysql_install_db --verbose --user=`whoami` --basedir="$(brew --prefix mysql)" --datadir=/usr  
/local/var/mysql --tmpdir=/tmp  
sudo chown -R your_user /usr/local/var/mysql/
```

第一句是安装，后面的是确保正常使用。然后是启动命令

```
mysql.server start
```

最好给mysql设个密码，方法如下

```
mysqladmin -u root password 'xxx'
```

如果想修改mysql的配置，在/usr/local/etc下建立一个my.cnf，例如增加log

```
[mysqld]  
general-log  
general_log_file = /usr/local/var/log/mysqld.log
```

## MongoDB

MongoDB可以说是最简单的一个，直接执行

```
brew install mongod
```

启动方法

```
mongod --fork
```

## PHPMyAdmin

phpmyadmin几乎是管理mysql最容易的web应用了吧，每次我都顺手装上。

- 去 [官网](#)下载最新的版本
- 解压到~/Sites/phpmyadmin下
- 在phpmyadmin目录下创建一个可写的config目录
- 打开http://localhost/phpmyadmin/setup，安装一个服务，最后保存（这里只需要输入帐号密码够了）
- 将config下生成的config.inc.php移到phpmyadmin根目录下
- 删除config

这样就装好了，虽然可能有点小复杂，但是来一次就习惯了。

这里很可能会遇到2002错误，就是找不到mysql.sock的问题，用下面方法解决

```
sudo mkdir /var/mysql  
sudo ln -s /tmp/mysql.sock /var/mysql/mysql.sock
```

## RockMongo

RockMongo是MongoDB很好用的一个web应用，安装也很容易

- 去 [官网](#)下载最新版本
- 解压到~/Sites/rockmongo下
- 运行http://localhost/rockmongo即可

## 把 site 放在 Dropbox 文件夹下

```
ln -s ~/Dropbox/sitename ~/Sites/sitename  
chmod a+rw ~/Dropbox  
chmod a+rw ~/Dropbox/sitename
```

## 完成

这样就在mac下配置好一个php开发环境了，enjoy it!