在阿里云 CentOS 服务器（ECS）上搭建 nginx + mysql + php-fpm 环境

作者：[王皓](http://ninghao.net/user/6)  
发布于：2014-03-22 04:48  
更新于：2014-03-28 17:29

阿里云的云服务器（ECS）可以选择多种操作系统，打算用它运行 [Drupal](http://ninghao.net/course/tag/drupal)或者 [WordPress](http://ninghao.net/course/tag/wordpress) ，你最好选择 Linux 系统，这篇文章的演示是基于阿里云的 [CentOS](http://www.centos.org/) 操作系统的服务器。我们在上面搭建一个 [nginx](http://nginx.org/) + mysql + php-fpm 的环境，这就是常说的 LNMP 。我们不过多解释什么是什么，而是着重讲流程与方法，想了解具体的细节，去搜索下吧：）这个手册是在阿里云上测试的，不过应该也适用于其它使用 CentOS 系统的服务器。

**背景**

宁皓网的《 [CentOS：在阿里云上运行网站](http://ninghao.net/course/514) 》这个课程里面演示了在阿里云服务器上搭建 AMP 环境，也就是 Apache + MySQL + PHP，并且介绍了 Linux 系统的基本的操作，还有文件目录权限的问题。这些非常重要，可以先看看。

这次想用 Nginx 代替 Apache 服务器，宁皓网现在用的是 Apache 服务器，最近出了点[问题](http://ninghao.net/blog/1366)。所以想更换服务器的系统，并且把 Web Server 换成 Nginx。顺便也去学习一下环境的优化，还有打算把 [memcache](http://memcached.org/) 也用上，也可能会加上 [varnish](https://www.varnish-cache.org/)，我会把整个过程写下来与大伙分享。下面开始第一步，创建一个阿里云服务器的实例。

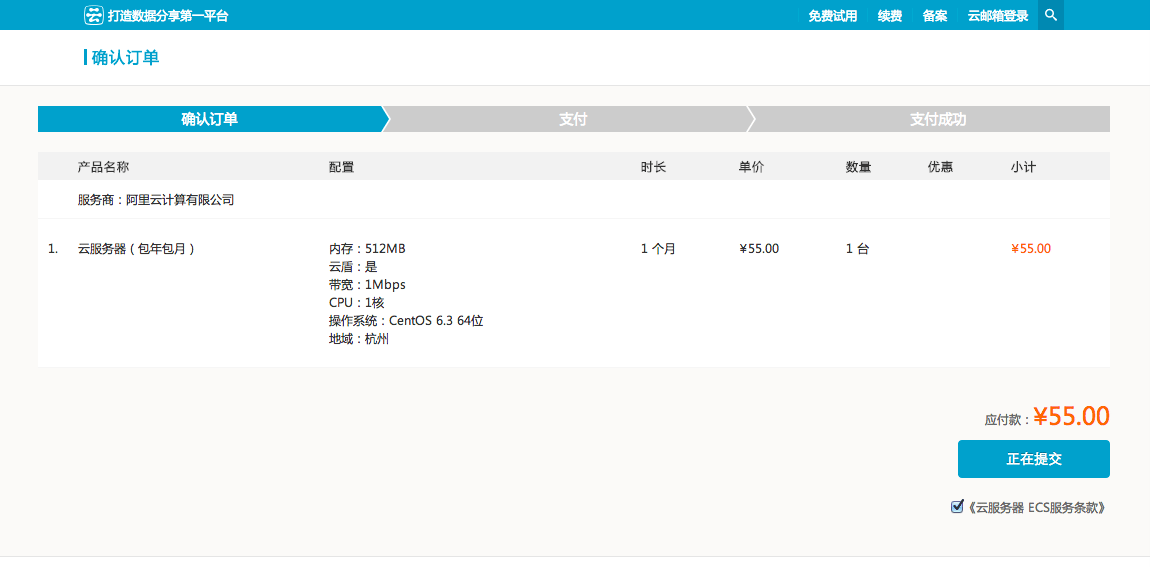
**创建服务器实例**

创建一个服务器实例其实就是去购买一台相应配置的云服务器。注册并登录到阿里云的网站，在 产品服务 里选择 云服务器 ECS ，点击页面上的 立即购买。可以 包年包月 的购买，也可以 按量付费，也就是用多少花多少，不过这种方式不支持网站的备案。

确定后，可以按需要去配置服务器，建议最初可以选择最低配置，因为后续升级也非常容易。选择 CPU，内存，带宽，不同的配置价格不同。你还要选择 地域 ，也就是云服务器所在的地方，目前只有 青岛 与 杭州。再去选择操作系统，这篇文章所做的测试用的是 *CentOS 6.3* 64 位。需要的话可以再增加一块数据盘，默认会带一个系统盘，一共 20 G ，足够我们测试用了。



点击 **立即购买**，预览一下所选的服务器，然后 **提交订单**。

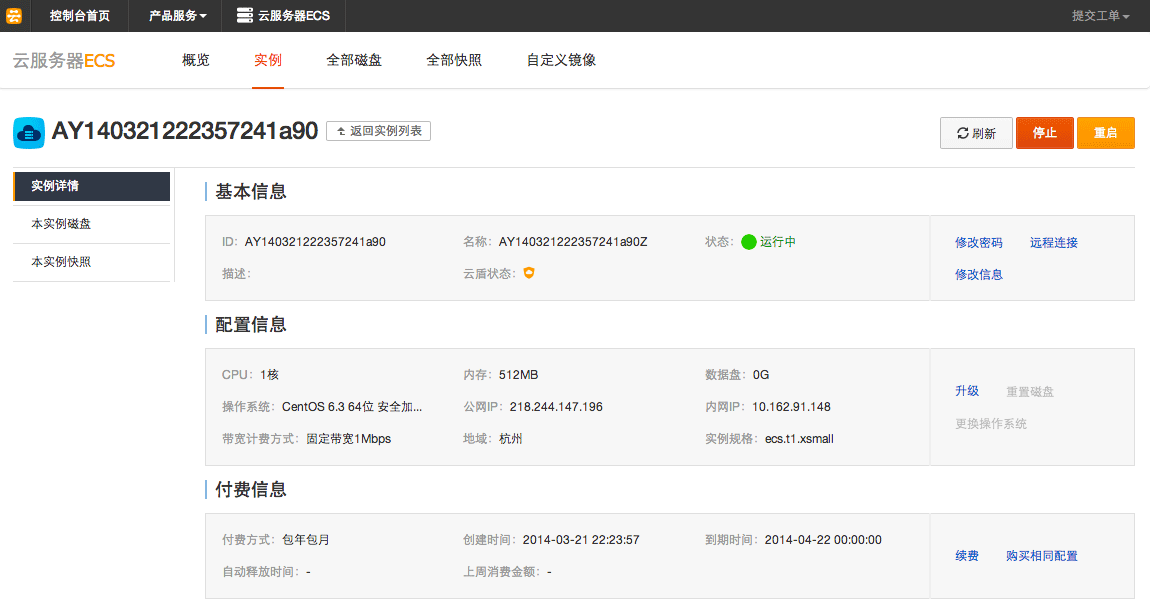


选择支付方式，然后点击 **确认支付**。



注意支付以后，可能会等一小会儿订单才能生效。至少我是等了一会儿，也可能是因为我用的手机版的支付宝。支付成功以后，打开你的个人账户页面，选择 管理控制台，在这里，你可以找到购买的云服务器。云服务器默认的 **root**（超级管理员） 用户的密码，会发送到你的手机上，你也可以点击 **修改密码**，去修改 root 用户密码。

在这个界面，你还可以 停止  或 重启 去服务器。也可以找到你的服务器的公众的 IP 地址。我购买的这台服务器地址是 218.244.147.196，使用这个地址配合 root 用户和密码，你可以使用工具登录到服务器，对它进行远程管理，比如安装软件，修改设置，重启服务等等。



**远程控制你的服务器**

远程控制 Linux 类型的系统的服务器，比如 CentOS 系统的服务器，一般不像 Windows 服务器那样，使用图形界面的远程控制。我们需要使用命令行工具，远程连接到服务器，然后使用命令去控制服务器。Windows 用户可以使用 [Putty](http://www.chiark.greenend.org.uk/~sgtatham/putty/)，Mac 用户可以使用系统自带的终端工具。然后用 ssh 命令，连接到你的服务器。像这样：

ssh root@218.244.147.196

@ 左边的 root 是连接时使用的用户的名字，@ 右边的数字是你的服务器的 IP 地址。这行命令的意思就是，使用 root 用户，登录到 218.244.147.196 这台服务器上。回车以后，会提示你输入 root 用户的密码（这个密码阿里云会发送到你手机上）。

**安装 Nginx**

想在 CentOS 系统上安装 Nginx ，你得先去添加一个资源库，像这样：

vim /etc/yum.repos.d/nginx.repo

使用 vim 命令去打开 /etc/yum.repos.d/nginx.repo ，如果 nginx.repo 不存在，就会去创建一个这样的文件，打开以后按一下小 i 键，进入编辑模式，然后复制粘贴下面这几行代码，完成以后按 esc 键退出，再输入 :wq （保存并退出）

[nginx]

name=nginx repo

baseurl=http://nginx.org/packages/centos/$releasever/$basearch/

gpgcheck=0

enabled=1

完成以后，我们就可以使用 yum 命令去安装 nginx 了，像这样：

yum install nginx

**安装好以后测试一下 nginx 服务：**

service nginx status

应该会返回：

nginx is stopped （nginx 已停止）

**再测试一下 nginx 的配置文件：**

nginx -t

应该会返回：

nginx: the configuration file /etc/nginx/nginx.conf syntax is ok

nginx: configuration file /etc/nginx/nginx.conf test is successful

*... syntax is ok*，... *test is successful*，说明配置文件没问题，同时这个结果里你可以找到 nginx 的配置文件 nginx.conf 所在的位置。

**操纵 nginx 服务**

操纵服务，可以使用使用 service 命令，它可以启动（start），重启（restart），或停止服务（stop），比如要启动 nginx 服务：

service nginx start

服务启动以后，你就可以在浏览器上使用服务器的 IP 地址，或者指向这个地址的域名访问服务器指定的目录了。你会看到类似下面的这些文字。

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. Commercial support is available at nginx.com. Thank you for using nginx.

**配置 nginx 虚拟主机**

安装完 nginx 以后，第一件想到的事应该就是去创建虚拟主机，虚拟主机允许我们在同一台服务器上运行多个网站，我们可以为不同的域名绑定不同的目录，访问这个域名的时候，会打开对应目录里面的东西。之前介绍过为 [Apache 服务器创建虚拟主机](http://ninghao.net/blog/1293)，下面来看一下为 nginx 配置虚拟主机。先进入到 nginx 配置文件目录：

cd /etc/nginx/conf.d

复制这个目录里的 *default.conf* ，复制以后的名字可以使用你的虚拟主机名字。比如创建一个 nginx.ninghao.net 的虚拟主机。复制文件可以使用 cp 命令，像这样：

cp default.conf nginx.ninghao.net.conf

再去编辑一下这个复制以后的配置文件，可以使用 vim 命令：

vim nginx.ninghao.net.conf

你会看到像这样的代码：

server {

listen 80;

server\_name localhost;

#charset koi8-r;

#access\_log /var/log/nginx/log/host.access.log main;

location / {

root /usr/share/nginx/html;

index index.html index.htm;

}

...

}

*server\_name* 就是主机名，也就是跟这个虚拟主机绑定在一块儿的域名，我事先把 *nginx.ninghao.net* 指向了服务器，这个虚拟主机就是为它准备的，所以，*server\_name* 后面的东西就是 *nginx.ninghao.net* 。紧接着 *server\_name* 下面可以是一个*root*，就是这个虚拟主机的根目录，也就是网站所在的目录。比如我们要把 *nginx.ninghao.net* 这个网站的文件放在*/home/www/nginx.ninghao.net* 下面，那么这个 *root* 就是这个路径。

然后去掉 location / 里面的 root 这行代码。再在 index 后面加上一种索引文件名，也就是默认打开的文件，这里要加上一个 index.php ，这样访问 nginx.ninghao.net 就可以直接打开 root 目录下面的 index.php 了。稍后我们再去安装 php 。修改之后，看起来像这样：

server {

listen 80;

server\_name **nginx.ninghao.net**;

root /home/www/nginx.ninghao.net;

#charset koi8-r;

#access\_log /var/log/nginx/log/host.access.log main;

location / {

index **index.php** index.html index.htm;

}

...

}

这个配置文件先修改到这，稍后，我们再回来继续修改一下它。保存一下，按 *esc* ，输入 *:wp*（保存并退出）。现在虚拟主机应该就可以使用了。主机的域名是 *nginx.ninghao.net*，访问它的时候，打开的是 */home/www/nginx.ninghao.net* 这个目录里面的东西，你可以在这个目录下放点东西。

重启 nginx 或者重新加载 nginx 可以让配置文件生效。

service nginx reload

现在，打开浏览器，输入你为虚拟主机设置的域名，看看是否能打开你指定的目录里的东西。

**配置 php-fpm**

要让 nginx 能够执行 php 文件，需要去安装一下 [php-fpm](http://php-fpm.org/)，它直接包含在了 CentOS 资源库里，所以直接使用 yum 命令可以安装它：

yum install php-fpm

完成以后，可以检查一下 php-fpm 的运行状态，使用 *service* 命令：

service php-fpm status

返回：

php-fpm is stopped（php-fpm 已停止）

启动 php-fpm 同样可以使用*service* 命令：

service php-fpm start

**让 nginx 可以执行 php**

现在我们应该就可以让 nginx 去执行 php 了。不过你需要修改一下 nginx 的配置文件，之前我们在配置虚拟主机的时候，创建了一个 *nginx.ninghao.net.conf* 的配置文件，需要去修改下 nginx 的这个配置文件，才能去执行 php 。使用 *vim* 命令去编辑它：

vim /etc/nginx/conf.d/nginx.ninghao.net.conf

注意你的配置文件不一定叫 nginx.ninghao.net.conf，应该是你自己命名的配置文件。打开以后，找到下面这段字样的代码：

# pass the PHP scripts to FastCGI server listening on 127.0.0.1:9000

#

#location ~ \.php$ {

# root html;

# fastcgi\_pass 127.0.0.1:9000;

# fastcgi\_index index.php;

# fastcgi\_param SCRIPT\_FILENAME /scripts$fastcgi\_script\_name;

# include fastcgi\_params;

#}

这是 nginx 默认给我们的用来执行 php 的配置，从 location 开始取消注释，会让这个配置生效，然后我们还得简单去修改一下：

# pass the PHP scripts to FastCGI server listening on 127.0.0.1:9000

#

location ~ \.php$ {

**# root html;**

fastcgi\_pass 127.0.0.1:9000;

fastcgi\_index index.php;

fastcgi\_param SCRIPT\_FILENAME **$document\_root**$fastcgi\_script\_name;

include fastcgi\_params;

}

注意 root 那里仍然是被注释掉的，还有 SCRIPT\_FILENAME 后面修改了一下，把 /scripts 换成了*$document\_root* 。保存并退出。然后重新启动 nginx：

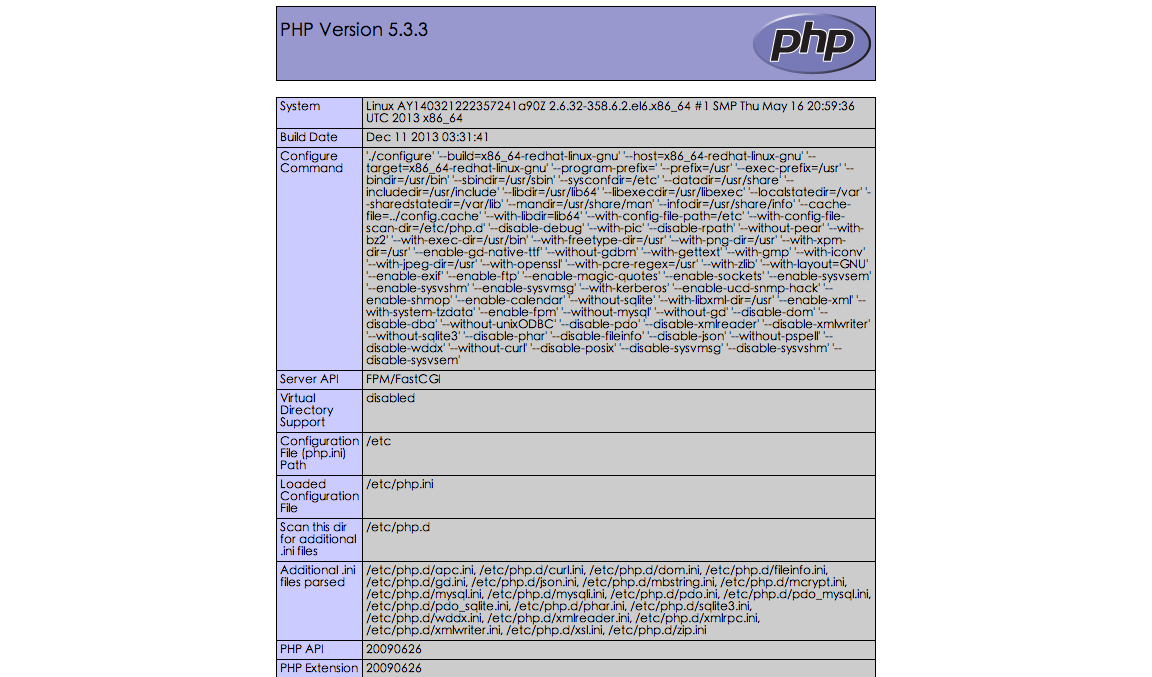
service nginx restart

**测试是否可以执行 php**

现在，我们已经安装了 php-fpm，并修改了 nginx 的配置文件让它可以去执行 php，下面，我们得去测试一下，可以使用 php 的 *phpinfo();* 函数，方法是在你的虚拟主机根目录下面，创建一个 php 文件，命名为 *phpinfo.php*，然后在这个文件里输入：

<?php phpinfo(); ?>

保存文件并退出。在浏览器里打开刚才创建的这个 php 文件。我这里应该是 [*http://nginx.ninghao.net/phpinfo.php*](http://nginx.ninghao.net/phpinfo.php)。打开以后，你应该能看到像下面这样的界面，如果能，说明 nginx 已经可以执行 php 了。



**为系统添加资源库**

前面我们使用 *yum* 命令去安装了一些东西，用起来很简单，*yum + 要安装的东西 + install*。它其实是一个资源包的管理工具，用它安装东西的时候，会检查这个东西所依赖的其它的东西，在安装的同时，也会去安装这些被依赖的东西。CentOS 本身带了一些资源库，但并不是所有的东西都在这些资源库里，所以，我们可以给系统手工的添加一些资源库，这样使用 yum 命令，就可以很方便的去安装在资源库里的东西了。

在这里可以找到可用的资源库：<http://iuscommunity.org/pages/Repos.html> ，你要根据自己的系统的版本去下载对应的资源库，然后安装在系统里面。查看系统的版本可以这样做：

cat /etc/redhat-release

我这里使用的是 CentOS 系统，所以返回了：

CentOS release 6.3 (Final)

说明我用的是 *CentOS 6* ，并且是 64 位的，打开上面的资源库地址，找到 *Release Packages* ，然后找到 *CentOS 6* 这个区域下面的链接，我们用这个 *epel* 资源库。打开这个地址以后，复制 *Packages* 下面的链接 *epel-release-6-5.noarch* 。然后回到我们连接到服务器的终端工具，Windows 用户应该用的是 *putty* 。用 cd 命令进入到一个目录里面，比如用户的根目录，可以这样：

cd ~

然后使用 *wget* 命令，把刚才复制的链接里的资源库下载下来：

wget http://dl.iuscommunity.org/pub/ius/stable/CentOS/6/x86\_64/epel-release-6-5.noarch.rpm

用*ls* 命令可以查看一下当前这个目录下面的东西，你应该可以看到 *epel-release-6-5.noarch.rpm*（也或者是你选择要下载的资源库）。下载好以后，去安装一下：

rpm -ivh epel-release-6-5.noarch.rpm

查看一下系统里面的资源库：

yum repolist

你会看到 *epel* 这个资源库，它就是我们刚才安装的。系统里的资源库地址是在：*/etc/yum.repos.d*，在这里，你同样可以找到 *epel.repo*。这样我们就可以使用 *yum* 命令去安装更多的东西了。

**安装 mysql**

mysql 可以管理网站用到的数据库，WordPress 和 Drupal 也都支持 mysql 数据库。所以我们的 Web 运行环境里，需要安装一个 mysql 。之前我们已经添加了资源库，所以可以直接使用 yum 命令去安装  mysql ：

yum install mysql-server

安装完成后，使用 service 命令启动 mysql 服务：

service mysqld start

然后我们需要简单配置一下 mysql ，默认安装以后 mysql 的 root 用户是没有密码的，对于生产环境来说，这肯定是不行的，另外还有一些安全相关的设置，可以使用下面这行命令去配置一下，它是一个向导，问你一些问题，你要给出答案，比如是否要设置 root 用户的密码， 密码是什么等等。

mysql\_secure\_installation

*Enter current password for root (enter for none):*  
解释：输入当前 root 用户密码，默认为空，直接回车。  
*Set root password? [Y/n]  y*  
解释：要设置 root 密码吗？输入 y 表示愿意。  
*Remove anonymous users? [Y/n]  y*  
解释：要移除掉匿名用户吗？输入 y 表示愿意。  
*Disallow root login remotely? [Y/n]  y*  
解释：不想让 root 远程登陆吗？输入 y 表示愿意。  
*Remove test database and access to it? [Y/n]  y*  
解释：要去掉 test 数据库吗？输入 y 表示愿意。  
*Reload privilege tables now? [Y/n]  y*  
解释：想要重新加载权限吗？输入 y 表示愿意。

**安装 php 扩展**

现在，我们有了可以提供 web 服务的 nginx ，并且安装了 php-fpm ，配置了 nginx 可以让它去执行 php ，也安装了数据库管理系统。不过在运行真正的网站的时候，我们还需要为 php 安装一些额外的扩展，比如 处理 mysql 数据库的 *mysql* 扩展，缓存功能的 *apc* 扩展，处理图像的*gd* 扩展等等。安装它们同样可以使用 yum 命令。

yum install php-pecl-apc php-mysql php-gd php-mcrypt php-pear php-mbstring php-xmlrpc php-dom

上面安装了一些 php 的扩展，如果你发现在安装网站的时候提示需要安装其它的扩展，同样可以使用 yum 命令去安装。安装完成以后，需要重启一下 php-fpm ：

service php-fpm restart

**目录与文件的权限问题**

网站上面有些目录或文件需要有写入权限，这样你得为这些目录和文件分配合适的权限，一般可以把它们的拥有者设置成 php 和 nginx 使用的用户，默认 nginx 的用户就是 *nginx* ，而 php-fpm 使用的用户默认是 *apache*。我们可以把它们改成一个统一的用户，可以修改 php-fpm 的用户为 *nginx* 。你可以使用下面的命令去查看一下 nginx 和 php-fpm 所使用的用户名：

ps aux|grep php

ps aux|grep nginx

修改所使用的用户，可以通过使用 nginx 和 php-fpm 的配置文件，nginx 的配置文件是： */etc/nginx/nginx.conf* ，php-fpm 的配置文件是：*/etc/php-fpm.conf*，还有在 */etc/php-fpm.d/\** 这个目录里的所有文件都是 php-fpm 的配置文件。默认这个目录里有一个 [*www.conf*](http://www.conf/) ，你可以编辑这个文件来修改 php-fpm 所使用的用户名称。使用 *vim* 命令：

vim /etc/php-fpm.d/www.conf

打开文件以后你可以搜索下关键词 *user* 或 *apache*，输入 *:/user* 或*:/apache* ，会搜索到结果，按小 *n* 可以继续向下查找，按大 *N* 可以向上查找。直到你看到：

user = apache

把它修改成：

user=nginx

esc 退出，再输入 *:wq* ，重启下 php-fpm，这样我们的 nginx 服务器还有 php-fpm 会使用同一个用户：*nginx*，你可以把要可以有写入权限的目录与文件的拥有者修改成 *nginx* 就行了。可以使用*chown* 命令：

chown -R nginx 目录名/文件名

**总结**

好来，现在我们就已经拥有一台 LNMP 环境的服务器了。这个操作手册里提供到的相关配置，还有很多可以优化一下，比如 nginx 的配置文件，可以针对 Drupal 来做一些优化。在后续的文章里面，我会再详细的去解释一下这些配置文件的使用。

phpmyadmin

2. 解压缩phpmyadmin软件包并拷贝到apache的根目录下

            # tar zxvf phpmyadmin-xxx  
            # mv phpmyadmin-xxx  /var/www/html/phpmyadmin

    3. 编辑apache的配置文件，加载php模块，使其可以解析php程序

            # vim /etc/httpd/conf.d/php.conf  
                    add an line: Loadmodule php5\_module /usr/lib64/httpd/modules/libphp5.so

    4. 编辑phpmyadmin配置文件，配置访问路径及mariadb用户名密码等信息

            # cp /var/www/html/phpmyadmin/libraries/config.default.php \

                /var/www/html/phpmyadmin/config.inc.php

            # vim /var/www/html/phpmyadmin/config.inc.php

                    $cfg['PmaAbsoluteUri'] = ''; ==> \

                    $cfg['PmaAbsoluteUri'] = 'http:// 120.27.34.165/phpmyadmin/';  
                    $cfg['blowfish\_secret'] = ''; ==> \

                    $cfg['blowfish\_secret'] = 'your secret  password';   
                    $cfg['Servers'][$i]['auth\_type'] = 'config'; ==> \

                    $cfg['Servers'][$i]['auth\_type'] = 'cookie';  
                    $cfg['Servers'][$i]['user'] = ''; ==> $cfg['Servers'][$i]['user'] = 'root';  
                    $cfg['Servers'][$i]['password'] = ''; ==> \

                    $cfg['Servers'][$i]['password'] = 'your mysql password';  
    5. 启动mariadb程序

            # systemctl start mariadb  
    6. 进行mariadb数据库初始化

            # mysql\_secure\_installation

    7. 启动Apache程序  
            # systemctl start httpd

cd /var/lib/php  
mkdir session  
chown nginx:nginx session  
chmod 770 session

下载thinkphp文件包

配置thinkphp

/Hprose/Server/config.php

<?php

return array(

'DB\_TYPE' => 'mysql', // 数据库类型

'DB\_HOST' => 'localhost', // 服务器地址

'DB\_NAME' => 'market', // 数据库名

'DB\_USER' => 'root', // 用户名

'DB\_PWD' => 'admin', // 密码

'DB\_PORT' => 3306, // 端口

'DB\_CHARSET'=> 'utf8', // 字符集

//自定义变量

'USERNAME\_LENGTH' => 8, //用户账号长度

'SMALLPAGESIZE' => 5, //少量分页长度

'PAGESIZE' => 10, //常规分页长度

'BIGPAGESIZE' => 20, //大量分页长度

'MAXRETURNNUMBER' => 50, //大量返回记录条数

'TOKENPATH' => '/',

);

浏览器出现allow 。。。。。。错误

/Application/Common/Conf/config.php

修改：'\_\_PATHHPROSE\_\_'=>'http://120.27.34.165/market'

方法一：通过controller来获取app

var appElement = document.querySelector('[ng-controller=mainController]');

然后在获取$scope变量：

var $scope = angular.element(appElement).scope();

如果改变了其中的变量之后，需要在页面表现出来，还需要调用apply()方法：

$scope.$apply(function($scope){

$scope.---=---

});

安装ffmpeg

# **yum install -y http://li.nux.ro/download/nux/dextop/el7/x86\_64/nux-dextop-release-0-1.el7.nux.noarch.rpm**

**yum install ffmpeg ffmpeg-devel**