## 1，apache多站点配置

|  |  |
| --- | --- |
| **1，apache/conf/extra/httpd-vhosts.conf文件添加虚拟站点**  <VirtualHost \*:80>      ServerAdmin webmaster@dummy-host.example.com     管理员邮箱  **DocumentRoot** "/PhpComeOn/htdocs/dummy-host.example.com"  网站根目录  **ServerName** dummy-host.example.com     主机名称      ServerAlias www.dummy-host.example.com   主机别名      ErrorLog "logs/dummy-host.example.com-error.log"   错误日志      CustomLog "logs/dummy-host.example.com-access.log" common   日常日志  </VirtualHost> | **2，配置httpd.conf文件添加扩展配置**   include conf/extra/httpd-vhosts.conf   ——添加扩展配置到apache配置文件中  3，**配置httpd.conf文件配置外部文件访问权限**  Directory节点下：      Deny from all     改为  Allow from all      Allow from 127.0.0.1  改为  #Allow from 127.0.0.1  **4，重启apache服务，配置系统hosts文件添加127.0.0.1 为步骤1中配置的主机名称** |

## 2，apache php环境配置

|  |  |  |
| --- | --- | --- |
| 1，安装需要的软件  sudo apt-get install apache2  php5  2，加载libphp.so  cat /etc/apache2/mods-enabled/php5.load | 3,安装mysql  apt-get install mysql-server  4，安装php的mysql扩展模块  apt-get install php5-mysql | 5,检查mysql扩展是否存在  cat /etc/php5/conf.d/mysql.ini  ===================================== |

## 3，Apache配置文件

|  |  |  |
| --- | --- | --- |
| 配置文件加载流程：  C:\Users\BARRET~1\AppData\Local\Temp\enhtmlclip\Image(8).png | mods-available目录下是可以用的模块  mods-enable目录是正在启用的模块，是软连接到available目录下的 | 修改apache端口号  httpd.conf修改 listen 端口号；修改，，ServerName localhost:端口号 |

## 4，Nginx php环境配置

1,由于需要大量的权限操作，方便起见临时提升权限，使用root账号  
sudo su  
2，安装apt源管理工具、添加nginx和php的安装源  
apt-get install python-software-properties  
add-apt-repository ppa:nginx/stable  
add-apt-repository ppa:ondrej/php5  
3，更新系统软件  
apt-get update  
4、安装mysql  
apt-get install mysql-server  
注：在安装过程中，会要求你输入MySQL的root账号的密码。  
5、安装php及对mysql的支持  
apt-get install php5 php5-fpm php5-mysql php-apc  
6、根据实际需要，选择性的安装php的各类功能模块（具体请自行百度）  
apt-get install php-pear php5-dev php5-curl  
apt-get install php5-gd php5-intl php5-imagick  
apt-get install php5-imap php5-mcrypt php5-memcache  
apt-get install php5-ming php5-ps php5-pspell  
apt-get install php5-recode php5-snmp php5-sqlite  
apt-get install php5-tidy php5-xmlrpc php5-xsl  
7、安装nginx  
apt-get install nginx  
8、配置php  
vi /etc/php5/fpm/php.ini  
找到：;cgi.fix\_pathinfo=1  
改为：cgi.fix\_pathinfo=0  
9、配置nginx  
mkdir /www  
chmod 755 /www  
创建一个 /www 目录设置为 755 权限  
vi /etc/nginx/sites-enabled/default  
找到：root /usr/share/nginx/html;  
改为：root /www;  
说明：将网站根目录设置到/www 目录下，注意不要写成 /www/ 这种格式，  
这个习惯性的小小洁癖给我带来了意想不到的困扰，后面细说。  
找到：index index.html index.htm;  
改为：index index.php index.html index.htm;  
说明：在默认索引文件中增加 index.php  
找到：location ~ .php$ { \* } 区块  
做如下调整（改动之处于行末有注释）：  
location ~ .php$ {  
try\_files $uri =404; #增加  
fastcgi\_split\_path\_info ^(.+.php)(/.+)$; #反注释  
## NOTE: You should have "cgi.fix\_pathinfo = 0;" in php.ini  
#  
## With php5-cgi alone:  
# fastcgi\_pass 127.0.0.1:9000;  
## With php5-fpm:  
fastcgi\_pass unix:/var/run/php5-fpm.sock; #反注释  
fastcgi\_index index.php; #反注释  
include fastcgi\_params; #反注释  
}  
10、解决php-fpm与nginx的小bug  
按上述步骤操作后，由于nginx与php-fpm之间的一个小bug，会导致这样的现象：  
网站中的静态页面 \*.html 都能正常访问，而 \*.php 文件虽然会返回200状态码，  
但实际输出给浏览器的页面内容却是空白。  
简而言之，原因是nginx无法正确的将 \*.php 文件的地址传递给php-fpm去解析，  
相当于php-fpm接受到了请求，但这请求却指向一个不存在的文件，于是返回空结果。  
为了解决这个问题，需要改动nginx默认的fastcgi\_params配置文件：  
vi /etc/nginx/fastcgi\_params  
在文件的最后增加一行：  
fastcgi\_param SCRIPT\_FILENAME $document\_root$fastcgi\_script\_name;  
关于这行的内容，多说几句，其中有两个参数：  
$document\_root 即是指网站的根目录，也就是我们在前面刚设置的 root /www;  
$fastcgi\_script\_name 则是指网站上的各个 \*.php 文件名（其实是文件的相对路径）  
这两个合在一起形成完整的 php file path，比如你的网站有个 /test/script.php 文件，  
nginx传递给php-fpm的完整路径就是：/www/test/script.php  
我在前面说配置nginx时，写成了 root /www/; 这种写法，对于nginx不成问题，  
但它在向php-fpm传递php文件路径时，会形成：/www//test/script.php  
这样的文件地址无效（获取不到正确的php文件），于是出现返回空白的问题。  
这个问题是如此之隐蔽，以至于我花了大量时间在查看LOG、寻找资料和调整设置上。  
11，重启各项服务  
service php5-fpm reload  
service nginx reload  
重新加载各项配置改动。