

企业实战LNMP高性能服务器：

LNMP WEB架构中， Nginx为一款高性能Web服务器，本身是不能处理PHP的，当接收到客户端浏览器发送HTTP Request请求时,Nginx服务器响应并处理web请求，静态资源CSS、图片、视频、TXT等静态文件请求，Nginx服务器可以直接处理并回应。

但是PHP动态页面请求Nginx不能直接处理， Nginx服务器会将PHP网页脚本通过接口传输协议（网关协议） PHP-FCGI (Fast-CGI) 传输给PHP-FPM（进程管理程序），PHP-FPM不做处理，然后PHP-FPM调用PHP解析器进程， PHP解析器解析PHP脚本信息。PHP解析器进程可以启动多个，可以实现多线程并发执行。

PHP解释器将解析后的脚本返回到PHP-FPM， PHP-FPM再通过Fast-CGI的形式将脚本信息传送给Nginx,Nginx服务器再通过Http Response的形式传送给浏览器,浏览器再进行解析与渲染然后进行呈现。

1. 安装mysql：

1.1 预编译：

```
cd /usr/src
wget http://mirrors.163.com/mysql/Downloads/MySQL-
5.5/mysql-5.5.60.tar.gz
tar xf mysql-5.5.60.tar.gz
cd mysql-5.5.60/
yum install gcc ncurses-devel libaio bison gcc-c++ git
cmake ncurses-devel ncurses -y

cmake . -DCMAKE_INSTALL_PREFIX=/usr/local/mysql55/ \
-DMYSQL_UNIX_ADDR=/tmp/mysql.sock \
-DMYSQL_DATADIR=/data/mysql \
-DSYSCONFDIR=/usr/local/mysql55/ \
-DMYSQL_USER=mysql \
-DMYSQL_TCP_PORT=3306 \
-DWITH_XTRADB_STORAGE_ENGINE=1 \
-DWITH_INNOBASE_STORAGE_ENGINE=1 \
-DWITH_PARTITION_STORAGE_ENGINE=1 \
```

```
-DWITH_BLACKHOLE_STORAGE_ENGINE=1 \
-DWITH_MYISAM_STORAGE_ENGINE=1 \
-DWITH_READLINE=1 \
-DENABLED_LOCAL_INFILE=1 \
-DWITH_EXTRA_CHARSETS=1 \
-DDEFAULT_CHARSET=utf8 \
-DDEFAULT_COLLATION=utf8_general_ci \
-DEXTRA_CHARSETS=all \
-DWITH_BIG_TABLES=1 \
-DWITH_DEBUG=0
```

1.2 编译/安装:

```
make && make install

cp support-files/my-large.cnf /usr/local/mysql55/my.cnf
cp support-files/mysql.server /etc/init.d/mysql
chmod +x /etc/init.d/mysql
mkdir -p /data/mysql
useradd -s /sbin/nologin mysql
chown -R mysql. /data/mysql
/usr/local/mysql55/scripts/mysql_install_db --user=mysql
--datadir=/data/mysql --basedir=/usr/local/mysql55
/etc/init.d/mysql start
```

1.3 添加系统服务:

```
# 加入service服务:
chkconfig --add mysql
chkconfig --level 35 mysql on
```

2. 安装nginx:

```
# 下载源码:
wget http://nginx.org/download/nginx-1.16.1.tar.gz
# 解压:
tar xf nginx-1.16.0.tar.gz
cd nginx-1.16.0/
# 预编译:
```

```
./configure --prefix=/usr/local/nginx --with-
http_stub_status_module
# 编译/安装
make && make install
# 修改nginx进程用户为nginx:
[root@node4 html]# grep "user"
/usr/local/nginx/conf/nginx.conf
user    nginx;
# 启动nginx:
/usr/local/nginx/sbin/nginx
```

3. 安装php:

```
# 安装依赖:
yum -y install gd curl curl-devel libjpeg libjpeg-devel
libpng libpng-devel freetype freetype-devel libxml2
libxml2-devel
# 解压包:
tar xf php-5.6.8.tar.bz2
cd php-5.6.8
# 预编译
./configure --prefix=/usr/local/php \
--enable-fpm \
--enable-debug \
--with-gd \
--with-jpeg-dir \
--with-png-dir \
--with-freetype-dir \
--enable-mbstring \
--with-curl \
--with-mysql=mysqlnd \
--with-mysqli=mysqlnd \
--with-pdo-mysql=mysqlnd \
--with-config-file-path=/usr/local/php/etc \
--with-zlib-dir
# 编译/安装
make && make install

cp php.ini-development /usr/local/php/etc/php.ini
cp /usr/local/php/etc/php-fpm.conf.default
/usr/local/php/etc/php-fpm.conf
```

```
cp sapi/fpm/init.d.php-fpm /etc/init.d/php-fpm
chmod +x /etc/init.d/php-fpm
# 修改php-fpm的用户为nginx:
[root@node4 html]# egrep "^(user|group)"
/usr/local/php/etc/php-fpm.conf
user = nginx
group = nginx

# 启动php-fpm:
/etc/init.d/php-fpm start
```

4. 上传wordpress网站代码:

```
# 解压包到nginx发布目录:
tar xf wordpress-4.9.4-zh_CN.tar.gz -C
/usr/local/nginx/html/
chown nginx. -R /usr/local/nginx/html/wordpress/
```

5. 创建wordpress虚拟主机:

```
# 指定应用的虚拟主机目录(主配置文件http指令块下添加):
include vhost/*.conf;
# 创建虚拟主机目录:
mkdir -p /usr/local/nginx/conf/vhost
# 创建虚拟主机配置文件:
vim /usr/local/nginx/conf/vhost/blog.wordpress.com.conf
server {
    listen      80;
    server_name blog.wordpress.com;
    #charset koi8-r;
    access_log  logs/wordpress.access.log  main;
    location / {
        root  html/wordpress;
        index index.php  index.html index.htm;
    }
    #error_page  404          /404.html;
    # redirect server error pages to the static page
/50x.html
    #
    error_page  500 502 503 504  /50x.html;
```

```
location = /50x.html {
    root    html;
}

location ~ \.php$ {
    root          html/wordpress;
    fastcgi_pass  127.0.0.1:9000;
    fastcgi_index index.php;
    fastcgi_param SCRIPT_FILENAME
$document_root$fastcgi_script_name;
    include        fastcgi_params;
}

}
```

6. 创建数据库:

```
# 启动数据库服务:
/etc/init.d/mysql start
# 进入数据库创建数据库，并授权:
mysql> create database wordpress charset utf8;
Query OK, 1 row affected (0.00 sec)

mysql> grant all on wordpress.* to "wordpress"@localhost
identified by "123456";
Query OK, 0 rows affected (0.10 sec)

mysql> flush privileges;
Query OK, 0 rows affected (0.00 sec)
```

7. 访问wordpress:

在物理机做好hosts解析 192.168.75.134 blog.wordpress.com

bbs.discuz.com



数据库名 将WordPress安装到哪个数据库？
用户名 与数据库授权信息保持一致
您的数据库用户名。
密码 您的数据库密码。
如果数据库是独立部署，则需要填写数据库所在的
IP地址或者主机名
如果localhost不能用，您通常可以从网站服
务提供商处得到正确的信息。
数据库主机
表前缀 如果您希望在同一个数据库安装多个
WordPress，请修改前缀。
提交



8. 创建discuz虚拟主机配置文件：

```
# 切换到vhost目录:
```

```
cp blog.wordpress.com.conf bbs.discuz.com.conf
# 修改域名和路径:
server {
    listen      80;
    server_name bbs.discuz.com;
    #charset koi8-r;
    access_log  logs/discuz.access.log  main;
    location / {
        root  html/discuz/upload;
        index index.php index.html index.htm;
    }
    #error_page 404          /404.html;
    # redirect server error pages to the static page
    /50x.html
    #
    error_page 500 502 503 504  /50x.html;
    location = /50x.html {
        root  html;
    }

    location ~ \.php$ {
        root           html/discuz/upload;
        fastcgi_pass  127.0.0.1:9000;
        fastcgi_index index.php;
        fastcgi_param SCRIPT_FILENAME
$document_root$fastcgi_script_name;
        include        fastcgi_params;
    }
}
```

9. 上传网站代码:

```
# 解压:
unzip Discuz_X3.1_SC_UTF8.zip -d
/usr/local/nginx/html/discuz/
# 授权:
chown nginx. -R /usr/local/nginx/html/discuz/
```

10. 创建数据库:

```
mysql> create database discuz charset utf8;
Query OK, 1 row affected (0.00 sec)

mysql> grant all on discuz.* to "discuz"@\localhost
identified by "123456";
Query OK, 0 rows affected (0.00 sec)

mysql> flush privileges;
Query OK, 0 rows affected (0.00 sec)
```

11. 访问discuz:

不安全 | bbs.discuz.com/install/

学资料

The screenshot shows the Discuz! Installation Wizard interface. At the top, it displays the title 'Discuz! 安装向导' and the version 'Discuz!X3.1 简体中文 UTF8 版 20150609'. Below the title, there is a section titled '中文版授权协议 适用于中文用户'. It contains the following text:
版权所有 (c) 2001-2013, 北京康盛新创科技有限责任公司保留所有权利。
感谢您选择康盛产品。希望我们的努力能为您提供一个高效快速、强大的站点解决方案，和强大的社区论坛解决方案。康盛公司网址为 <http://www.comsenz.com>, 产品官方讨论区网址为 <http://www.discuz.net>.
用户须知: 本协议是您与康盛公司之间关于您使用康盛公司提供的各种软件产品及服务的法律协议。无论您是个人或组织、盈利与否、用途如何（包括以学习和研究为目的），均需仔细阅读本协议，包括免除或者限制康盛责任的免责条款及对您的权利限制。请您审阅并接受或不接受本服务条款。如您不同意本服务条款及/或康盛随时对其的修改，您应不使用或主动取消康盛公司提供的康盛产品。否则，您的任何对康盛产品中的相关服务的注册、登陆、下载、查看等使用行为将被视为您对本服务条款全部的完全接受，包括接受康盛对服务条款随时所做的任何修改。
本服务条款一旦发生变更，康盛将在网页上公布修改内容。修改后的服务条款一旦在网站管理后台公布即有效代替原来的服务条款。您可随时登陆康盛官方论坛查阅最新版服务条款。如果您选择接受本条款，即表示您同意接受协议各项条件的约束。如果您不同意本服务条款，则不能获得使用本服务的权利。您若有违反本条款规定，康盛公司有权随时中止或终止您对康盛产品的使用资格并保留追究相关法律责任的权利。
At the bottom of the page, there are two buttons: '我同意' (I Agree) and '我不同意' (I Disagree).

环境检查

项目	Discuz! 所需配置	Discuz! 最佳	当前服务器
操作系统	不限制	类Unix	✓ Linux
PHP 版本	5.1	5.3	✓ 5.6.8
附件上传	不限制	2M	✓ 2M
GD 库	1.0	2.0	✓ bundled (2.1.0 compatible)
磁盘空间	10M	不限制	✓ 12165M

目录、文件权限检查

目录文件

./config/config_global.php
./config/config_ucenter.php
./config
.data
.data/cache
.data/avatar
.data/plugindata

所需状态

可写
可写
可写
可写
可写
可写
可写
可写

当前状态

可写
可写
可写
可写
可写
可写
可写
可写

均为绿色，表示没问题

填写数据库信息

数据库服务器:

localhost

数据库服务器所在地址，如果在远程，则写远程服务器IP或者主机名

数据库名:

discuz

数据库服务器地址，一般为 localhost

数据库用户名:

discuz

与数据库授权信息保持一致

数据库密码:

123456

同一数据库运行多个论坛时，请修改前缀

数据表前缀:

pre_

系统信箱 Email:

admin@admin.com

用于发送程序错误报告

填写管理员信息

管理员账号:

admin

管理员后台密码

管理员密码:

管理员密码不能为空

重复密码:

管理员 Email:

admin@admin.com

您的论坛已完成安装，点此访问

创建成功

← → × ⓘ 不安全 | bbs.discuz.com/forum.php

实用工具 教学资料

设为首页 收藏本站

社区动力 DISCUZ!

论坛

请输入搜索内容 帖子 热搜: 活动 交友 discuz

家 》 论坛

今日: 0 | 昨日: 0 | 帖子: 0 | 会员: 1 | 欢迎新会员: admin

Discuz!

默认版块