# 概述

## 引入

访问百度这个网站很容易，但是百度这个网站究竟是如何搭建的，都有哪些组件组成，各个组件之间是如何协作的。

而且百度这个网站访问量非常大，这么大的访问量如何承载

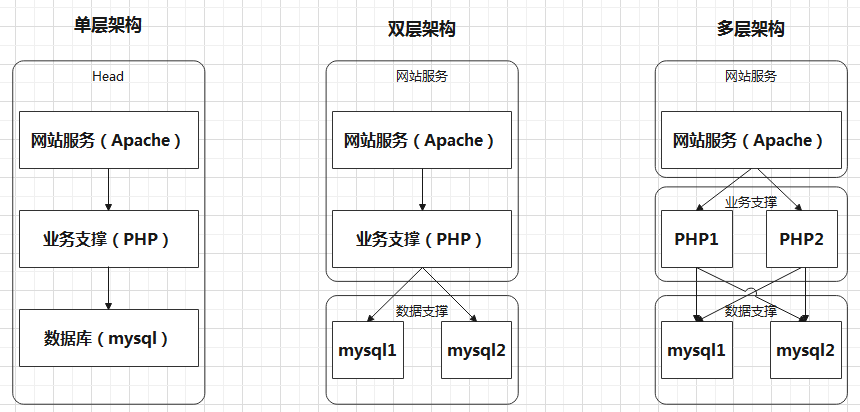
## LAMP

LAMP是一套网站解决方案，使用其就可以搭建出一套网站

* L：Linux最底层操作系统
* A： Apache提供网站服务，服务名叫httpd端口号为80
* M：mysql提供数据存储服务，服务名叫mysqld端口号为3306
* P： PHP业务支撑，负责运行业务逻辑代码

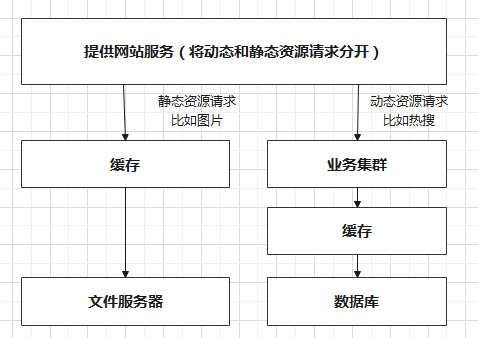
## 架构

* 单层架构：将三个组件部署到一台机器，优点是节省成本方便维护，缺点承载量有限
* 双层架构：将数据库独立出来，因为数据访问压力大的时候可以再加一台分压
* 多层架构：将业务支撑和数据库都独立出来，面对大访问量时可以动态扩展



## 网站结构

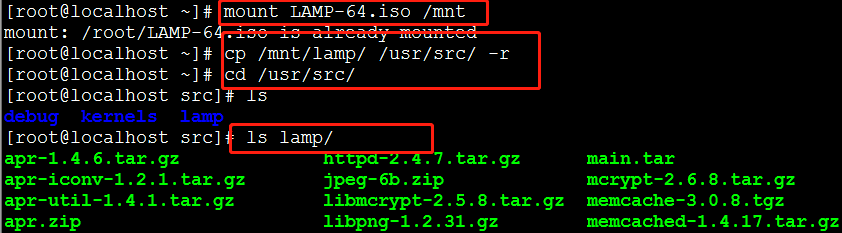
* 网站服务：将动态资源请求和静态资源请求分开处理
* 静态资源：比如请求一张图片，从文件服务器拿到缓存，下次直接从缓存里拿
* 动态资源：比如请求热搜业务
  + 去数据库里拿出所有新闻，依据访问量进行排序，拿出前十个新闻。
  + 将结果放到缓存里，下次再请求热搜业务直接从缓存里拿取



# 编译安装Apache

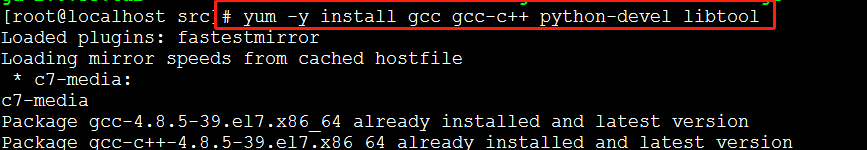
## 上传源码

使用xshell将LAMP-64.ISO上传到/root，挂载源码光盘，把源码拷贝到/usr/src/目录下



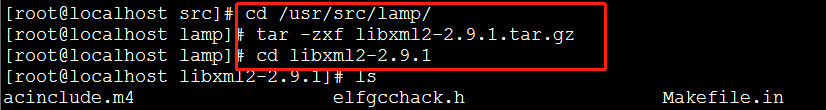
## 编译环境

安装gcc和gcc-c++ python-devel libtool编译环境

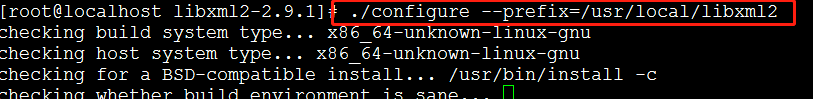


## 安装libxml2

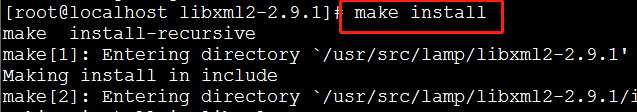
进入/usr/src/lamp目录下，解压libxml2源码压缩包



进入到libxml2-2.9.1目录下，执行下图配置命令，指定安装路径为/usr/local/libxml2

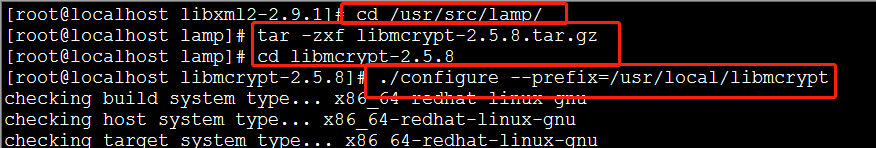


执行make install 将libxml安装到/usr/local/libxml2

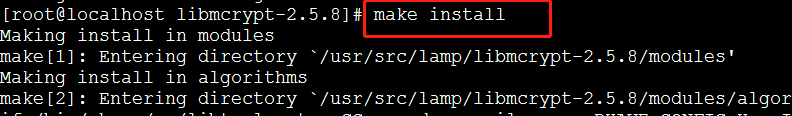


## 安装libmcrypt

解压libmcrypt源码，进入目录，执行configure配置安装到/usr/local/libmcrypt

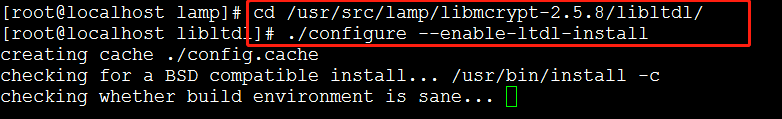


使用make install安装libmcrypt到/usr/local/libmcrypt目录

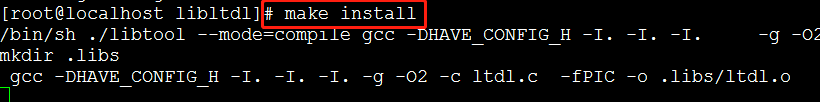


## 安装libltdl

在libmcrypt目录下进入到libltdl，不是上级目录

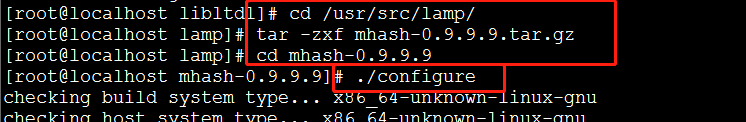


使用make install安装到/usr/local

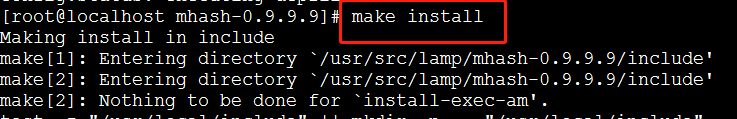


## 安装mhash

进入到/usr/src/lamp目录下，解压mhash源码，进入目录，执行configure脚本

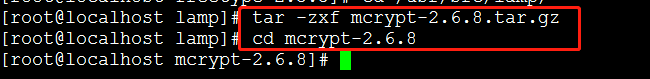


执行make install 安装到/usr/local

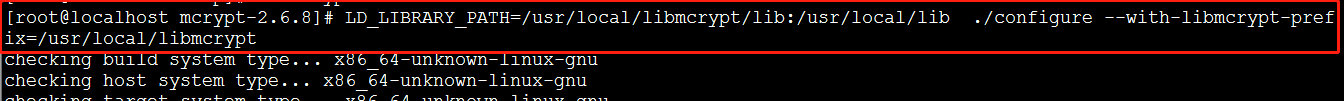


## 安装mcrypt

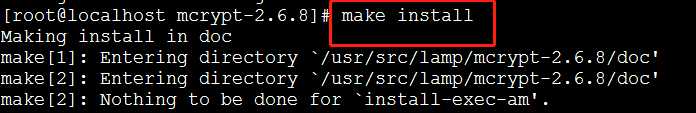
进入到/usr/src/lamp目录下，解压mcrypt进入目录，执行下面命令



LD\_LIBRARY\_PATH=/usr/local/libmcrypt/lib:/usr/local/lib ./configure --with-libmcrypt-prefix=/usr/local/libmcrypt

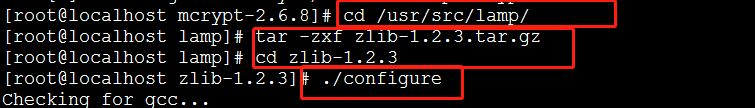


执行make install安装到/usr/local/libmcrypt

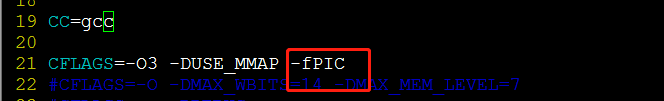


## 安装zlib

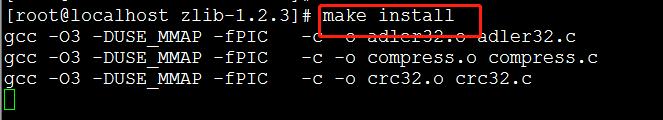
进入到/usr/src/lamp目录下，解压zlib进入目录，执行下面命令



使用vim Makefile打开文件，在第21行增加-fPIC

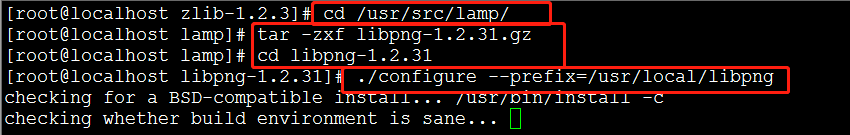


使用make install安装到/usr/local/

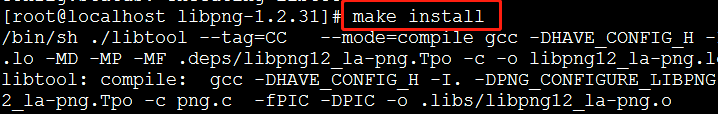


## 安装libpng

进入到/usr/src/lamp目录下，解压libpng进入目录，执行下面命令

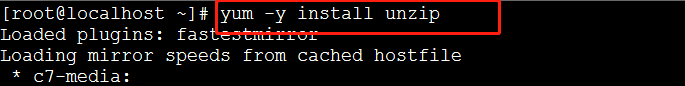


使用make install安装到/usr/local/libpng目录

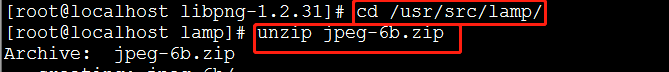


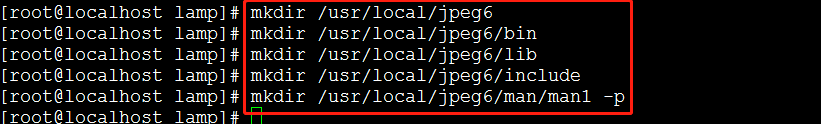
## 安装libjpeg

安装解压zip的软件unzip

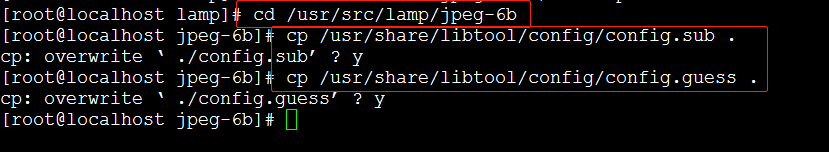


进入到/usr/src/lamp目录下，解压libpng进入目录

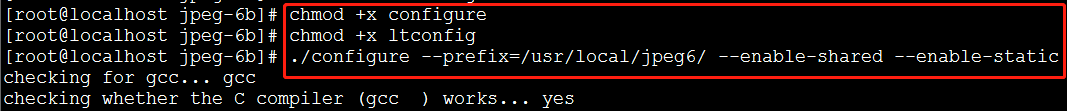
创建下面目录



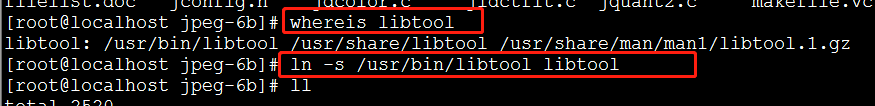
使用下面命令拷贝配置文件



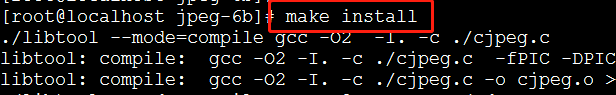
执行下面命令配置编译环境



使用whereis查看libtool命令路径，使用ln制作软链接到libtool路径



执行make install安装到/usr/local/jpeg6

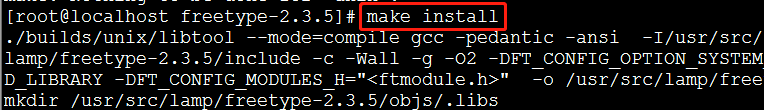


## 安装freetype

进入到/usr/src/lamp目录下，解压freetype进入目录，执行命令配置安装

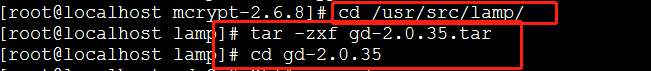


执行make install安装到/usr/local/freetype



## 安装gd

进入到/usr/src/lamp目录下，解压gd进入目录

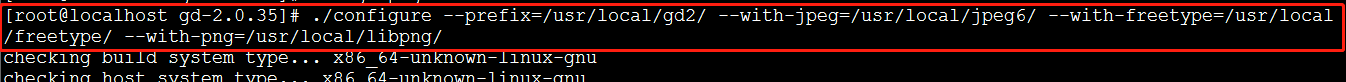


使用vim gd\_png.c打开文件，更改第16行如下图所示

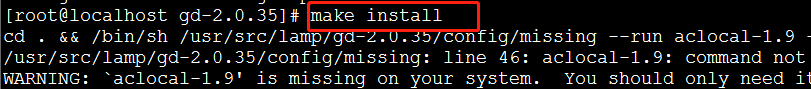


使用下面命令配置编译环境

./configure --prefix=/usr/local/gd2/ --with-jpeg=/usr/local/jpeg6/ --with-freetype=/usr/local/freetype/ --with-png=/usr/local/libpng/

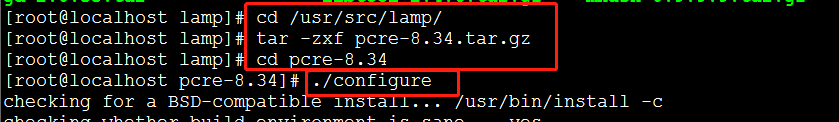


使用make install 安装到/usr/local/gd2

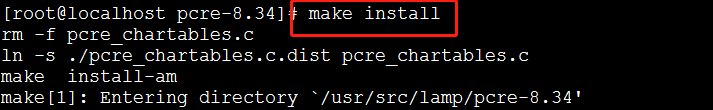


## 安装pcre

进入到/usr/src/lamp目录下，解压pcre进入目录，执行configure配置编译环境

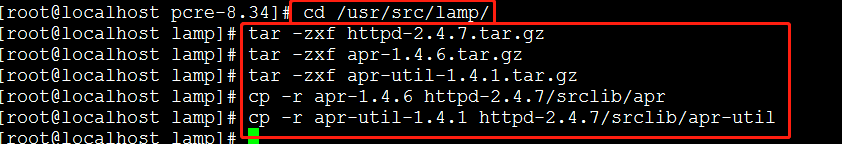


执行make install安装到/usr/local



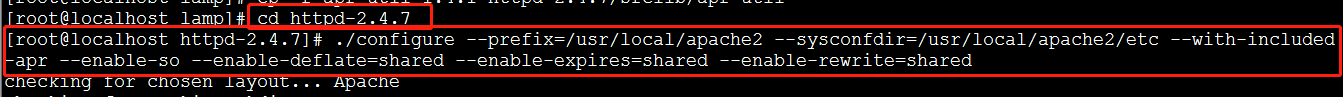
## 安装apache

执行下面命令构建源码目录

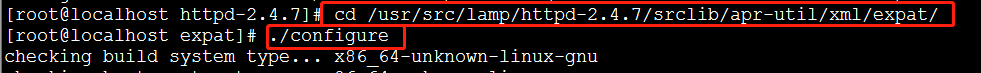


使用下面命令配置编译环境

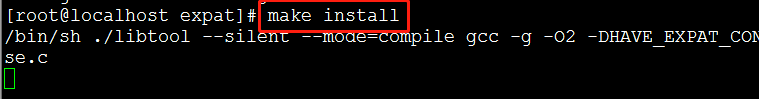
./configure --prefix=/usr/local/apache2 --sysconfdir=/usr/local/apache2/etc --with-included-apr --enable-so --enable-deflate=shared --enable-expires=shared --enable-rewrite=shared



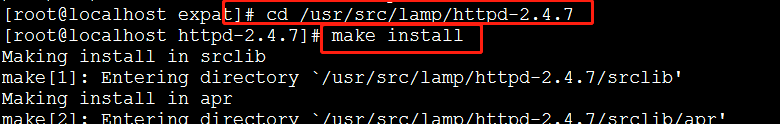
进入到expat编译安装expat源码



安装expat到/usr/local下

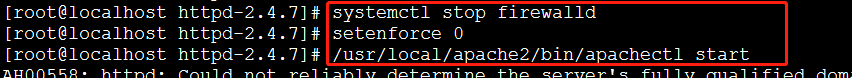


进入Apache源码目录，make install安装到/usr/local/apache2

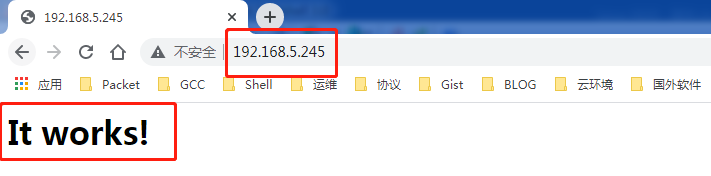


## 测试apache

关闭防护墙和selinux，启动apache

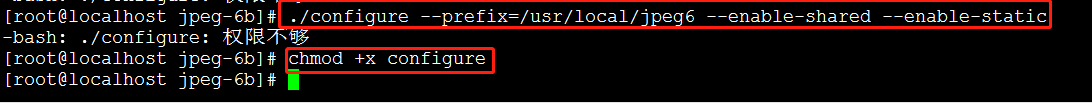


浏览器访问服务器IP，出现it works表示成功

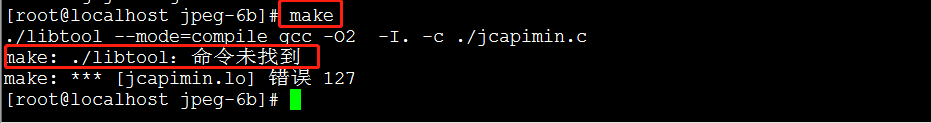


## 错误总结

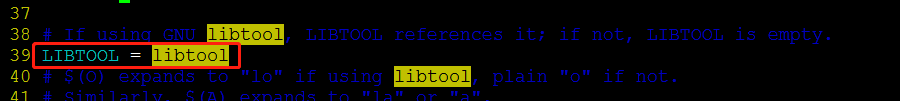
编译jpeg6b时出现权限不够的提示，使用chmod +x configure增加权限即可



使用make编译源码，提示libtool未找到



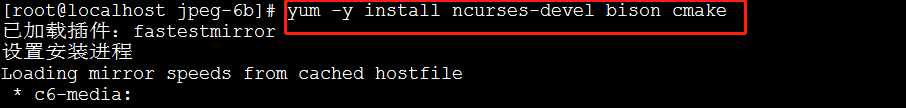
使用vim打开Mafilefile文件，在第39行将./libtool改为libtool



# 编译安装MySQL

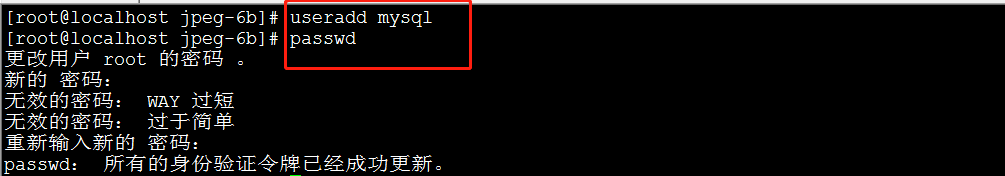
## Yum安装依赖

使用yum 安装ncurses-devel bison cmake依赖项



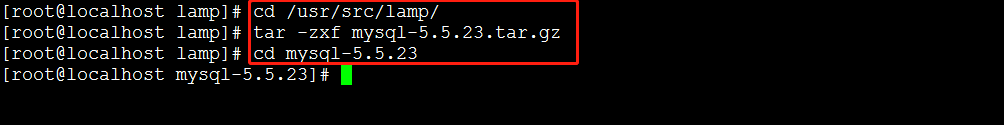
## 创建mysql用户

使用useradd创建mysql用户，并设置密码



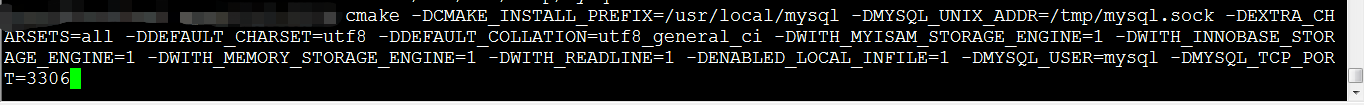
## 编译mysql源码

进入到/usr/src/lamp目录下，解压mysql并进入目录



使用下面命令配置编译环境

cmake -DCMAKE\_INSTALL\_PREFIX=/usr/local/mysql -DMYSQL\_UNIX\_ADDR=/tmp/mysql.sock -DEXTRA\_CHARSETS=all -DDEFAULT\_CHARSET=utf8 -DDEFAULT\_COLLATION=utf8\_general\_ci -DWITH\_MYISAM\_STORAGE\_ENGINE=1 -DWITH\_INNOBASE\_STORAGE\_ENGINE=1 -DWITH\_MEMORY\_STORAGE\_ENGINE=1 -DWITH\_READLINE=1 -DENABLED\_LOCAL\_INFILE=1 -DMYSQL\_USER=mysql -DMYSQL\_TCP\_PORT=3306

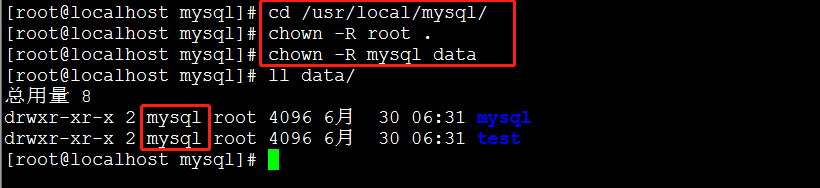


使用make install安装到/usr/local/mysql/目录下



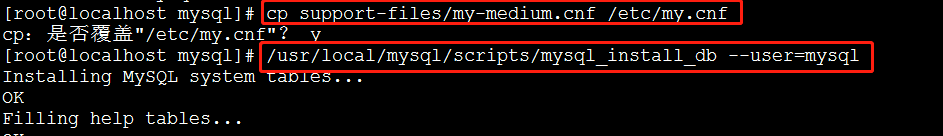
## 设置属主

进入到/usr/local/mysql目录下，将data目录属主改为mysql

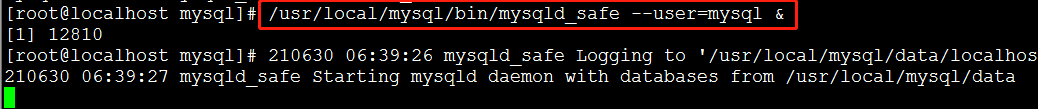


## 初始化数据库

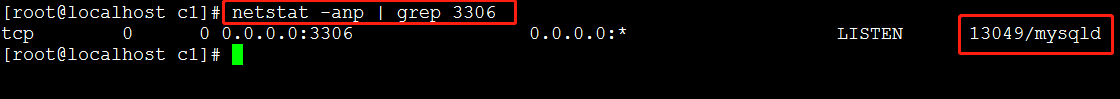
拷贝配置文件到/etc，使用mysql\_install\_db初始化数据库



使用mysqld\_safe运行mysql数据库

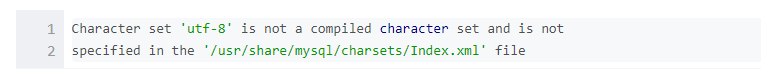


使用netstat查看3306端口是否被监听



## 排错

出现下面报错，导致mysql不能启动，原因为上面的cmake长命令输错



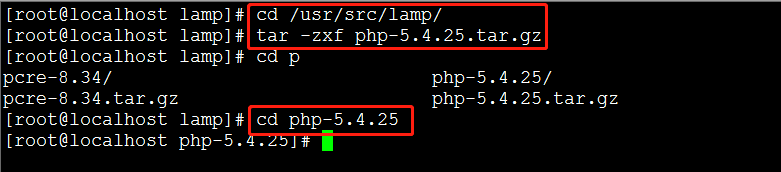
主要错误为把mysql中的l写成了数字mysq1，下面为正确的配置命令

cmake -DCMAKE\_INSTALL\_PREFIX=/usr/local/mysql -DMYSQL\_UNIX\_ADDR=/tmp/mysql.sock -DEXTRA\_CHARSETS=all -DDEFAULT\_CHARSET=utf8 -DDEFAULT\_COLLATION=utf8\_general\_ci -DWITH\_MYISAM\_STORAGE\_ENGINE=1 -DWITH\_INNOBASE\_STORAGE\_ENGINE=1 -DWITH\_MEMORY\_STORAGE\_ENGINE=1 -DWITH\_READLINE=1 -DENABLED\_LOCAL\_INFILE=1 -DMYSQL\_USER=mysql -DMYSQL\_TCP\_PORT=3306

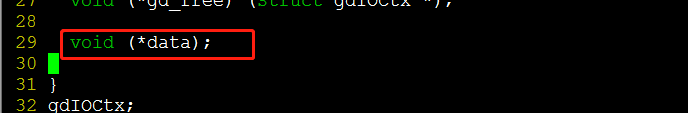
# 编译安装PHP

## 编译源码

解压PHP进入目录

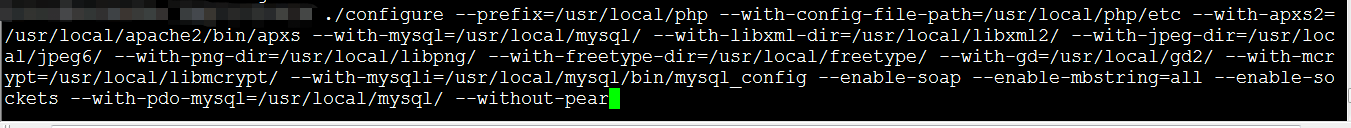


使用vim /usr/local/gd2/include/gd\_io.h打开文件，在第29行添加下面内容

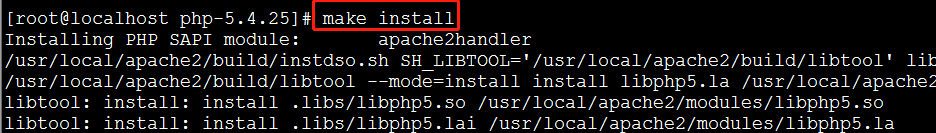


使用下面命令配置编译环境

./configure --prefix=/usr/local/php --with-config-file-path=/usr/local/php/etc --with-apxs2=/usr/local/apache2/bin/apxs --with-mysql=/usr/local/mysql/ --with-libxml-dir=/usr/local/libxml2/ --with-jpeg-dir=/usr/local/jpeg6/ --with-png-dir=/usr/local/libpng/ --with-freetype-dir=/usr/local/freetype/ --with-gd=/usr/local/gd2/ --with-mcrypt=/usr/local/libmcrypt/ --with-mysqli=/usr/local/mysql/bin/mysql\_config --enable-soap --enable-mbstring=all --enable-sockets --with-pdo-mysql=/usr/local/mysql/ --without-pear

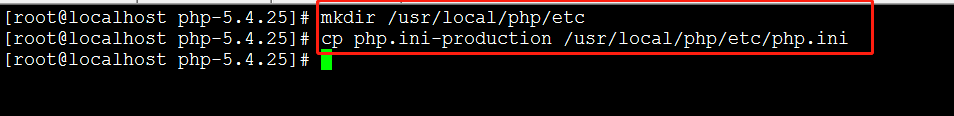


使用make install安装到/usr/local/php



## 配置PHP

创建目录/usr/local/php/etc，拷贝配置文件php.ini

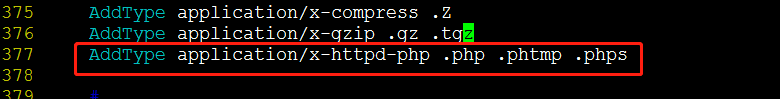


使用vim /usr/local/php/etc/php.ini，更改212行配置



## 配置Apache

使用vim /usr/local/apache2/etc/httpd.conf打开配置文件，在377行添加下图内容

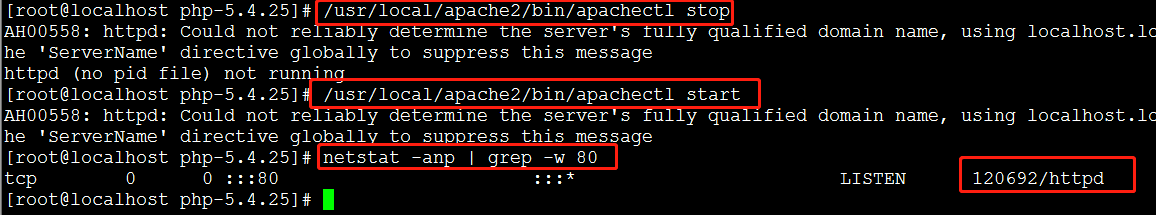


在248行添加下图内容

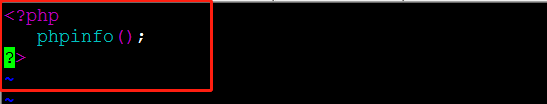


## 测试Apache和php

按下图所示，重启Apache，查看80端口是否监听，关闭防火墙和selinux



vim /usr/local/apache2/htdocs/test.php，输入下图内容，测试php和apache之间是否正常

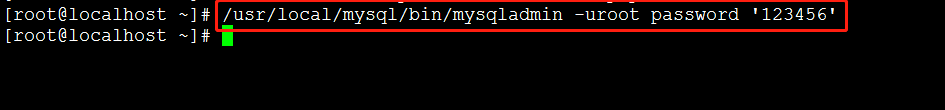


打开浏览器访问[IP]/test.php，出现php版本证明成功

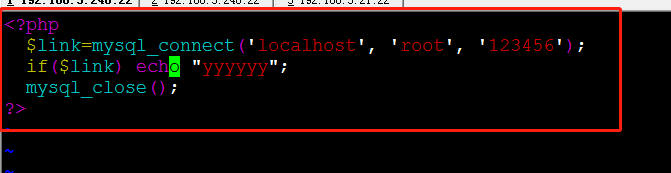


## 测试PHP和mysql

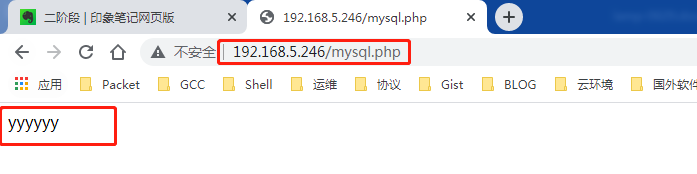
更改mysql密码为123456



使用vim /usr/local/apache2/htdocs/mysql.php打开文件，输入下面内容



打开浏览器访问[IP]/mysql.php，出现yyyyyy证明成功



## 排错

提示can not found libxml2-config，please check your libxml2 install报错，导致不能继续make编译提示找不到makefile

因为libxml2编译的时候把libxml2中的字母l写成了数字1

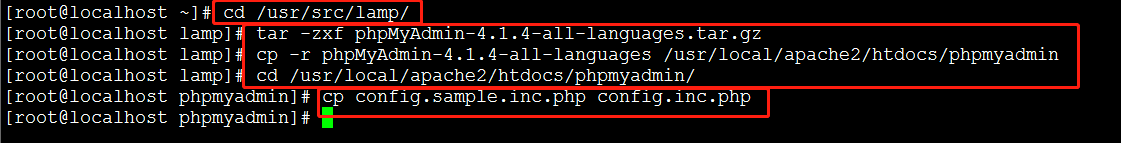
删除libxml2的源代码和删除/usr/local/libxm\* ，重新编译安装libxml2，正确的配置命令：

./configure --prefix=/usr/local/php --with-config-file-path=/usr/local/php/etc --with-apxs2=/usr/local/apache2/bin/apxs --with-mysql=/usr/local/mysql/ --with-libxml-dir=/usr/local/libxml2/ --with-jpeg-dir=/usr/local/jpeg6/ --with-png-dir=/usr/local/libpng/ --with-freetype-dir=/usr/local/freetype/ --with-gd=/usr/local/gd2/ --with-mcrypt=/usr/local/libmcrypt/ --with-mysqli=/usr/local/mysql/bin/mysql\_config --enable-soap --enable-mbstring=all --enable-sockets --with-pdo-mysql=/usr/local/mysql/ --without-pear

# 搭建phpmyadmin

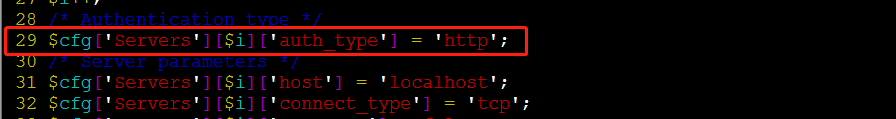
## 解压拷贝

按下图所示，解压拷贝phpmyadmin到/usr/local/apache2/htdocs下



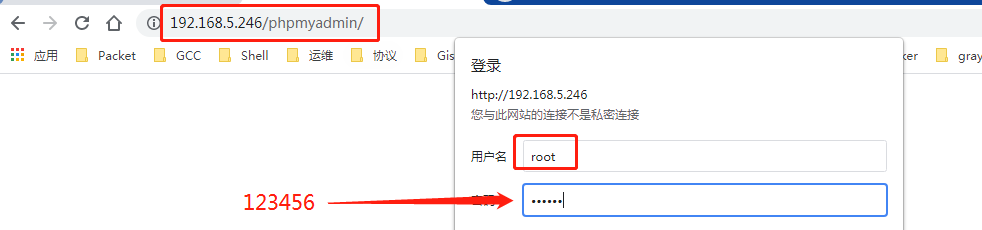
## 编辑配置

使用vim /usr/local/apache2/htdocs/phpmyadmin/config.inc.php按下图所示修改配置

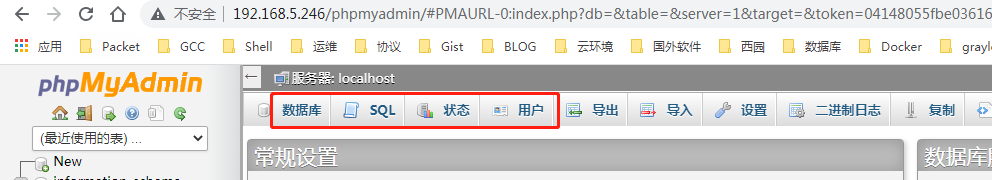


## 测试

浏览器访问[IP]/phpmyadmin，输入用户名密码



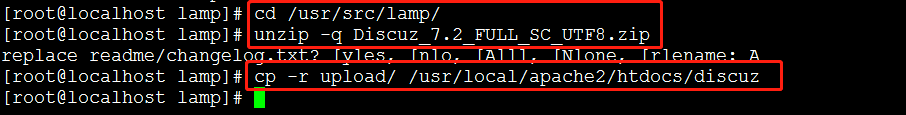
登录后出现下图界面证明成功



# 搭建论坛

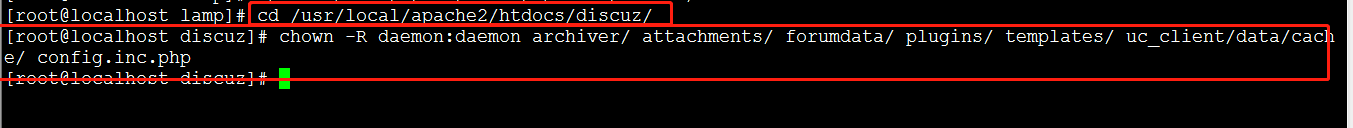
## 解压拷贝

解压拷贝论坛源码到/usr/local/apache2/htdocs/目录，按下图所示



## 更改属主

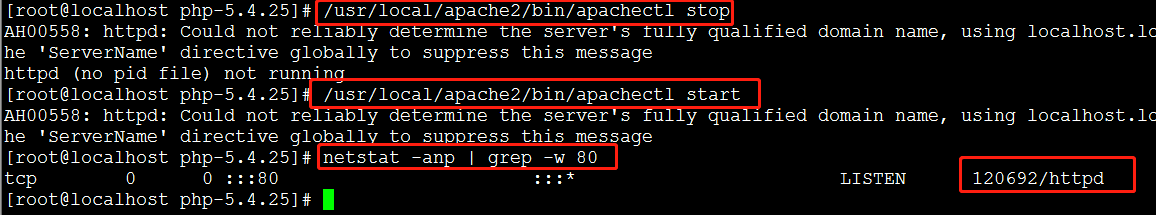
进入discuz目录，更改目录属主



## 更改配置

## 测试

按下图所示，重启Apache，查看80端口是否监听，关闭防火墙和selinux



浏览器访问[IP]/discus/install/，出现安装界面成功

