你只有非常努力，才能看起来毫不费力！

# LAMP 编译环境搭建

yum -y install gcc

yum install gcc-c++

# Apache 安装

## 安装 zlib

[root@localhost resource]# tar -xf zlib-1.2.5.tar.gz

[root@localhost resource]# cd zlib-1.2.5

[root@localhost zlib-1.2.5]# ./configure

[root@localhost zlib-1.2.5]# make && make install

**提示：zlib是提供数据压缩用的函式库。**

## 安装 apr

[root@localhost zlib-1.2.5]# cd ..

[root@localhost resource]# tar -xf apr-1.5.2.tar.gz

[root@localhost resource]# cd apr-1.5.2

[root@localhost apr-1.5.2]# ./configure --prefix=/usr/local/apr

[root@localhost apr-1.5.2]# make

[root@localhost apr-1.5.2]# make install

## 安装 apr-iconv

[root@localhost apr-1.5.2]# cd ..

[root@localhost resource]# tar -xf apr-iconv-1.2.1.tar.gz

[root@localhost resource]# cd apr-iconv-1.2.1

[root@localhost apr-iconv-1.2.1]# ./configure --prefix=/usr/local/apr-iconv --with-apr=/usr/local/apr/

[root@localhost apr-iconv-1.2.1]# make && make install

## 安装 apr-util

[root@localhost apr-iconv-1.2.1]# cd ..

[root@localhost resource]# tar -xf apr-util-1.5.4.tar.gz

[root@localhost resource]# cd apr-util-1.5.4

[root@localhost apr-util-1.5.4]# ./configure --prefix=/usr/local/apr-util/ --with-apr=/usr/local/apr/ --with-apr-iconv=/usr/local/apr-iconv/bin/apr-iconv

[root@localhost apr-util-1.5.4]# make && make install

**提示：APR(Apache portable Run-time libraries，Apache可移植运行库)的目的如其名称一样，主要为上层的应用程序提供一个可以跨越多操作系统平台使用的底层支持接口库，Apache需要使用APR，完整的APR实际上包含了三个开发包：apr、apr-util以及apr-iconv。**

## 安装 pcre

[root@localhost apr-util-1.5.4]# cd ..

[root@localhost resource]# tar -xf pcre-8.35.tar.gz

[root@localhost resource]# cd pcre-8.35

[root@localhost pcre-8.35]# ./configure --prefix=/usr/local/pcre && make && make install

make && make install

**注意：PCRE(Perl Compatible Regular Expressions)是一个Perl库，包括 perl 兼容的正则表达式库。**

## 安装 httpd

[root@localhost pcre-8.35]# cd ..

[root@localhost resource]# tar xf httpd-2.4.17.tar.gz

[root@localhost resource]# cd httpd-2.4.17

[root@localhost httpd-2.4.17]# ~~./configure --prefix=/usr/local/lamp/apache2 \~~

~~> --enable-modules=all \~~

~~> --enable-so \~~

~~> --with-apr=/usr/local/apr/ \~~

~~> --with-apr-util=/usr/local/apr-util/ \~~

~~> --with-pcre=/usr/local/pcre/~~

从编译参数.txt复制

[root@localhost httpd-2.4.17]# make && make install

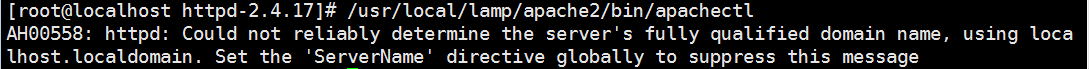
## 测试 Apache

ps -A | grep httpd

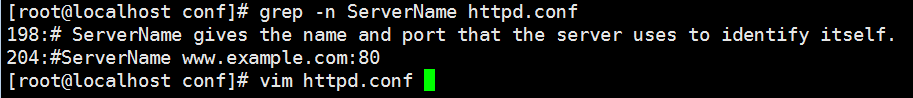
如果结果为空,表示没有启动,启动方式如下:

[root@localhost httpd-2.4.17]# /usr/local/lamp/apache2/bin/apachectl

如下情况：



解决方案：

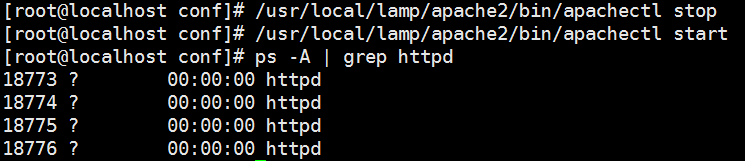




修改为：



接下来：



在虚拟机访问：

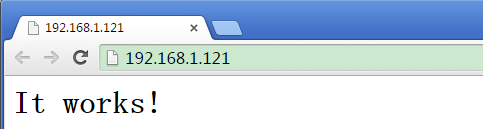
在浏览器打开 http://192.168.1.121



在其它机器访问:

在浏览器打开 http://192.168.1.121

如果看到

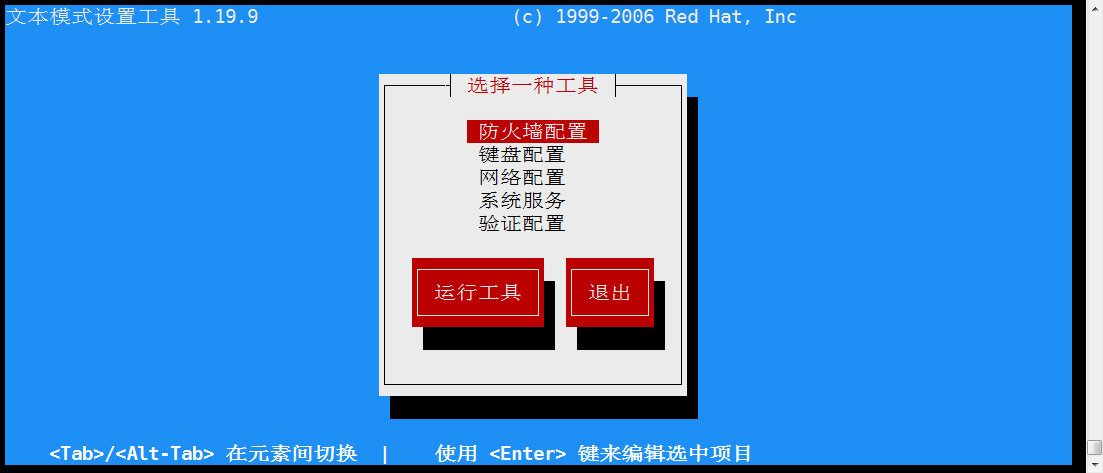


表示成功了

如果不能看到,一般来说是防火墙禁止了,应该开启虚拟机的防火墙

防火墙配置

[root@localhost httpd-2.4.17]# setup







然后重启防火墙？

应该怎么重新启动？

# 安装 PHP

## 安装 xml

[root@localhost httpd-2.4.17]# cd ..

[root@localhost resource]# tar xf libxml2-2.7.2.tar.gz

[root@localhost resource]# cd libxml2-2.7.2

[root@localhost libxml2-2.7.2]# ./configure

[root@localhost libxml2-2.7.2]# make && make install

**提示：XML库是一种支持对XML（标准通用标记语言下的一个应用）格式文档进行存储和查询等操作的数据管理系统。在系统中，开发人员可以对数据库中的XML文档进行查询、导出和指定格式的序列化。**

## 安装 jpeg

[root@localhost libxml2-2.7.2]# cd ..

[root@localhost jpeg-8b]# tar xf jpegsrc.v8b.tar.gz

[root@localhost jpeg-8b]# cd jpeg-8b/

[root@localhost jpeg-8b]# ./configure --prefix=/usr/local/jpeg --enable-shared --enable-static

[root@localhost jpeg-8b]# make && make install

## 安装 png

[root@localhost jpeg-8b]# cd ..

[root@localhost resource]# tar xf libpng-1.4.3.tar.gz

[root@localhost resource]# cd libpng-1.4.3

[root@localhost libpng-1.4.3]# ./configure --prefix=/usr/local/png --enable-shared --enable-static

[root@localhost libpng-1.4.3]# make && make install

## 安装 freetype 字体库

[root@localhost libpng-1.4.3]# cd ..

[root@localhost resource]# tar xf freetype-2.4.1.tar.gz

[root@localhost resource]# cd freetype-2.4.1

[root@localhost freetype-2.4.1]# ./configure --prefix=/usr/local/freetype --enable-shared

[root@localhost freetype-2.4.1]# make && make install

## 安装 gd

[root@localhost freetype-2.4.1]# cd ..

[root@localhost resource]# tar xf libgd-2.1.1.tar.gz

[root@localhost resource]# cd libgd-2.1.1

[root@localhost libgd-2.1.1]# ./configure --prefix=/usr/local/gd --with-jpeg=/usr/local/jpeg/ --with-png=/usr/local/png --with-zlib --with-freetype=/usr/local/freetype

[root@localhost libgd-2.1.1]# make && make install

**提示：GD库，是php处理图形的扩展库。**

依赖关系解决:

## 安装 bison

[root@localhost gd-2.0.35]# rpm -ivh /mnt/cdrom/Packages/bison-2.4.1-5.el6.i686.rpm

## 安装 autoconf

[root@localhost mcrypt-2.6.8]# cd ..

[root@localhost resource]# tar xf autoconf-2.69.tar.gz

[root@localhost resource]# cd autoconf-2.69

[root@localhost autoconf-2.69]# ./configure && make && make install

**提示：Autoconf是一个用于包，以适应多种Unix类系统的 shell脚本的工具。**

## 安装libXpm-devel

[root@localhost resource]# yum install libXpm-devel

## 安装 php

[root@localhost libiconv-1.14]# cd ..

[root@localhost lamp]# tar xvf php-5.6.24.tar.gz

[root@localhost lamp]# cd php-5.6.24

[root@localhost php-5.6.24]# ~~./configure --prefix=/usr/local/lamp/php \~~

~~> --with-apxs2=/usr/local/lamp/apache2/bin/apxs \~~

~~> --with-mysql=mysqlnd \~~

~~> --with-pdo-mysql=mysqlnd \~~

~~> --with-mysqli=mysqlnd \~~

~~> --with-freetype-dir=/usr/local/freetype \~~

~~> --with-gd=/usr/local/gd/ \~~

~~> --with-zlib \~~

~~> --with-jpeg-dir=/usr/local/jpeg \~~

~~> --with-png-dir=/usr/local/png \~~

~~> --enable-mbstring=all \~~

~~> --enable-mbregex \~~

~~> --enable-shared \~~

~~> --with-libxml-dir=/usr/local~~

~~> --with-xpm-dir=/usr/lib~~

到配置参数.txt文件复制

[root@localhost php-5.6.24]# make

[root@localhost php-5.6.24]# make install

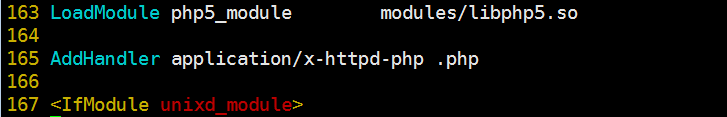
不开心、不开心、等了这么久。。。。。。

### 拷贝PHP配置文件

[root@localhost php-5.6.24]# cp php.ini-development /usr/local/lamp/php/lib/php.ini

### 让 Apache 支持 php

[root@localhost php-5.6.24]# vim /usr/local/lamp/apache2/conf/httpd.conf



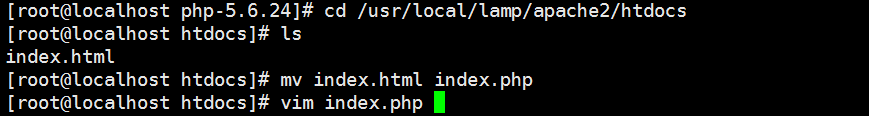
AddHandler application/x-httpd-php .php

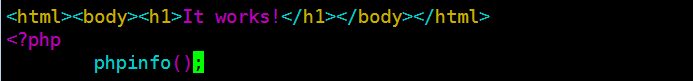
重启

[root@localhost php-5.6.24]# killall httpd

[root@localhost php-5.6.24]# /usr/local/lamp/apache2/bin/apachectl

测试





# 安装 MySQL

## 安装依赖

### 安装 ncurses-devel

[root@localhost php-5.6.24]# rpm -ivh /mnt/cdrom/Packages/ncurses-devel-5.7-3.20090208.el6.i686.rpm

### 安装 cmake

[root@localhost mysql-5.6.25]# rpm -ivh /mnt/cdrom/Packages/cmake-2.6.4-5.el6.i686.rpm

## 安装 MySQL

[root@localhost php-5.6.24]# cd ..

[root@localhost lamp]# tar xvf mysql-5.6.25.tar.gz

[root@localhost lamp]# cd mysql-5.6.25

[root@localhost mysql-5.6.25]# ~~cmake - \ DCMAKE\_INSTALL\_PREFIX=/usr/local/lamp/mysql -DMYSQL\_UNIX\_ADDR=/tmp/mysql.sock~~

~~-DDEFAULT\_CHARSET=utf8 -DDEFAULT\_COLLATION=utf8\_general\_ci -DWITH\_EXTRA\_CHARSETS:STRING=utf8,gbk \~~

~~-DWITH\_MYISAM\_STORAGE\_ENGINE=1 -DWITH\_INNOBASE\_STORAGE\_ENGINE=1 -DWITH\_READLINE=1 -DENABLED\_LOCAL\_INFILE=1 \~~

~~-DMYSQL\_DATADIR=/usr/local/lamp/mysql/data -DMYSQL\_USER=mysql~~

从配置参数.txt文件复制

[root@localhost mysql-5.6.25]# make && make install

## 添加用户

用户无需登录系统，所以指定用户登陆后使用的 shell 是 nologin，禁止登录，-r 表示系统账号，系统账号 uid 较小

[root@localhost mysql-5.6.25]# useradd -s /sbin/nologin -r mysql

## 修改目录权限，将 data 文件夹所有者改为 mysql

data 文件夹用来存放数据，必须保证 mysql 用户可以写操作，由于整个安装过程都是 root 用户执行，所以其它文件夹都是输入 root 用户

[root@localhost mysql-5.6.25]# chown mysql.mysql /usr/local/lamp/mysql/data -R

## 复制配置文件

[root@localhost mysql-5.6.25]# cp support-files/my-default.cnf /etc/my.cnf

## 创建 mysql 测试数据库和系统数据库

[root@localhost mysql-5.6.25]# cd /usr/local/lamp/mysql

[root@localhost mysql]# /usr/local/lamp/mysql/scripts/mysql\_install\_db --user=mysql --basedir=/usr/local/lamp/mysql --datadir=/usr/local/lamp/mysql/data &

## 测试

[root@localhost mysql]# /usr/local/lamp/mysql/bin/mysqld\_safe

**上面这一行是启动MySQL服务进程的。**

[root@localhost mysql]# /usr/local/lamp/mysql/bin/mysql -uroot

**上面这一行是登陆mysql的。**

**mysql>** show databases;

## 修改密码，增加安全性

**mysql>** SET PASSWORD FOR 'root'@'localhost' = PASSWORD('newpass');

## 创建允许远程登录的用户

**给远程root用户授权：**

**mysql>** select password('654321');

**mysql>** GRANT ALL PRIVILEGES ON \*.\* TO 'root'@'%' IDENTIFIED BY PASSWORD '\*2A032F7C5BA932872F0F045E0CF6B53CF702F2C5';

**给远程yunchao用户授权：**

**mysql>** GRANT ALL PRIVILEGES ON \*.\* TO 'yunchao'@'%' IDENTIFIED BY ‘654321’;

在防火墙中允许外部机器访问 3306 端口





重启防火墙

# service 管理

便于通过 service 管理复制启动文件

## Apache

由于 gnome 需要 Apache,所以无法直接拆卸,使用删除启动文件的方式废弃默认的 Apache 服务器

**centos6.5自带的httpd 在 /etc/init.d/httpd**

[root@localhost lamp]# rm -rf /etc/init.d/httpd

[root@localhost lamp]# cp /usr/local/lamp/apache2/bin/apachectl /etc/init.d/httpd

**service httpd start ----->> /etc/init.d/httpd start**

## MySQL

[root@localhost mysql]# cp /usr/local/lamp/mysql/support-files/mysql.server /etc/init.d/mysqld # 复制启动文件

[root@localhost mysql]# chmod +x /etc/init.d/mysqld #添加可执行权限

做了上面的步骤就可以使用 service mysqld 来控制 mysql 服务器了

比如：service mysqld start 可以启动 mysql 服务器

切换至将服务设置为开机自启动