

实验 04：通过 WordPress 建设网站

一、实验目的

- 1、了解 LAMP；
- 2、掌握使用 WordPress 建设网站。

二、实验学时

2 学时

三、实验类型

综合性



四、实验需求

1、硬件

每人配备计算机 1 台。

2、软件

安装 VMware WorkStation Pro 或 Oracle VM VirtualBox 软件，安装 Mobaxterm 软。

3、网络

本地主机与虚拟机能够访问互联网，虚拟机网络不使用 DHCP 服务。

4、工具

预先下载 WordPress 安装程序。

五、实验任务

- 1、完成 LAMP 环境的部署；
- 2、完成通过 WordPress 建设并发布个人网站。

六、实验内容及步骤

- 1、本实验需要 VM 1 台。
- 2、本实验 VM 配置信息如下表所示。

虚拟机配置	操作系统配置
虚拟机名称：VM-Lab-04-Task-01-172.31.0.141 内存：2GB CPU：1 颗，1 核心 虚拟磁盘：20GB 网卡：1 块，NAT	主机名：Lab-04-Task-01 IP 地址：172.31.0.141 子网掩码：255.255.255.0 网关：172.31.0.254 DNS：172.31.0.254

- 3、本实验拓扑图。

无

4、本实验操作演示视频。

本实验操作演示视频为视频集的第 4 集：

<https://www.bilibili.com/video/BV1iH4y1c7ft?p=4>

七、实验内容及步骤

1、创建虚拟机

根据规划创建虚拟机，使用 MobaXterm 远程连接虚拟机，具体步骤参考《实验 01：安装 openEuler》。

2、配置安全策略

配置 SELINUX 和防火墙策略。

```
# 配置防火墙, 允许 80-82/tcp 端口访问
[root@Lab-04-Task-01 ~]# firewall-cmd --zone=public --add-port=80-82/tcp --permanent
[root@Lab-04-Task-01 ~]# firewall-cmd --reload
[root@Lab-04-Task-01 ~]# firewall-cmd --list-all

# 配置 SELinux, 允许 80-82 端口提供服务
# 安装 SELinux 的配置工具 semanage
[root@Lab-04-Task-01 ~]# yum install -y policycoreutils-python-utils

# 查看 SELinux 允许通过的 HTTP 端口(默认支持 80、81)
[root@Lab-04-Task-01 ~]# semanage port -l | grep http
# -----允许的 HTTP 端口-----
http_cache_port_t          tcp      8080, 8118, 8123, 10001-10010
http_cache_port_t          udp      3130
http_port_t                 tcp      80, 81, 443, 488, 8008, 8009, 8443, 9000
pegasus_http_port_t        tcp      5988
pegasus_https_port_t       tcp      5989
# -----允许的 HTTP 端口-----

# 增加 82 端口的访问策略
[root@Lab-04-Task-01 ~]# semanage port -a -t http_port_t -p tcp 82

# 重新启动
[root@Lab-04-Task-01 ~]# reboot
```

3、安装 Apache 服务

```
# 安装 Apache HTTP Server
[root@Lab-04-Task-01 ~]# yum install -y httpd

# 启动 Apache 服务, 设置开机自启
[root@Lab-04-Task-01 ~]# systemctl start httpd
```

```
[root@Lab-04-Task-01 ~]# systemctl enable httpd

# 查看 Apache 服务状态
[root@Lab-04-Task-01 ~]# systemctl status httpd

# 使用浏览器访问 http://172.31.0.141, 出现欢迎界面, 说明 Apache Httpd
Server 安装成功
```

4、安装 PHP

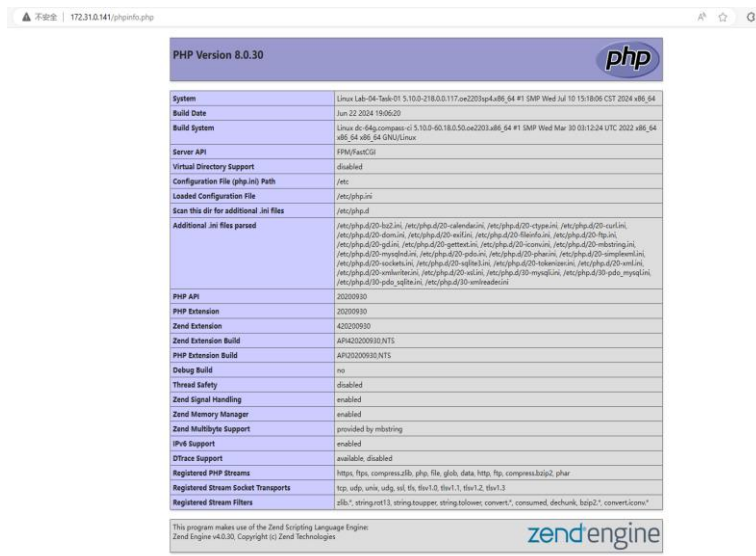
```
# 更新 yum 源
[root@Lab-04-Task-01 ~]# yum update -y
# 删除缓存中无用软件包并更新软件源
[root@Lab-04-Task-01 ~]# yum clean all
[root@Lab-04-Task-01 ~]# yum makecache

# 下载安装 PHP 相应的模块
[root@Lab-04-Task-01 ~]# yum install -y php php-curl php-dom php
p-exif php-fileinfo php-fpm php-gd php-hash php-mbstring php-my
sqli php-openssl php-pcre php-xml libsodium
# 查看 PHP 版本
[root@Lab-04-Task-01 ~]# php -v
```

验证 php 环境。

```
# 在 Apache 网站的根目录创建测试文件
[root@Lab-04-Task-01 ~]# echo "<?php phpinfo(); ?>" > /var/www/
html/phpinfo.php
# 重启 Apache 服务
[root@Lab-04-Task-01 ~]# systemctl restart httpd
```

通过浏览器访问 <http://172.31.0.141/phpinfo.php>, 出现如图 4-1 所示, 表示环境搭建成功。



PHP Version 8.0.30	
System	Linux Lab-04-Task-01 5.10.0-218.0.117.el220ppc64le_64 #1 SMP Wed Jul 10 15:18:09 CST 2024 x86_64
Build Date	Jun 22 2024 19:06:29
Build System	Linux 4i-64g-composer-1 5.10.0-60.18.0.el203.x86_64 #1 SMP Wed Mar 30 08:12:34 UTC 2022 x86_64 x86_64 x86_64 GNU/Linux
Server API	FPM/FastCGI
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc
Loaded Configuration File	/etc/php.ini
Scan this dir for additional ini files	/etc/php.d
Additional ini files parsed	/etc/php.d/20-bcmath.ini, /etc/php.d/20-calendar.ini, /etc/php.d/20-ctype.ini, /etc/php.d/20-curl.ini, /etc/php.d/20-dom.ini, /etc/php.d/20-gd.ini, /etc/php.d/20-gettext.ini, /etc/php.d/20-iconv.ini, /etc/php.d/20-igmp.ini, /etc/php.d/20-intl.ini, /etc/php.d/20-ldap.ini, /etc/php.d/20-ldap_sasl.ini, /etc/php.d/20-mbstring.ini, /etc/php.d/20-mcrypt.ini, /etc/php.d/20-mysqlnd.ini, /etc/php.d/20-openssl.ini, /etc/php.d/20-pdo.ini, /etc/php.d/20-pdo_mysql.ini, /etc/php.d/20-pdo_odbc.ini, /etc/php.d/20-pdo_sqlite.ini, /etc/php.d/20-sockets.ini, /etc/php.d/20-tokenizer.ini, /etc/php.d/20-xml.ini, /etc/php.d/20-xmlrpc.ini, /etc/php.d/20-zip.ini, /etc/php.d/20-zlib.ini
PHP API	20200930
PHP Extension	20200930
Zend Extension	42020930
Zend Extension Build	API620200930.NTS
PHP Extension Build	API620200930.NTS
Debug Build	no
Thread Safety	disabled
Zend Signal Handling	enabled
Zend Memory Manager	enabled
Zend Multibyte Support	provided by mbstring
IPv6 Support	enabled
DTrace Support	available, disabled
Registered PHP Streams	https, ftps, compress.zlib, php, file, glob, data, http, ftp, compress.bzip2, phar
Registered Stream Socket Transports	tcp, udp, unix, udg, ssl, tls, tlsv1.0, tlsv1.1, tlsv1.2, tlsv1.3
Registered Stream Filters	zlib.*, string.rot13, string.toupper, string.tolower, convert.*, consumed, dechunk, bzip2.*, convert.iconv.*

This program makes use of the Zend Scripting Language Engine
 Zend Engine v4.0.30, Copyright (c) Zend Technologies

zendengine

图 4-1 验证 php 环境搭建成功界面

```
[root@Lab-04-Task-01 ~]# rm -rf /var/www/html/phpinfo.php
```

5、安装 MySQL

```
# 安装 MySQL
[root@Lab-04-Task-01 ~]# yum install -y mysql-server
# 查看 MySQL 的版本号
[root@Lab-04-Task-01 ~]# mysql -V
mysql Ver 8.0.29 for Linux on x86_64 (Source distribution)

# 启动 MySQL
[root@Lab-04-Task-01 ~]# systemctl start mysqld

# 设置 MySQL 开机自启动
[root@Lab-04-Task-01 ~]# systemctl enable mysqld
[root@Lab-04-Task-01 ~]# systemctl daemon-reload

# 查看 MySQL 的初始密码
[root@Lab-04-Task-01 ~]# grep "password" /var/log/mysql/mysqld.log
2023-09-01T06:42:35.564870Z 6 [Warning] [MY-010453] [Server] root
@localhost is created with an empty password ! Please consider swi
tching off the --initialize-insecure option.

# MySQL 的安全性配置
[root@Lab-04-Task-01 ~]# mysql_secure_installation
```

6、部署 WordPress

(1) 获取 WordPress 应用程序。

```
# 进入根目录/var/www/html/, 下载解压 WordPress。
[root@Lab-04-Task-01 ~]# cd /var/www/html/
# 安装下载工具
[root@Lab-04-Task-01 html]# yum install -y wget tar
# 下载 WordPress 安装包
[root@Lab-04-Task-01 html]# wget https://cn.wordpress.org/latest-zh_CN.tar.gz
# 解压 WordPress 安装包
[root@Lab-04-Task-01 html]# tar zxvf latest-zh_CN.tar.gz

# 复制数据库连接配置文件
[root@Lab-04-Task-01 html]# cd wordpress/
[root@Lab-04-Task-01 wordpress]# cp -a wp-config-sample.php wp-config.php

# 赋予 Apache 对相关目录的操作权限
[root@Lab-04-Task-01 wordpress]# chown -R apache:apache /var/www/html/
[root@Lab-04-Task-01 wordpress]# chmod -R 755 /var/www/html/
[root@Lab-04-Task-01 wordpress]# chown -R :apache /var/www/html/wordpress
```

(2) 创建 WordPress 数据库和数据库管理用户。

```
# 连接数据库
[root@Lab-04-Task-01 ~]# mysql -uroot -p
# 创建数据库用户 wordpressuser, 并赋予用户权限、设置用户密码
mysql> CREATE USER 'wordpressuser'@'%' IDENTIFIED BY 'Wordpress@123';
Query OK, 1 row affected (0.02 sec)
# 创建 WordPress 数据库
mysql> CREATE DATABASE wordpress;
Query OK, 1 row affected (0.02 sec)
# 设置权限, 允许 wordpressuser 用户远程访问 wordpress 数据库
mysql> GRANT ALL PRIVILEGES ON wordpress.* TO 'wordpressuser'@'%';
# 刷新权限使更改生效, 退出
mysql> flush privileges;
mysql> exit;
```

(3) 编辑数据库连接配置。

```
# 编辑 wp-config.php 文件。根据已配置的 WordPress 数据库信息, 修改 MySQL 相关配置信息
[root@Lab-04-Task-01 wordpress]# vi wp-config.php
-----wp-config.php-----
# 修改如下配置项
# /** WordPress 数据库的名称 */
define( 'DB_NAME', 'wordpress' );
# /** MySQL 数据库用户名 */
define( 'DB_USER', 'wordpressuser' );
# /** MySQL 数据库密码 */
define( 'DB_PASSWORD', 'Wordpress@123' );
# /** MySQL 主机 */
define( 'DB_HOST', 'Mysql 主机 IP' );
-----wp-config.php-----
```

(4) 重启 Apache 服务。

```
[root@Lab-04-Task-01 ~]#systemctl restart httpd
```

7、初始化配置

打开浏览器访问 <http://172.31.0.141/wordpress>, 根据安装向导填写安装信息, 点击安装 WordPress 按钮, 完成安装并创建站点, 如图 4-2 所示。



图 4-2 WordPress 安装页

8、访问测试

等待初始化完成，刷新网页页面，出现如图 4-3 所示，说明通过 WordPress 建设网站成功。

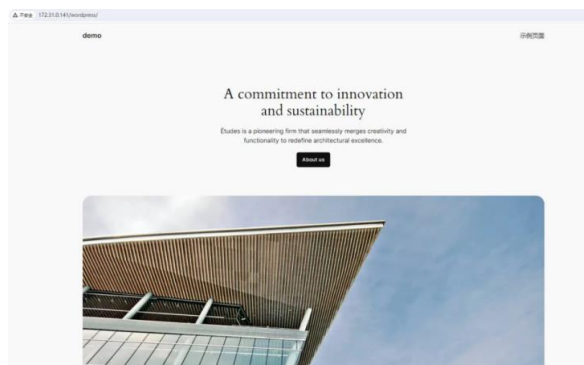


图 4-3 测试站点

八、实验考核

实验考核分为【实验随堂查】和【实验线上考】两个部分。

实验随堂查：每个实验设置 2-5 考核点。完成实验任务后，任课教师随机选择一个考核点，学生现场进行演示和汇报讲解。

实验线上考：每个实验设置 10 道客观题。通过线上考核平台（如课堂派）进行作答。

1、实验随堂查

本实验随堂查设置 2 个考核点，具体如下：

考核点 1：可以访问 wordpress 默认站点

考核点 2：通过 wordpress 发布个人站点且能够访问

考核点 3：实现只用 IP 地址访问 wordpress 站点，如 <http://172.31.0.21>

2、实验线上考

本实验线上考共 10 题，题型为单选、多选、判断、填空等题型。