

部署文档

作者: 李煦阳

部署服务器操作系统: centos7

项目地址: [前台链接](#), [后台链接](#)

一、环境需求

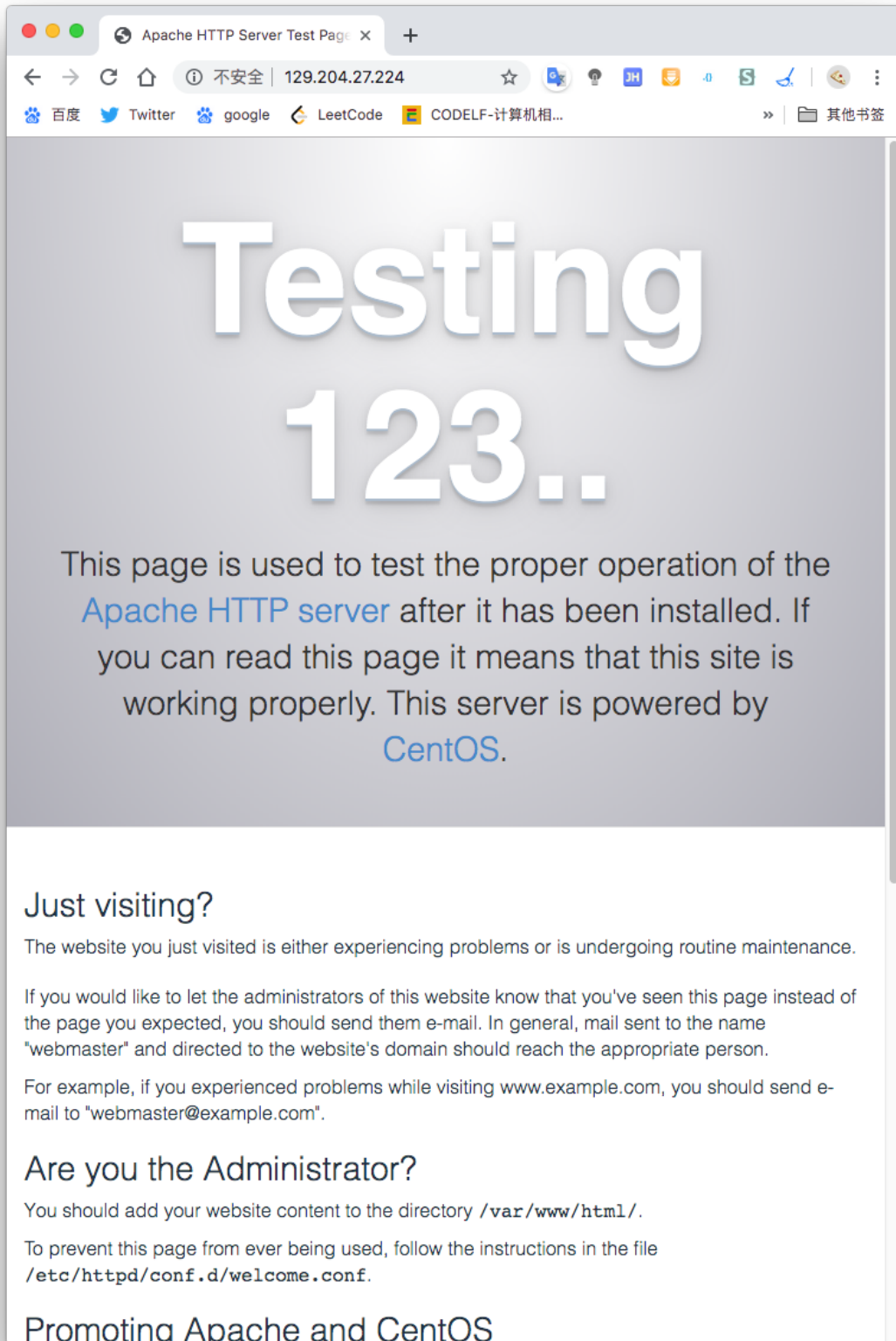
必要环境

- apache
- php
- mysql
- git
- composer

二、环境部署步骤

1. apache的安装

```
1 | sudo yum install httpd
2 | sudo systemctl enable httpd.service
3 | sudo systemctl is-enabled httpd.service
4 | # 观察到enabled, 直接通过ip访问服务器, 或看见apache默认欢迎页, 则成功
```



关于apache的一些常用命令，在下面经常用到：

```
1 # 在服务器上启动 Apache 服务的命令为：
2 sudo systemctl start httpd.service
3 # 重新启动 Apache：
4 sudo systemctl restart httpd.service
5 # 停止 Apache：
6 sudo systemctl stop httpd.service
```

若在阿里云上，请先在服务器控制台开启相应接口

本项目未使用url美化规则，未使用虚拟域名，所以无需对apache配置进行更改，无需新建 `.htaccess` 文件

2. mysql的安装

```
1 sudo yum update
2 wget http://repo.mysql.com/mysql57-community-release-e17-8.noarch.rpm
3 sudo yum install mysql57-community-release-e17-8.noarch.rpm
4 sudo yum update
5 ##### 若速度缓慢，可以再次出更改yum镜像源
6 sudo yum install mysql-server
```

更改yum镜像源的代码：

```
1 cd /etc/yum.repos.d
2 mv CentOS-Base.repo CentOS-Base.repo.bk
3 wget http://mirrors.163.com/.help/CentOS6-Base-163.repo
4 # yum clean all
5 # yum update
6 yum makecache
```

3. mysql的配置

- [初始密码配置](#) (mysql version before 5.76)

```
1 systemctl stop mysqld
2 systemctl set-environment MYSQLD_OPTS="--skip-grant-tables"
3 systemctl start mysqld
4 mysql -u root
5 mysql> UPDATE mysql.user SET authentication_string =
  PASSWORD('MyNewPassword')
6     -> WHERE User = 'root' AND Host = 'localhost';
7 mysql> FLUSH PRIVILEGES;
8 mysql> quit
9 systemctl stop mysqld
10 systemctl unset-environment MYSQLD_OPTS
11 systemctl start mysqld
12 mysql -u root -p
```

- 远程访问配置

```
1 use mysql
2 # 若出现报错，初始密码配置未成功
3 # set global validate_password_policy=0;
4 # set password=password("MyNewPassword");
5 # use mysql;
6 mysql> update user set host = '%' where user = 'root';
7 mysql> select host, user from user;
8 FLUSH PRIVILEGES;
```

此时即可通过mysql端口（默认3306）使用root账户从任意的ip访问该数据库

- [其他配置](#)：安全配置、自启；比较简单，见连接，未处理。

4. 安装php

```
1 rpm -Uvh https://mirror.webtatic.com/yum/el7/epel-release.rpm
2 rpm -Uvh https://mirror.webtatic.com/yum/el7/webtatic-release.rpm
3 yum install php71w php71w-cli php71w-common php71w-devel php71w-embedded
  php71w-fpm php71w-gd php71w-mbstring php71w-mysqlnd php71w-opcache php71w-pdo
  php71w-xml php71w-ldap php71w-mcrypt
4 service php-fpm start
```

5. 安装git，并进行全局配置

```
1 yum install git
2 git config --global user.name <myusername>
3 git config --global user.email <myemail>
4 # ...
```

6. composer安装，配置

```
1 curl -SS https://getcomposer.org/installer | php
2 mv composer.phar /usr/local/bin/composer
3 # composer -v
```

三、项目一般部署步骤

在配置好环境后，放在github上的项目一般要通过以下几个步骤进行部署。

- git clone 到web根目录
- 导入数据库文件
- 进行项目初始化php init
- 重写数据库配置
- 通过composer安装依赖

以上流程都是确定性的，所以我们使用一键部署脚本将五个操作合并

四、项目一键自动部署脚本分析

脚本名称：start.sh，放在项目根目录

脚本的调用方式为 ./start.sh <mysqluser> <mysqluserpassword>

1. 对环境进行检查。（php, mysql）
2. 读取了输入的mysql用户参数与密码参数，将数据库建库文件重定向至mysql中。
3. 并调用init脚本，完成yii项目的必要文件的生成。
4. 根据mysql的用户名，密码，重写项目的主main-local.php文件。
5. 检查composer是否安装，未安装则安装。并更改composer镜像。
6. 使用composer install, composer update下载依赖。

脚本内容：

```
1 #!/usr/bin/env bash
2 # Unsafe auto-deploy script.
3 if ! [ -x "$(command -v php)" ]; then
4     echo 'Error: php is not installed.' >&2
```

```

5     exit 1
6 fi
7
8 if ! [ -x "$(command -v mysql)" ]; then
9     echo 'Error: mysql is not installed.' >&2
10    exit 1
11 fi
12
13 user=root
14 password="default"
15 if [ -n "$1" ]; then
16     user=$1
17     echo Using mysql user: ${user}
18 else
19     echo Using default mysql user: ${user}
20 fi
21
22 if [ -n "$2" ]; then
23     password=$2
24 else
25     echo Using empty password for user ${user}
26 fi
27
28 echo Initiating database: TestShell
29 cat ./db/sh_test.sql | mysql -u ${user} -p"${password}"
30
31 echo PHP INIT
32 php init
33
34 echo PROJECT CONFIGING
35
36 (
37 cat << EOF
38 <?php
39 return [
40     'components' => [
41         'db' => [
42             'class' => 'yii\db\Connection',
43             'dsn' => 'mysql:host=localhost;dbname=TestShell',
44             'username' => '${user}',
45             'password' => '${password}',
46             'charset' => 'utf8',
47         ],
48         'mailer' => [
49             'class' => 'yii\swiftmailer\Mailer',
50             'viewPath' => '@common/mail',
51             // send all mails to a file by default. You have to set
52             // 'useFileTransport' to false and configure a transport
53             // for the mailer to send real emails.
54             'useFileTransport' => true,
55         ],
56     ],
57 ];
58 EOF
59 ) > ./common/config/main-local.php
60
61 if ! [ -x "$(command -v composer)" ]; then
62     curl -ss https://getcomposer.org/installer | php

```

```

63 mv composer.phar /usr/local/bin/composer
64 fi
65 # composer 全局安装
66
67 echo COMPOSER UPDATE
68 # 更改镜像源为aliyun
69 composer config -g repo.packagist composer
https://mirrors.aliyun.com/composer/
70 # composer install
71 composer install -vvv
72 # composer update
73 composer update -vvv

```

应声明的是，改脚本读取密码的方式不够安全，对用户提示的方式还可以修改。

在部署过程中，由于composer update的速度缓慢，出现ssh断开的情况，所以又重新调用了composer install，composer update，完成对未成功下载的项目依赖的安装。

使用自动脚本自动创建数据库：

