阿里云
云服务器 ECS

建站教程

文档版本：20181101

为了无法计算的价值 | 阿里云

法律声明

阿里云提醒您在阅读或使用本文档之前仔细阅读、充分理解本法律声明各条款的内容。如果您阅读或使用本文档，您的阅读或使用行为将被视为对本声明全部内容的认可。

1. 您应当通过阿里云网站或阿里云提供的其他授权通道下载、获取本文档，且仅能用于自身的合法合规的业务活动。本文档的内容视为阿里云的保密信息，您应当严格遵守保密义务；未经阿里云事先书面同意，您不得向任何第三方披露本手册内容或提供给任何第三方使用。
2. 未经阿里云事先书面许可，任何单位、公司或个人不得擅自摘抄、翻译、复制本文档内容的部分或全部，不得以任何方式或途径进行传播和宣传。
3. 由于产品版本升级、调整或其他原因，本文档内容有可能变更。阿里云保留在没有任何通知或者提示下对本文档的内容进行修改的权利，并在阿里云授权通道中不时发布更新后的用户文档。您应当实时关注用户文档的版本变更并通过阿里云授权渠道下载、获取最新版的用户文档。
4. 本文档仅作为用户使用阿里云产品及服务的参考性指引，阿里云以产品及服务的”现状“、“有缺陷”和“当前功能”的状态提供本文档。阿里云在现有技术的基础上尽最大努力提供相应的介绍及操作指引，但阿里云在此明确声明对本文档内容的准确性、完整性、适用性、可靠性等不作任何明示或暗示的保证。任何单位、公司或个人因为下载、使用或信赖本文档而发生任何差错或经济损失的，阿里云不承担任何法律责任。在任何情况下，阿里云均不对任何间接性、后果性、惩戒性、偶然性、特殊性或刑罚性的损害，包括用户使用或信赖本文档而遭受的利润损失，承担责任（即使阿里云已被告知该等损失的可能性）。
5. 阿里云网站上所有内容，包括但不限于著作、产品、图片、档案、资讯、资料、网站架构、网站画面的安排、网页设计，均由阿里云和/或其关联公司依法拥有其知识产权，包括但不限于商标权、专利权、著作权、商业秘密等。非经阿里云和/或其关联公司书面同意，任何人不得擅自使用、修改、复制、公开传播、改变、散布、发行或公开发表阿里云网站、产品程序或内容。此外，未经阿里云事先书面同意，任何人不得为了任何营销、广告、促销或其他目的使用、公布或复制阿里云的名称（包括但不限于单独为或以组合形式包含”阿里云”、Aliyun”、“万网”等阿里云和/或其关联公司品牌，上述品牌的附属标志及图案或任何类似公司名称、商号、商标、产品或服务名称、域名、图案标示、标志、标识或通过特定描述使第三方能够识别阿里云和/或其关联公司）。
6. 如若发现本文档存在任何错误，请与阿里云取得直接联系。

通用约定

格式	说明	样例
	该类警示信息将导致系统重大变更甚至故障，或者导致人身伤害等结果。	 **禁止：** 重置操作将丢失用户配置数据。
	该类警示信息可能导致系统重大变更甚至故障，或者导致人身伤害等结果。	 **警告：** 重启操作将导致业务中断，恢复业务所需时间约10分钟。
	用于补充说明、最佳实践、窍门等，不是用户必须了解的内容。	 **说明：** 您也可以通过按**Ctrl** + **A**选中全部文件。
>	多级菜单递进。	设置 > 网络 > 设置网络类型
粗体	表示按键、菜单、页面名称等UI元素。	单击 **确定**。
`courier` 字体	命令。	执行 `cd /d C:/windows` 命令，进入Windows系统文件夹。
斜体	表示参数、变量。	`bae log list --instanceid` *`Instance_ID`*
[]或者[a|b]	表示可选项，至多选择一个。	`ipconfig` *`[-all|-t]`*
{}或者{a|b}	表示必选项，至多选择一个。	`swich` *`{stand | slave}`*

目录

法律声明.....I
通用约定.....I
1 搭建WordPress网站..... 1
2 部署LNMP..... 7
2.1 搭建LNMP环境（CentOS 6）.....7
3 部署Java Web.....17
3.1 手工部署Java Web项目.....17
4 在Linux实例上搭建Magento电子商务网站（CentOS 7）.....22
5 手动建站（Windows环境）..... 30
6 部署LAMP..... 36
7 搭建Joomla基础管理平台..... 46
8 部署Ghost博客（CentOS 7）..... 61
9 搭建FTP站点.....78
9.1 Linux实例搭建FTP站点.....78
9.2 Windows实例搭建FTP站点.....83

1 搭建WordPress网站

WordPress是使用PHP语言开发的博客平台，在支持PHP和MySQL数据库的服务器上，您可以用WordPress架设自己的网站，也可以用作内容管理系统（CMS）。建站时需要准备域名、空间和程序。使用WordPress镜像创建ECS实例，不需要部署Web环境，解决了空间和程序的问题，只要注册域名，完成备案，网站就可以直接上线，降低了建站的门槛，即买即用。

本文档介绍如何使用阿里云云市场的 *WordPress*纯净版免费镜像 或者 *WordPress*商用主题镜像 创建实例，并上线网站。

适用对象

适用于刚开始使用阿里云建站的企业或个人用户。

基本流程

使用 *WordPress*纯净版免费镜像 或者 *WordPress*商用主题镜像 在ECS实例上搭建网站的步骤如下：

第1步：购买WordPress镜像

第2步：安装企业官网模板主题

第3步：修改主题元素

第4步：购买域名

第5步：备案

第6步：解析域名

第1步：购买WordPress镜像

假设您第一次使用阿里云ECS服务，按以下步骤创建一台基于 *WordPress*纯净版免费镜像 或者 *WordPress*商用主题镜像 的ECS实例。

1. 登录 阿里云。如果尚未注册，单击 免费注册。
2. 进入云市场，找到 *WordPress*纯净版免费镜像 或者 *WordPress*商用主题镜像，并单击 立即购买。
3. 在云服务器ECS 自定义购买 页面，完成如下 基础配置：

   a. 选择 计费方式：如果您需要备案网站，必须选择 包年包月，并在页面底部设置 购买时长 不少于3个月。如果不需要备案，您可以根据自己的需求选择计费方式。

   b. 选择 地域：目前支持该镜像的地域包括华北1、华北2、华北3、华北5、华东1、华东2、华南1。请根据网站用户的分布和您自己的地理位置选择合适的地域。如何选择地域与可用区，请参见 地域与可用区。

   Note:

- 实例创建成功后，不能更改地域和可用区。
- 不同地域提供的可用区数量、实例规格族、存储类型、实例价格等也会有所差异。请根据您的业务需求选择地域。

c. 选择 实例：根据您网站的预期访问量选择实例规格（CPU、内存）。一般企业网站，通用或入门级的1核2 GiB或者2核4 GiB实例规格能满足需求。关于实例规格的详细介绍，请参见 实例规格族。

d. 选择 镜像：已经设置为 镜像市场 的 **WordPress纯净版免费镜像 WordPress4.9.4纯净版** 或者 **WordPress商用主题镜像 php-wordPress1.0.8**。您也可以根据需求，单击 重新选择镜像 从镜像市场选择其他WordPress镜像。

e. 选择 存储：

- 系统盘：必填项。您可以选择云盘类型和云盘容量。本示例中，系统盘类型选择 高效云盘，大小采用镜像文件大小，即46 GiB。
- 数据盘：选填项。建议您创建一块50 GiB的高效云盘作为数据盘，不加密。

4. 单击 下一步：网络和安全组，并完成 网络和安全组 配置：

a. 选择 网络：选择 专有网络。如果您未创建专有网络和交换机，选择 默认专有网络 和 默认交换机。

Note:
如果您已经创建过经典网络类型的ECS实例，而且选择的实例类型也支持经典网络，那么，您可以选择 经典网络。

b. 设置 公网带宽：因为创建的实例需要访问公网，所以，您需要选择 分配公网IP地址，并根据预期的网站出网流量，选择 按固定带宽 或 按使用流量 计费，并设置带宽。建议选择 固定带宽，而且带宽值建议不低于2 Mbps。

c. 选择 安全组：如果在当前地域内未创建过安全组，您可以使用默认安全组，并选择 **HTTP 80** 端口。

> **Note:**
> 如果您已经创建了安全组，必须在安全组中添加规则，放行入方向允许对HTTP 80端口的访问。详细信息，请参见 添加安全组规则。

5. 单击 下一步：系统配置，完成系统配置：

a. 设置 登录凭证：建议您在这里直接设置实例的登录密码。请务必牢记登录名和密码。

b. 设置 实例名称、描述 和 主机名，为了便于以后管理。

6. 略过 下一步：分组设置，单击 确认订单：
   - 确认 所选配置。如果需要修改配置，单击 图标，重新选择配置。
   - 确认购买时长（比如本例中的3个月），并决定是否开启 自动续费 功能。
7. 阅读并确认《云服务器 ECS 服务条款》和《镜像商品使用条款》。
8. 单击 确认下单。
9. 确认订单并付款。

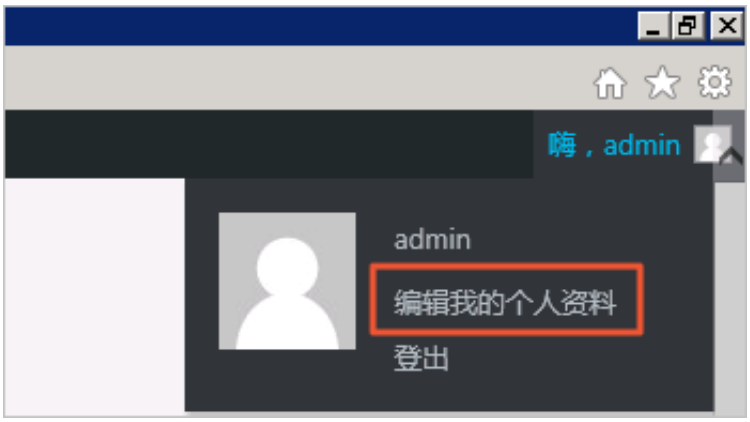
实例开通后，单击 管理控制台 回到ECS管理控制台查看新建的ECS实例。在相应地域的 实例列表 里，您能查看新建实例的实例名称、公网IP地址、内网IP地址或私有IP地址等信息。

第2步：安装企业官网模板主题

为了满足企业官网展示的需求，您需要安装适合的主题模板。这部分主要介绍如何安装WordPress主题模板。

1. 远程连接*Windows*实例。
2. 打开IE浏览器，使用 `http://实例公网IP地址/wp-admin` 进入WordPress登录页面。
3. 输入用户名 **admin** 和初始密码 **admin123**，并单击 登录。
4. 在 仪表盘 页面的左侧导航栏中，选择 外观 > 主题 。
5. 在 主题 页面上选择已经安装的主题。

> **Note:**
> 登录后，建议您在 编辑我的个人资料 的 账户管理 中修改密码。



第3步：修改主题元素

安装主题后，您可以修改主题元素，打造您自己的专属网站。具体操作，请参见镜像供应商飞色网络的 帮助中心。

第4步：购买域名

为了便于您的使用易记的域名访问您的网站，您可以给自己的网站设定一个单独的域名。您可以参考 域名注册流程 完成操作。

第5步：备案

假设您是第一次备案域名，具体操作流程，请参见 首次备案流程图文引导。其他情况下，请参见 备案导航。

第6步：解析域名

域名解析是使用域名访问您的网站的必备环节。具体操作流程，请参见 设置域名解析。

扩展服务容量

随着业务的扩展，您还可以借助阿里云强大的产品平台，横向和纵向扩展服务容量，例如：

- 扩展单个ECS实例的vCPU和内存规格，增强服务器的处理能力。详细信息，请参见 升降配概述。
- 增加多台ECS实例，并利用负载均衡，在多个实例中进行负载的均衡分配。详细信息，请参见 购买相同配置实例 和 什么是负载均衡。
- 利用弹性伸缩（Auto Scaling），根据业务量自动增加或减少ECS实例的数量。详细信息，请参见 什么是弹性伸缩。
- 利用对象存储OSS（Object Storage Service），存储静态网页和海量图片、视频等。详细信息，请参见 什么是OSS。

2 部署LNMP

2.1 搭建LNMP环境（CentOS 6）

本文档介绍如何使用一台普通配置的云服务器ECS实例搭建LNMP平台的web环境。

- Linux：自由和开放源码的类UNIX操作系统。
- Nginx：轻量级网页服务器、反向代理服务器。
- MySQL：关系型数据库管理系统。
- PHP：主要适用于Web开发领域的一种脚本语言。

适用对象

适用于熟悉Linux操作系统，刚开始使用阿里云进行建站的个人用户。

基本流程

使用云服务器 ECS 搭建LNMP平台的操作步骤如下：

1. 准备编译环境
2. 安装nginx
3. 安装mysql
4. 安装php-fpm
5. 测试访问

步骤一：准备编译环境

本文主要说明手动安装LNMP平台的操作步骤，您也可以在 云市场 购买LNMP镜像直接启动ECS，以便快速建站。

1. 系统版本说明

```
# cat /etc/redhat-release
CentOS release 6.5 (Final)
```

> **说明：**
> 这是本文档实施时参考的系统版本。您的实际使用版本可能与此不同，下文中的nginx，mysql，及php版本，您也可以根据实际情况选择相应版本。

2. 关闭SELINUX

修改配置文件，重启服务后永久生效。

```
# sed -i 's/SELINUX=.*/SELINUX=disabled/g' /etc/selinux/config
```

命令行设置立即生效。

```
# setenforce 0
```

3. 安全组设置

在ECS安全组放行需访问的端口和访问白名单，下面的示例表示允许所有IP访问服务器的80端口。您可以根据实际情况放行允许访问的客户端IP。

步骤二：安装**nginx**

Nginx是一个小巧而高效的Linux下的Web服务器软件，是由 Igor Sysoev 为俄罗斯访问量第二的 Rambler.ru 站点开发的，已经在一些俄罗斯的大型网站上运行多年，目前很多国内外的门户网站、行业网站也都在是使用Nginx，相当稳定。

1. 添加运行nginx服务进程的用户。

```
# groupadd -r nginx
```

```
# useradd -r -g nginx  nginx
```

2. 下载源码包解压编译。

```
# wget http://nginx.org/download/nginx-1.10.2.tar.gz
# tar xvf nginx-1.10.2.tar.gz -C /usr/local/src
# yum groupinstall "Development tools"
# yum -y install gcc wget gcc-c++ automake autoconf libtool libxml2
-devel libxslt-devel perl-devel perl-ExtUtils-Embed pcre-devel
openssl-devel
# cd /usr/local/src/nginx-1.10.2
# ./configure \
--prefix=/usr/local/nginx \
--sbin-path=/usr/sbin/nginx \
--conf-path=/etc/nginx/nginx.conf \
--error-log-path=/var/log/nginx/error.log \
--http-log-path=/var/log/nginx/access.log \
--pid-path=/var/run/nginx.pid \
--lock-path=/var/run/nginx.lock \
--http-client-body-temp-path=/var/tmp/nginx/client \
--http-proxy-temp-path=/var/tmp/nginx/proxy \
--http-fastcgi-temp-path=/var/tmp/nginx/fcgi \
--http-uwsgi-temp-path=/var/tmp/nginx/uwsgi \
--http-scgi-temp-path=/var/tmp/nginx/scgi \
--user=nginx \
--group=nginx \
--with-pcre \
--with-http_v2_module \
--with-http_ssl_module \
--with-http_realip_module \
--with-http_addition_module \
--with-http_sub_module \
--with-http_dav_module \
--with-http_flv_module \
--with-http_mp4_module \
--with-http_gunzip_module \
--with-http_gzip_static_module \
--with-http_random_index_module \
--with-http_secure_link_module \
--with-http_stub_status_module \
--with-http_auth_request_module \
--with-mail \
--with-mail_ssl_module \
--with-file-aio \
--with-ipv6 \
--with-http_v2_module \
--with-threads \
--with-stream \
--with-stream_ssl_module
# make && make install
# mkdir -pv /var/tmp/nginx/client
```

3. 添加SysV启动脚本。

```
# vim /etc/init.d/nginx
#!/bin/sh
#
# nginx - this script starts and stops the nginx daemon
#
# chkconfig:   - 85 15
# description: Nginx is an HTTP(S) server, HTTP(S) reverse \
```

```
#               proxy and IMAP/POP3 proxy server
# processname: nginx
# config:      /etc/nginx/nginx.conf
# config:      /etc/sysconfig/nginx
# pidfile:     /var/run/nginx.pid
# Source function library.
. /etc/rc.d/init.d/functions
# Source networking configuration.
. /etc/sysconfig/network
# Check that networking is up.
[ "$NETWORKING" = "no" ] && exit 0
nginx="/usr/sbin/nginx"
prog=$(basename $nginx)
NGINX_CONF_FILE="/etc/nginx/nginx.conf"
[ -f /etc/sysconfig/nginx ] && . /etc/sysconfig/nginx
lockfile=/var/lock/subsys/nginx
start() {
    [ -x $nginx ] || exit 5
    [ -f $NGINX_CONF_FILE ] || exit 6
    echo -n $"Starting $prog: "
    daemon $nginx -c $NGINX_CONF_FILE
    retval=$?
    echo
    [ $retval -eq 0 ] && touch $lockfile
    return $retval
}
stop() {
    echo -n $"Stopping $prog: "
    killproc $prog -QUIT
    retval=$?
    echo
    [ $retval -eq 0 ] && rm -f $lockfile
    return $retval
killall -9 nginx
}
restart() {
    configtest || return $?
    stop
    sleep 1
    start
}
reload() {
    configtest || return $?
    echo -n $"Reloading $prog: "
    killproc $nginx -HUP
RETVAL=$?
    echo
}
force_reload() {
    restart
}
configtest() {
$nginx -t -c $NGINX_CONF_FILE
}
rh_status() {
    status $prog
}
rh_status_q() {
    rh_status >/dev/null 2>&1
}
case "$1" in
    start)
```

```
        rh_status_q && exit 0
    $1
        ;;
    stop)
        rh_status_q || exit 0
        $1
        ;;
    restart|configtest)
        $1
        ;;
    reload)
        rh_status_q || exit 7
        $1
        ;;
    force-reload)
        force_reload
        ;;
    status)
        rh_status
        ;;
    condrestart|try-restart)
        rh_status_q || exit 0
            ;;
    *)
      echo $"Usage: $0 {start|stop|status|restart|condrestart|try-
restart|reload|force-reload|configtest}"
        exit 2
esac
```

4. 赋予脚本执行权限。

```
# chmod +x /etc/init.d/nginx
```

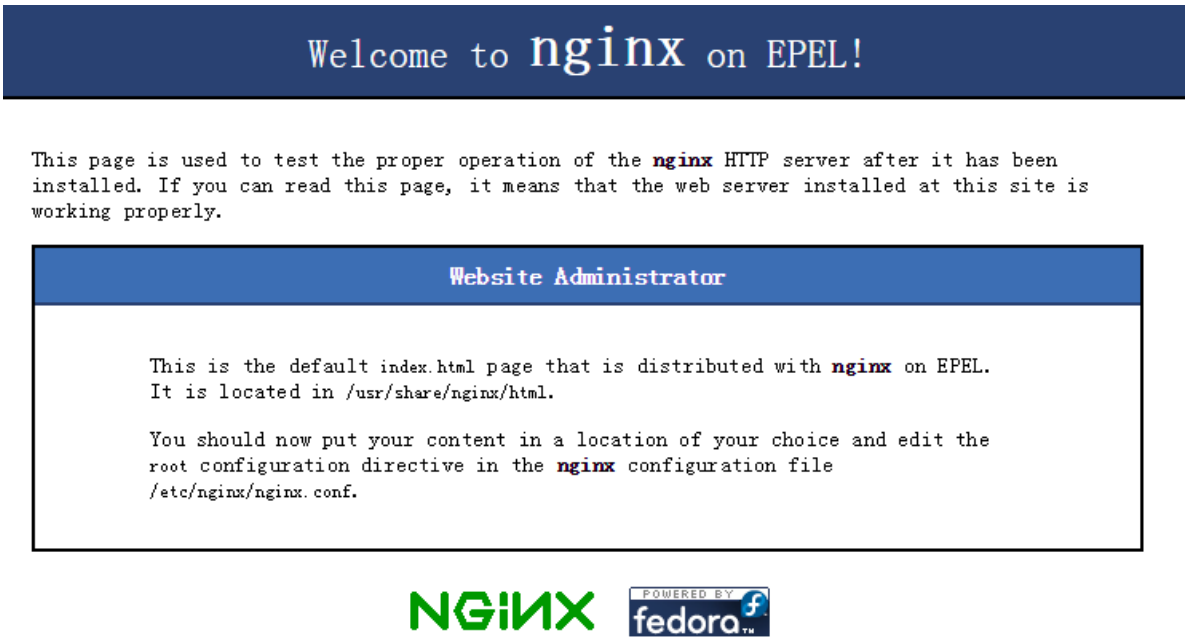
5. 添加至服务管理列表，设置开机自启。

```
# chkconfig --add nginx
# chkconfig  nginx on
```

6. 启动服务。

```
# service nginx start
```

7. 浏览器访问可看到默认欢迎页面。



步骤三：安装**mysql**

1. 准备编译环境。

```
# yum groupinstall "Server Platform Development"  "Development tools
" -y
# yum install cmake -y
```

2. 准备mysql数据存放目录。

```
# mkdir /mnt/data
# groupadd -r mysql
# useradd -r -g mysql -s /sbin/nologin mysql
# id mysql
uid=497(mysql) gid=498(mysql) groups=498(mysql)
```

3. 更改数据目录属主属组。

```
# chown -R mysql:mysql /mnt/data
```

4. 解压编译在 *MySQL官网* 下载的稳定版源码包，这里使用的是5.6.24版本。

```
# tar xvf mysql-5.6.24.tar.gz -C  /usr/local/src
# cd /usr/local/src/mysql-5.6.24
# cmake . -DCMAKE_INSTALL_PREFIX=/usr/local/mysql \
-DMYSQL_DATADIR=/mnt/data \
-DSYSCONFDIR=/etc \
-DWITH_INNOBASE_STORAGE_ENGINE=1 \
-DWITH_ARCHIVE_STORAGE_ENGINE=1 \
-DWITH_BLACKHOLE_STORAGE_ENGINE=1 \
-DWITH_READLINE=1 \
-DWITH_SSL=system \
-DWITH_ZLIB=system \
-DWITH_LIBWRAP=0 \
-DMYSQL_TCP_PORT=3306 \
```

```
-DMYSQL_UNIX_ADDR=/tmp/mysql.sock \
-DDEFAULT_CHARSET=utf8 \
-DDEFAULT_COLLATION=utf8_general_ci
# make && make install
```

5. 修改安装目录的属组为mysql。

```
# chown -R mysql:mysql /usr/local/mysql/
```

6. 初始化数据库。

```
# /usr/local/mysql/scripts/mysql_install_db --user=mysql --datadir=/
mnt/data/
```

说明：

在CentOS 6.5版操作系统的最小安装完成后，在/etc目录下会存在一个my.cnf，需要将此文件更名为其他的名字，如：/etc/my.cnf.bak，否则，该文件会干扰源码安装的MySQL的正确配置，造成无法启动。

7. 拷贝配置文件和启动脚本。

```
# cp /usr/local/mysql/support-files/mysql.server /etc/init.d/mysqld
# chmod +x /etc/init.d/mysqld
# cp support-files/my-default.cnf /etc/my.cnf
```

8. 设置开机自动启动。

```
# chkconfig mysqld  on
# chkconfig --add mysqld
```

9. 修改配置文件中的安装路径及数据目录存放路径。

```
# echo -e "basedir = /usr/local/mysql\ndatadir = /mnt/data\n" >> /
etc/my.cnf
```

10. 设置PATH环境变量。

```
# echo "export PATH=$PATH:/usr/local/mysql/bin" > /etc/profile.d/
mysql.sh
# source /etc/profile.d/mysql.sh
```

11. 启动服务。

```
# service mysqld start
# mysql -h 127.0.0.1
```

步骤四：安装**php-fpm**

Nginx本身不能处理PHP，作为web服务器，当它接收到请求后，不支持对外部程序的直接调用或者解析，必须通过FastCGI进行调用。如果是PHP请求，则交给PHP解释器处理，并把结果返回给客

户端。PHP-FPM是支持解析php的一个FastCGI进程管理器。提供了更好管理PHP进程的方式，可以有效控制内存和进程、可以平滑重载PHP配置。

1. 安装依赖包。

```
# yum install libmcrypt libmcrypt-devel mhash mhash-devel libxml2
libxml2-devel bzip2 bzip2-devel
```

2. 解压官网下载的源码包，编译安装。

```
# tar xvf php-5.6.23.tar.bz2 -C /usr/local/src
# cd /usr/local/src/php-5.6.23
# ./configure --prefix=/usr/local/php \
--with-config-file-scan-dir=/etc/php.d \
--with-config-file-path=/etc \
--with-mysql=/usr/local/mysql \
--with-mysqli=/usr/local/mysql/bin/mysql_config \
--enable-mbstring \
--with-freetype-dir \
--with-jpeg-dir \
--with-png-dir \
--with-zlib \
--with-libxml-dir=/usr \
--with-openssl \
--enable-xml \
--enable-sockets \
--enable-fpm \
--with-mcrypt \
--with-bz2
# make && make install
```

3. 添加php和php-fpm配置文件。

```
# cp /usr/local/src/php-5.6.23/php.ini-production /etc/php.ini
# cd /usr/local/php/etc/
# cp php-fpm.conf.default php-fpm.conf
# sed -i 's@;pid = run/php-fpm.pid@pid = /usr/local/php/var/run/php-
fpm.pid@' php-fpm.conf
```

4. 添加php-fpm启动脚本。

```
# cp /usr/local/src/php-5.6.23/sapi/fpm/init.d.php-fpm /etc/init.d/
php-fpm
# chmod +x /etc/init.d/php-fpm
```

5. 添加php-fpm至服务列表并设置开机自启。

```
# chkconfig --add php-fpm
# chkconfig --list php-fpm
```

```
# chkconfig php-fpm on
```

6. 启动服务。

```
# service php-fpm start
```

7. 添加nginx对fastcgi的支持，首先备份默认的配置文件。

```
# cp /etc/nginx/nginx.conf /etc/nginx/nginx.confbak
# cp /etc/nginx/nginx.conf.default /etc/nginx/nginx.conf
```

编辑/etc/nginx/nginx.conf，在所支持的主页面格式中添加php格式的主页，类似如下：

```
location / {
  root   /usr/local/nginx/html;
  index  index.php index.html index.htm;
}
```

取消以下内容前面的注释：

```
location ~ \.php$ {
  root           /usr/local/nginx/html;
  fastcgi_pass    127.0.0.1:9000;
  fastcgi_index   index.php;
  fastcgi_param  SCRIPT_FILENAME  /usr/local/nginx/html/$fastcgi_sc
ript_name;
  include        fastcgi_params;
}
```

重新载入nginx的配置文件。

```
# service nginx reload
```

在/usr/local/nginx/html/新建index.php的测试页面，内容如下。

```
# cat index.php
<?php
$conn=mysql_connect('127.0.0.1','root','');
if ($conn){
echo "LNMP platform connect to mysql is successful!";
}else{
echo "LNMP platform connect to mysql is failed!";
}
phpinfo();
?>
```

浏览器访问测试，如看到以下内容则表示LNMP平台构建完成。

LNMP platform connect to mysql is successful!

PHP Version 5.6.23

System	Linux iZuf86k0f52wt2c81bplgZZ 2.6.32-573.22.1.el6.x86_64 #1 SMP Wed Mar 23 03:35:39 UTC 2016 x86_64
Build Date	Dec 12 2016 21:27:46
Configure Command	'./configure' '--prefix=/usr/local/php' '--with-config-file-scan-dir=/etc/php.d' '--with-config-file-path=/etc' '--with-mysql=/usr/local/mysql' '--with-mysqli=/usr/local/mysql/bin/mysql_config' '--enable-mbstring' '--with-freetype-dir' '--with-jpeg-dir' '--with-png-dir' '--with-zlib' '--with-libxml-dir=/usr' '--with-openssl' '-enable-xml' '--enable-sockets' '--enable-fpm' '--with-mcrypt' '--with-bz2'
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

3 部署Java Web

3.1 手工部署Java Web项目

本文档介绍如何使用一台基本配置的云服务器 ECS 实例部署 Java web 项目。适用于刚开始使用阿里云进行建站的个人用户。

项目配置

此处列出验证本文档操作时使用的软件版本。操作时，请您以实际软件版本为准。

- 操作系统：CentOS 7.4
- Tomcat 版本：Tomcat 8.5.23
- JDK 版本：JDK 1.8.0_141

安装前准备

- CentOS 7.4 系统默认开启了防火墙。您可以关闭防火墙，也可以参考官网文档在防火墙里添加规则，放行 80、443 或 8080 端口入方向规则。
  - 关闭防火墙：

    ```
    systemctl stop firewalld.service
    ```

  - 关闭防火墙开机自启动功能：

    ```
    systemctl disable firewalld.service
    ```

- 创建一般用户 www，运行 tomcat：

  ```
  useradd www
  ```

- 在安全组中放行 8080 端口。具体操作，请参考 添加安全组规则。
- 创建网站根目录：

  ```
  mkdir -p /data/wwwroot/default
  ```

- 新建 Tomcat 测试页面：

  ```
  echo Tomcat test > /data/wwwroot/default/index.jsp
  ```

```
chown -R www.www /data/wwwroot
```

下载源代码

下载 Apache

```
wget https://mirrors.aliyun.com/apache/tomcat/tomcat-8/v8.5.23/bin/
apache-tomcat-8.5.23.tar.gz
```

源代码版本会不断升级。您可以在 *https://mirrors.aliyun.com/apache/tomcat/tomcat-8/* 目录下获取合适的安装包地址。

下载 JDK

```
wget http://mirrors.linuxeye.com/jdk/jdk-8u141-linux-x64.tar.gz
```

源代码版本会不断升级。您可以在 *http://mirrors.linuxeye.com/jdk/* 目录下获取合适的安装包地址。

安装 JDK

按以下步骤安装 JDK。

1. 新建一个目录：

   ```
   mkdir /usr/java
   ```

2. 解压 jdk-8u141-linux-x64.tar.gz 到 */usr/java*。

   ```
   tar xzf jdk-8u141-linux-x64.tar.gz -C /usr/java
   ```

3. 设置环境变量：

   a. 打开 /etc/profile：`vi /etc/profile`。

   b. 按 `i` 键进入编辑模式。

   c. 在 /etc/profile 文件中添加以下信息：

   ```
   #set java environment
   export JAVA_HOME=/usr/java/jdk1.8.0_141
   export CLASSPATH=$JAVA_HOME/lib/tools.jar:$JAVA_HOME/lib/dt.jar:$
   JAVA_HOME/lib
   export PATH=$JAVA_HOME/bin:$PATH
   ```

   d. 按 `Esc` 键退出编辑模式，输入 `:wq` 保存并关闭文件。

4. 加载环境变量： `source /etc/profile`。

5. 查看 jdk 版本。当出现 jdk 版本信息时，表示 JDK 已经安装成功。

```
java -version
```

```
java -version
java version "1.8.0_141"
Java(TM) SE Runtime Environment (build 1.8.0_141-b15)
Java HotSpot(TM) 64-Bit Server VM (build 25.141-b15, mixed mode)
```

安装 Tomcat

按以下步骤安装 Tomcat。

1. 依次运行以下命令解压 apache-tomcat-8.5.23.tar.gz，重命名 Tomcat 目录，并设置用户权限。

```
tar xzf apache-tomcat-8.5.23.tar.gz
mv apache-tomcat-8.5.23 /usr/local/tomcat/
chown -R www.www /usr/local/tomcat/
```

说明：

在 */usr/local/tomcat/* 目录里：

- bin：存放 Tomcat 的一些脚本文件，包含启动和关闭 Tomcat 服务脚本。
- conf：存放 Tomcat 服务器的各种全局配置文件，其中最重要的是 server.xml 和 web.xml。
- webapps：Tomcat 的主要 Web 发布目录，默认情况下把 Web 应用文件放于此目录。
- logs：存放 Tomcat 执行时的日志文件。

2. 配置 server.xml 文件：

   a. 切换到 */usr/local/tomcat/conf/* 目录：`cd /usr/local/tomcat/conf/`。

   b. 重命名 server.xml 文件：`mv server.xml server.xml_bk`。

   c. 创建一个新的 server.xml 文件：

      A. 运行命令 `vi server.xml`。

      B. 单击 `i` 键进入编辑模式。

      C. 添加以下内容：

```
<?xml version="1.0" encoding="UTF-8"?>
<Server port="8006" shutdown="SHUTDOWN">
<Listener className="org.apache.catalina.core.JreMemoryLeakPrevent
ionListener"/>
<Listener className="org.apache.catalina.mbeans.GlobalReso
urcesLifecycleListener"/>
<Listener className="org.apache.catalina.core.ThreadLocalLeakPreve
ntionListener"/>
<Listener className="org.apache.catalina.core.AprLifecycleListener
"/>
```

```
<GlobalNamingResources>
<Resource name="UserDatabase" auth="Container"
 type="org.apache.catalina.UserDatabase"
 description="User database that can be updated and saved"
 factory="org.apache.catalina.users.MemoryUserDatabaseFactory"
 pathname="conf/tomcat-users.xml"/>
</GlobalNamingResources>
<Service name="Catalina">
<Connector port="8080"
 protocol="HTTP/1.1"
 connectionTimeout="20000"
 redirectPort="8443"
 maxThreads="1000"
 minSpareThreads="20"
 acceptCount="1000"
 maxHttpHeaderSize="65536"
 debug="0"
 disableUploadTimeout="true"
 useBodyEncodingForURI="true"
 enableLookups="false"
 URIEncoding="UTF-8"/>
<Engine name="Catalina" defaultHost="localhost">
<Realm className="org.apache.catalina.realm.LockOutRealm">
<Realm className="org.apache.catalina.realm.UserDatabaseRealm"
  resourceName="UserDatabase"/>
</Realm>
<Host name="localhost" appBase="/data/wwwroot/default" unpackWARs
="true" autoDeploy="true">
<Context path="" docBase="/data/wwwroot/default" debug="0"
reloadable="false" crossContext="true"/>
<Valve className="org.apache.catalina.valves.AccessLogValve"
directory="logs"
prefix="localhost_access_log." suffix=".txt" pattern="%h %l %u %t
 &amp;quot;%r&amp;quot; %s %b" />
</Host>
</Engine>
</Service>
</Server>
```

3. 设置 JVM 内存参数：

   a. 运行命令 `vi /usr/local/tomcat/bin/setenv.sh`，创建 /usr/local/tomcat/bin/setenv.sh。

   b. 按 `i` 键进入编辑模式。

   c. 添加以下内容：

   ```
   JAVA_OPTS='-Djava.security.egd=file:/dev/./urandom -server -
   Xms256m -Xmx496m -Dfile.encoding=UTF-8'
   ```

   d. 按 `Esc` 键退出编辑模式，输入 `:wq` 保存并退出文件。

4. 设置 Tomcat 自启动脚本。

   a. 下载脚本：`wget https://github.com/lj2007331/oneinstack/raw/master/init.d/Tomcat-init`

b. 重命名 Tomcat-init：`mv Tomcat-init /etc/init.d/tomcat`

c. 添加执行权限：`chmod +x /etc/init.d/tomcat`

d. 运行以下命令，设置启动脚本 JAVA_HOME。

```
sed -i 's@^export JAVA_HOME=.*@export JAVA_HOME=/usr/java/jdk1.8.
0_141@' /etc/init.d/tomcat
```

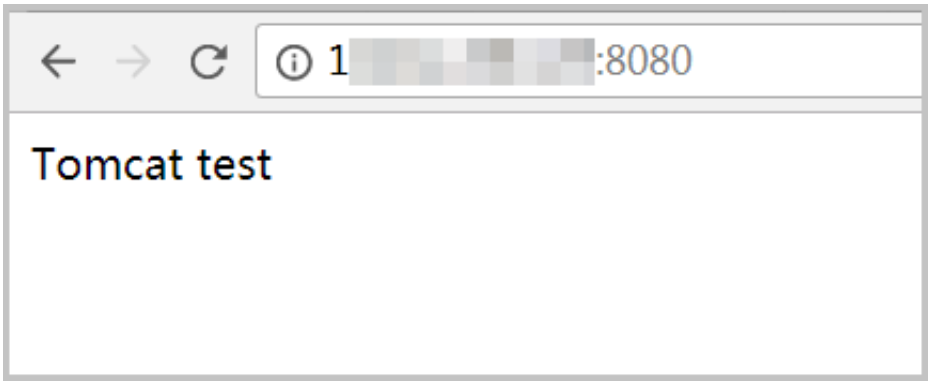
5. 设置自启动。

```
chkconfig --add tomcat
chkconfig tomcat on
```

6. 启动 Tomcat。

```
service tomcat start
```

7. 在浏览器地址栏中输入 http://ip:8080 进行访问。出现如图所示页面时表示安装成功。



4 在Linux实例上搭建Magento电子商务网站（CentOS 7）

Magento是一款开源电商网站框架，其丰富的模块化架构体系及拓展功能可为大中型站点提供解决方案。它使用PHP开发，支持版本范围从PHP 5.6到PHP 7.1，并使用MySQL存储数据。本文主要说明如何在阿里云ECS实例上搭建Magento电子商务网站，使用的操作系统为Linux CentOS 7.2 64位。

适用对象

适用于熟悉ECS，熟悉Linux系统，刚开始使用阿里云进行建站的用户。

资源

本文描述的操作涉及的Linux ECS实例配置包括：2 vCPU、4 GiB内存、Cent OS 7.2 64位操作系统、VPC网络、分配的公网IP地址。

> 说明：
> 用于搭建Magento 2的服务器，内存不能小于2 GiB。

根据本文搭建的Magento电子商务网站，使用的软件版本信息如下：

- MySQL 5.7
- PHP 7.0
- Magento 2.1

前提条件

您已经创建了一台VPC网络类型的Linux ECS实例，详细操作，请参见 步骤 2#创建ECS实例。配置包括：2 vCPU、4 GiB内存、Cent OS 7.2 64位操作系统、VPC网络、分配公网IP地址。

ECS实例所在安全组中已经添加了如下表所示的安全组规则。详细操作，请参见 步骤 2#创建ECS实例 和 添加安全组规则。

服务	规则方向	授权策略	协议类型	端口范围	授权类型	授权对象	优先级
HTTP	入方向	允许	自定义TCP	80/80	地址段访问	0.0.0.0/0	1
MySQL	入方向	允许	自定义TCP	3306/3306	地址段访问	0.0.0.0/0	1

步骤1：安装配置LAMP平台

本部分内容说明如何手动安装LAMP平台。您也可以在 云市场 购买LAMP镜像直接启动ECS实例，以便快速建站。

1. 依次运行以下命令更新包和存储库，并安装Apache Web服务器和MySQL服务器。

```
[ECS]$ yum update -y
[ECS]$ yum install httpd -y
[ECS]$ rpm -Uvh http://dev.mysql.com/get/mysql57-community-release-
el7-8.noarch.rpm
[ECS]$ yum -y install mysql-community-server
```

2. 启动HTTP和MySQL服务并设置开机自启动。

```
[ECS]$ systemctl start httpd
[ECS]$ systemctl enable httpd
[ECS]$ systemctl start mysqld
[ECS]$ systemctl enable mysqld
```

3. 编辑Apache配置文件：

   a. 运行命令 `vim /etc/httpd/conf/httpd.conf`。

   b. 按 `i` 键进入编辑模式。

   c. 做以下修改：

   - 在 `Include conf.modules.d/*.conf` 之后添加 `LoadModule rewrite_module modules/mod_rewrite.so`。
   - 将以下内容的 `AllowOverride None` 改为 `AllowOverride all`。

```
Options Indexes FollowSymLinks
#
# AllowOverride controls what directives may be placed in .
htaccess files.
# It can be "All", "None", or any combination of the keywords:
# Options FileInfo AuthConfig Limit
#
AllowOverride None
```

   d. 按 `Esc` 键退出编辑，并输入 `:wq` 保存并退出。

4. 查看/var/log/mysqld.log文件，获取安装MySQL时自动设置的root用户密码。

```
# grep 'temporary password' /var/log/mysqld.log
2016-12-13T14:57:47.535748Z 1 [Note] A temporary password is
generated for root@localhost: p0/G28g>lsHD
```

5. 运行下面的命令可以从如下4个方面提高MySQL的安全性：

   - 设置root账号密码

- 禁止root账号远程登录
- 删除匿名用户账号
- 删除test库以及对test库的访问权限

详细说明可参见 官方文档。

```
# mysql_secure_installation
Securing the MySQL server deployment.
Enter password for user root: #输入第4步中获取的root用户密码
The 'validate_password' plugin is installed on the server.
The subsequent steps will run with the existing configuration of the
 plugin.
Using existing password for root.
Estimated strength of the password: 100
Change the password for root ? ((Press y|Y for Yes, any other key
for No) : Y #是否更改root用户密码，输入Y
New password: #输入密码，长度为8至30个字符，必须同时包含大小写英文字母、数字和特殊符号。特殊符号可以是()` ~!@#$%^&*-+=|{}[]:;‘<>,.?/
Re-enter new password: #再次输入密码
Estimated strength of the password: 100
Do you wish to continue with the password provided?(Press y|Y for
Yes, any other key for No) : Y
By default, a MySQL installation has an anonymous user, allowing
 anyone to log into MySQL without having to have a user account
created for them. This is intended only for testing, and to make
the installation go a bit smoother. You should remove them before
moving into a production environment.
Remove anonymous users? (Press y|Y for Yes, any other key for No) :
Y #是否删除匿名用户，输入Y
Success.
Normally, root should only be allowed to connect from 'localhost'.
This ensures that someone cannot guess at the root password from the
 network.
Disallow root login remotely? (Press y|Y for Yes, any other key for
No) : Y #禁止root远程登录，输入Y
Success.
By default, MySQL comes with a database named 'test' that anyone can
 access.
This is also intended only for testing, and should be removed before
 moving into a production
environment.
Remove test database and access to it? (Press y|Y for Yes, any other
 key for No) : Y #是否删除test库和对它的访问权限，输入Y
- Dropping test database...
Success.
- Removing privileges on test database...
Success.
Reloading the privilege tables will ensure that all changes
made so far will take effect immediately.
Reload privilege tables now? (Press y|Y for Yes, any other key for
No) : Y #是否重新加载授权表，输入Y
Success.
```

```
All done!
```

6. 依次运行以下命令，安装PHP 7和一些所需的额外PHP扩展。

```
# yum install -y http://dl.iuscommunity.org/pub/ius/stable/CentOS/7/
x86_64/ius-release-1.0-14.ius.centos7.noarch.rpm
# yum -y update
# yum -y install php70u php70u-pdo php70u-mysqlnd php70u-opcache
php70u-xml php70u-gd php70u-mcrypt php70u-devel php70u-intl php70u-
mbstring php70u-bcmath php70u-json php70u-iconv
```

7. 查看PHP版本，以验证PHP是否已经成功安装。

```
# php -v
PHP 7.0.13 (cli) (built: Nov 10 2016 08:44:18) ( NTS )
Copyright (c) 1997-2016 The PHP Group
Zend Engine v3.0.0, Copyright (c) 1998-2016 Zend Technologies
 with Zend OPcache v7.0.13, Copyright (c) 1999-2016, by Zend
Technologies
```

8. 编辑配置文件/etc/php.ini：

   a. 运行命令 `vim /etc/php.ini`。

   b. 按 `i` 进入编辑模式。

   c. 在文件最后添加以下配置：

```
memory_limit = 128M #根据实际情况增加内存限制
date.timezone = Asia/Shanghai #设置时区为上海。
```

9. 重启Web服务进程。

```
# systemctl restart httpd
```

步骤2：创建数据库

按以下步骤创建数据库。

1. 创建数据库及用户：为Magento数据创建一个数据库和一个数据库用户，数据库和用户名可根据实际情况修改。

```
# mysql -u root -p
Enter password:
mysql> CREATE DATABASE magento; #根据实例情况替换magento
Query OK, 1 row affected (0.00 sec)
mysql> GRANT ALL ON magento.* TO YourUser@localhost IDENTIFIED BY '
YourPass'; #根据实际情况替换YourUser和YourPass
Query OK, 0 rows affected, 1 warning (0.00 sec)
mysql> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0.00 sec)
```

2. 运行 `exit` 退出MySQL。

3. （可选）验证新建的Magento数据库和用户是否可用。

```
# mysql -u YourUser -p
mysql> show databases;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| magento            |
+--------------------+
2 rows in set (0.00 sec)
mysql> exit
```

步骤3：安装配置**Composer**

Composer是PHP一个包管理和包依赖管理的工具。按以下步骤安装配置Composer。

1. 安装Composer。

```
# curl -sS https://getcomposer.org/installer | php
All settings correct for using Composer
Downloading 1.2.4...
Composer successfully installed to: /root/composer.phar
Use it: php composer.phar
```

2. 配置Composer全局使用。

```
# mv /root/composer.phar /usr/bin/composer
```

3. 测试命令是否可用。

```
# composer -v
______
/ ____/___  ____ ___  ____  ____  ________  _____
/ /   / __ \/ __ `__ \/ __ \/ __ \/ ___/ _ \/ ___/
/ /___/ /_/ / / / / / /_/ / /_/ (__  )  __/ /
\____/\____/_/ /_/ /_/ .___/\____/____/\___/_/
                    /_/
Composer version 1.2.4 2016-12-06 22:00:51
```

步骤4：安装配置**Magento**

您可以使用不同的方法安装Magento，也可以选择是否安装示例数据。如果安装Magento仅用于测试，您可以选择安装示例数据。如果是在生产环境中安装Magento，建议您安装全新的Magento，从头开始配置。

本部分介绍如何使用git下载Magento，然后使用Composer安装Magento。

1. 依次运行以下命令，通过 `git clone` 下载Magento。

```
# yum -y install git
# cd /var/www/html/
```

```
# git clone https://github.com/magento/magento2.git
```

2. （可选）将Magento切换到稳定版本。

   默认情况git下载安装Magento是一个最新的开发版本。如果您在生产环境中使用，建议切换到稳定版本，否则未来将无法升级安装。

```
# cd magento2 &&  git checkout tags/2.1.0 -b 2.1.0
Switched to a new branch '2.1.0'
```

3. 将安装文件移到Web服务器根目录下。否则，您只能通过 `http://[ECS实例公网IP地址]/magento2` 访问您的Magento站点。

```
# shopt -s dotglob nullglob && mv /var/www/html/magento2/* /var/www/html/ && cd ..
```

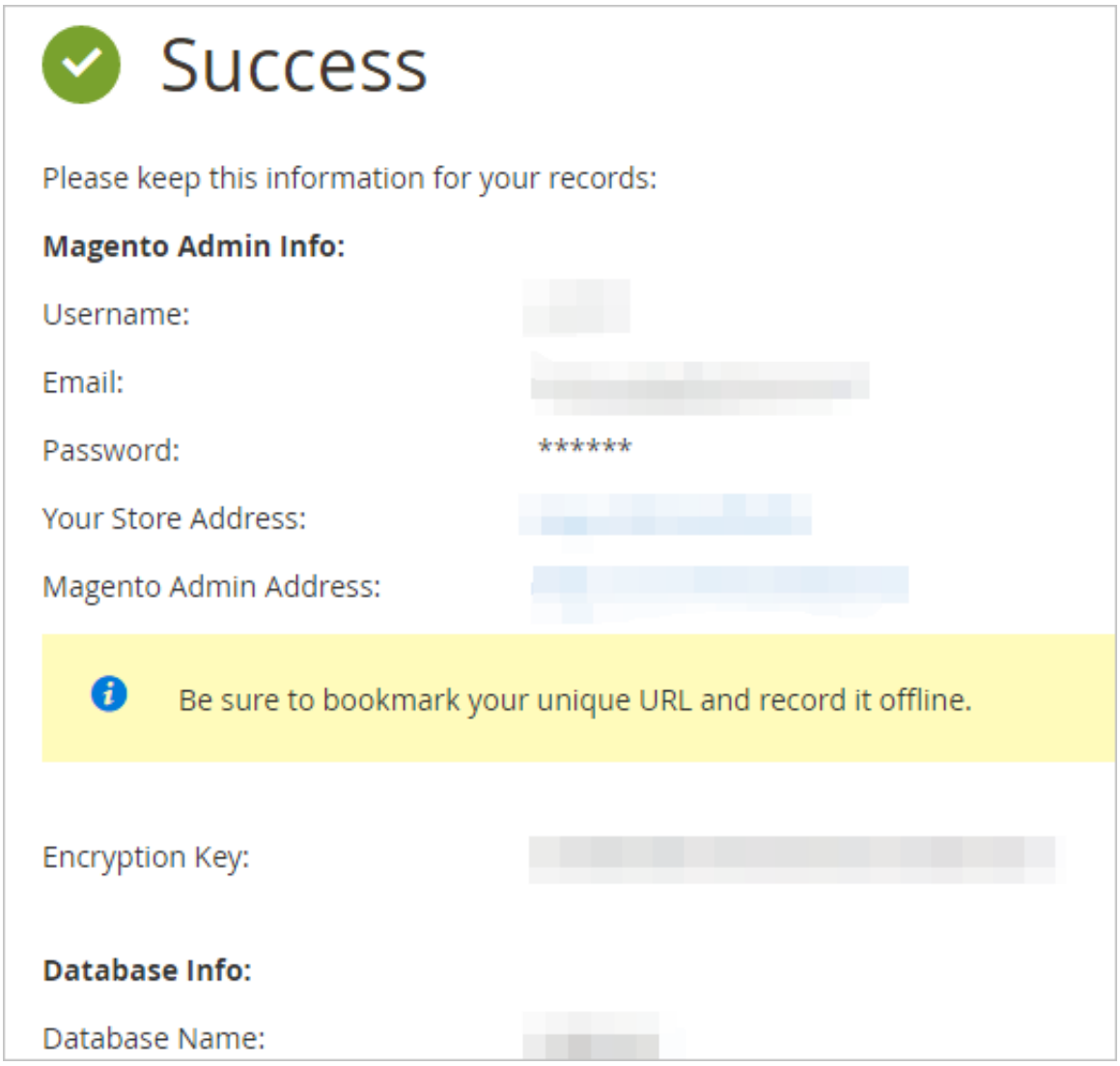
4. 设置Magento文件适当的权限。

```
# chown -R :apache /var/www/html
# find /var/www/html -type f -print0 | xargs -r0 chmod 640
# find /var/www/html -type d -print0 | xargs -r0 chmod 750
# chmod -R g+w /var/www/html/{pub,var}
# chmod -R g+w /var/www/html/{app/etc,vendor}
# chmod 750 /var/www/html/bin/magento
```

5. 运行 `composer install` 安装Magento。

6. 测试：在浏览器中访问 `http://[ECS实例公网IP地址]`，如果出现以下页面，说明Magento安装成功。

7. 单击 **Agree and Setup Magento** 开始配置Magento：按实际情况填写连接数据库信息、Web访问设置、定制商店、创建管理员账号。出现如下图所示的界面时，说明Magento配置完成。



步骤5：添加**cron**作业

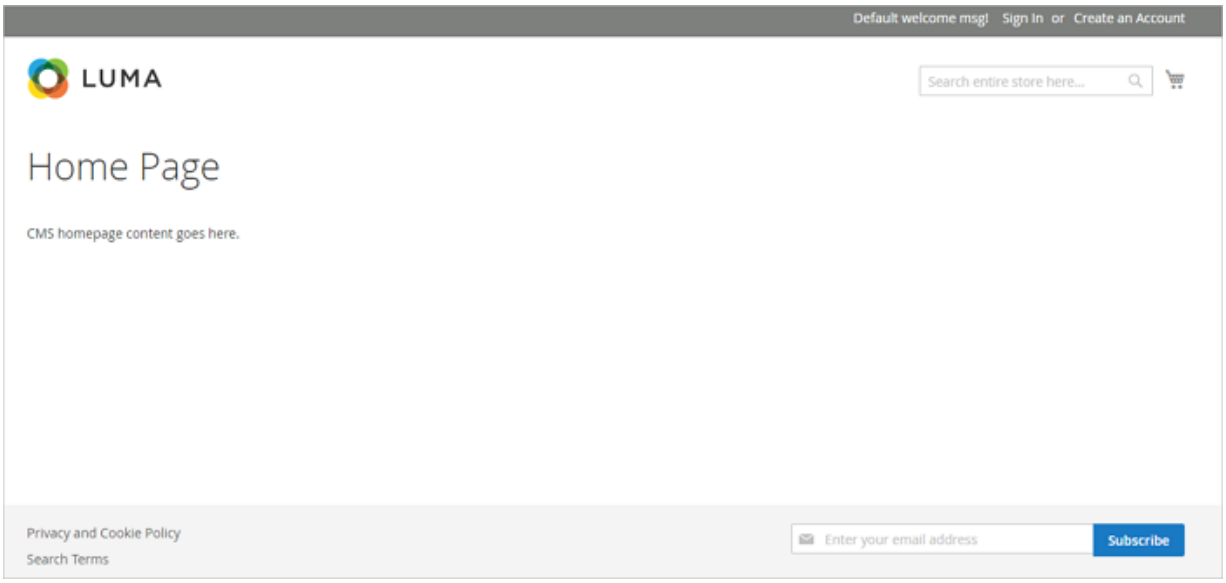
1. 运行 `crontab -u apache -e` 设置cron运行调度工作。
2. 添加以下内容。

```
*/10 * * * * php -c /etc /var/www/html/bin/magento cron:run
*/10 * * * * php -c /etc /var/www/html/update/cron.php
*/10 * * * * php -c /etc /var/www/html/bin/magento setup:cron:run
```

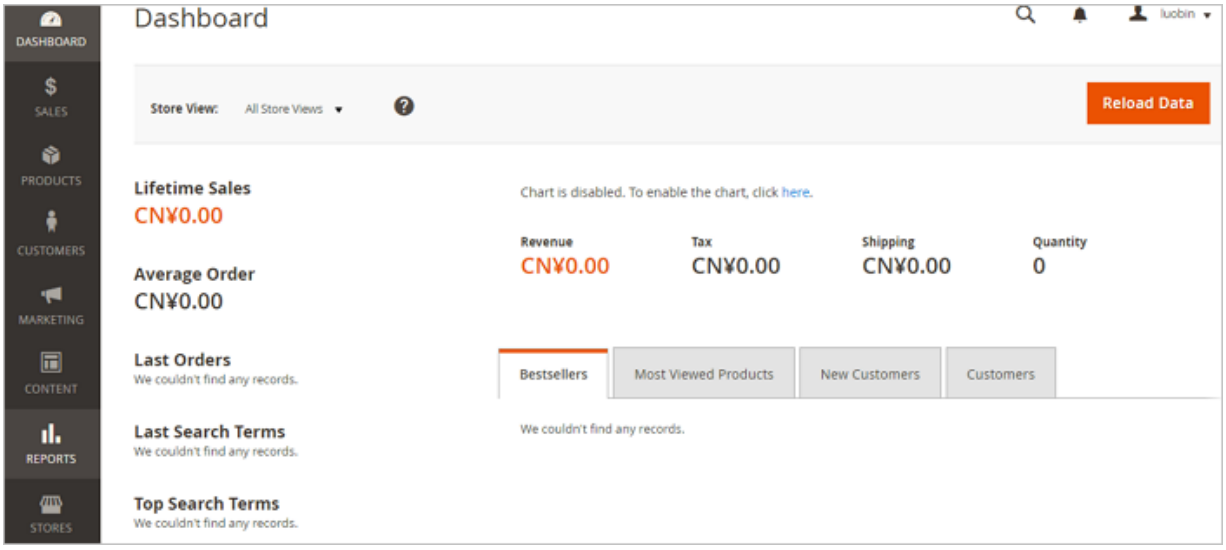
关于Magento上使用cron作业，请参见 *Magento*官方文档。

后续操作

访问 `http://[ECS实例公网IP]` 可以看到如下图所示的默认主页。



访问 `http://[ECS实例公网IP]/admin`，使用您在安装过程中设置的用户名和密码，成功登录管理面板后可看到如下界面。



更多Magento配置信息，请参见 *Magento*官方文档。

5 手动建站（Windows环境）

本节介绍如何使用阿里云镜像，一键部署 Web 环境，包括安装 IIS 组件（不包括 FTP 组件）、PHP 环境、重定向 Rewrite、MySQL、phpwind。该示例不需要更换系统盘。

创建实例以及配置安全组时，请避免以下情况：

- 请务必选择至少2GB或更高内存的实例规格，1 核 1GB 的实例规格无法启动 Mysql。
- 请务必完成安全组的详细配置，开通相关的端口。其中：端口21（FTP服务），端口3389，端口80请务必全部开通。

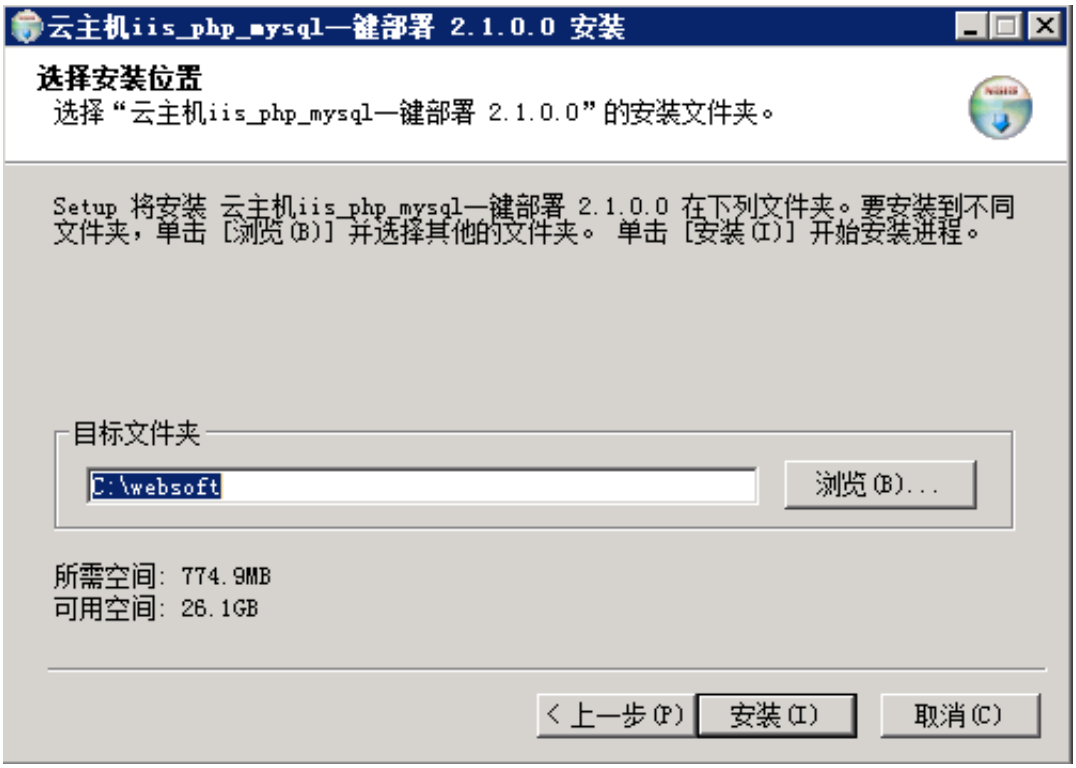
操作步骤

1. 在浏览器中打开阿里云的 云市场。
2. 搜索 阿里云**windows**一键安装**web**环境，然后购买该软件。
3. 登录 阿里云管理控制台。选择 产品与服务 > 云市场。
4. 单击 已购买的服务。在 阿里云**Windows**一键安装**Web**环境 的右侧，单击 下载，一键下载安装包。

5. 解压缩安装包。通过远程连接，将解压后的安装包拷贝到云服务器 ECS 实例上。

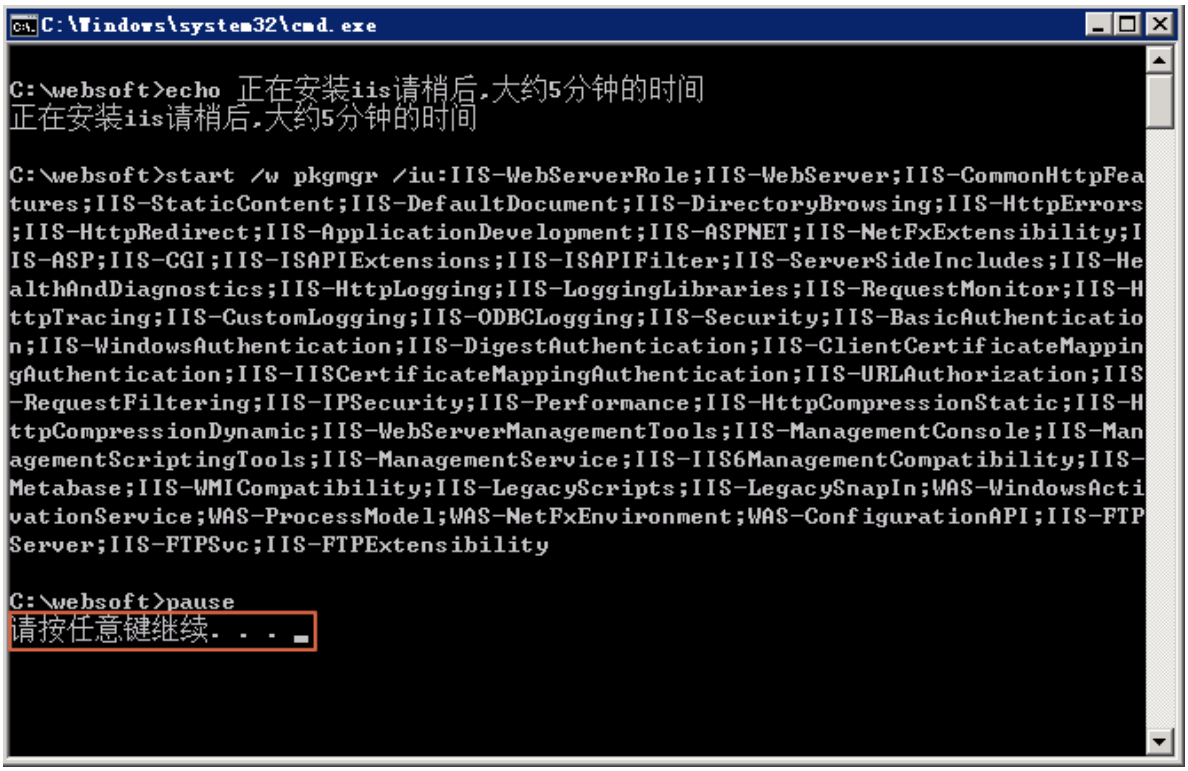
> 说明：
> 远程连接时，建议不要在控制台中完成。为了方便直接将软件进行本地机与远程机的复制粘贴，请参考 连接*Windows*实例 进行操作。

6. 安装包启动后，单击 下一步。
7. 指定安装目录，默认使用 **C:\websoft** 。然后单击 安装。



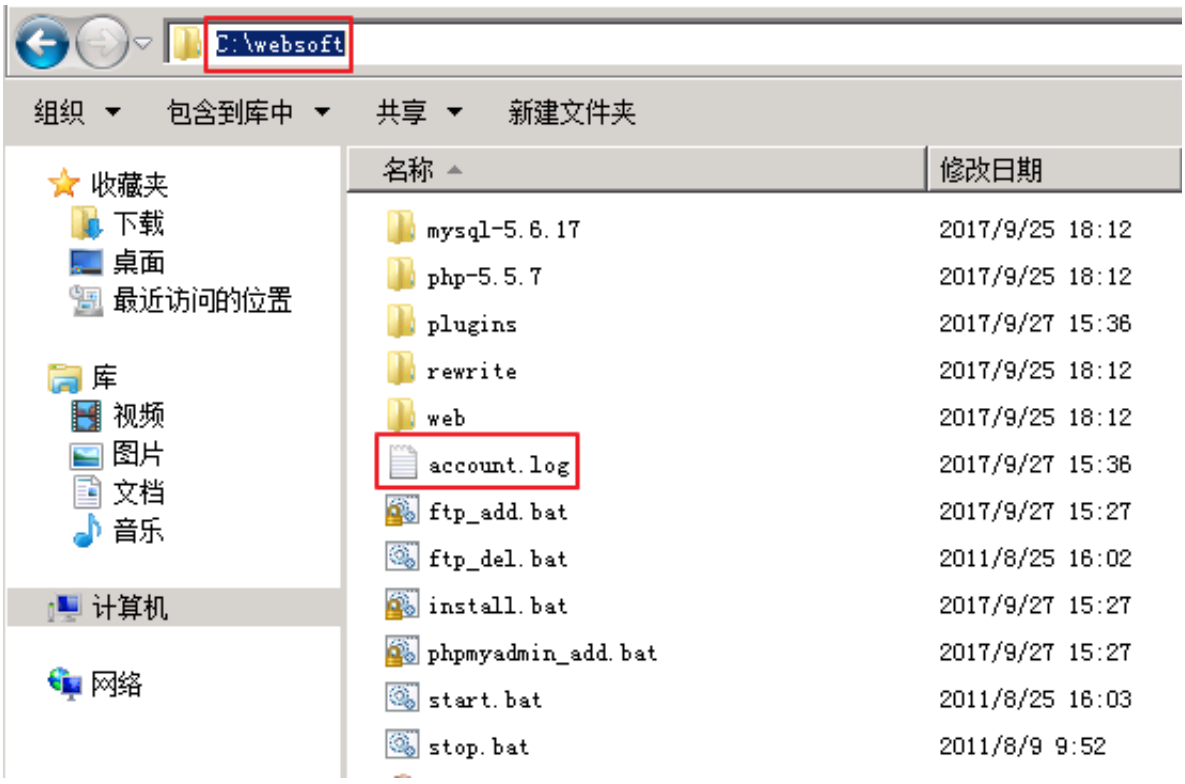
8. 系统会依次自动安装 IIS 组件（不包括 FTP 组件）、PHP 环境、重定向 Rewrite、MySQL、phpwind。

> 说明：
> 在安装过程中，每完成一项，需要您手动 按任意键继续，如下图所示。



9. 安装 MySQL 时，会报错 服务名无效（因之前未安装过MySQL），请直接忽略。

10. 将安装目录下的 account.log 中的 MySQL 密码复制粘贴到相应的数据库密码项中，并填写数据库名称、phpwind 管理员的登录密码，然后单击 **创建数据**。



11. phpwind 安装程序自动在 MySQL 中创建库和数据表。完成后直接进入 phpwind 登录页。phpwind 安装完毕。

12. 开始安装 FTP 服务。

13. 创建 FTP 用户，并将 FTP 站点的路径设置到 phpwind 的根目录，按任意键继续。

14. 安装 phpMyAdmin 。默认访问端口号为 8080，并在 IIS 中创建站点和连接池。

15. 完成后按任意键启动浏览器，打开 phpMyAdmin 站点，输入安装目录下的 account.log 文件中的 MySQL 账号信息进行登录。

登录后即可看到所有 MySQL 中的数据库，包括之前命名的 phpwind 数据库。如下图所示。

16.安装全部完毕。按任意键关闭窗口。

17.安装完成后，您可以打开 IIS，查看所有部署的服务和站点内容。

您已经成功部署了 Web 环境，可以制作和发布站点了。

6 部署LAMP

LAMP指Linux+Apache+MySQL/MariaDB+Perl/PHP/Python，是一组常用来搭建动态网站或者服务器的开源软件。它们本身都是各自独立的程序，但是因为常被放在一起使用，拥有了越来越高的兼容度，共同组成了一个强大的Web应用程序平台。

部署方式

您可以使用三种方式在云服务器ECS上部署LAMP：

- 镜像部署：方便快捷，适合不太了解Linux命令的用户。
- 一键安装包部署：适合对Linux命令有基本了解的用户。
- 手动部署：可以满足用户个性化部署需求，适合对Linux命令有基本了解的用户。

本文介绍如何在云服务器ECS上手动部署LAMP。

软件版本说明

本文操作的镜像和软件版本说明如下 ：

- 操作系统：CentOS 7.2 64位
- Apache：2.4.23
- MySQL：5.7.17
- PHP：7.0.12

前提条件

在部署之前，需要确认：

- 实例所在安全组已经 放行了服务所需的端口：

服务	SSH	HTTP	MySQL
端口	TCP 22	TCP 80	TCP 3306

- 已经远程连接到实例。

准备工作

设置防火墙

CentOS 7.2系统默认开启防火墙firewalld。您可以关闭firewalld放行80、22等端口。

说明：

您也可以参考 *firewalld* 官方文档 在防火墙里放行这些端口。

1. 运行命令关闭防火墙。

```
systemctl stop firewalld.service
```

2. 运行命令关闭防火墙开机自启动。

```
systemctl disable firewalld.service
```

安装软件

运行命令下载软件、编辑和解压文件。

```
yum install -y wget vim unzip
```

操作步骤

按以下步骤部署LAMP。

步骤1：编译安装Apache

1. 运行命令安装相关依赖包。

```
yum install -y gcc gcc-c++ autoconf libtool
```

2. 依次运行以下命令安装apr。

```
cd /usr/local/src/
wget http://oss.aliyuncs.com/aliyunecs/onekey/apache/apr-1.5.0.tar.
gz
tar zxvf apr-1.5.0.tar.gz
cd apr-1.5.0
./configure --prefix=/usr/local/apr
make && make install
```

3. 依次运行以下命令安装apr-util。

```
cd /usr/local/src/
wget http://oss.aliyuncs.com/aliyunecs/onekey/apache/apr-util-1.5.3.
tar.gz
tar zxvf apr-util-1.5.3.tar.gz
cd apr-util-1.5.3
./configure --prefix=/usr/local/apr-util --with-apr=/usr/local/apr
make && make install
```

4. 依次运行以下命令安装pcre。

```
cd /usr/local/src/
wget http://zy-res.oss-cn-hangzhou.aliyuncs.com/pcre/pcre-8.38.tar.
gz
tar zxvf pcre-8.38.tar.gz
cd pcre-8.38
./configure --prefix=/usr/local/pcre
```

```
make && make install
```

5. 依次运行以下命令编译安装Apache。

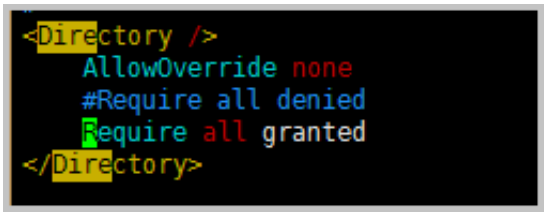
```
cd /usr/local/src/
wget http://zy-res.oss-cn-hangzhou.aliyuncs.com/apache/httpd-2.4.23.
tar.gz
tar zxvf httpd-2.4.23.tar.gz
cd httpd-2.4.23
./configure \
--prefix=/usr/local/apache --sysconfdir=/etc/httpd \
--enable-so --enable-cgi --enable-rewrite \
--with-zlib --with-pcre=/usr/local/pcre \
--with-apr=/usr/local/apr \
--with-apr-util=/usr/local/apr-util \
--enable-mods-shared=most --enable-mpms-shared=all \
--with-mpm=event
make && make install
```

6. 修改httpd.conf配置文件参数：

   a. 运行 `cd /etc/httpd/` 切换到/etc/httpd/目录。

   b. 运行 `vim httpd.conf` 打开httpd.conf文件，按 `i` 键进入编辑模式。

   c. 找到 `Directory` 参数，注释掉 `Require all denied`，并添加 `Require all granted`。

   d. 找到 `ServerName` 参数，添加 `ServerName localhost:80`。

```
#
#
# ServerAdmin: Your address, where problems with the server should be
# e-mailed.  This address appears on some server-generated pages, such
# as error documents.  e.g. admin@your-domain.com
#
ServerAdmin you@example.com

#
# ServerName gives the name and port that the server uses to identify itself.
# This can often be determined automatically, but we recommend you specify
# it explicitly to prevent problems during startup.
#
# If your host doesn't have a registered DNS name, enter its IP address here.
#
#ServerName www.example.com:80
ServerName localhost:80
#
# Deny access to the entirety of your server's filesystem. You must
# explicitly permit access to web content directories in other
# <Directory> blocks below.
```

e. 设置 PidFile 路径：在文件最后添加 PidFile "/var/run/httpd.pid"。

f. 按 Esc 键退出编辑模式，输入 :wq 保存并关闭 httpd.conf 文件。

7. 依次执行以下命令启动Apache服务并验证。

```
cd /usr/local/apache/bin/
./apachectl start
netstat -tnlp                                    #查看服务是否开启
```

如果返回以下结果，说明Apache服务已经成功启动。

```
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
PID/Program name
tcp        0      0 0.0.0.0:80              0.0.0.0:*               LISTEN
571/httpd
tcp        0      0 0.0.0.0:22              0.0.0.0:*               LISTEN
3916/sshd
```

在本地机器的浏览器中输入ECS实例公网IP地址，如果出现如图所示信息，说明Apache服务安装成功。

8. 设置开机自启动。

   a. 运行 `vim /etc/rc.d/rc.local` 打开rc.local文件，按 `i` 进入编辑模式。

   b. 添加 `/usr/local/apache/bin/apachectl start`。

```
#!/bin/bash
# THIS FILE IS ADDED FOR COMPATIBILITY PURPOSES
#
# It is highly advisable to create own systemd services or udev rules
# to run scripts during boot instead of using this file.
#
# In contrast to previous versions due to parallel execution during boot
# this script will NOT be run after all other services.
#
# Please note that you must run 'chmod +x /etc/rc.d/rc.local' to ensure
# that this script will be executed during boot.

touch /var/lock/subsys/local
/usr/local/apache/bin/apachectl start
~
```

   c. 按 `Esc` 键退出编辑模式，输入 `:wq` 保存并关闭rc.local文件。

9. 设置环境变量。

   a. 运行 `vi /root/.bash_profile` 打开文件，按 `i` 进入编辑模式。

   b. 将 `PATH=$PATH:$HOME/bin` 修改为 `PATH=$PATH:$HOME/bin:/usr/local/apache/bin`。

   c. 按 `Esc` 键退出编辑模式，输入 `:wq` 保存并关闭文件。

   d. 运行 `source /root/.bash_profile` 重新执行文件。

步骤2：编译安装**MySQL**

1. 依次执行以下命令检查系统中是否存在使用rpm安装的MySQL或者MariaDB。

```
rpm -qa | grep mysql
```

```
rpm -qa | grep mariadb
```

如果已经安装，则运行下面第一个命令卸载。若第一个命令卸载不成功，则运行下面第二个命令强制卸载。

```
rpm -e 软件名     #注意：这里的软件名必须包含软件的版本信息，如rpm -e mariadb
-libs-5.5.52-1.el7.x86_64。一般使用此命令即可卸载成功。
rpm -e --nodeps 软件名    #注意：这里的软件名必须包含软件的版本信息，如rpm -e
 --nodeps mariadb-libs-5.5.52-1.el7.x86_64。
```

卸载后，再用 `rpm -qa|grep mariadb` 和 `rpm -qa|grep mysql` 查看结果。如下图所示。

```
[root@iZm5e24s637uue17hkz1gkZ ~]# rpm -qa | grep mysql
[root@iZm5e24s637uue17hkz1gkZ ~]# rpm -qa | grep mariadb
[root@iZm5e24s637uue17hkz1gkZ ~]#
[root@iZm5e24s637uue17hkz1gkZ ~]#
```

2. 依次运行以下命令安装 MySQL。

```
yum install -y libaio-*                              #安装依赖
mkdir -p /usr/local/mysql
cd /usr/local/src
wget http://zy-res.oss-cn-hangzhou.aliyuncs.com/mysql/mysql-5.7.17-
linux-glibc2.5-x86_64.tar.gz
tar -xzvf mysql-5.7.17-linux-glibc2.5-x86_64.tar.gz
mv mysql-5.7.17-linux-glibc2.5-x86_64/* /usr/local/mysql/
```

3. 依次运行以下命令建立mysql组和用户，并将mysql用户添加到mysql组。

```
groupadd mysql
useradd -g mysql -s /sbin/nologin mysql
```

4. 运行命令初始化MySQL数据库。

```
/usr/local/mysql/bin/mysqld --initialize-insecure --datadir=/usr/
local/mysql/data/ --user=mysql
```

5. 更改MySQL安装目录的属性：`chown -R mysql:mysql /usr/local/mysql`。

6. 依次运行以下命令设置开机自启动。

```
cd /usr/local/mysql/support-files/
cp mysql.server  /etc/init.d/mysqld
chmod +x /etc/init.d/mysqld              # 添加执行权限
vim /etc/rc.d/rc.local
```

在 rc.local 文件中添加 `/etc/init.d/mysqld start`。

7. 设置环境变量。

   a. 运行 `vi /root/.bash_profile` 打开文件，按 `i` 进入编辑模式。

b. 将 `PATH=$PATH:$HOME/bin:/usr/local/apache/bin` 修改为 `PATH=$PATH:$HOME/bin:/usr/local/apache/bin:/usr/local/mysql/bin:/usr/local/mysql/bin` 。

> 说明：
> 此处是在编译安装 Apache的环境变量的基础上再进行修改。

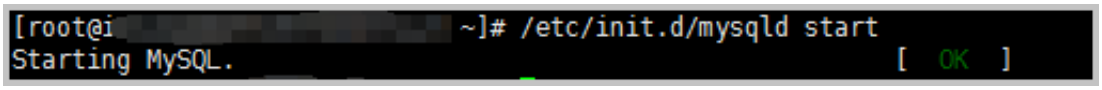
c. 按 `Esc` 键退出编辑模式，输入 `:wq` 保存并关闭文件。

d. 运行 `source /root/.bash_profile` 重新执行文件。

8. 启动 MySQL 数据库。

```
/etc/init.d/mysqld start
```

出现如下截图所示信息，表示MySQL启动成功。

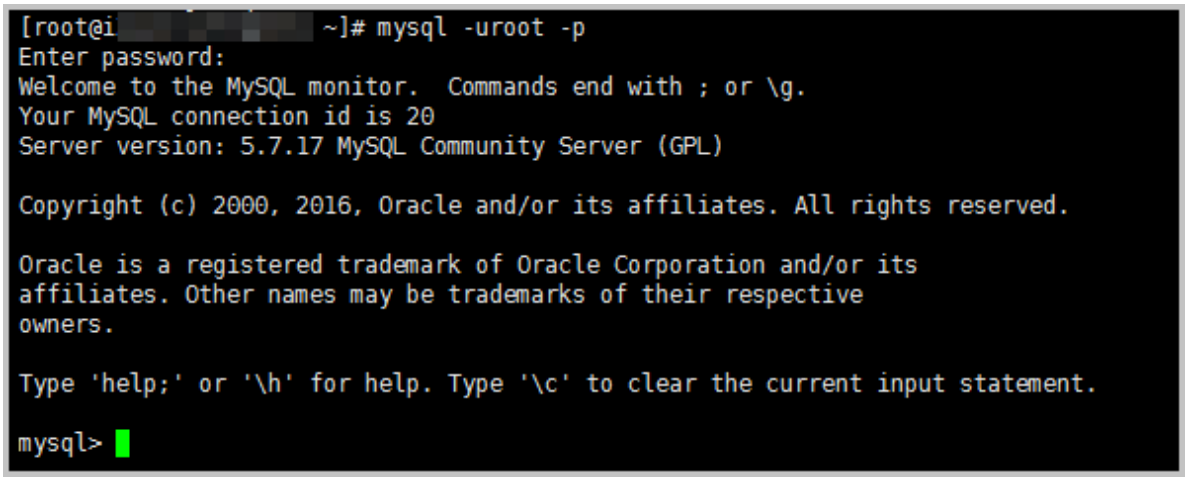


9. 修改MySQL的root用户密码：初始化后MySQL为空密码可直接登录，为了保证安全性需要修改MySQL的root用户密码。运行以下命令，并按界面提示设置密码。

```
mysqladmin -u root password
```

10. 测试登录MySQL数据库。

```
mysql -uroot -p #-p和密码之间无空格
```



11. 运行 `\q` 退出MySQL。

步骤3. 编译安装 PHP

1. 依次运行以下命令安装依赖。

```
yum install epel-release php-mcrypt libmcrypt libmcrypt-devel
libxml2-devel  openssl-devel  libcurl-devel libjpeg.x86_64 libpng
.x86_64 freetype.x86_64 libjpeg-devel.x86_64 libpng-devel.x86_64
freetype-devel.x86_64  libjpeg-turbo-devel   libmcrypt-devel   mysql
-devel  -y
 wget http://zy-res.oss-cn-hangzhou.aliyuncs.com/php/php-7.0.12.tar.
gz
 tar zxvf php-7.0.12.tar.gz
 cd php-7.0.12
 ./configure \
 --prefix=/usr/local/php \
 --enable-mysqlnd \
 --with-mysqli=mysqlnd --with-openssl \
 --with-pdo-mysql=mysqlnd \
 --enable-mbstring \
 --with-freetype-dir \
 --with-jpeg-dir \
 --with-png-dir \
 --with-zlib --with-libxml-dir=/usr \
 --enable-xml  --enable-sockets \
 --with-apxs2=/usr/local/apache/bin/apxs \
 --with-mcrypt  --with-config-file-path=/etc \
 --with-config-file-scan-dir=/etc/php.d \
 --enable-maintainer-zts \
 --disable-fileinfo
 make && make install
```

2. 运行命令复制配置文件。

```
cp php.ini-production /etc/php.ini
```

3. 编辑Apache配置文件 httpd.conf，以Apache支持PHP。

   a. 运行 `vim /etc/httpd/httpd.conf` 打开文件，按 `i` 进入编辑模式。

   b. 在配置文件最后添加如下二行代码。

```
AddType application/x-httpd-php  .php
AddType application/x-httpd-php-source  .phps
```

   c. 定位到 `DirectoryIndex index.html`，修改为 `DirectoryIndex index.php index.html`。

   > 说明：
   >
   > 如果文件中没有 `DirectoryIndex index.html`，则添加上述代码。

   d. 按 `Esc` 键退出编辑模式，输入 `:wq` 保存并关闭文件。

4. 重启Apache服务：

```
/usr/local/apache/bin/apachectl restart
```

5. 测试是否能够正常解析PHP。

a. 依次运行以下命令，找开index.php文件。

```
cd  /usr/local/apache/htdocs/
vim index.php
```

b. 按 `i` 键进入编辑模式，并添加以下内容。

```
<?php
phpinfo();
?>
```

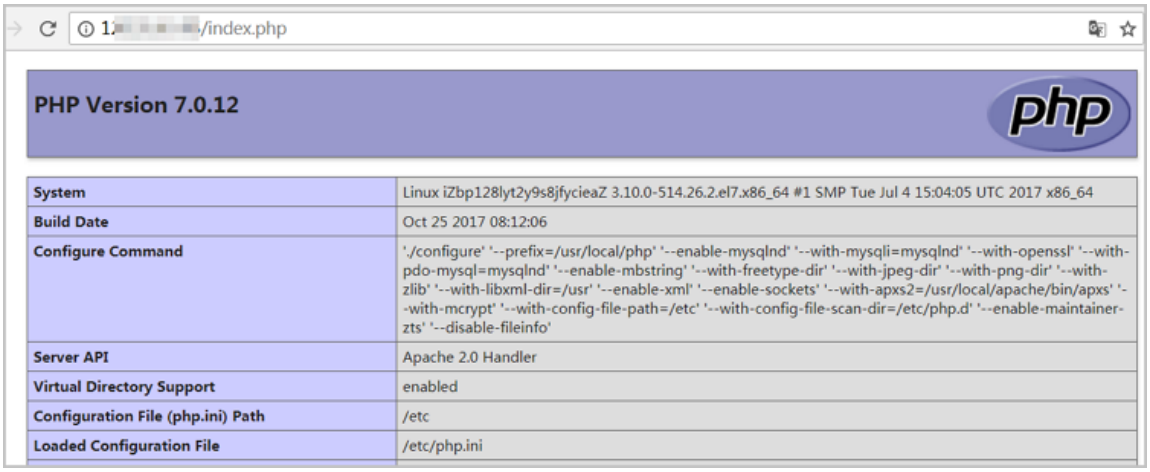
c. 按 `Esc` 键退出编辑模式，并输入 `:wq` 保存并关闭文件。

d. 重启Apache服务：

```
/usr/local/apache/bin/apachectl restart
```

e. 在本地机器的浏览器里输入 `http://实例公网 IP/index.php`。

如果出现以下页面表示PHP解析成功。



步骤4. 安装phpMyAdmin

依次运行以下命令安装phpMyAdmin。

```
mkdir -p /usr/local/apache/htdocs/phpmyadmin
cd /usr/local/src/
wget http://oss.aliyuncs.com/aliyunecs/onekey/phpMyAdmin-4.1.8-all-languages.zip
unzip phpMyAdmin-4.1.8-all-languages.zip
mv phpMyAdmin-4.1.8-all-languages/*  /usr/local/apache/htdocs/phpmyadmin
```

在本地机器浏览器输入 `http://实例公网 IP/phpmyadmin` 访问phpMyAdmin登录页面。如果出现以下页面，说明phpMyAdmin安装成功。输入MySQL的用户名和密码即可登录。

相关链接

您可通过 云中沙箱平台 上体验本文档描述的操作。

更多开源软件尽在 云市场。

7 搭建Joomla基础管理平台

Joomla是一套知名的内容管理系统。Joomla是使用PHP语言加上Mysql数据开发的软件系统，Joomla的最新版本是3.x，这一版本实现了许多技术上的优化调整，是目前的稳定版本。

本文主要说明如何在阿里云ECS上搭建Joomla基础管理平台。使用的操作系统为Linux CentOS 6.5 64位。

适用对象

适用于熟悉 ECS，熟悉 Linux 系统， ECS 实例搭建刚开始使用阿里云进行建站的用户。

基本流程

使用云服务器 ECS 搭建 Joomla 平台的操作步骤如下：

购买 ECS 实例，如果需要备案网站，请选择包年包月付费模式。对于个人使用的小型网站，一台云服务器 ECS 实例可以满足需求。

这里只介绍新购实例。如果您有镜像，可以使用自定义镜像创建实例。

> 说明：
> 这个文档中描述的实例将结合 云市场 的 joomla 镜像 使用，而这个产品目前仅支持 CentOS、Ubuntu 和 Aliyun Linux。

操作步骤

1. 登录 云服务器管理控制台。如果尚未注册，单击 免费注册。
2. 选择 云服务器ECS > 实例。单击 创建实例。

3. 选择付费方式：包年包月或按量付费。因为目前只有包年包月的 ECS 可以备案，如果您需要备案网站，请选择 包年包月。

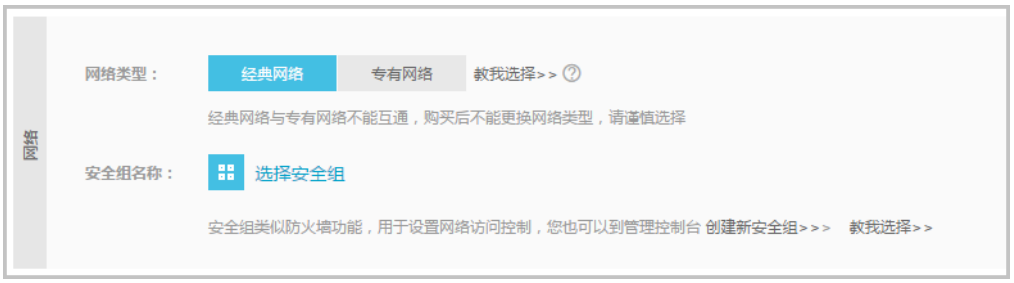
4. 选择地域。所谓地域，是指实例所在的地理位置。您可以根据所在的地理位置选择地域。地域与用户距离越近，延迟相对越少，下载速度相对越快。

   例如，如果您的网站访问者都分布在北京地区，则可以选择 华北 2。

   说明：

   - 实例创建完成后，不支持更换地域。
   - 不同地域提供的可用区数量、实例系列、存储类型、实例价格等也会有所差异。请根据您的业务需求进行选择。

5. 选择网络类型。对于建站的用户，选择 经典网络 即可。然后选择安全组。



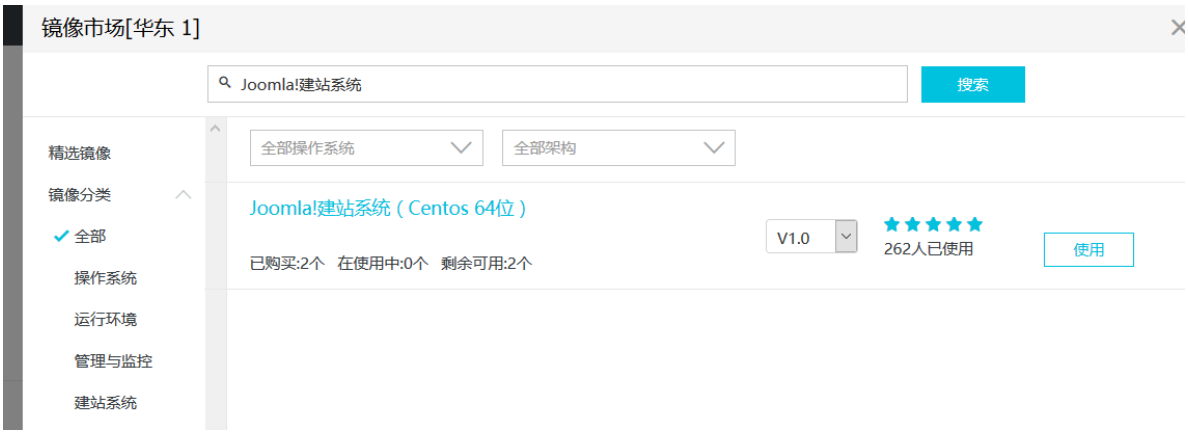
6. 选择实例，根据您网站的访问量选择实例规格（CPU、内存）。对于个人网站，1 核 2GB 或 2 核 4GB 一般能够满足需求。关于实例规格的详细介绍，请参考实例规格族。实例系列 II 是实例系列 I 的升级版，提供更高的性能，推荐使用。

7. 选择网络带宽。因为创建的实例需要访问公网，如果选择 0 Mbps，则不分配公网 IP，实例将无法访问公网，所以，无论是 按固定带宽 还是 按使用流量 付费，带宽都不能选择 0 Mbps。

   - 按固定带宽付费。

   - 按使用流量付费。

8. 选择镜像。您可以在镜像里面点击镜像市场，再点击从镜像市场选择，搜索 *Joomla!*建站系统，然后点击使用就可以使用镜像。



9. 选择 系统盘 和 数据盘。您可以创建全新的磁盘作为数据盘，也可以选择 用快照创建磁盘，将快照的数据直接复制到磁盘中作为数据盘。

10. 设置实例的登录密码和实例名称。请务必牢记密码。您也可以在创建完成后再设置密码。

11. 设置购买的时长和数量。

12. 单击页面右侧价格下面的 立即购买。

13. 确认订单并付款。

实例创建好之后，您会收到短信和邮件通知，告知您的实例名称、公网 IP 地址、内网 IP 地址等信息。您可以使用这些信息登录和管理实例。

很多重要的信息都是通过绑定手机的短信接收，并且重要的操作（如重启、停止等）都需要手机接收验证码，因此请务必保持绑定手机通信畅通。

部署 Web 环境

通过 ECS 更换系统盘，来更换所需要的镜像，这里选择 php 运行环境（centos 64 位 | php5.4| nginx1.4|joomla）。

- 镜像版本说明操作系统：centos 6.5 64 位。

  镜像版本 V1.0 软件明细：Nginx1.4.7-PHP 5.4.27-MySQL5.5.37-FTP2.2.2- Joomla!3.3.3 1.2、镜像安装说明。
- 镜像环境里相应软件的安装，是基于阿里云 linux 版的一键安装包源码 1.3.0 版本，在此基础上修改、优化了相应功能，编译安装完成。
- 在镜像环境中，/root/sh-1.3.0-centos-joomla.zip 是安装镜像环境的脚本。您可以在 centos 6.5 系统中自行采用此脚本安装，安装后的环境跟镜像里初始化的环境一致。

   说明：

  如果采用此脚本安装镜像环境，需要 chmod 777 -R sh-1.3.0-centos-joomla 赋予 777 安装权限。
- 在镜像环境中出于安全考虑，joomla 默认设置页面只容许 127.0.0.1 访问，/root/目录下提供一个 joomla_opennet.sh 的脚本 。用户运行此脚本后，可以通过外网访问 joomla 的默认设置页面。
- 在镜像环境中，/root/sh-1.3.0-centos-joomla 是安装环境的主目录，镜像中的环境是在此目录下编译安装的。

mysql 以及 ftp 的密码

1. 密码存储位置： /alidata/account.log 文件中。
2. 查看密码.

   进入服务器的系统中，可以在任意的目录下，执行以下命令

   ```
   cat /alidata/account.log
   ```

    说明：

   cat 后有空格。
3. 修改 ftp 的密码。

用 root 用户登录系统，然后执行下面命令。

```
passwd www 然后输入您的 ftp 新密码。
```

4. 修改 mysql 的密码。

```
mysqladmin -uroot -p 旧密码 password 新密码
```

说明：

-p 和旧密码之间没有空格，password 和新密码之间有空格。

软件目录及配置列表

软件的主目录：/alidata

web 主目录：/alidata/www

ftp 主目录：/alidata/www

nginx 主目录：/alidata/server/nginx

nginx 配置文件主目录：/alidata/server/nginx/conf

php 主目录：/alidata/ server/php

php 配置文件主目录：/alidata/ server/php/etc

mysql 主目录：/alidata/server/mysql

mysql 配置文件：/etc/my.cnf

joomla 中文支持包存放目录：/alidata/res

日志目录：

/alidata/log/nginx 为 nginx 存放日志主目录

/alidata/log/php 为 php 存放日志主目录

/alidata/log/mysql 为 mysql 存放日志主目录 init 目录

/alidata/init 为当用户用镜像创建系统后，当且仅当用户在第一次启动系统的时候，调用此目录下的脚本来初始化 ftp 及 mysql 的密码（随机密码）。

软件操作命令汇总

/etc/init.d/mysqld start|stop|restart

/etc/init.d/php-fpm start|stop|restart

/etc/init.d/vsftpd start|stop|restart

/etc/init.d/nginx start|stop|restart

关于卸载

关于卸载镜像环境中安装的软件，可以参考如下命令。

```
cd /root/sh-1.3.0-centos-joomla
./uninstall.sh
```

说明：

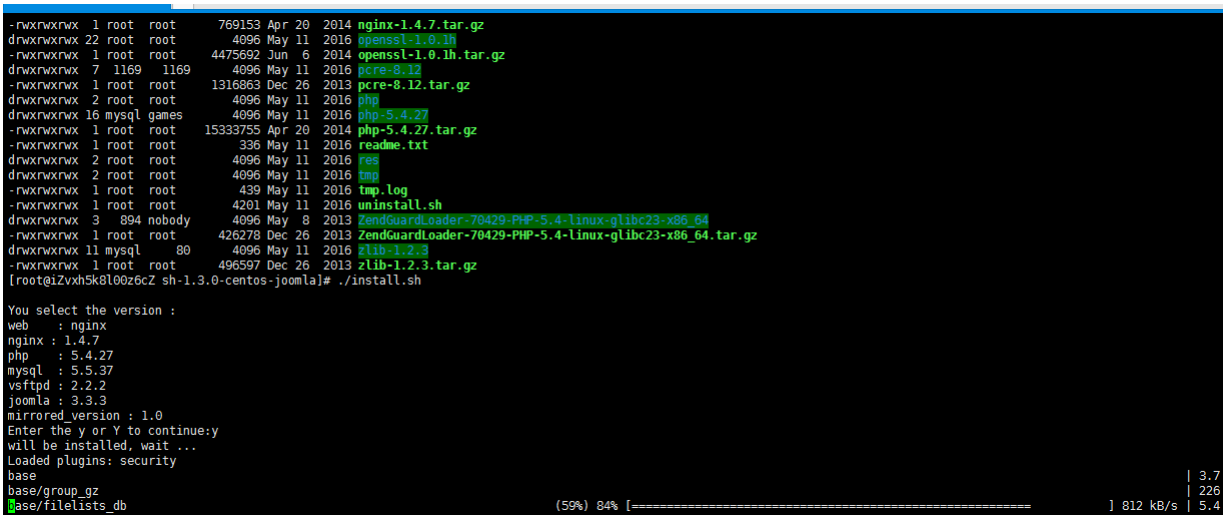
- 执行以上操作会清理环境的 /alidata 目录，请卸载前自行备份好相应数据。
- 如果不小心删除了 /root/sh-1.3.0-centos-joomla,可以解压缩 /root/sh-1.3.0-centos-joomla.zip 参考以下命令。

```
cd
unzip sh-1.3.0-centos-joomla.zip
chmod 777 -R sh-1.3.0-centos-joomla
cd sh-1.3.0-centos-joomla
./uninstall
```

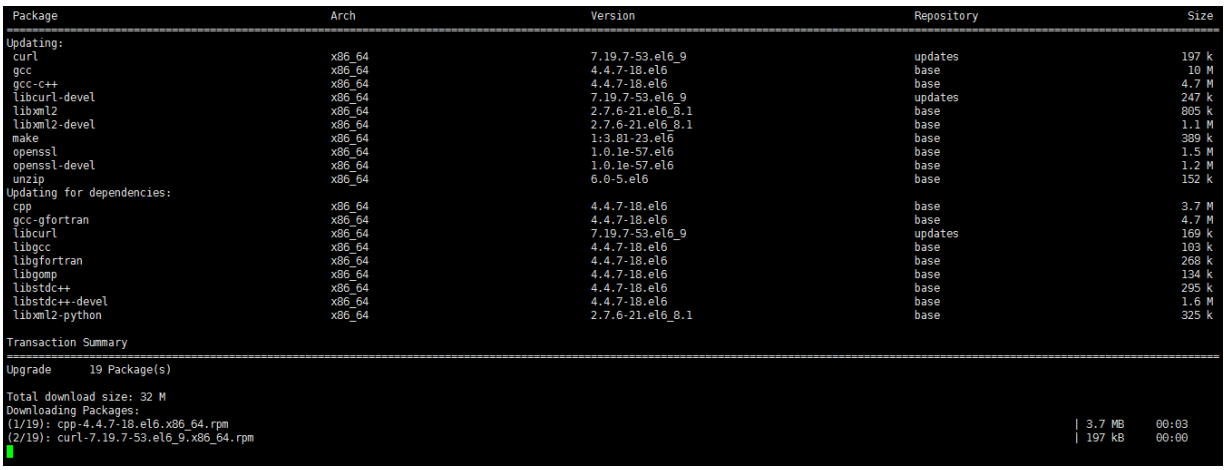
在 centos6.5 系统中自行安装

/root/sh-1.3.0-centos-joomla.zip 是安装镜像环境的脚本。值得注意的是，如果采用此脚本安装镜像环境，需要 chmod 777 -R sh-1.3.0-centos-joomla 赋予 777 安装权限,然后cd sh-1.3.0-centos-joomla目录下执行 ./install 开始安装。

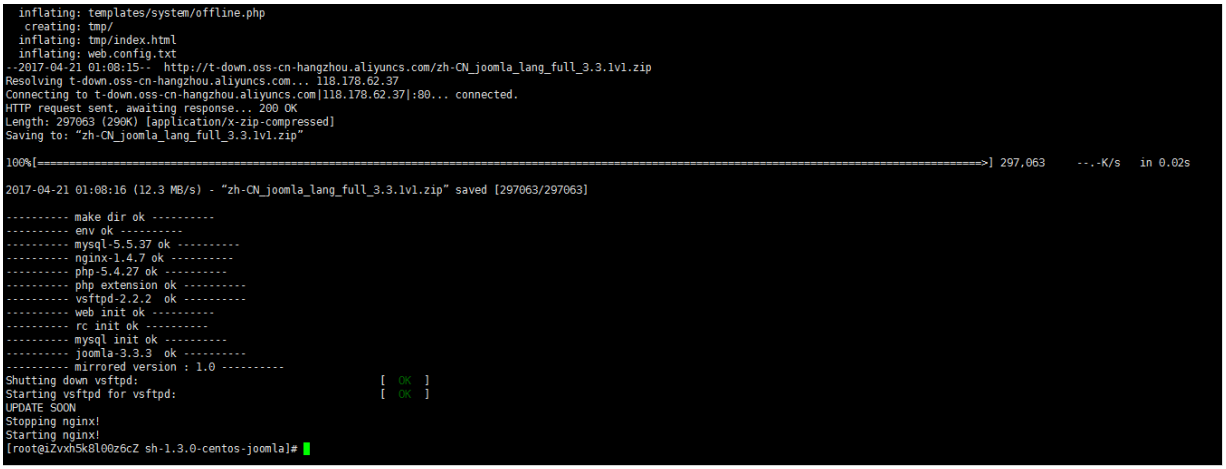
根据提示输入 y。



持续安装中。



安装结束出现以下界面。




```
[root@iZvxh5k8l00z6cZ ~]# netstat -luntp
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address               Foreign Address             State       PID/Program name
tcp        0      0 0.0.0.0:80                  0.0.0.0:*                   LISTEN      28115/nginx
tcp        0      0 0.0.0.0:21                  0.0.0.0:*                   LISTEN      28106/vsftpd
tcp        0      0 0.0.0.0:22                  0.0.0.0:*                   LISTEN      941/sshd
tcp        0      0 127.0.0.1:9000              0.0.0.0:*                   LISTEN      28034/php-fpm
tcp        0      0 0.0.0.0:3306                0.0.0.0:*                   LISTEN      26553/mysqld
udp        0      0 120.26.213.174:123          0.0.0.0:*                               949/ntpd
udp        0      0 10.117.50.27:123            0.0.0.0:*                               949/ntpd
udp        0      0 127.0.0.1:123               0.0.0.0:*                               949/ntpd
udp        0      0 0.0.0.0:123                 0.0.0.0:*                               949/ntpd
[root@iZvxh5k8l00z6cZ ~]#
```

80、21、9000、3306 等端口都已开启。

配置外网访问

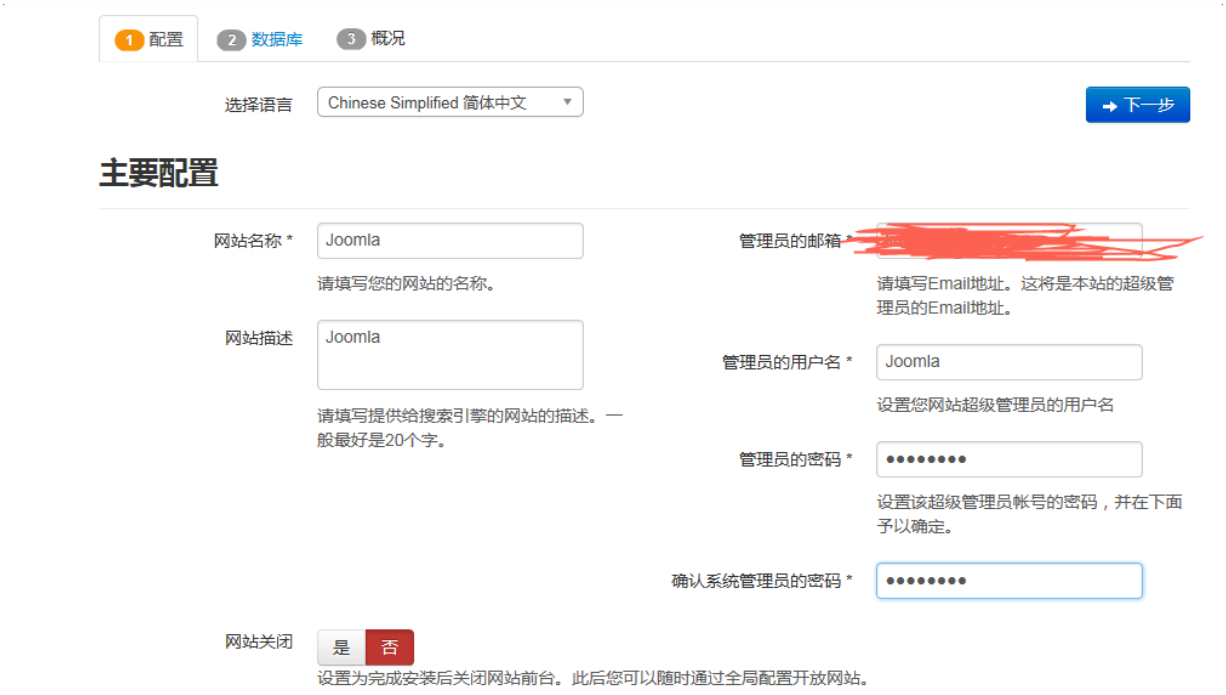
在镜像环境中处于安全考虑，joomla 默认页面只允许 127.0.0.1 访问，/root/ 目录下提供了一个 joomla_opennet.sh 的脚本。用户运行之后，可通过外网访问 joomla 的默认设置页面。

运行脚本文件。

```
/root/joomla_opennet.sh
```

配置joomla

初次使用镜像，运行 /root/joomla_opennet.sh 文件，在游览器中输入 *http://ip*，回车即可看到 joomla 的初始化界面。



选择语言，并填写相关内容，单击 下一步。

选择mysql数据库，填写相关权限后，单击 下一步。

查看相关配置是否符合，确认完毕单击 安装。

安装完毕

进入服务器 /alidata/www/default 目录下删除 installation 目录。

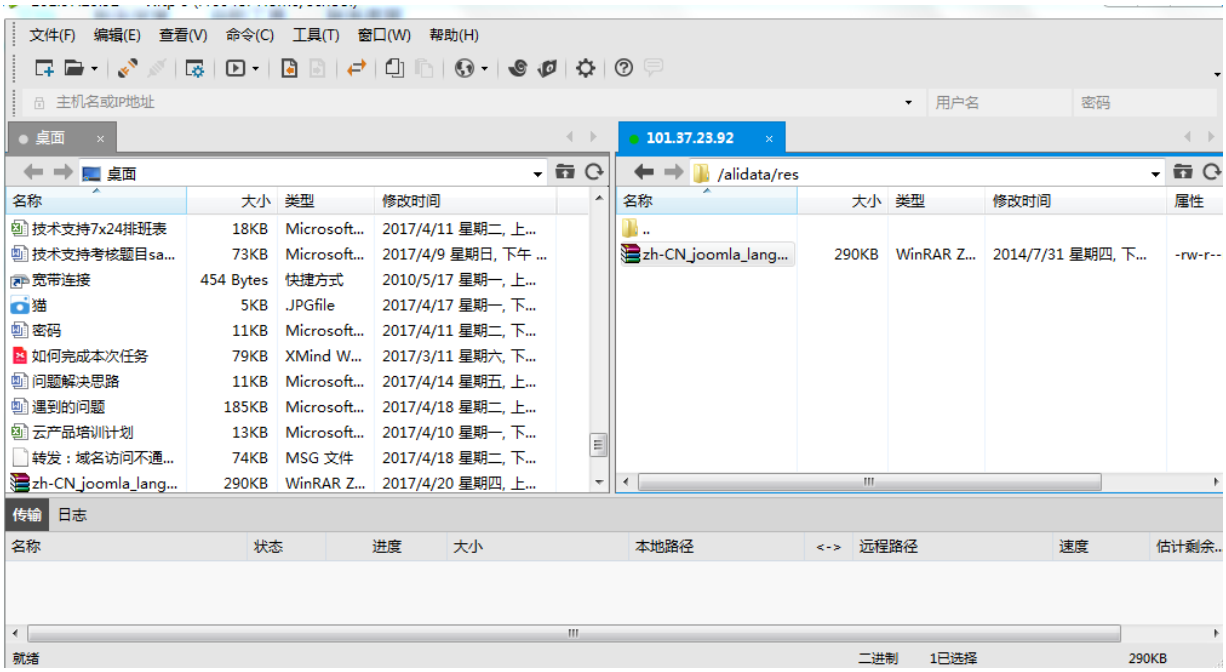
```
cd /alidata/www/default
rm -rf installation/
```

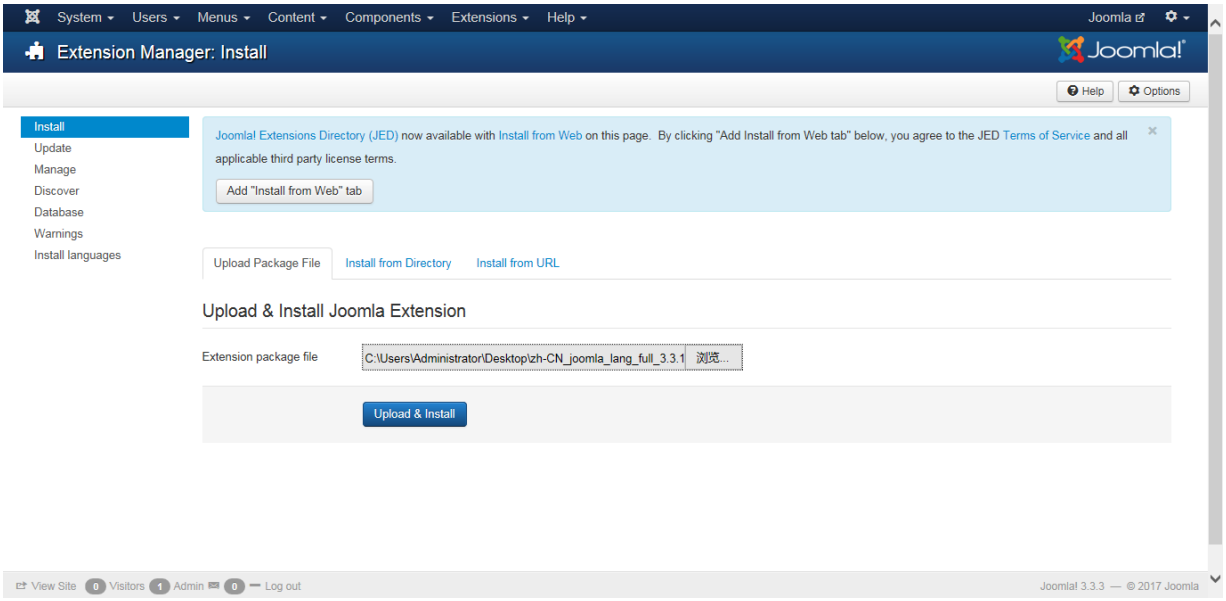
至此，joomla 搭建完成。

访问前端网站 *http://ip*，访问后台管理 *http://ip/administrator*。

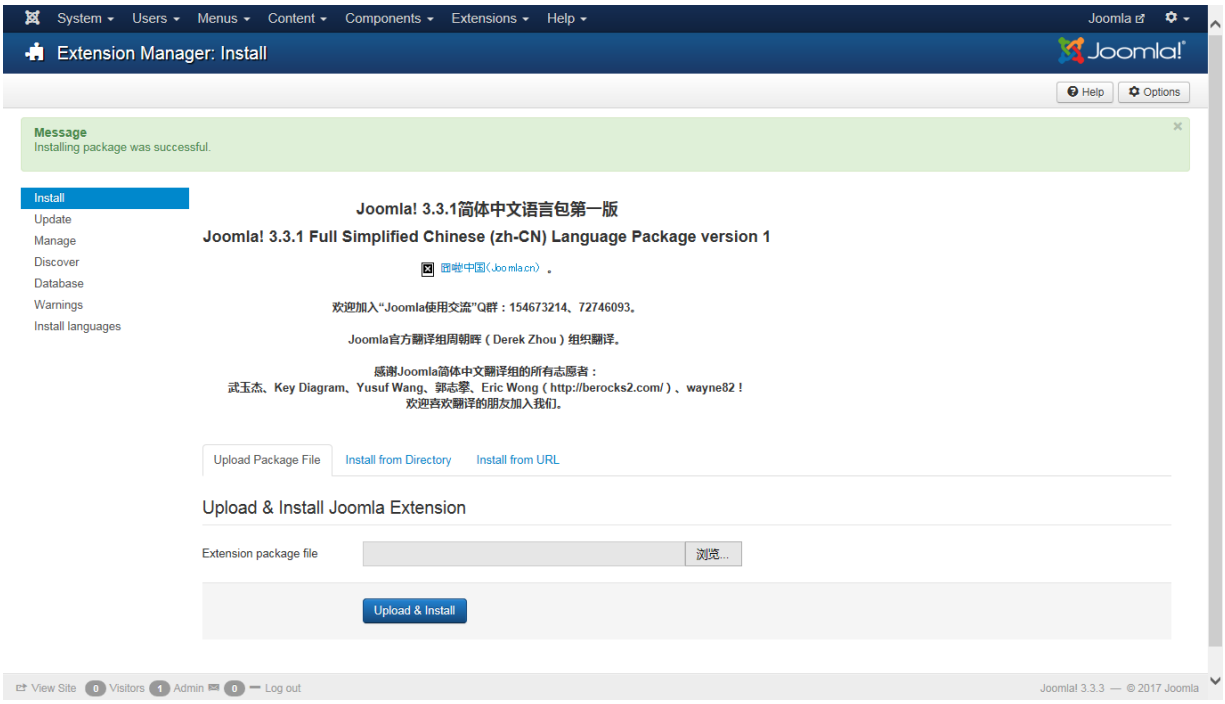
关于 Joomla 支持中文。

Joomla 安装完成之后默认前台后台都是英文界面，中文语言需要手动安装。登陆 Joomla 之后在 Extensions（扩展）> **Extension** Manager（扩展管理）打开扩展配置页面后，上传简体中文包，中文包在服务器的/alidata/res目中，将中文包下载到本地后上传。

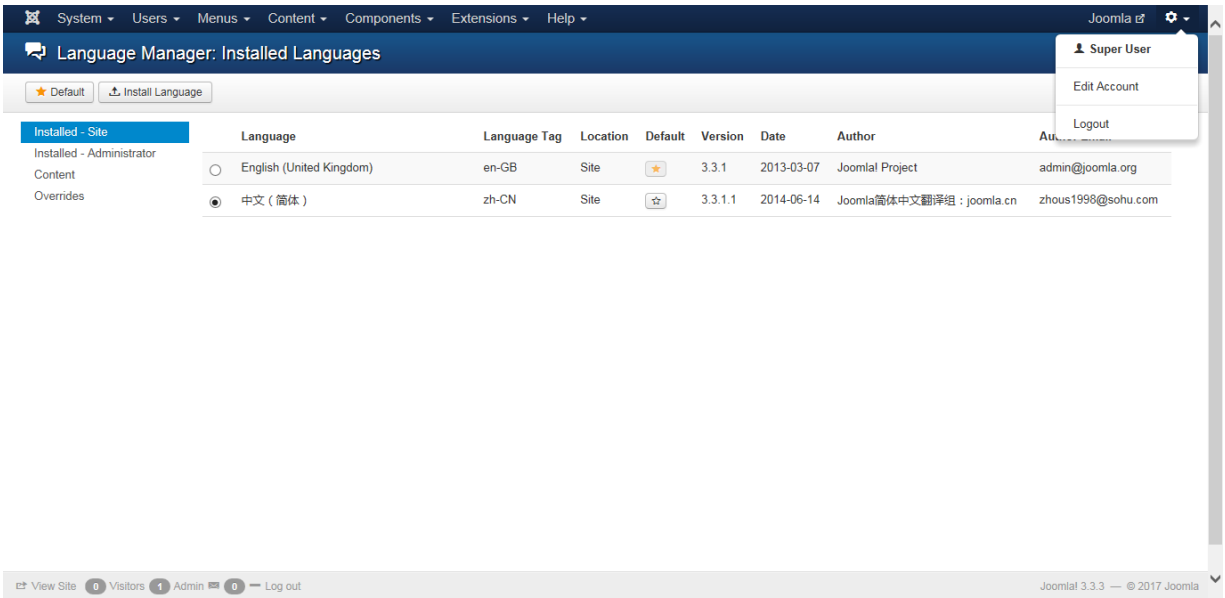




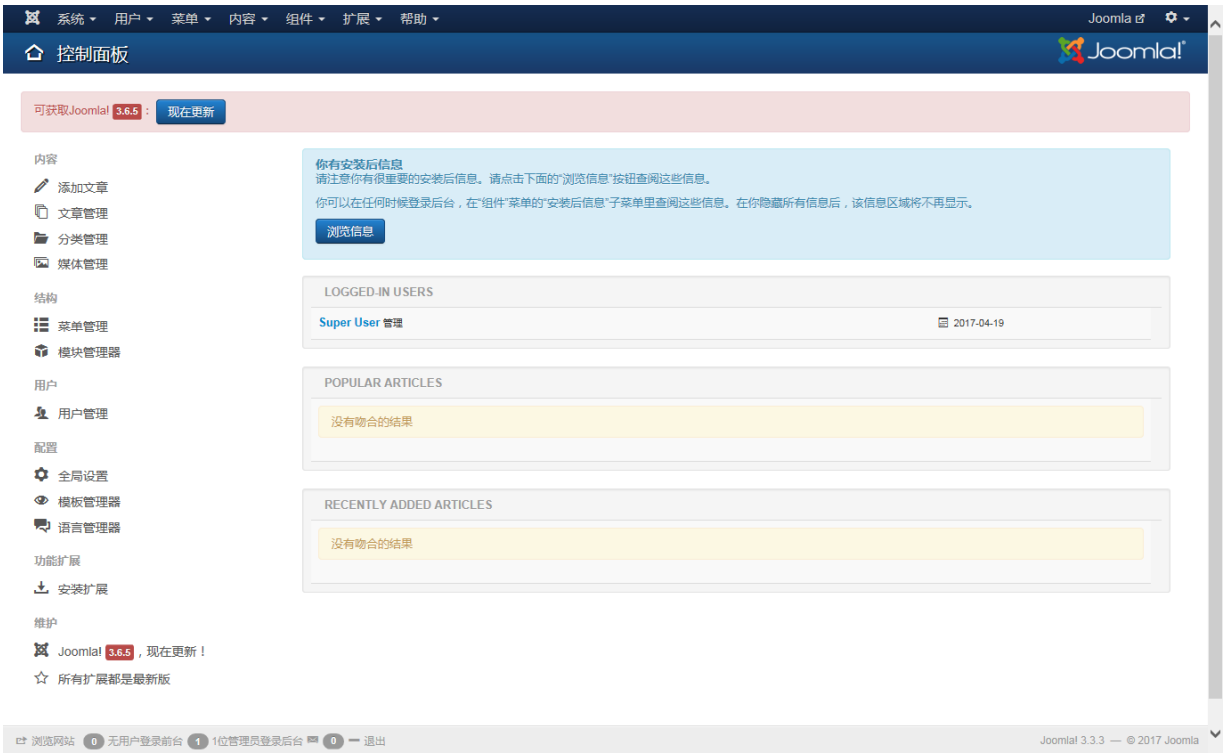
单击 **Update & Install** 上传。



在 Extensions（扩展）> **Language** Manager（语言管理）中，设置前端后台的默认语言，设置完后并单击右上角 **Logout** 重新登陆。



登陆后就能进入中文界面了。


系统
用户
菜单
内容
组件
扩展
帮助
Joomla
控制面板
Joomla!
可获取Joomla! 3.6.5：
现在更新
内容
添加文章
文章管理
分类管理
媒体管理
结构
菜单管理
模块管理器
用户
用户管理
配置
全局设置
模板管理器
语言管理器
功能扩展
安装扩展
维护
Joomla! 3.6.5，现在更新！
所有扩展都是最新版
你有安装后信息
请注意你有很重要的安装后信息。请点击下面的"浏览信息"按钮查阅这些信息。
你可以在任何时候登录后台，在"组件"菜单的"安装后信息"子菜单里查阅这些信息。在你隐藏所有信息后，该信息区域将不再显示。
浏览信息
LOGGED-IN USERS
Super User 管理
2017-04-19
POPULAR ARTICLES
没有吻合的结果
RECENTLY ADDED ARTICLES
没有吻合的结果
浏览网站 0 无用户登录前台 1 1位管理员登录后台 0 退出
Joomla! 3.3.3 — © 2017 Joomla

8 部署Ghost博客（CentOS 7）

Ghost是一个免费的开源博客平台，使用JavaScript编写，基于Node.js，旨在简化个人博客和在线出版物的在线发布过程。

此外，将来随着业务的扩展，您可以利用阿里云强大的产品平台，平滑地横向和纵向扩展服务容量，例如：

- 扩展单个 ECS 实例的 CPU 和内存规格，增强服务器的处理能力。
- 增加多台 ECS 实例，并利用负载均衡，在多个实例中进行负载的均衡分配。
- 利用弹性伸缩（Auto Scaling），根据业务量自动增加或减少 ECS 实例的数量。
- 利用对象存储 OSS（Object Storage Service），存储静态网页和海量图片、视频等。

适用对象

本文档介绍如何使用一台基本配置的云服务器 ECS 实例搭建 Ghost。适用于刚开始使用阿里云进行建站的个人用户。

基本流程

使用云服务器 ECS 搭建 Ghost 网站的操作步骤如下：

1. 购买 ECS 实例。
2. 部署 Web 环境。
3. 安装 Ghost。
4. 购买域名。
5. 备案域名。
6. 解析。

步骤 1：购买 Linux 实例

对于个人使用的小型网站，一台云服务器ECS实例可以满足需求。

这里只介绍新购实例。如果您有镜像，可以 使用自定义镜像创建实例。

操作步骤

1. 登录 云服务器管理控制台。如果尚未注册，单击 **免费注册**。
2. 选择 **云服务器 ECS > 实例**。单击 **创建实例**。

3. 选择付费方式：**包年包月** 或 **按量付费**。关于两种付费方式的区别，请参见 计费模式。

   如果选择 **按量付费**，请确保账户余额至少有100元。如无余额，请进入 **充值页面** 充值后再开通。

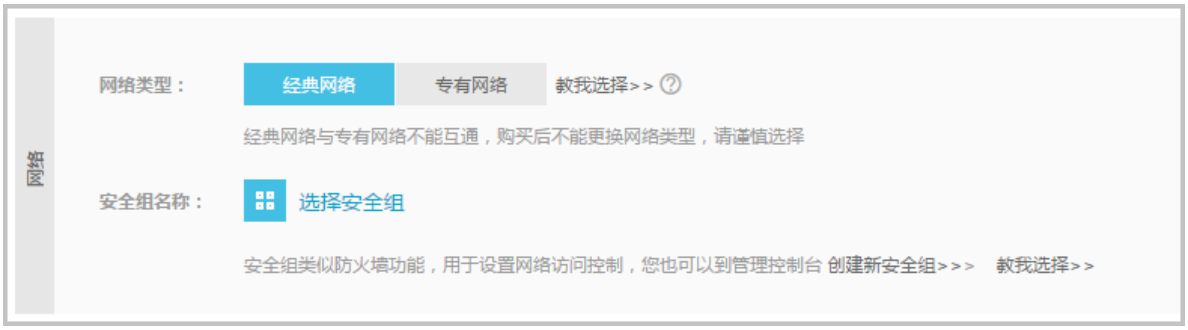
> **说明：**
>
> 对于按量付费的实例，即使停止实例，也会继续收费。如果您不再需要该按量付费的实例，请及时 释放实例。

4. 选择地域。所谓地域，是指实例所在的地理位置。您可以根据您的用户所在的地理位置选择地域。与用户距离越近，延迟相对越少，下载速度相对越快。例如，您的用户都分布在北京地区，则可以选择 **华北2**。

> **说明：**

- 实例创建完成后，不支持更换地域。
- 不同地域提供的可用区数量、实例系列、存储类型、实例价格等也会有所差异。请根据您的业务需求进行选择。

5. 选择网络类型。对于建站的用户，选择 经典网络 即可。然后选择安全组。



6. 选择实例，根据您网站的访问量选择实例规格（CPU、内存）。对于个人网站，1 核 2GB 或 2 核 4GB 一般能够满足需求。关于实例规格的详细介绍，请参考 实例规格族。

   实例系列 II 是实例系列 I 的升级版，提供更高的性能，推荐使用。

7. 选择网络带宽。如果选择 0 MB，则不分配外网 IP，该实例将无法访问公网。如果您选择了 按使用流量，同时选择 0 MB 固定带宽，则同样不分配外网 IP，而且 不支持 0 MB 带宽升级，因此请谨慎选择。

   - 按固定带宽付费。

- 按使用流量付费。

8. 选择镜像。如果用于建站，可以选择公共镜像中的 Linux 操作系统，如 CentOS。

9. 选择 系统盘。您还可以选择 用快照创建磁盘，非常方便地把快照的数据直接复制到磁盘中。

10. 设置实例的登录密码和实例名称。请务必牢记密码。您也可以在创建完成后再设置密码。

11.设置购买的时长和数量。

12.单击页面右侧价格下面的 立即购买。

13.确认订单并付款。

实例创建好之后，您会收到短信和邮件通知，告知您的实例名称、公网 IP 地址、内网 IP 地址等信息。您可以使用这些信息登录和管理实例。

很多重要的信息都是通过绑定手机的短信接收，并且重要的操作（如重启、停止等）都需要手机接收验证码，因此请务必保持绑定手机通信畅通。

步骤 2：部署 Web 环境

本节介绍如何部署 Web 环境，以安装 Nginx为例：

软件包中包含的软件及版本如下：

- nginx：1.10.2

> 说明：
> 这是写文档时参考的软件版本。您下载的版本可能与此不同。

准备工作

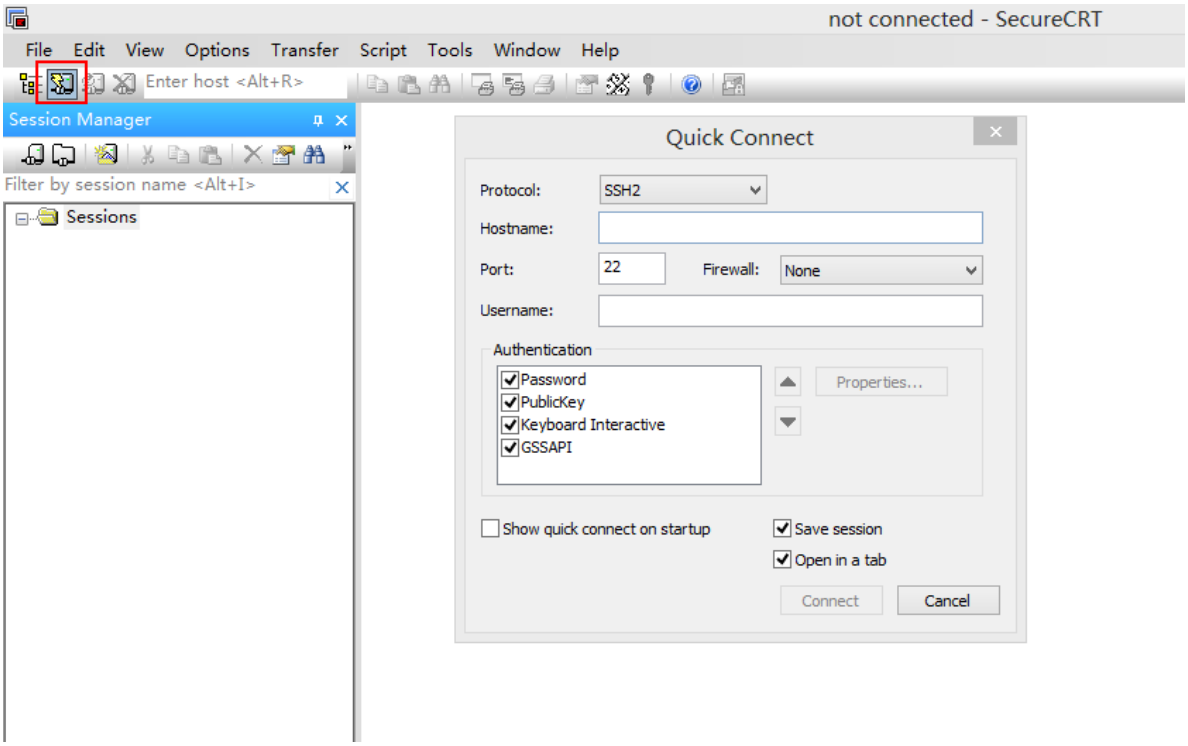
部署之前，请确保：

- 您的实例可以连接公网。
- 已经安装用于连接 Linux 实例的工具，如 SecureCRT。本文将以这个工具为例介绍操作步骤。

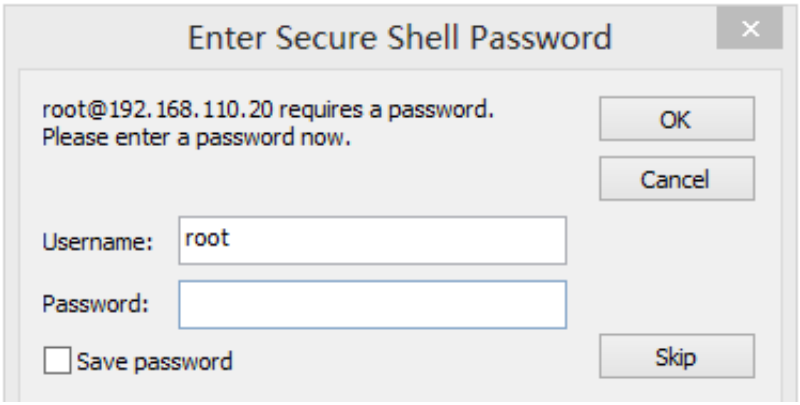
操作步骤

1. 确保您安装了连接 Linux 实例的工具，如 SecureCRT。
2. 打开 SecureCRT ，设置登录实例所需的信息。
3. 设置连接名称。

4. 协议选择 **SSH**。
5. 输入主机 IP 地址和用户名。
6. 单击 确定 保存。



7. 输入用户名 root 和登录密码。



8. 添加Nginx软件库。

```
[root@localhost ~]#rpm -Uvh http://nginx.org/packages/centos/7/
noarch/RPMS/nginx-release-centos-7-0.el7.ngx.noarch.rpm
```

9. 安装Nginx。

```
[root@localhost ~]#yum -y install nginx
```

10. 设置Nginx服务器自动启动。

```
[root@localhost ~]# systemctl enable nginx.service
```

11. 启动Nginx并查看Nginx服务状态。

```
[root@localhost ~]#systemctl start nginx.service
[root@localhost ~]#systemctl status nginx.service
```

12. 在浏览器中输入IP地址，可以看到默认的Nginx的网页。

至此，Nginx搭建完成。

步骤 3：安装 Ghost

请先下载最新版的Ghost，网址： *https://ghost.org/zip/ghost-latest.zip*

操作步骤

1. 更新系统。

   确保你的服务器系统处于最新状态。

```
[root@localhost ~]# yum -y update
```

2. 安装Node.js。

a. 安装EPEL。

```
[root@localhost ~]# yum install epel-release -y
```

b. 安装Node.js 和 npm。

```
[root@localhost ~]# yum install nodejs npm --enablerepo=epel
```

c. 安装进程管理器以便控制Node.js应用程序，这个进程管理器可以保持应用程序一直在运行，运行以下命令进行安装。

```
[root@localhost ~]# npm install pm2 -g
```

d. 安装后可以通过 node -v 和 npm -v 命令来检查 Node.js 的版本。

3. 安装Ghost。

a. 创建Ghost安装目录。

```
[root@localhost ~]# mkdir -p /var/www/ghost
```

b. 进入Ghost安装目录，下载最新的Ghost版本。

```
[root@localhost ~]# cd /var/www/ghost
[root@localhost ghost]# curl -L https://ghost.org/zip/ghost-latest
.zip -o ghost.zip
```

c. 解压Ghost安装包。

```
[root@localhost ghost]# yum install unzip -y
[root@localhost ghost]# unzip ghost.zip
```

d. 使用npm安装Ghost。

```
[root@localhost ghost]# npm install -production
```

e. 安装完成后用 npm start 命令启动ghost，检查有没有安装成功。

f. 从示例配置文件复制并新建 Ghost 配置文件 config.js。

```
[root@localhost ghost]# cp config.example.js config.js
```

g. 配置config.js文件中的URL为自己的域名。

```
[root@localhost ghost]# vim config.js
```

```
var path = require('path'),
    config;

config = {
    // ### Production
    // When running Ghost in the wild, use the production environment.
    // Configure your URL and mail settings here
    production: {
        url: 'http://myghostblog.com',
        mail: {},
        database: {
            client: 'sqlite3',
            connection: {
                filename: path.join(__dirname, '/content/data/ghost.db')
            },
            debug: false
        },

        server: {
            host: '127.0.0.1',
            port: '2368'
        }
    },
```

h. 使用进程管理器来配置Ghost永久运行。

```
[root@localhost ghost]# NODE_ENV=production pm2 start index.js --
name "ghost"
```

i. 开启/停止/重启ghost。

```
[root@localhost ghost]# pm2 start ghost
[root@localhost ghost]# pm2 stop ghost
[root@localhost ghost]# pm2 restart ghost
```

4. 安装Nginx。

a. 添加Nginx软件库。

```
[root@localhost ~]# rpm -Uvh http://nginx.org/packages/centos/7/
noarch/RPMS/nginx-release-centos-7-0.el7.ngx.noarch.rpm
```

b. 安装Nginx。

```
[root@localhost ~]# yum -y install nginx
```

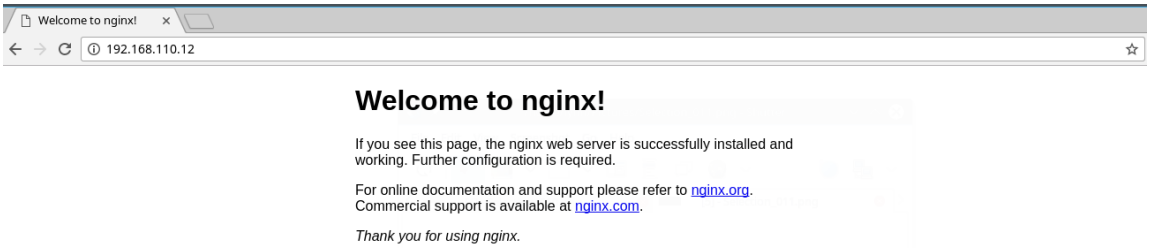
c. 设置Nginx服务器自动启动。

```
[root@localhost ~]# systemctl enable nginx.service
```

d. 启动Nginx并查看Nginx服务状态。

```
[root@localhost ~]#systemctl start nginx.service
[root@localhost ~]#systemctl status nginx.service
```

e. 在浏览器中输入IP地址，可以看到默认的Nginx的网页。



5. 配置Nginx作为Ghost的反向代理。

a. 进入Nginx配置目录，新建Ghost博客的Nginx配置文件。

```
[root@localhost ~]#vim /etc/nginx/conf.d/ghost.conf
```

b. 将以下内容输入到ghost.conf中，把 **server_name** 改成实际的域名。

```
upstream ghost {
    server 127.0.0.1:2368;
}

server {
    listen      80;
    server_name myghostblog.com;

    access_log  /var/log/nginx/ghost.access.log;
    error_log   /var/log/nginx/ghost.error.log;

    proxy_buffers 16 64k;
    proxy_buffer_size 128k;

location / {
        proxy_pass  http://ghost;
        proxy_next_upstream error timeout invalid_header http_500 http_502 http_503 http_504;
        proxy_redirect off;

        proxy_set_header    Host            $host;
        proxy_set_header    X-Real-IP       $remote_addr;
        proxy_set_header    X-Forwarded-For $proxy_add_x_forwarded_for;
        proxy_set_header    X-Forwarded-Proto https;
    }

}
```

c. 修改默认的配置文件default.conf为default.conf.bak，使Nginx只应用ghost.conf。

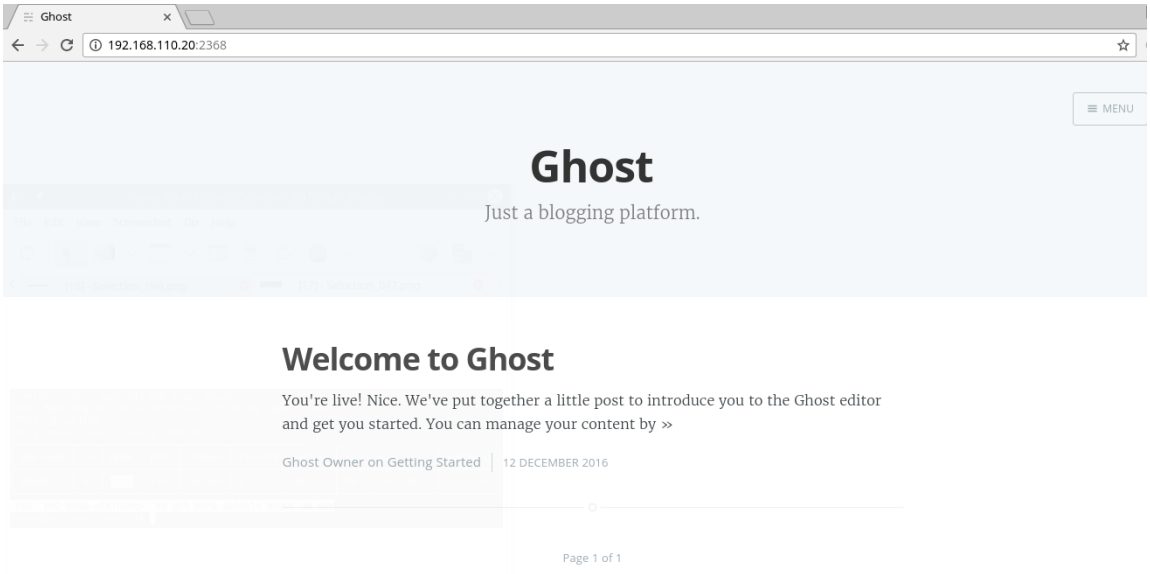
```
[root@localhost ~]#mv default.conf default.conf.bak
```

d. 重启Nginx服务。

```
[root@localhost conf.d]# systemctl restart nginx.service
```

6. 访问Ghost博客。

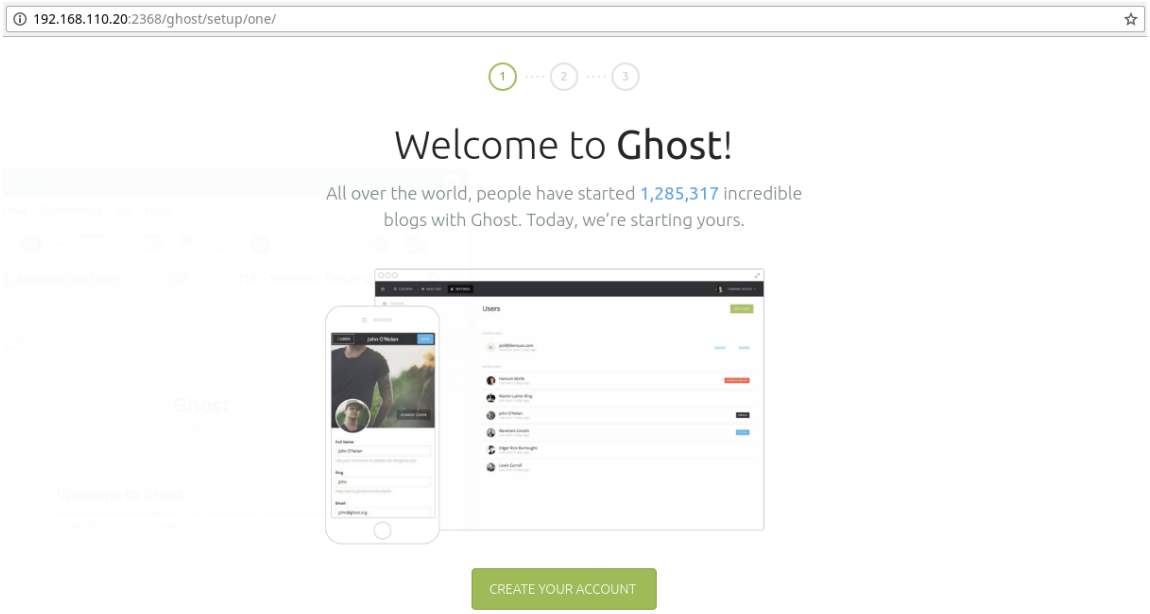
a. 在浏览器输入 *http://IP* 或 *http://域名* 即可访问Ghost。



说明：

如果访问出现502，请检查是否由于防火墙的问题引起，可以关闭防火墙。

b. 需要对博客进行编辑修改，在浏览器输入 *http://IP/ghost* 即可。



步骤 4：购买域名

您可以给自己的网站设定一个单独的域名。您的用户可以使用易记的域名访问您的网站，而不需要使用复杂的 IP 地址。

建议通过 阿里云购买域名。

操作步骤

1. 在 购买域名 页面，搜索想用的域名，如尚未被注册，则可以购买。选择要购买的域名及期限，然后结算。

2. 在确认订单的时候，需要选择域名的所有者是个人还是企业。为方便操作，建议暂时先选择个人，以后可以在会员中心进行修改。本文档将以个人用户为例。

3. 如果这是您首次购买域名，需要 创建消息模板。

4. 比较便捷的方式是选择用会员信息自动填写。请务必填写真实信息。

5. 完成后需要进行实名认证。上传个人身份证正面扫描件。审核一般需要 3 ~ 5 个工作日。

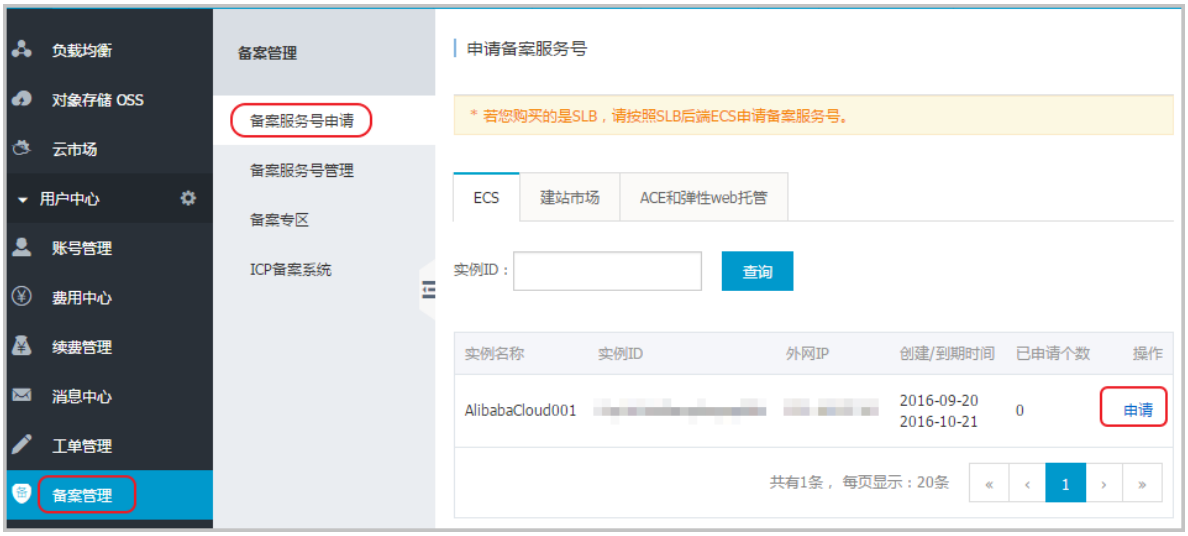
步骤 **5**：备案

对于域名指向中国境内服务器的网站，必须进行网站备案。在域名获得备案号之前，网站是无法开通使用的。

阿里云有代备案系统，方便您进行备案。备案免费，一般审核时间为20天左右。请您耐心等待。

操作步骤

1. 首先给购买的ECS实例 申请备案服务号。这个服务号在备案时会用到。选择 备案管理 > 备案服务号申请，然后单击 申请。



2. 在弹出的提示信息对话框中，单击 确定。

3. 申请成功后，页面自动跳转到备案服务号管理页面，显示与 ECS 实例绑定的备案号。然后单击 备案专区，了解备案相关信息。

4. 首次备案的用户，需要在 ICP代备案管理系统 注册一个备案账号。注意，该账号不是阿里云账号，而是申请备案专用的账号。

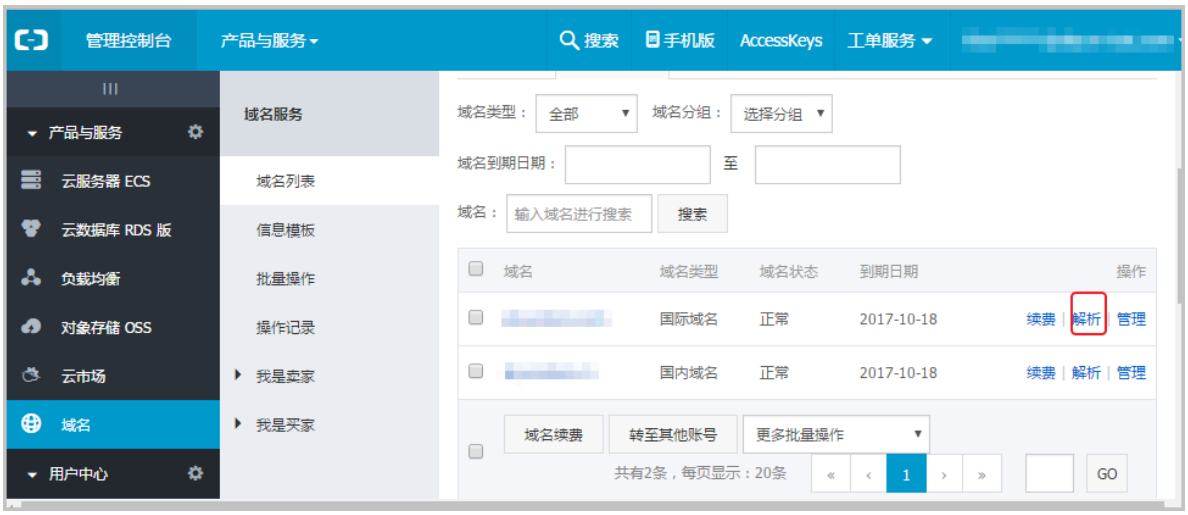
   关于首次备案的详细步骤，请参考 首次备案图文引导。

步骤 6：配置域名解析

您需要在阿里云万网上配置域名解析之后，用户才能通过域名访问您的网站。

操作步骤

1. 登录 域名管理控制台。
2. 在域名列表中找到要解析的域名，然后单击 解析。



3. 单击 新手引导设置。

4. 输入您的 Linux 实例的公网 IP 地址。然后单击 提交。

5. 设置成功，会出现如下信息。

恭喜您！您可以使用域名访问自己的网站了！

9 搭建FTP站点

9.1 Linux实例搭建FTP站点

vsftpd 是 Linux 下的一款小巧轻快、安全易用的 FTP 服务器软件，是一款在各个 Linux 发行版中最受推崇的 FTP 服务器软件。本文以 CentOS 7.2 64位操作系统为例，说明如何在 Linux 实例上安装 vsftpd。

Linux 实例搭建 FTP 站点具体操作步骤如下：

- 步骤一： 安装 vsftpd
- 步骤二： 配置 vsftpd
- 步骤三： 设置安全组
- 步骤四： 客户端测试

步骤一： 安装 **vsftpd**

1. 远程连接 并登录到 Linux 实例。
2. 运行以下命令安装 vsftpd。

```
yum install -y vsftpd
```

```
[root@iZbp1g1kolvxh5k8l00z6cZ ~]# yum install -y vsftpd
```

出现下图表示安装成功。

```
Total download size: 169 k
Installed size: 348 k
Downloading packages:
vsftpd-3.0.2-21.el7.x86_64.rpm                              | 169 kB  00:00:00
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
  Installing : vsftpd-3.0.2-21.el7.x86_64                                   1/1
  Verifying  : vsftpd-3.0.2-21.el7.x86_64                                   1/1

Installed:
  vsftpd.x86_64 0:3.0.2-21.el7

Complete!
[root@iZbp1g1kolvxh5k8l00z6cZ ~]#
```

3. 运行以下命令打开及查看 `etc/vsftpd`。

```
cd /etc/vsftpd
```

```
ls
```

```
[root@iZbp1g1kolvxh5k8l00z6cZ ~]# cd /etc/vsftpd/
[root@iZbp1g1kolvxh5k8l00z6cZ vsftpd]# ls
ftpusers  user_list  vsftpd.conf  vsftpd_conf_migrate.sh
[root@iZbp1g1kolvxh5k8l00z6cZ vsftpd]#
```

说明：

- `/etc/vsftpd/vsftpd.conf` 是核心配置文件。
- `/etc/vsftpd/ftpusers` 是黑名单文件，此文件里的用户不允许访问 FTP 服务器。
- `/etc/vsftpd/user_list` 是白名单文件，是允许访问 FTP 服务器的用户列表。

4. 运行以下命令设置开机自启动。

```
systemctl enable vsftpd.service
```

5. 运行以下命令启动 FTP 服务。

```
systemctl start vsftpd.service
```

6. 运行以下命令查看 FTP 服务端口。

```
netstat -antup | grep ftp
```

```
[root@iZbp1g1kolvxh5k8l00z6cZ vsftpd]# systemctl enable vsftpd.service
[root@iZbp1g1kolvxh5k8l00z6cZ vsftpd]# systemctl start vsftpd.service
[root@iZbp1g1kolvxh5k8l00z6cZ vsftpd]# netstat -antup | grep ftp
tcp6       0      0 :::21                   :::*                    LISTEN      9379/vsftpd
```

步骤二： 配置 vsftpd

vsftpd 安装后默认开启了匿名 FTP 的功能，使用匿名 FTP，用户无需输入用户名密码即可登录 FTP 服务器，但没有权限修改或上传文件。

文本介绍了以下几个配置 vsftpd 的方法以及相关的参数说明，您可以根据具体需要进行参考。

- 配置匿名用户上传文件权限
- 配置本地用户登录
- vsftpd.conf 的配置文件参数说明

配置匿名用户上传文件权限

修改 `vsftpd.conf` 的配置文件的选项，可以赋予匿名 FTP 更多的权限。

1. 修改 `/etc/vsftpd/vsftpd.conf`：

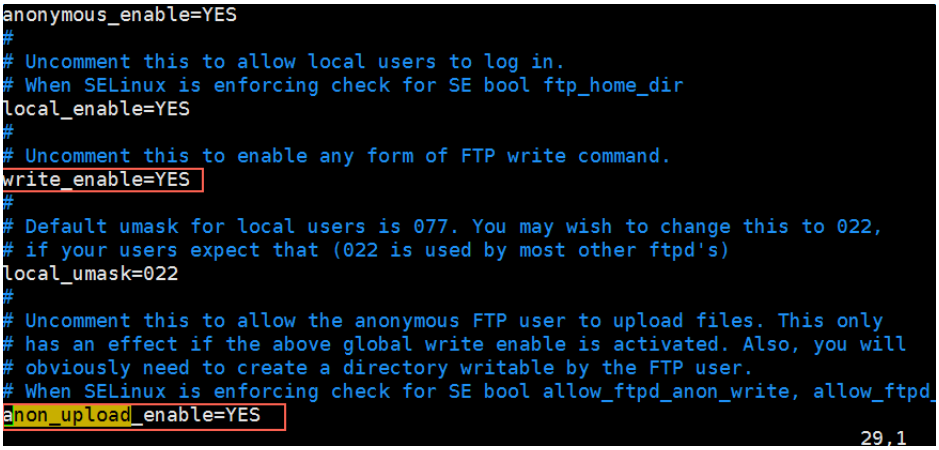
   a. 运行 `vim /etc/vsftpd/vsftpd.conf`。

b. 按 **i** 键进入编辑模式。

c. 将写权限修改为 `write_enable=YES`。

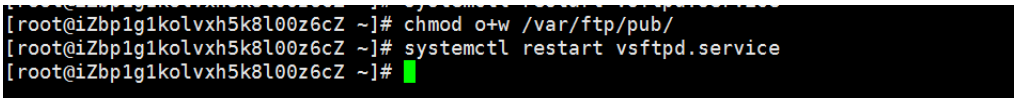
d. 将匿名上传权限修改为 `anon_upload_enable=YES`。

e. 按 **Esc** 键退出编辑模式，然后输入 `:wq` 保存并退出文件。



2. 运行以下命令更改 `/var/ftp/pub` 目录的权限，为 FTP 用户添加写权限，并重新加载配置文件。

```
chmod o+w /var/ftp/pub/
systemctl restart vsftpd.service
```



配置本地用户登录

本地用户登录就是指用户使用 Linux 操作系统中的用户账号和密码登录 FTP 服务器。

vsftpd 安装后默只支持匿名 FTP 登录，用户如果试图使用 Linux 操作系统中的账号登录服务器，将会被 vsftpd 拒绝，但可以在 vsftpd 里配置用户账号和密码登录。具体步骤如下：

1. 运行以下命令创建 ftptest 用户。

```
useradd ftptest
```

2. 运行以下命令修改 ftptest 用户密码。

```
passwd ftptest
```

```
[root@iZbp1g1kolvxh5k8l00z6cZ ~]# useradd ftptest
[root@iZbp1g1kolvxh5k8l00z6cZ ~]# passwd ftptest
Changing password for user ftptest.
New password:
BAD PASSWORD: The password fails the dictionary check - it is too simplistic/systematic
Retype new password:
passwd: all authentication tokens updated successfully.
[root@iZbp1g1kolvxh5k8l00z6cZ ~]#
```

3. 修改 `/etc/vsftpd/vsftpd.conf`：

   a. 运行 `vim /etc/vsftpd/vsftpd.conf`。

   b. 按键 **i** 进入编辑模式。

   c. 将是否允许匿名登录 FTP 的参数修改为 `anonymous_enable=NO`。

   d. 将是否允许本地用户登录 FTP 的参数修改为 `local_enable=YES`。

   e. 按键 **Esc** 退出编辑模式，然后按键 `:wq` 保存并退出文件。

```
# Allow anonymous FTP? (Beware - allowed by default if you comment this out).
anonymous_enable=NO
#
# Uncomment this to allow local users to log in.
# When SELinux is enforcing check for SE bool ftp_home_dir
local_enable=YES
#
# Uncomment this to enable any form of FTP write command.
write_enable=YES
#
# Default umask for local users is 077. You may wish to change this to 022,
# if your users expect that (022 is used by most other ftpd's)
local_umask=022
#
```

4. 运行以下命令重新加载配置文件。

```
systemctl restart vsftpd.service
```

vsftpd.conf 的配置文件参数说明

运行命令 `cat /etc/vsftpd/vsftpd.conf` 查看配置文件内容。

用户登录控制：

参数	说明
anonymous_enable=YES	接受匿名用户
no_anon_password=YES	匿名用户login时不询问口令

参数	说明
anon_root=(none)	匿名用户主目录
local_enable=YES	接受本地用户
local_root=(none)	本地用户主目录

用户权限控制：

参数	说明
write_enable=YES	可以上传(全局控制)
local_umask=022	本地用户上传文件的umask
file_open_mode=0666	上传文件的权限配合umask使用
anon_upload_enable=NO	匿名用户可以上传
anon_mkdir_write_enable=NO	匿名用户可以建目录
anon_other_write_enable=NO	匿名用户修改删除
chown_username=lightwiter	匿名上传文件所属用户名

步骤三： 设置安全组

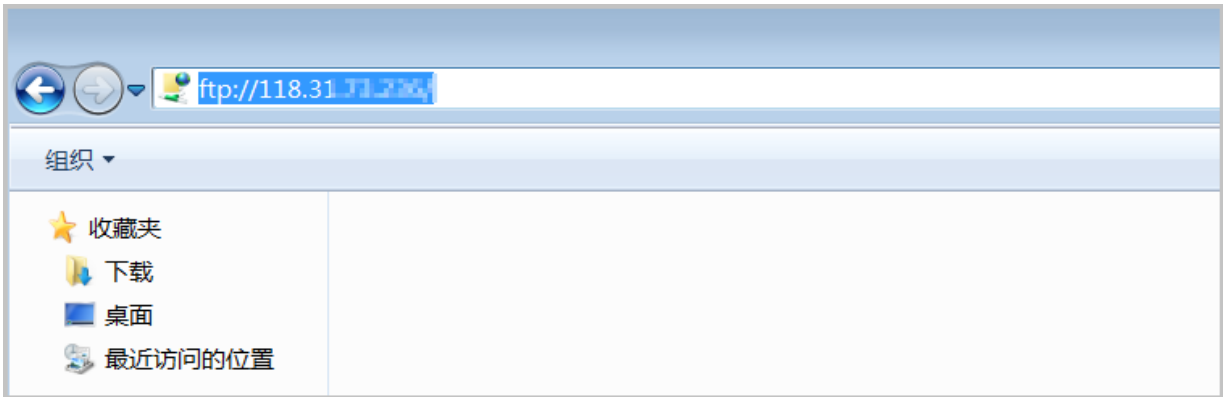
搭建好 FTP 站点后，您需要在实例的安全组的入方向添加一条放行 FTP 端口的规则，具体步骤参见 添加安全组规则。

步骤四： 客户端测试

打开客户端的 计算机，在路径栏输入 `ftp://服务器 IP 地址:FTP 端口`（如果不填端口则默认访问21端口），例如：`ftp://0.0.0.0:20`。弹出输入用户名和密码的对话框表示配置成功，正确的输入用户名和密码后，即可对 FTP 文件进行相应权限的操作。

说明：

客户端使用此方法访问 FTP 站点时，需要对 IE 浏览器进行设置，才能打开 FTP 的文件夹。 打开 IE 浏览器，选择 设置 > **Internet** 选项 > 高级。勾选 启用 **FTP** 文件夹视图，取消勾选 使用被动 **FTP**。



后续操作

您可以参考 安全加固方案 https://help.aliyun.com/knowledge_detail/37452.html对 FTP 服务进行安全加固。

9.2 Windows实例搭建FTP站点

本文介绍了如何使用 Windows 实例搭建 FTP 站点。此方法适用于 Windows Server 2008 及以上系统，本文以 Windows Server 2008 R2 为例。

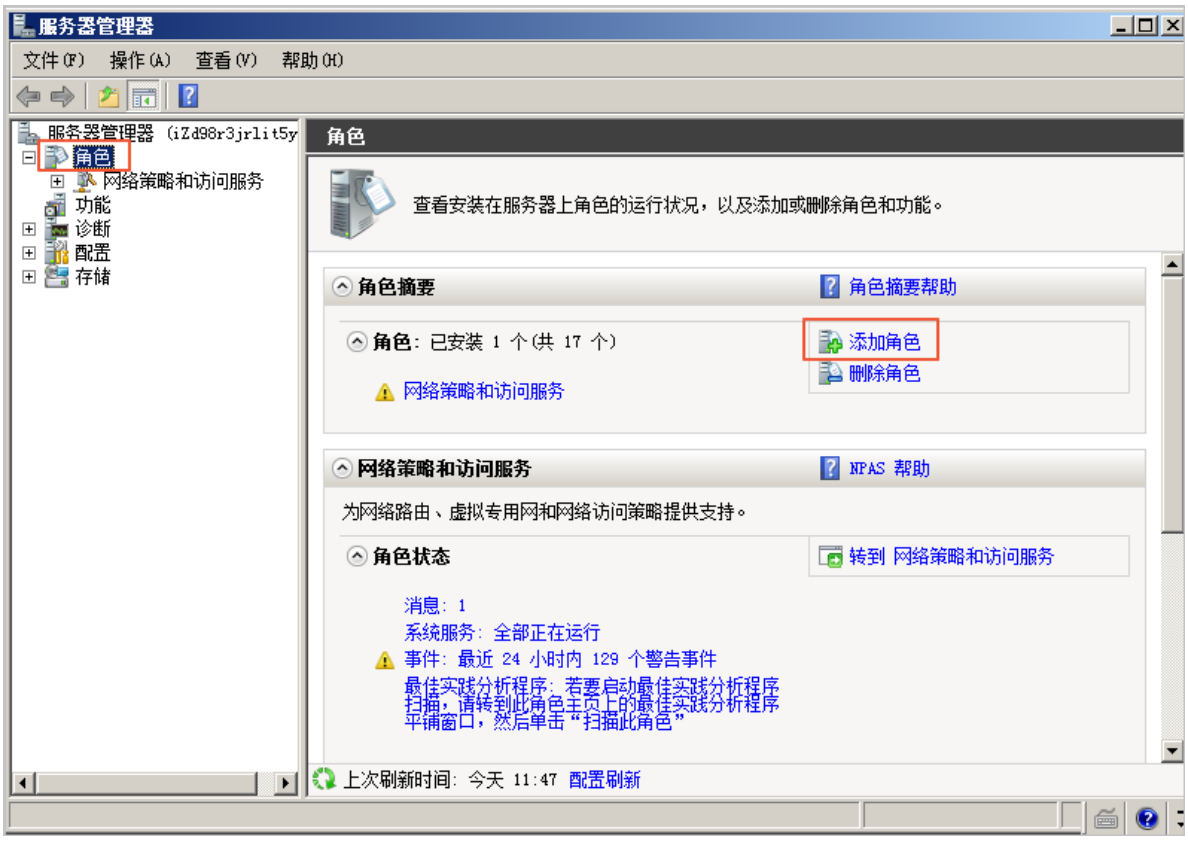
Windows 实例搭建 FTP 站点具体操作步骤如下：

- 步骤一# 添加 *IIS* 以及 *FTP* 服务角色
- 步骤二# 创建 *FTP* 用户名及密码
- 步骤三# 设置共享文件的权限
- 步骤四# 添加及设置 *FTP* 站点
- 步骤五# 设置安全组及防火墙
- 步骤六# 客户端测试

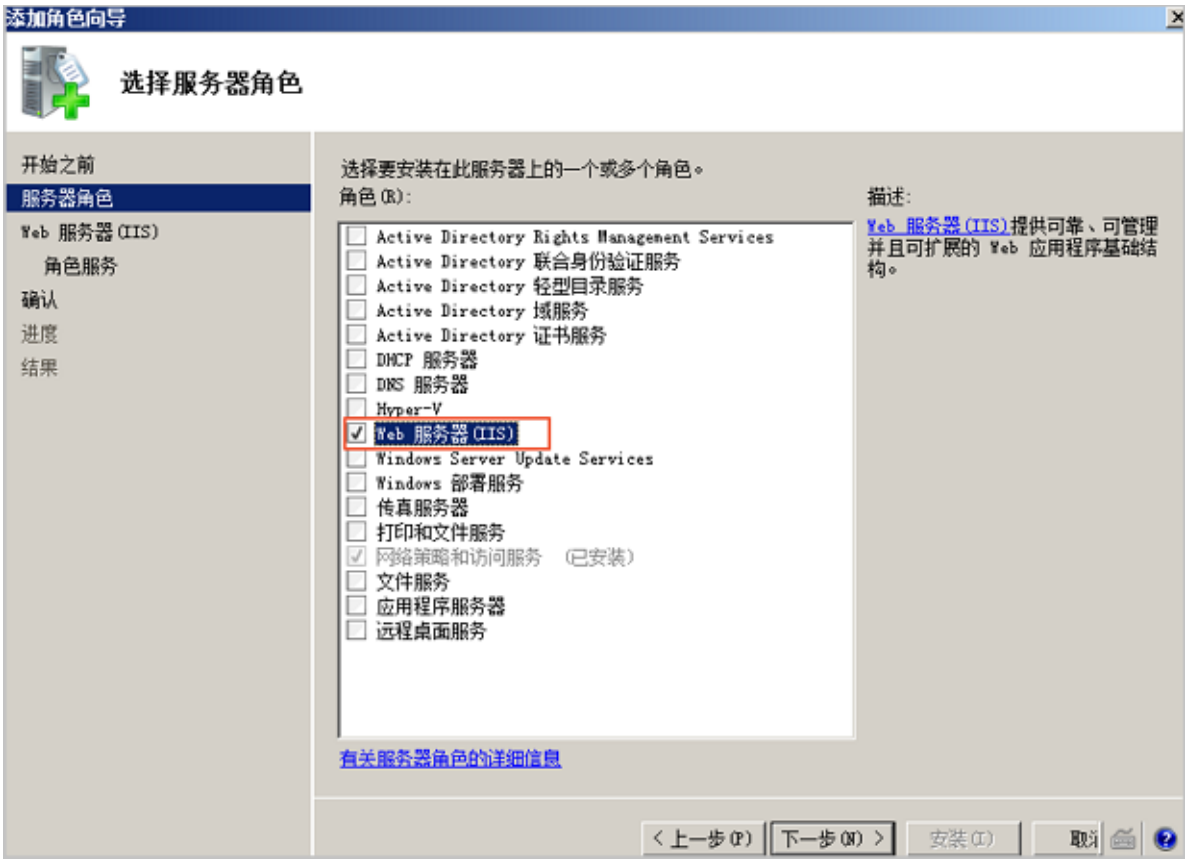
步骤一： 添加 **IIS** 以及 **FTP** 服务角色

在创建 FTP 站点前，首先需要安装 IIS 及 FTP 服务。

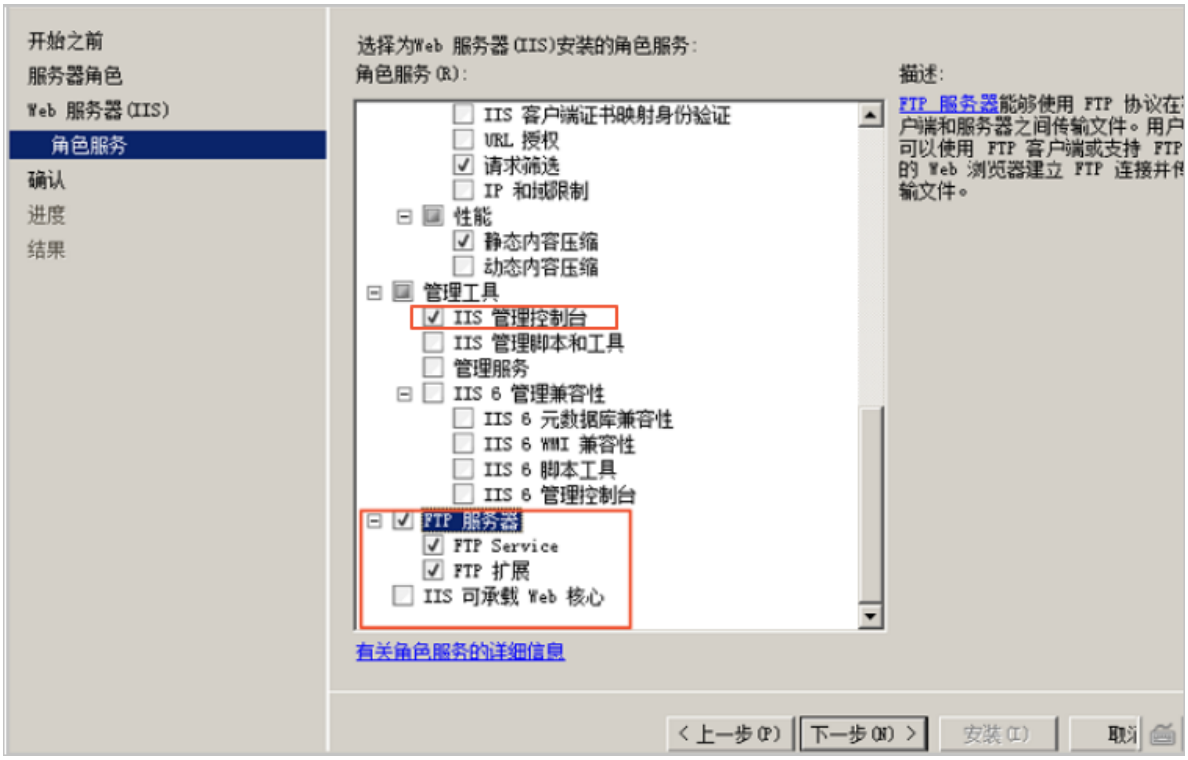
1. 远程连接 并登录到 Windows 实例。
2. 选择 开始 > 所有程序 > 管理工具 > 服务管理器。
3. 单击 角色，然后单击 添加角色。



4. 在弹出的对话框中，选择 下一步。
5. 选择 **Web** 服务器（IIS），然后单击 下一步。



6. 选择 **IIS管理控制台** 以及 **FTP** 服务器，选择 下一步，单击 安装。

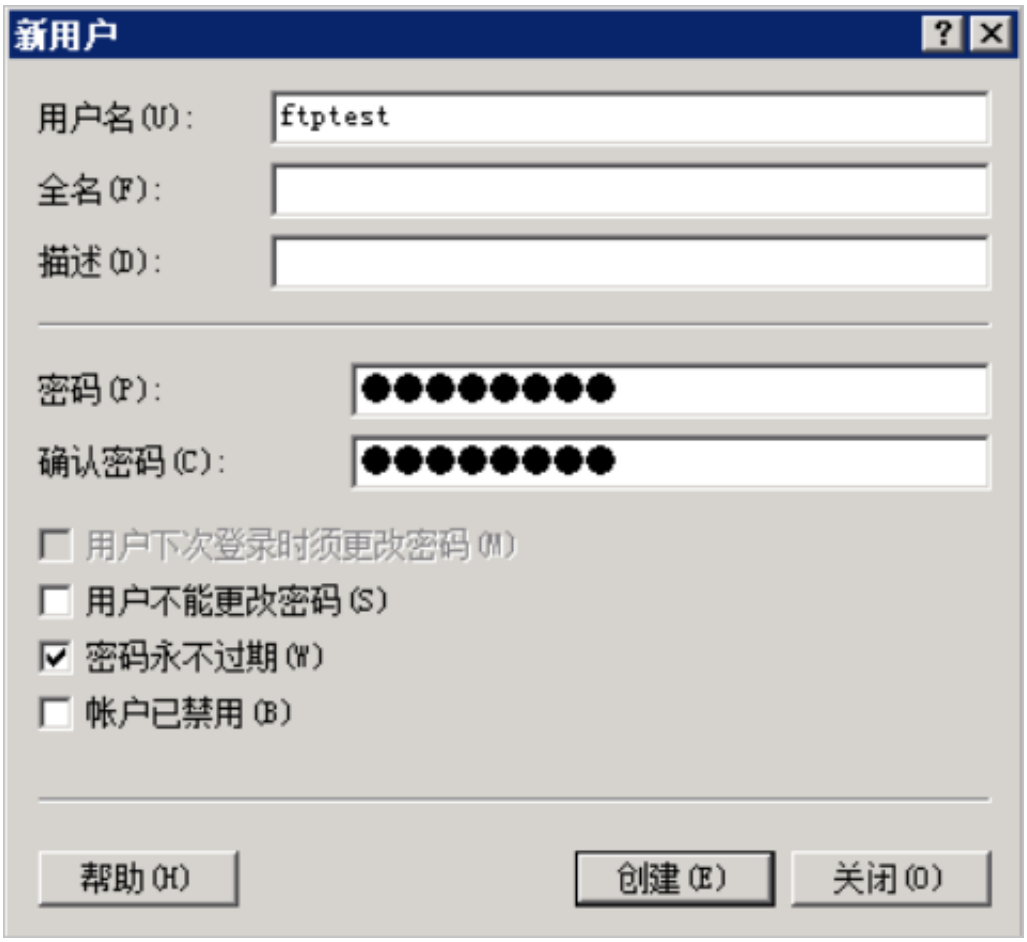


步骤二： 创建 **FTP** 用户名及密码

创建 Windows 用户名和密码，用于 FTP 使用。如果您希望匿名用户可以访问，此步可省略。

1. 选择 开始 > 管理工具 > 服务器管理器。
2. 单击 配置 > 本地用户和组 > 用户，并在右侧空白处单击右键，再选择 添加用户，本文例子中 用户名 使用 ftptest。

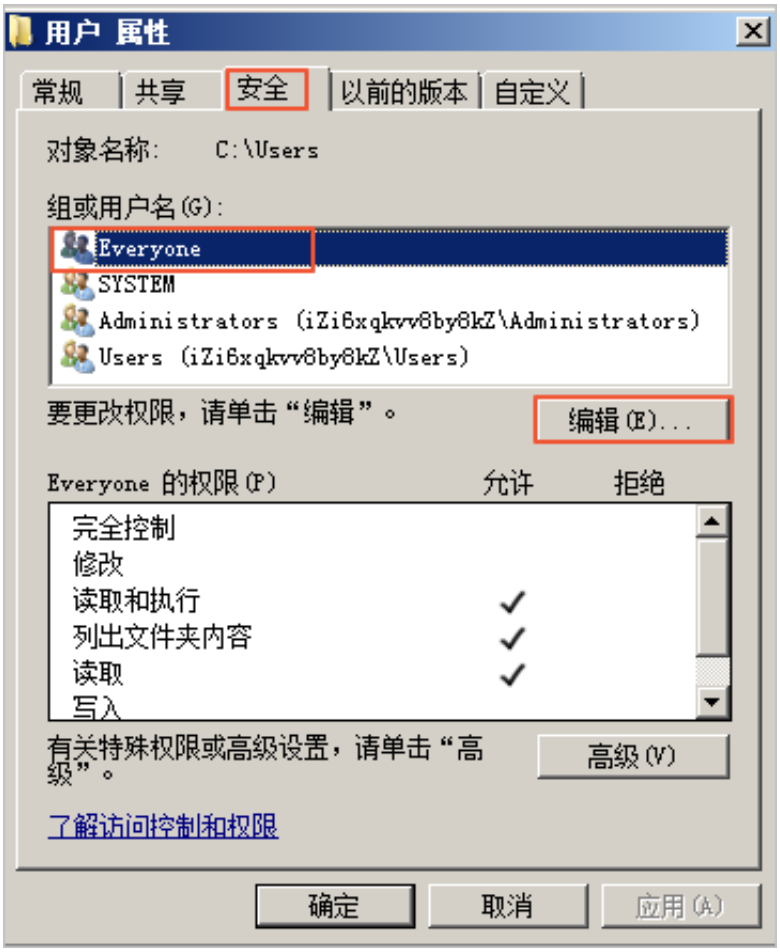
> 说明：
> 密码必须包括大写字母、小写字母和数字。否则会显示无法通过密码策略。



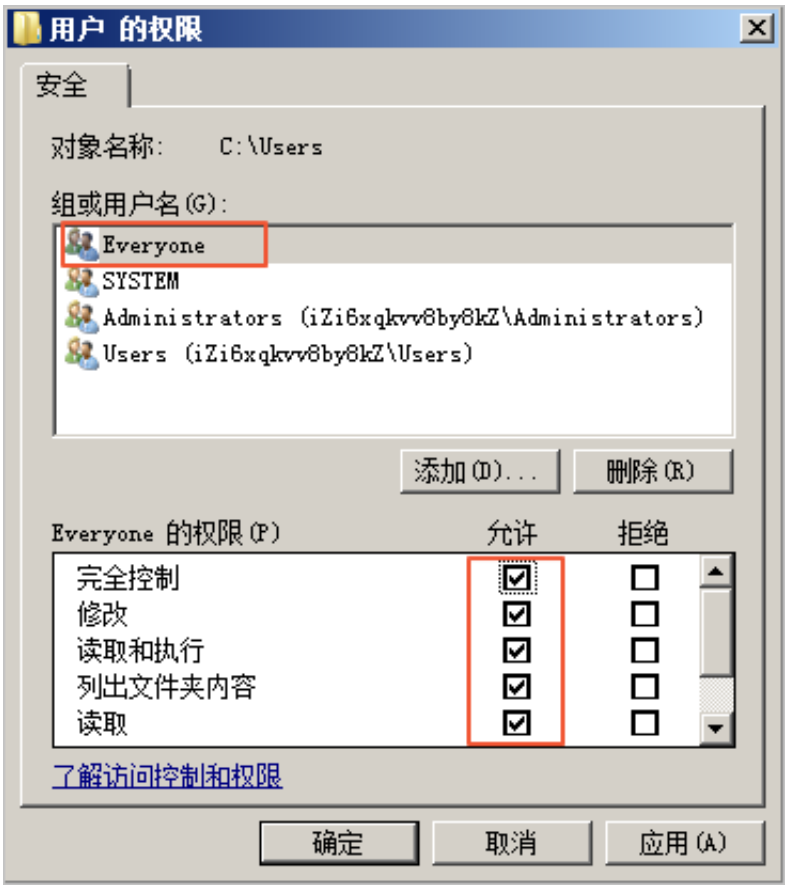
步骤三： 设置共享文件的权限

您需要为在 FTP 站点共享给用户的文件夹设置访问以及修改等权限。

1. 在服务器磁盘上创建一个供 FTP 使用的文件夹，右键单击文件夹，选择 属性。
2. 单击 安全，选择 **Everyone**，然后选择 编辑。



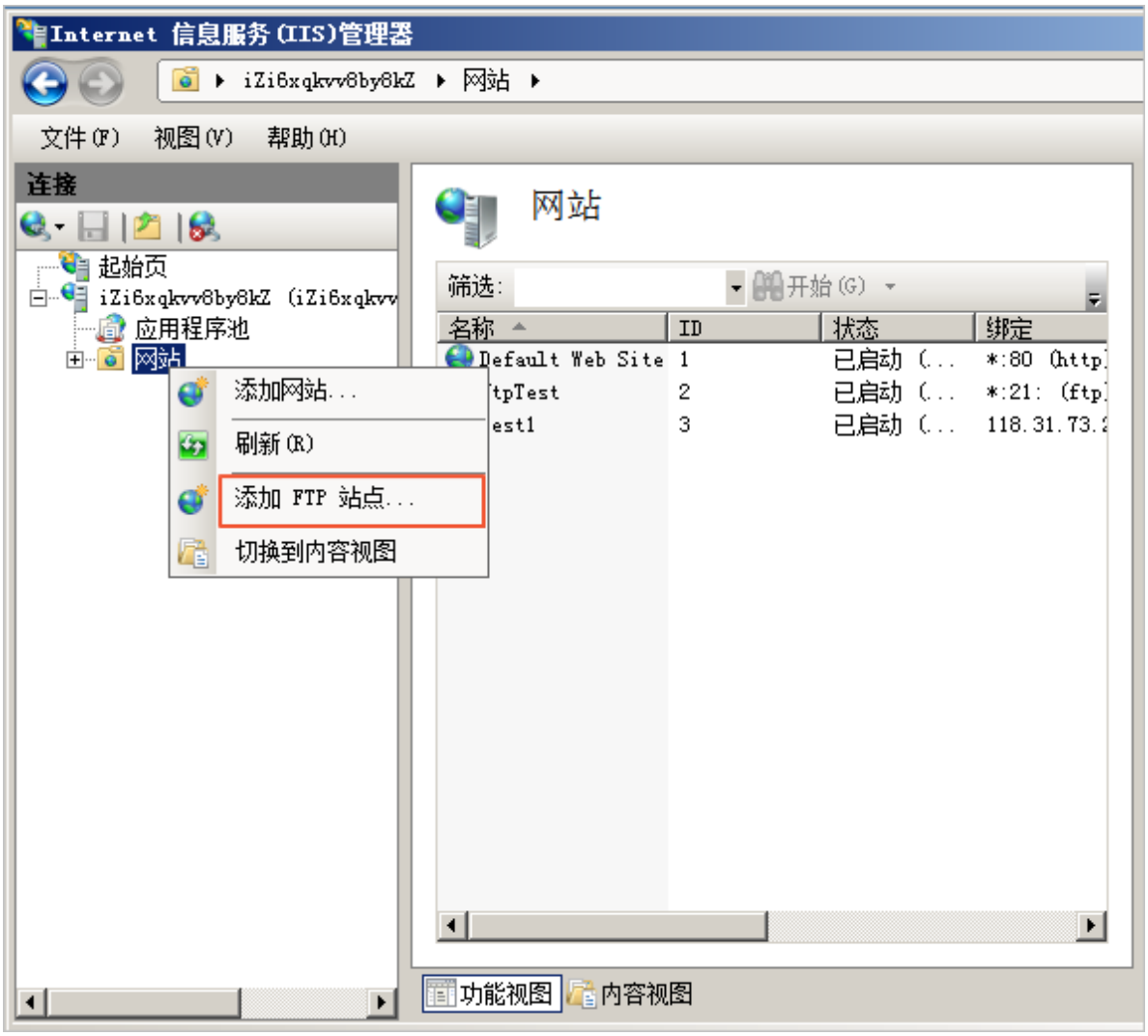
3. 选择 **Everyone**，然后根据需要，选择 **Everyone** 的权限，本文例子中允许所有权限。



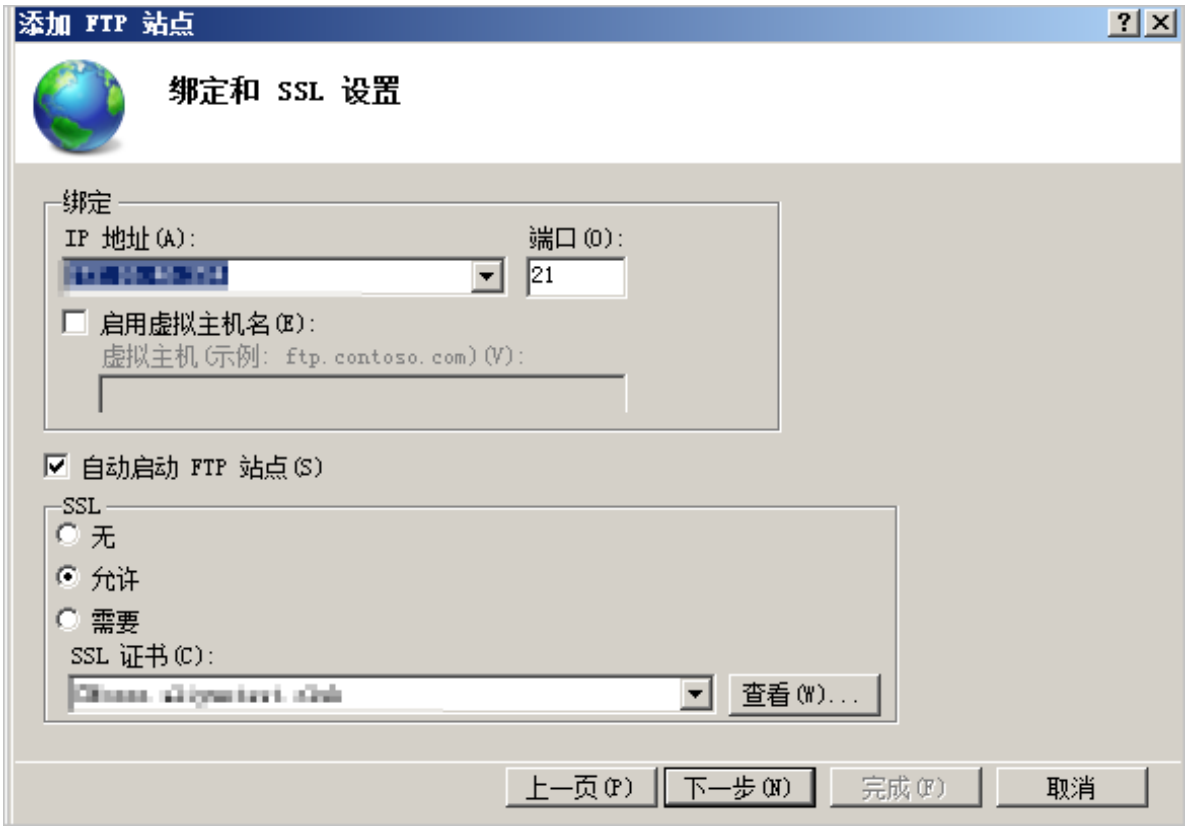
步骤四： 添加及设置 **FTP** 站点

安装 FTP，设置好共享文件夹权限后，您需要创建 FTP 站点。

1. 选择 开始 > 所有程序 > 管理工具 > **Internet** 信息服务（IIS）管理器。
2. 右键单击 网站，选择 添加 **FTP** 站点。



3. 在弹出的窗口，填写 FTP 站点名称与共享文件夹的物理路径，然后单击 下一步。
4. IP 地址默认选择 全部未分配。端口号可自行设置，FTP 默认端口号为 21。
5. 选择 SSL 设置。
   - 允许：允许 FTP 服务器支持与客户端的非 SSL 和 SSL 连接。
   - 需要：需要对 FTP 服务器和客户端之间的通信进行 SSL 加密。
   - 无： 不需要 SSL 加密选择 无。



6. 选择要使用的一种或多种身份验证方法。

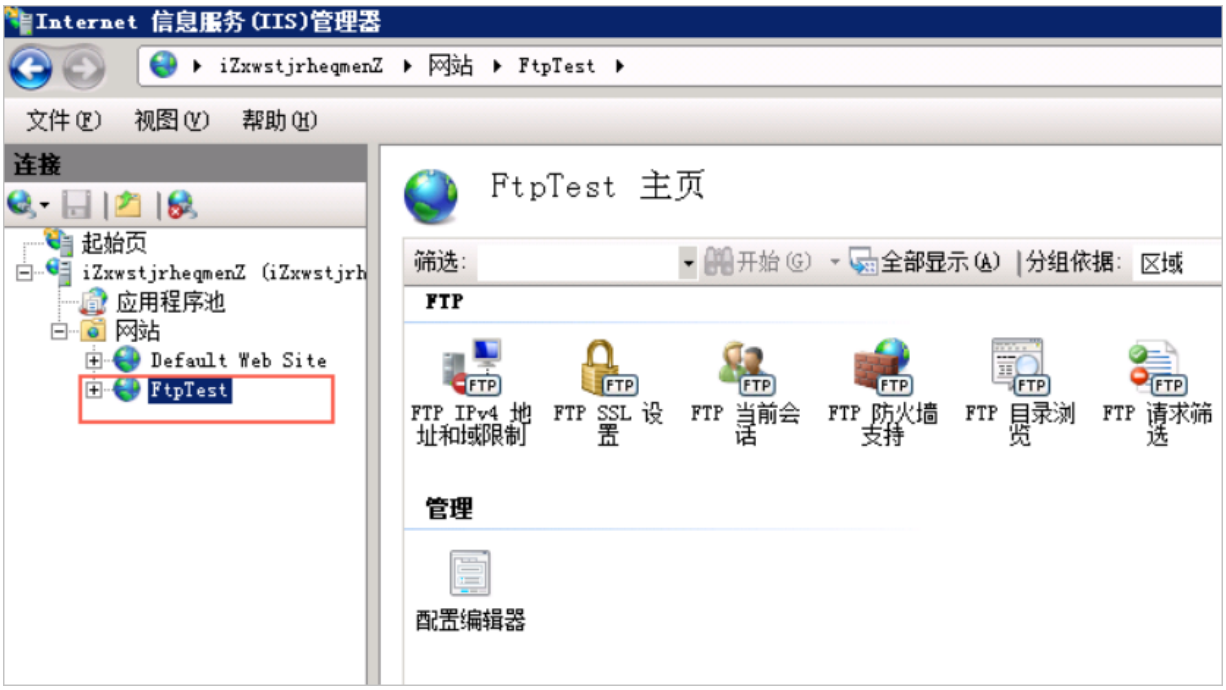
   - 匿名：允许任何仅提供用户名 **anonymous** 或 **ftp** 的用户访问内容。
   - 基本：需要用户提供有效用户名和密码才能访问内容。由于基本身份验证通过网络传输未加密的密码，因此请仅在清楚客户端和 FTP 服务器之间的连接是安全的情况下（例如，使用安全套接字层 (SSL) 时）使用此身份验证方法。

7. 从 **允许访问** 列表中，选择以下选项之一：

   - **所有用户**：所有用户（不论是匿名用户还是已标识的用户）均可访问相应内容。
   - **匿名用户**：匿名用户可访问相应内容。
   - **指定角色或用户组**：仅特定角色或用户组的成员才能访问相应内容。请在对应的框中键入角色或用户组。
   - **指定用户**：仅指定用户才能访问相应内容。请在对应的框中键入用户名。

8. 选择经过授权的用户的 **读取** 和 **写入** 权限。然后单击 **完成**。

完成后可以看到搭建的 FTP 站点。



步骤五： 设置安全组及防火墙

搭建好 FTP 站点后，您需要在实例安全组的入方向添加一条放行 FTP 端口的规则，具体步骤参见添加安全组规则，具体配置可以参见 安全组规则的典型应用。

服务器防火墙默认放行 TCP 21 端口用于 FTP 服务。如果选用其他端口，您需要在防火墙中添加一条放行此端口的入站规则。

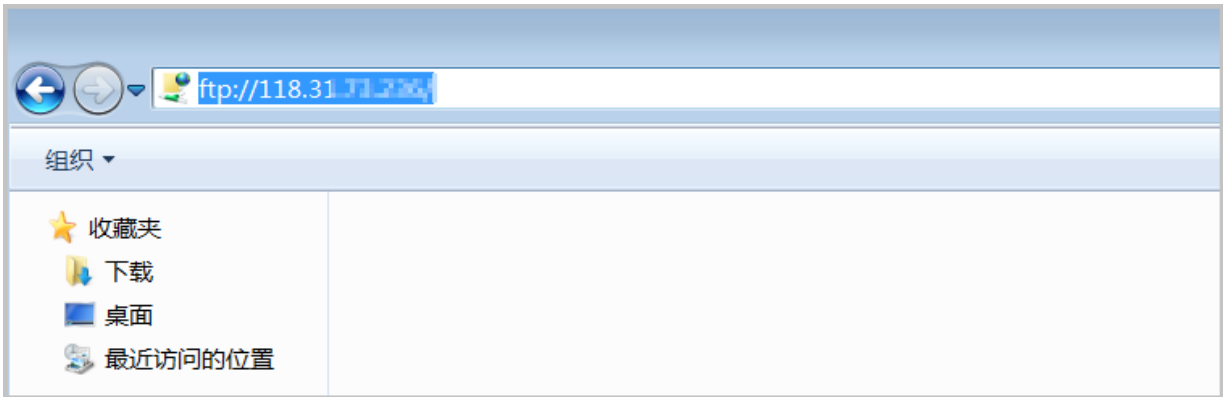
具体方法参见 设置 *ECS* 实例远程连接防火墙。

其他防火墙设置参见 微软官方文档。

步骤六： 客户端测试

打开客户端的 计算机，在路径栏输入 `ftp://服务器 IP 地址:FTP 端口`（如果不填端口则默认访问21端口），例如：`ftp://0.0.0.0:20`。弹出输入用户名和密码的对话框表示配置成功，正确的输入用户名和密码后，即可对 FTP 文件进行相应权限的操作。

> **说明：**
> 客户端使用此方法访问 FTP 站点时，需要对 IE 浏览器进行设置，才能打开 FTP 的文件夹。 打开 IE 浏览器，选择 设置 > **Internet**选项 > 高级。勾选 启用 **FTP** 文件夹视图，取消勾选 使用被动 **FTP**。



后续操作

您可以参考 安全加固方案 对 FTP 服务进行安全加固。

如果您想基于 FTP 协议来管理存储在 OSS 上的文件，安装 *OSS FTP*。OSS FTP 接收普通 FTP 请求后，将对文件、文件夹的操作映射为对 OSS 的操作。