云服务器 ECS

建站教程

建站教程

建站教程汇总

为了方便您搭建网站，本文汇总了阿里云云市场上最常用的应用镜像的使用教程，并提供了教程和云市场镜像的链接，让您一键触达，轻松建站。

网站类型	推荐 OS	镜像及其内含资源	说明
搭建 WordPress 博客	- CentOS - Ubuntu - Aliyun Linux	**Aliyun Linux 一键安装 Web 环境** - Nginx: 1.4.4 - Apache: 2.2.29、2.4.10 - MySQL: 5.1.73、5.5.40、5.6.21 - PHP: 5.2.17、5.3.29、5.4.23、5.5.7 - PHP 扩展: Memcached、Zend Engine/Opcache - JDK: 1.7.0 - Tomcat: 7.0.54 - FTP:（yum/apt-get 安装） - PHPWind: 8.7 GBK - PHPMyAdmin: 4.1.8	- WordPress 是一款常用的搭建个人博客网站的软件。 - 暂不支持自动挂载 I/O 优化的数据盘。
使用资源编排ROS部署 LNMP 环境	- CentOS - Ubuntu - Aliyun Linux	**ROS** - Nginx - MySQL - PHP 5.4.16	- LNMP 分别代表 Linux、Nginx、MySQL、PHP。 - 资源编排服务 ROS 是阿里云官网提供的免费服务，无需下载安装。ROS 通过一个 JSON 格式的模板文件，创建一组阿里云资源。
镜像部署 LNMP 环境	- CentOS - Ubuntu - Aliyun Linux	由所购镜像决定 - Nginx - MySQL - PHP - 其余由所购镜像决定	- 适用于已经购买实例，但想使用镜像重新部署环境的用户。
部署 Java Web 环境	- CentOS 7.3	**JAVA 镜像**	- Tomcat 是开源且免

		- Tomcat - Nginx - MySQL5.7 - Tomcat8	费的 Java Web 服务器，常用作 Web 开发工具。Tomcat 可以托管由 servlet，JSP 页面（动态内容），HTML 页面，JS，Stylesheet，图片（静态内容）组成的 Java Web 应用程序。 - 适合新手，利用云市场丰富的JAVA 镜像资源快捷部署环境。
部署 Java Web 环境	- CentOS 7.3	**OneinStack** - Lnmp - Lamp - Lnmpa - Lnmt	- 适合新手，利用云市场资源快捷部署环境。 - 适合对 Linux 命令有基本了解的用户，满足个性化部署的要求。
搭建 Magento 电商网站	- CentOS 7. 2 - Ubuntu 16.04	**LAMP 镜像** - ApacheMySQL - PHP 7.0.13	- 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。
搭建 PHPWind 论坛系统	- CentOS 6.8	**PHPWind** - Nginx - MySQL5.6 - PHP 5.2, 5.3, 5.4, 5.5, 5.6, 7.0 - PHPWind 9.0.1	- PHPWind 是一款采用 PHP + MySQL 方式运行的开源社区程序。轻架构，高效率简易开发，实现快速搭建并轻松管理社区站点。 - 适用于要搭建论坛的**经典网络**用户。
部署 Drupal 环境	- CentOS 7.2 - Ubuntu 16.04	**LAMP 镜像** - LAMP: 7.0.12 - Apache: 2.4.25 - MySQL: 5.7.17 - PHP: 7.1.1 - Drupal: 8.1.1	- Drupal 是一款采用 PHP 语言编写的开源内容管理框架（CMF），由内容管理系统（CMS）和PHP开发框架（Framework）共同构成。 - 适用于熟悉ECS，Linux系统，刚开始使用ECS 实例建站的用户。
部署 Windows 环境	- Windows	由所购镜像决定	- 适用于已经购买实例、但想使用镜像重新部署环境的用户。
部署 Docker 项目	- CentOS 7.2	由所购镜像决定	- Docker 是一个开源工具，其能将一个 Web 应用封装在一个轻量级，便携且独立的容器里，几乎可以运行在任何服务环境下。 - 适用于熟悉Linux系统，刚开始使用 ECS 实例的开发者。

部署 WDCP 系统	- Linux	由所购镜像决定	- WDCP (WDlinux Control Panel)，是一套通过 Web 控制和管理服务器的 Linux 服务器管理系统以及虚拟主机管理系统。
部署 RabbitMQ 项目	- CentOS 7.3	**RabbitMQ 镜像** - RabbitMQ-Server: 3.6.9 - Erlang: 19.3 - JDK: 1.8.0_121	- RabbitMQ 是一个开源的 AMQP 实现，支持：Python、Ruby、.NET、Java、JMS、C、PHP、ActionScript、XMPP、STOMP、AJAX 等多种客户端。用于在分布式系统中存储转发消息，具有不俗的易用性、扩展性、高可用性。
部署 GitLab 项目	- Linux	**GitLab** - GitLab 7.4.3	- GitLab 通过利用 Ruby on Rails，实现自托管的 Git 项目仓库，可通过 Web 界面轻松访问公开或者私人项目。
部署 PostgreSQL 项目	- CentOS 7.2	由所购 PostgreSQL 镜像决定	- 阿里云版数据库 PostgreSQL 具有 NoSQL 兼容，高效查询，插件化管理，安全稳定的特性。 - 适用于熟悉 ECS，Linux 系统，PostgreSQL 的用户。
部署 SVN 环境	- Centos7	**SVN** - ApachePHP: 5.4.27 - FTP: 2.2.2 - SVN: 1.8.8	- SVN (Subversion) 作为一个开源的版本控制系统, 能管理随时间改变的数据。
搭建 ThinkPHP 框架	- CentOS 6.8	**ThinkPHP** - ThinkPHP: 3.2.3	- ThinkPHP 是一款免费开源的，快速、简单的面向对象的轻量级 PHP 开发框架，遵循 Apache2 开源协议发布，是为了敏捷 Web 应用开发和简化企业应用开发而诞生的。
部署 AMH 虚拟主机系统	Linux	**PHP 运行环境** - AMH 4.2	- AMH 是一套通过 Web 控制和管理服务器的 Linux 服务器管理系统以及虚拟主机管理系统。
部署常用数据库	- Windows Server 2012 64bit	由所购镜像决定	- 常用数据库通常包含：Oracle，MySQL，SQL Server等。

搭建 Joomla 平台	- CentOS 6.5 64bit	由所购镜像决定	- Joomla 是一套知名的内容管理系统。采用 PHP + MySQL 方式开发软件系统。 - 适用于熟悉 ECS，Linux 系统，刚开始使用 ECS 实例建站的用户。
部署 Moodle 系统	- Centos 7.0 64bit	**Moodle** - Apache: 2.4.6 - PHP: 7.0 - MySQL: 5.6 - Moodle: 3.1.2	- Moodle 是一个开源课程管理系统，采用 PHP + MySQL 方式运行软件，遵循 GNU 公共许可协议。Moodle 平台界面简单精巧，用户可以根据需要随时调整界面，增减内容。

熟悉搭建云服务器环境的用户还可以参阅**详细的建站教程**，浏览并选购**云市场**产品，个性化搭建网络环境。

零基础入门—网站建站教程（新手必备）

前言

相信很多新用户会有这样的疑惑，我要做个网站，到底要使用什么产品，如何能快速完成网站建站呢？搭建网站有两种选择，一种是直接购买建站模板，另一种则是自行建站。两类建站方式对比如下：

建站方式	**优势**	**适用人群**
购买建站模板	即买即用，轻松便捷，后台管理方便，且有专人进行网站维护。	预算相对充足、希望节省人力、使用需求迫切的用户。
DIY自行建站	服务器购买、网站搭建、网站维护全程自主，弹性灵活。	希望自行设计、有动手意愿的用户。

如果需要购买模板，阿里云云市场提供丰富的建站，并且支持网站定制。如：

- **云市场企业官网定制**：适合看重网站个性化的企业用户，成本较高。
- **云市场网站模板**：适合无特殊需求的个人和小企业用户，支持 PC、手机、微信等多种渠道，选择多样，成本较低。
- **网站基础环境搭建服务**，省心省力。

以下介绍自行搭建网站的流程，适用于刚接触云计算或对云服务器和建站不太了解的但有建站需求的个人/小企业用户，内容比较基础，但覆盖了新手用户从服务器选择到完成建站整个流程的**零基础入门教程**，强烈推荐给

新手用户。

自建步骤

自行建站的四个基本步骤：

1. 服务器选择

不同网站类型需要的服务器配置不同，首先确认网站规模与访问人数，一般情况下，小型网站只需要选择基础配置即可。购买服务器流程可参考通过购买页面创建一台ECS实例。

想了解不同配置对应的价格，推荐使用ECS价格计算器。

云服务器ECS主要计费方式为预付（包括包年包月和按周付费）和按量付费，详细的计费规则可以参考产品计价。

2. 域名购买和备案

域名购买：输入想要的域名，未被占用即可注册，具体操作请参考域名注册流程。域名后缀通常为.COM或.CN，其它后缀介绍和区别请参考域名区别。

注意：域名注册成功后需进行实名认证。流程请参考域名如何实名制认证。

域名备案：域名购买完成后，还需进行备案方可使用。备案步骤如下：

i. 备案准备：因各省管局要求存在差异，所需资料也不尽相同，请根据各省市备案规则准备资料，或访问**工信部备案管理系统**（www.miitbeian.gov.cn） 了解细则。详情请参考备案准备。

首次备案接入：若之前尚未进行过工信部备案，请参考首次备案流程图文引导。

其它备案场景请参考备案导航。

3. 网站部署

常见网站类型有以下几种：

个人博客

- WordPress 常用于搭建个人博客网站，尤其适用于首次使用阿里云进行建站的新用户。详情请参考使用 WordPress 搭建个人手机博客。

- 若已有网站代码，可通过阿里云 Windows 一键安装web环境来部署。

论坛网站

- 通过 phpwind 镜像快速搭建论坛网站请参考使用 phpwind 镜像搭建论坛。
- 通过 Discuz 搭建论坛网站请参考使用 Discuz 镜像搭建论坛。

其他建站教程

通过镜像一键部署 Linux 环境

通过镜像部署 Java Web 环境

部署 Linux 主机管理系统 WDCP

- 快速搭建 Moodle 课程管理系统

初级建站推荐使用网上一键安装包，例如WordPress、discuz这类，比较适合新手操作。

4. 域名解析

设置域名解析后，外部用户可通过域名访问网站。详情请参考设置域名解析快速入门。

如需将域名指向一个IP地址，添加A记录即可。详情请参考如何添加不同记录类型的解析。

至此，建站操作已基本完成，接下来您可使用域名测试访问是否正常。

常见问题与解决方案

在服务器使用或者网站搭建过程中，可能会遇到一些使用问题，因此，列举了以下最常见的问题和对应的解决方案。

服务器远程连接

Windows无法远程连接：

- 文档解决方案：无法连接 Windows 实例。
- 视频教程：如何远程连接 Windows 实例。

Linux无法远程连接：

- 文档解决方案：无法连接 Linux 实例。
- 视频教程：如何远程连接 Linux 实例。

安全组和快照

- 如何设置安全组默认规则？
- 安全组应用案例合集。
- 如何创建自定义快照？

网站无法访问

- Windows 实例带宽和CPU跑满或跑高怎么办？
- 网站无法访问的常见原因及检查方法有哪些？
- Windows 实例网络访问丢包延时高怎么办？
- Linux 实例网站访问丢包延时高如何解决？
- 域名解析已经生效，无法打开网站常规的原因有哪些？

相关服务

- 若在上云前希望了解如何选择适合自身业务特点的阿里云产品和配置，可点此查看架构设计&上云咨询服务。
- 若您需要将云下自建机房、托管机房等环境下的业务迁移到阿里云上， 并希望获取专业上云方案实施服务，可点此查看上云方案实施服务。
- 若您需要在阿里云服务器上部署站点环境、安装应用程序，可点此查看网站基础环境搭建服务。
- 基于阿里云服务器 ECS，若需要专业工程师协助对系统/数据库/站点进行基础设置，可使用云资源管理基础设置服务。

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。

登录 阿里云。如果尚未注册，单击 免费注册。

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**。

在云服务器ECS **自定义购买** 页面，完成如下 **基础配置**：

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。

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**。

说明：

i. 实例创建成功后，不能更改地域和可用区。
ii. 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。

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**。

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。

选择 **存储**：

i. **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。
ii. **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。

单击 **下一步：网络和安全组**，并完成 **网络和安全组** 配置：

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**。

说明：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**。

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。

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**。

> **说明**：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。

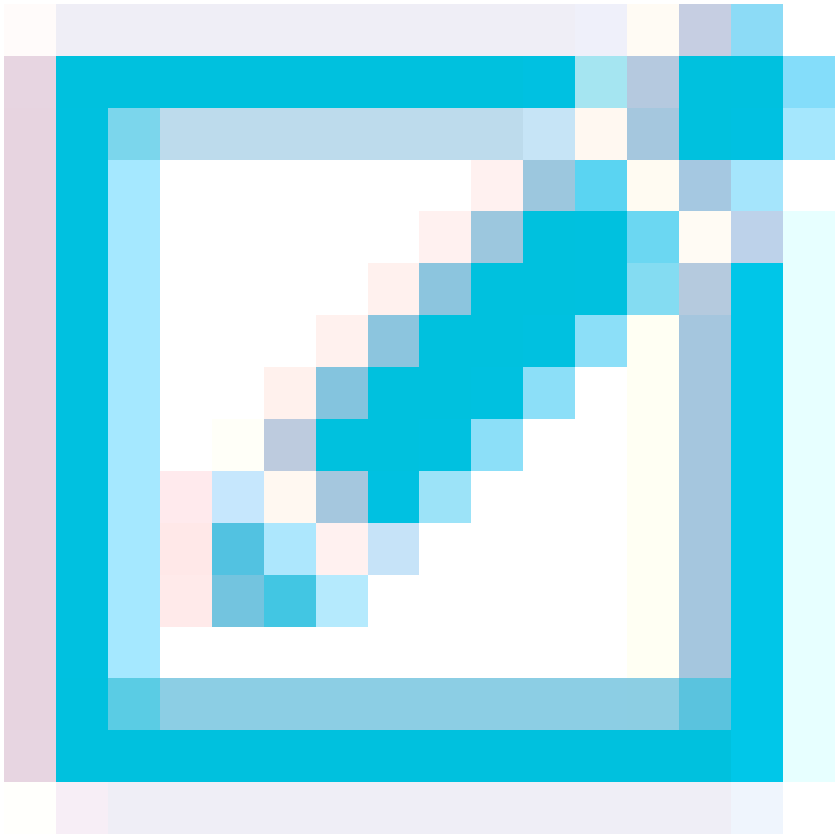
单击 **下一步：系统配置**，完成系统配置：

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。

设置 **实例名称**、**描述** 和 **主机名**，为了便于以后管理。

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。

阅读并确认《云服务器 ECS 服务条款》和《镜像商品使用条款》。

单击 **确认下单**。

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。

远程连接Windows实例。

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。

输入用户名 **admin** 和初始密码 **admin123**，并单击 **登录**。

在 **仪表盘** 页面的左侧导航栏中，选择 **外观** > **主题**。

在 **主题** 页面上选择已经安装的主题。

说明：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。

部署LNMP

一键部署LNMP环境

LNMP分别代表Linux、Nginx、MySQL、PHP。本文介绍如何使用阿里云 资源编排服务（ROS） 一键在ECS实例搭建LNMP环境。

ROS是阿里云官网提供的免费服务，无需下载安装。您可以使用ROS创建JSON格式的资源栈模板文件，或者使用ROS提供的 模板样例 创建一组阿里云资源。在本教程中，我们会使用ROS控制台提供的 LNMP_basic 模板，自动创建一台ECS实例，并在实例上部署LNMP环境。

前提条件

创建按量付费资源时，账号余额不能低于100.00元，可以是现金、可用信用额度或者可用于开通产品的代金券

。

操作步骤

登录 ROS管理控制台。

> **说明**：如果您是首次使用ROS，必须先开通ROS服务。ROS服务免费，开通服务不会产生任何费用。

在左侧导航栏中，选择 **关键帮助** > **ECS实例相关信息**，获取您需要的ECS实例规格、可用区ID（ZoneId）和镜像ID（ImageId）。

在左侧导航栏中，单击 **模板样例**。

从模板样例中，找到 **LNMP_basic**。

单击 **预览** 按钮查看模板的JSON文件。JSON文件各个顶级字段的解释如下表所示。

顶级字段	解释
"ROSTemplateFormatVersion" : "2015-09-01"	定义模板版本。
"Description": "Deploy LNMP(Linux+Nginx+MySQL+PHP) stack on 1 ECS instance. *** WARNING *** Only support CentOS-7."	解释说明模板。
"Parameters" : { }	定义模板的一些参数。本示例中，模板定义的参数包括：镜像ID、实例规格等，并指定了默认值。
"Resources" : { }	定义这个模板将要创建的阿里云资源。本示

	例中，申明将要创建一个ECS实例和一个安全组，这里申明的资源属性可以引用Parameters中定义的参数。
"Outputs": { }	定义资源创建完成后，栈需要输出的资源信息。本示例中，资源创建完成后将输出ECS实例ID、公网IP地址和安全组ID。

说明：关于ROS资源栈模板的更多信息，请参见 *资源编排* 的 模板结构说明。

单击 **创建Stack**。

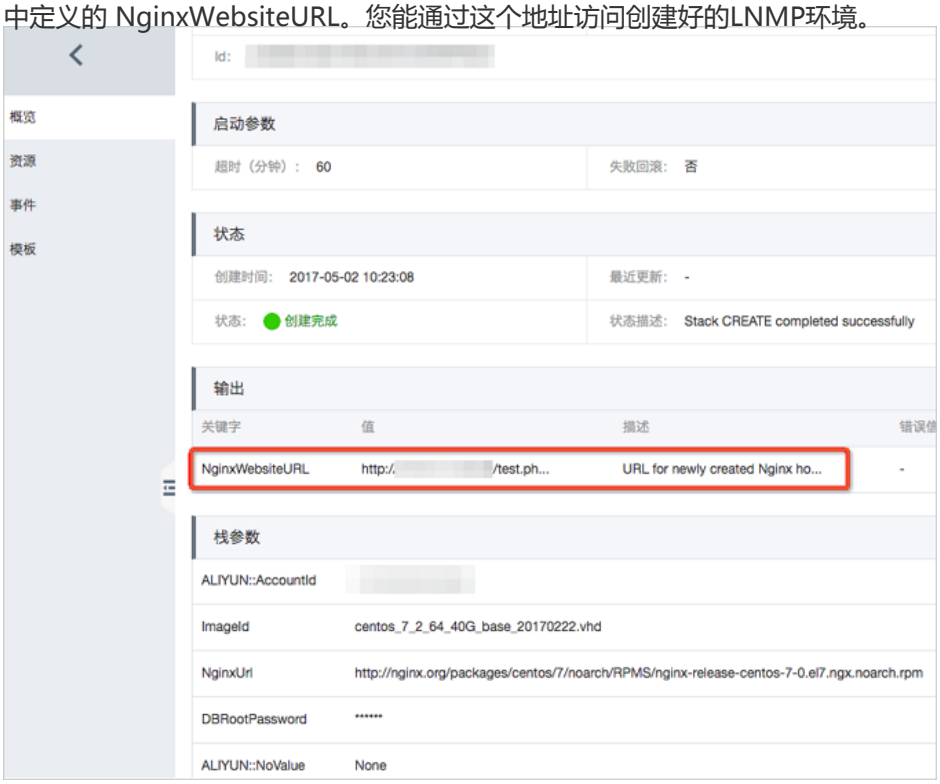
在 **直接输入** 部分，在 **所在region** 的下拉框中选择具体地域，并在页面右下角单击 **下一步**。本例选择 **华东2**。

在 **启动栈** 部分，设置参数：

- **栈名**：设置一个栈名，不可重复，而且创建之后不能修改。
- **创建超时**：设置一个时间。如果在设置的时间段内资源未创建成功，则判断超时。您可以选择是否 **失败回滚**。如果选择失败回滚，那么创建过程中发生任何失败（包括创建超时），ROS都会删除已经创建成功的资源。
- **NginxDownloadUrl**：使用默认的Nginx下载地址。
- **DBPassword** 和 **Please Confirm DBPassword**：设置并确认访问MySQL数据库的密码。根据模板定义，密码只能包括英文字母和数字。
- **ZoneId**：填写您需要创建资源的可用区。详见第2步。
- **DBUser**：创建一个用户，访问MySQL数据库。
- **DBRootPassword** 和 **Please Confirm DBRootPassword**：设置并确认MySQL root账号的密码。根据模板定义，密码只能包括英文字母和数字。
- **InstanceType**：填写您需要的ECS实例规格。详见第2步。
- **SystemDiskCategory**：选择云盘类型，作为系统盘。
- **InstancePassword** 和 **Please Confirm InstancePassword**：设置并确认实例的登录密码。根据模板定义，密码只能包括大写或小写英文字母和数字。

单击 **创建**，页面将提示 **创建请求提交成功。**

在左侧导航栏中，单击 **资源栈管理** 查看栈的状态。当栈创建成功后，在 **输出** 部分查看 Outputs 中定义的 NginxWebsiteURL。您能通过这个地址访问创建好的LNMP环境。



说明：

- 在 **资源** 列表中查看栈中所有资源。
- 在 **事件** 列表中查看ROS创建这个资源栈过程中产生的操作记录。任何涉及资源栈的操作失败了，列表中都会显示资源操作失败的原因。
- 在 **模板** 列表中查看资源栈的原始模板。

参考信息

您还可以使用ROS提供的其他模板样例搭建环境，比如Java Web测试环境、Node.js测试开发环境、Ruby Web开发测试环境或Hadoop/Spark分布式系统。

更多模板，请参见 模板样例。

更多ROS信息，请参见 ROS帮助中心 和 ROS云栖博客。

镜像部署 Linux 环境

您可以根据业务需要，选择下列任意一种方式部署云服务器 ECS 实例的使用环境：

- **镜像部署**
- **手动部署**

下表列出了两种部署方式的特点。一般推荐镜像部署。如果您需要个性化定制部署，建议使用手动部署。

对比项	镜像部署	手动部署
部署所需时间	3-5分钟，快速部署上云	1-2天。选择适合的操作系统、中间件、数据库、各类软件、插件、脚本，再进行安装和配置
专业性 IOPS	由运维过万级用户的优质服务商提供	依赖开发人员的开发水平
个性化	支持主流应用场景	可满足个性化的部署要求
安全性	经过严格安全审核，集成最稳定安全的版本	依赖开发人员的开发水平
售后服务	专业售后工程师团队支持	依赖运维人员的经验，或由外包团队支持

注意： 本文档只介绍通用的操作步骤。一般镜像软件安装包都包含了操作指南，请阅读镜像操作指南进行具体的安装和配置。

阿里云的云市场提供了丰富的镜像资源。镜像集成了操作系统和应用程序。在创建实例时，您可以选择包含了

应用环境的镜像，创建后无需再部署环境。

注意：云服务器 ECS 不支持虚拟化软件（如 KVM、Xen、VMware 等）的安装部署。

操作步骤

说明：本节介绍的方法适用于已经购买实例、但想使用镜像重新部署环境的用户。此外，您也可以在创建实例的时候就选择镜像，请参考创建实例。

如果您想使用镜像市场的镜像来替换当前实例的操作系统，可以通过本节介绍的更换系统盘的方法来实现。

更换系统盘的时候，**数据盘**的数据则不会受到影响。因此建议您将系统盘的个人数据备份到数据盘中，或采用其他方式进行备份。

更换系统盘后，IP 地址不会改变。

如果您购买的实例已经开始运行，但是您想使用镜像市场中的镜像重新部署环境，操作步骤如下：

登录云服务器管理控制台。

找到需要重新部署环境的实例。

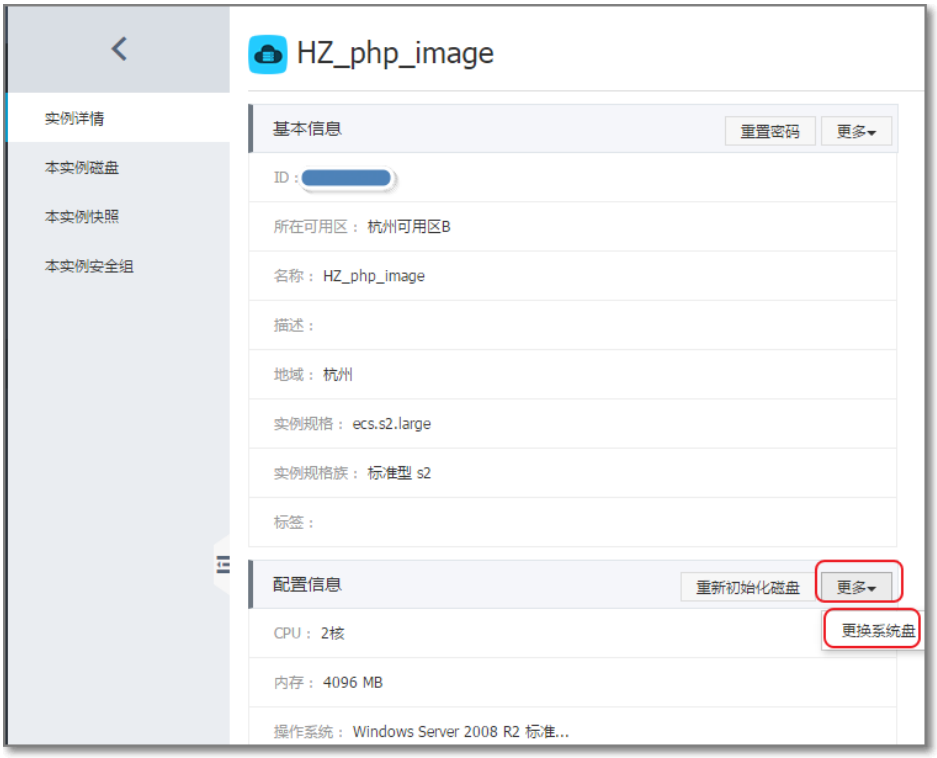
如果该实例刚刚创建，可以直接停止实例。如果实例已经运行了一段时间，您想保留其中的数据，请在操作前将数据备份到数据盘中。
注意：在更换镜像后，系统盘的数据会全部被清空，服务器的自动备份的快照也可能会被删除（取决于您的设置，请参见 自动快照随磁盘释放）。因此务必做好数据备份工作。

停止实例。

确认停止实例。

实例停止后，单击实例名称，或者单击右侧的 **管理**。

在左侧的 **配置信息** 中，单击 **更多** > **更换系统盘**。



在提示消息中，单击 **确定，更换系统盘**。

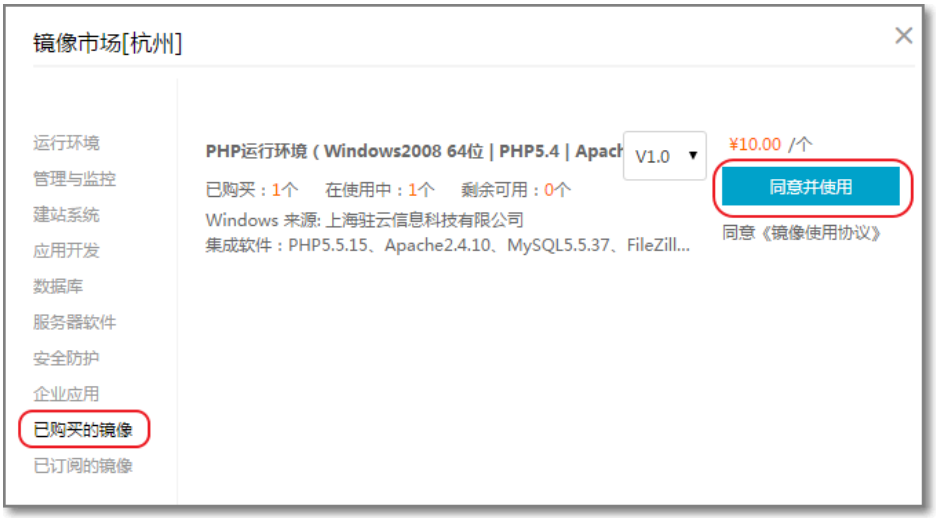
单击 **镜像市场**，然后单击 **从镜像市场选择（含操作系统）**。

镜像市场列表的左侧是镜像的分类。您可以根据分类，选择想使用的镜像。找到需要的镜像后，单击镜像右下方的 **同意并使用**。

注意在左侧最下方，有两个按钮：**已购买的镜像**和**已订阅的镜像**。如果您已经购买过镜像，可以直接单击**已购买的镜像**，选择镜像。

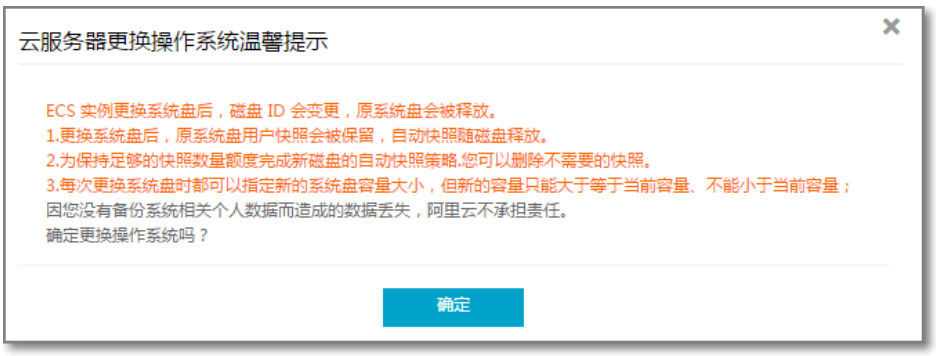
下图是选择已经购买的镜像的示例。单击 **同意并使用**。

注意：在此页面，不要单击镜像连接，否则会直接引导您到购买镜像页面，引起误解。



继续选择系统盘，输入登录密码，然后单击 **去支付**。

您会看到更换操作系统的提示。单击 **确定**。



您成功使用镜像部署了环境。现在可以启动、并登录实例，开始使用您的环境了。

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)
```

注：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

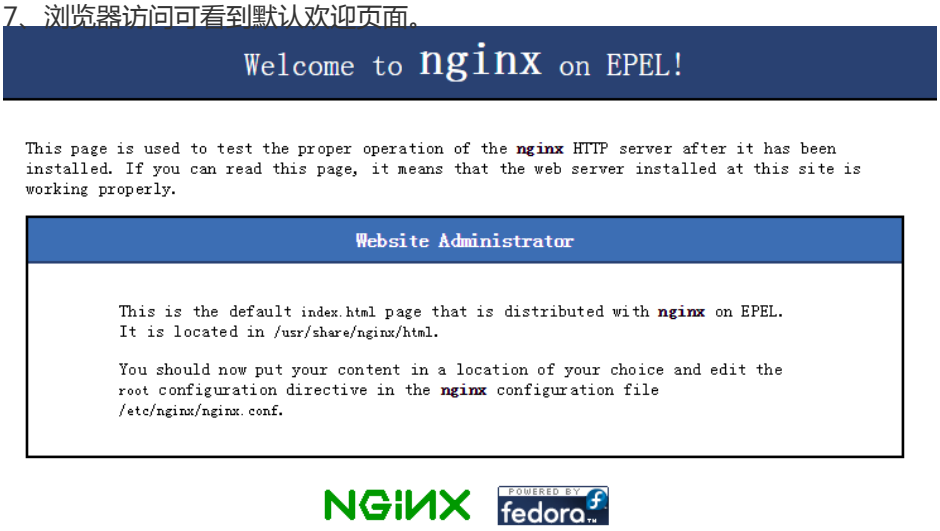
5、添加至服务管理列表，设置开机自启。

```
# 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/
```

注：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_script_name;
include fastcgi_params;
}
```

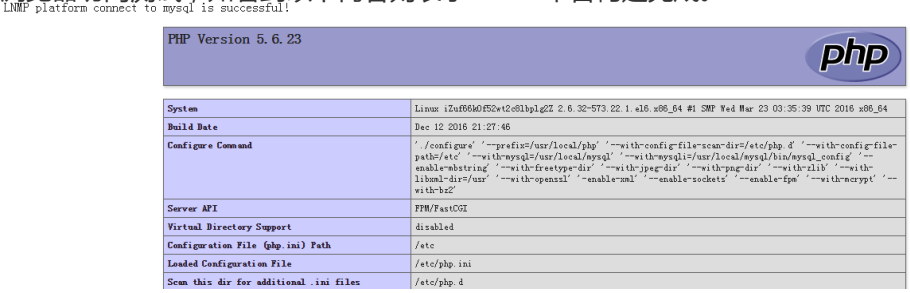
重新载入nginx的配置文件。

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。



部署Java Web

镜像部署Java Web项目

Tomcat 作为一个开源且免费的 Java Web 服务器，常用来作为 Web 开发的工具。它可以托管由 Servlet、JSP 页面（动态内容）、HTML 页面、JS、样式表、图片（静态内容）组成的 Java Web 应用程序。

部署方式

在阿里云服务器下部署 Java 提供 3 种部署方式：

- JAVA 镜像部署
- 一键安装包部署
- 手动部署（源码编译安装/YUM安装）

一般推荐使用镜像部署，尤其适合新手，使用更加快捷方便（阿里云的云市场提供了丰富的镜像软件，点击查看）。而安装包部署以及手动部署适合对 Linux 命令有基本了解的用户，可以满足用户个性化部署的要求。本文主要介绍镜像和手工部署的方式。

镜像部署

单击 JAVA 环境（CentOS7.3 Nginx Tomcat8 JDK）进入镜像详情页。

单击 **立即购买**，按提示步骤购买 ECS 实例。

登录 ECS 管理控制台。

在左边导航栏中，单击 **实例**，进入 ECS 实例列表页。

选择所购 ECS 实例所在的地域，找到已购的 ECS 实例，在 **IP 地址** 列获取该实例的公网 IP 地址。

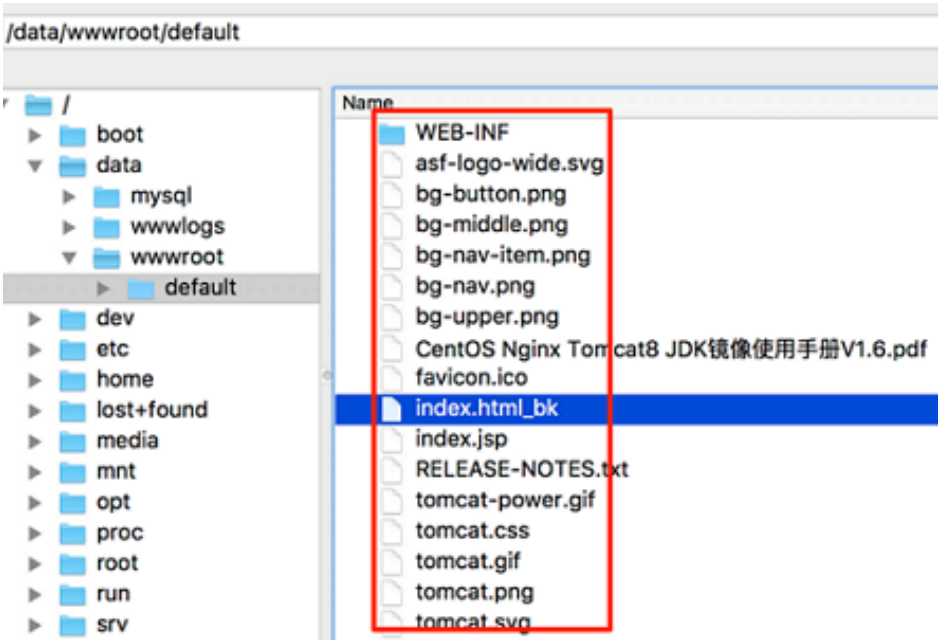
在浏览器地址栏中输入http://公网 IP 地址后，收藏在线文档。

注意：若输入公网后无法显示下述页面，请检查安全组公网入方向已开通80端口。

使用 Putty 登录 Linux 服务器，参考连接Linux实例。

说明：若创建实例时未设置密码，root需重置实例密码。

使用 Winscp 工具将 Java 代码放入 /data/wwwroot/default 中。



默认 Tomcat 是以一般 www 用户运行，将网站代码权限改为 www，执行命令：

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

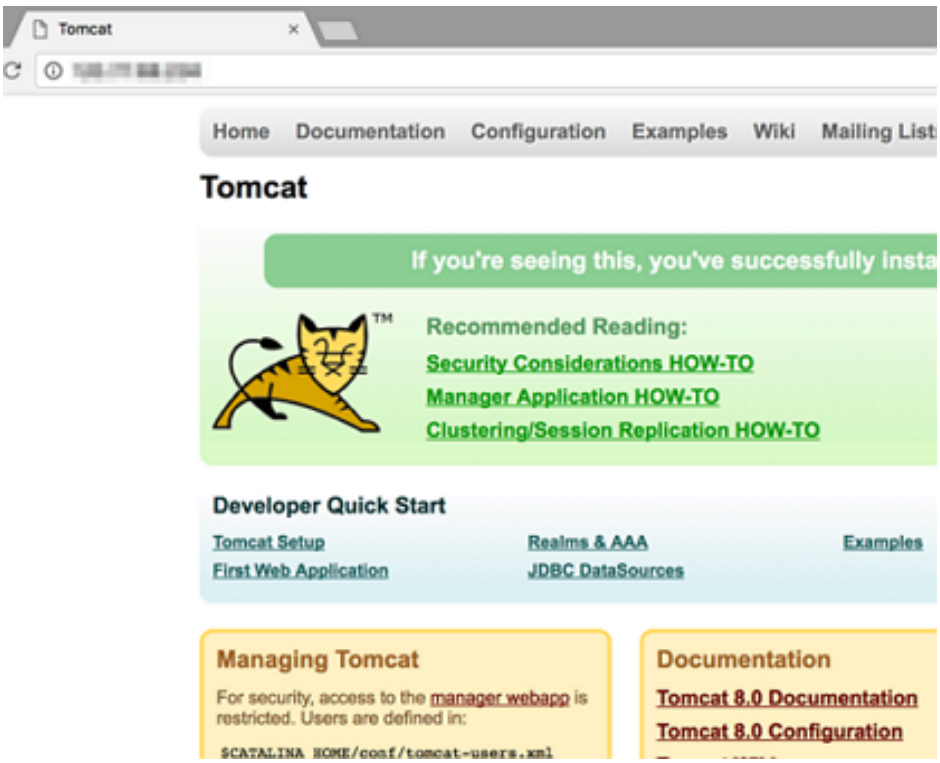
```
[root@i-2wz3l6d2wdoh2fl13nwhgZ ~]# cd /data/wwwroot/default/
[root@i-2wz3l6d2wdoh2fl13nwhgZ default]# ls -l | more
total 2956
-rw-r--r-- 1 root root   26447 Jan 19 06:25 asf-logo-wide.svg
-rw-r--r-- 1 root root     713 Jan 19 06:21 bg-button.png
-rw-r--r-- 1 root root    1918 Jan 19 06:21 bg-middle.png
-rw-r--r-- 1 root root    1392 Jan 19 06:21 bg-nav-item.png
-rw-r--r-- 1 root root    1401 Jan 19 06:21 bg-nav.png
-rw-r--r-- 1 root root    3103 Jan 19 06:21 bg-upper.png
-rw-r--r-- 1 root root 2821382 Mar 11 20:20 CentOS Nginx Tomcat8 JDK镜像使用手册
V1.6.pdf
-rw-r--r-- 1 root root   21630 Jan 19 06:21 favicon.ico
-rw-r--r-- 1 root root    2739 Mar 11 20:22 index.html_bk
-rw-r--r-- 1 root root   12279 Jan 19 06:25 index.jsp
-rw-r--r-- 1 root root    6741 Jan 19 06:25 RELEASE-NOTES.txt
-rw-r--r-- 1 root root    5581 Jan 19 06:25 tomcat.css
-rw-r--r-- 1 root root    2066 Jan 19 06:21 tomcat.gif
-rw-r--r-- 1 root root    5103 Jan 19 06:21 tomcat.png
-rw-r--r-- 1 root root    2376 Jan 19 06:21 tomcat-power.gif
-rw-r--r-- 1 root root   67795 Jan 19 06:25 tomcat.svg
drwxr-xr-x 2 root root    4096 Apr 13 20:20 WEB-INF
[root@i-2wz3l6d2wdoh2fl13nwhgZ default]# chown -R www.www /data/wwwroot/default/
[root@i-2wz3l6d2wdoh2fl13nwhgZ default]# ls -l | more
total 2956
-rw-r--r-- 1 www www   26447 Jan 19 06:25 asf-logo-wide.svg
-rw-r--r-- 1 www www     713 Jan 19 06:21 bg-button.png
-rw-r--r-- 1 www www    1918 Jan 19 06:21 bg-middle.png
```

重启 Tomcat。

```
service tomcat restart
```

```
[root@i-2wz3l6d2wdoh2fl13nwhgZ default]# service tomcat restart
Stoping Tomcat
Using CATALINA_BASE:   /usr/local/tomcat
Using CATALINA_HOME:   /usr/local/tomcat
Using CATALINA_TMPDIR: /usr/local/tomcat/temp
Using JRE_HOME:        /usr/java/jdk1.8.0_121
Using CLASSPATH:       /usr/local/tomcat/bin/bootstrap.jar:/usr/local/tomcat/bin
/tomcat-juli.jar
waiting for processes to exit
Starting tomcat
Using CATALINA_BASE:   /usr/local/tomcat
Using CATALINA_HOME:   /usr/local/tomcat
Using CATALINA_TMPDIR: /usr/local/tomcat/temp
Using JRE_HOME:        /usr/java/jdk1.8.0_121
Using CLASSPATH:       /usr/local/tomcat/bin/bootstrap.jar:/usr/local/tomcat/bin
/tomcat-juli.jar
Tomcat started.
Tomcat is running with pid: 10741
```

在浏览器地址栏中输入公网 IP 地址，完成验证。



手工部署Java Web项目

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。

配置要求

这里列出的软件版本仅代表写作本文档使用的版本。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

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

说明：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。

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

说明：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。

新建一个目录：

```
mkdir /usr/java
```

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

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

设置环境变量：

编辑 /etc/profile：vi /etc/profile。

按 i 键进入编辑模式。

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
```

按 Esc 键退出编辑模式，输入 :wq 保存并关闭文件。

加载环境变量：source /etc/profile。

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 "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。

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 目录中存放 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。

配置 server.xml 文件：

切换到 /usr/local/tomcat/conf/ 目录：cd /usr/local/tomcat/conf/。

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

创建一个新的 server.xml 文件：

运行命令 vi server.xml。

按 i 键进入编辑模式。

添加以下内容：

```
<?xml version="1.0" encoding="UTF-8"?>
<Server port="8006" shutdown="SHUTDOWN">
<Listener className="org.apache.catalina.core.JreMemoryLeakPreventionListener"/>
<Listener className="org.apache.catalina.mbeans.GlobalResourcesLifecycleListener"/>
<Listener className="org.apache.catalina.core.ThreadLocalLeakPreventionListener"/>
<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>
```

设置 JVM 内存参数：

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

ii. 按 i 键进入编辑模式。

添加以下内容：

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

按 Esc 键退出编辑模式，输入 :wq 保存并退出文件。

设置 Tomcat 自启动脚本。

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

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

添加执行权限：chmod +x /etc/init.d/tomcat

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
```

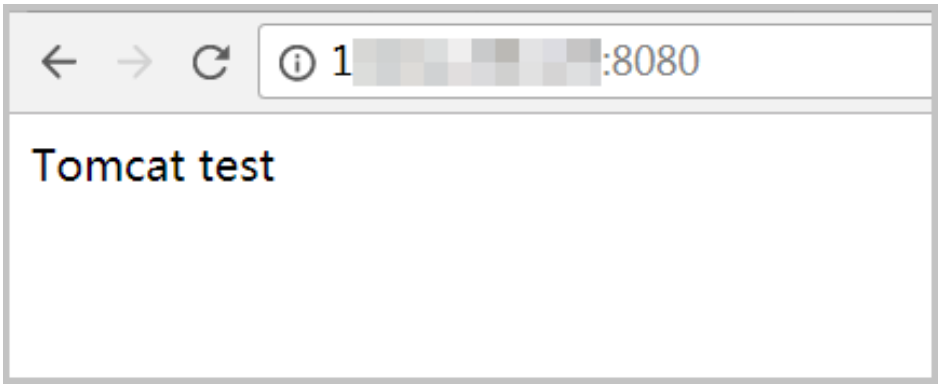
设置自启动。

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

启动 Tomcat。

```
service tomcat start
```

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。



创建基于ECS和RDS的WordPress环境

您可以在资源编排服务ROS (Resource Orchestration Service)中通过模版创建一组阿里云资源。 ROS 的控制台已经提供了一些常用的模版样例。本文将使用一个 ROS 模版创建基于 ECS 和 RDS 的 WordPress 环境。

说明：ROS 模版是一个 JSON 格式文本文件，您可以在这个文本中定义自己的阿里云资源。

前提条件

阿里云规定创建资源时，账号需要有超过 100 元的现金、可用信用额度或者可用于开通产品的代金券。

操作步骤

登录 ROS 管理控制台。

说明：如果您是首次使用 ROS ，那么需要接受 ROS 的协议，同意开通 ROS 服务。ROS 服务是免费，开通服务不会产生任何费用。

在控制台左侧导航栏中，单击 **模版样例**，页面显示 ROS 提供的常用模版。

从模版示例中找到 **wordpress_instance**，通过这个模版将创建基于 ECS 和 RDS 的 WordPress 环境。

每个模版样例下方都有一个 **预览** 和 **创建** 按钮，单击 **预览** 按钮将显示 JSON 模版，单击 **创建 stack** 。

这个 JSON 文本包含五个顶级字段：

定义模版版本版本："ROSTemplateFormatVersion" : "2015-09-01"。

定义对模版的解释说明:"Description": "一个简配的ecs实例，包括一个安全组，用户只需要指定imageId"。

定义模版的一些参数，本例中定义了镜像ID的参数，实例规格的参数，并指定了默认值："Parameters" : { }。

定义这个模版将要创建的阿里云资源，本例中申明将要创建一个 ECS 实例和一个安全组；这里申明的资源属性可以引用Parameters中定义的参数："Resources" : { }。

定义资源创建完成后，通过 ROS 的栈输出资源信息。本例中，将输出 ECS 实例的 ID ，公网 IP 和安全组 ID："Outputs": { }

说明：您可以在线编写模板，也可以通过 URL 地址获取模板。

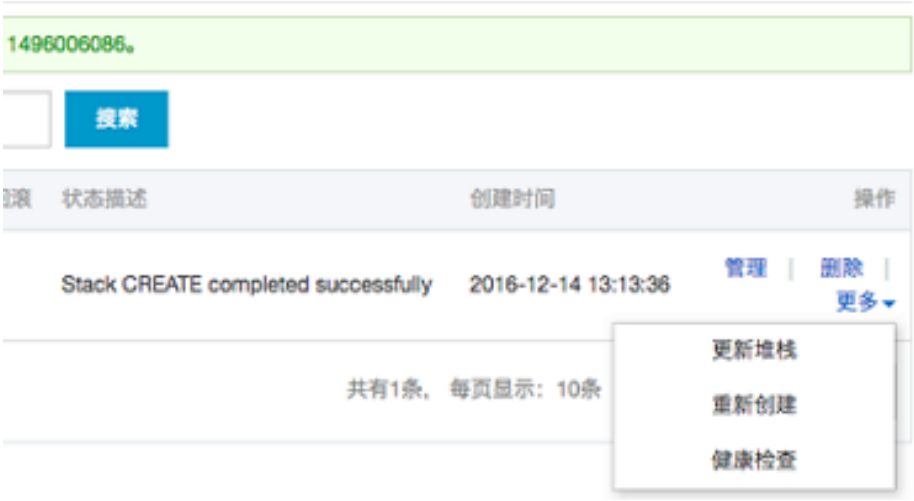
在 **创建 Stack** 的页面中，**所在region** 的下拉框中选择具体地域，本例选择 **华东1**，在页面右下角单击 **下一步**。

填写所有带 * 的选项，完成后单击 **创建**，页面将提示 **创建请求提交成功**。

单击左侧导航栏的 **资源栈管理** 查看 stack 的状态。当栈创建成功后，Outputs中定义的那些值，就会输出。

说明：通过 **资源** 可以查看 stack 中的所有资源；通过 **事件** 可以查看 ROS 创建这个资源栈时的操作记录。任何涉及资源栈的操作失败了，会显示具体操作哪个资源失败的原因；通过 **模版** 可以查看资源栈的原始模版。

以上示例只是通过 ROS 的文本模版快速创建资源， ROS 也可以通过用户指定的模版 URL 地址创建资源。除此之外，ROS 同时有管理资源的能力。用户可以删除自己的资源组，或者只删除 stack 而保留资源，还能根据自己的需求更新一个资源栈，重新检查资源栈的状态等等。



若您想了解 ROS 中资源创建的其它操作，详情请参见快速入门 。

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、定位到 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。

注意：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**。

注意：

- 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

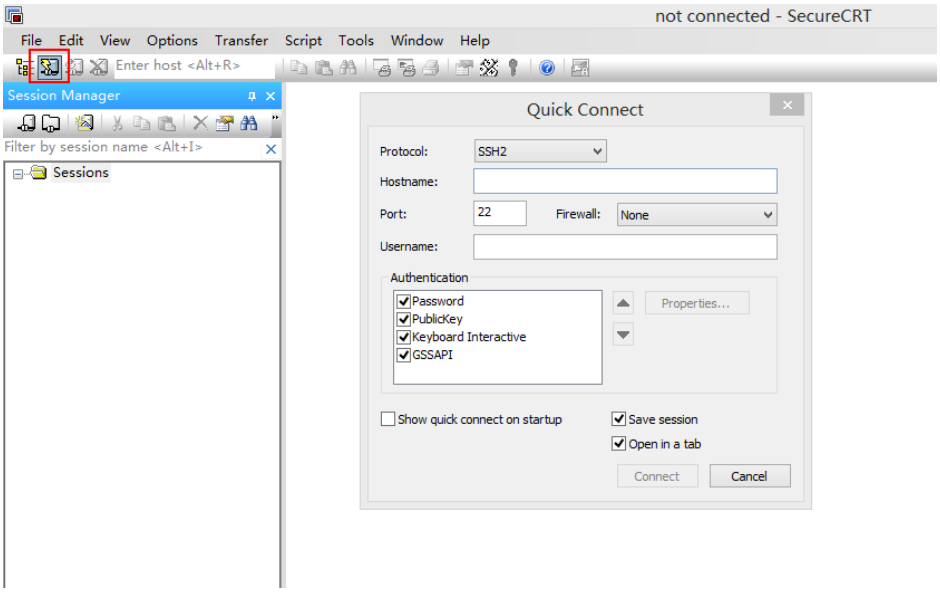
部署之前，请确保：

- 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

- 确保您安装了连接 Linux 实例的工具，如 SecureCRT。
- 打开 SecureCRT ，设置登录实例所需的信息。
- 设置连接名称。
- 协议选择 **SSH**。
- 输入主机 IP 地址和用户名。

然后单击 **确定** 保存。



- 输入用户名 root 和登录密码。

- 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

- 安装Nginx：

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

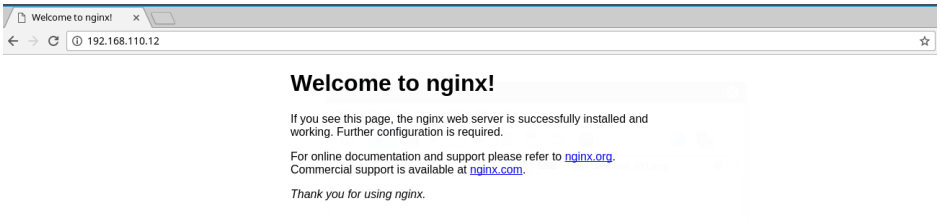
- 设置Nginx服务器自动启动：

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

- 启动Nginx并查看Nginx服务状态：

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

- 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

- 安装EPEL：

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

- 安装Node.js 和 npm：

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

- 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

- 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

- 创建Ghost安装目录：

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

- 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
```

- 解压Ghost安装包：

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

- 使用npm安装Ghost：

[root@localhost ghost]# npm install -production

安装完成后用 npm start 命令启动ghost，检查有没有安装成功

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

- 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'
        }
    },
```

- 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"

- 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

- 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

- 安装Nginx：

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

- 设置Nginx服务器自动启动：

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

- 启动Nginx并查看Nginx服务状态：

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

- 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

- 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

- 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;
    }

}
~
```

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

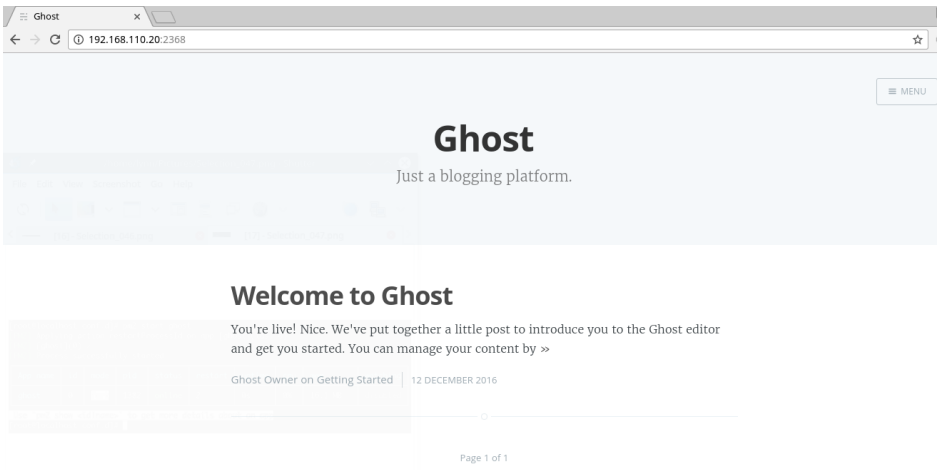
[root@localhost ~]#mv default.conf default.conf.bak

- 重启Nginx服务：

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

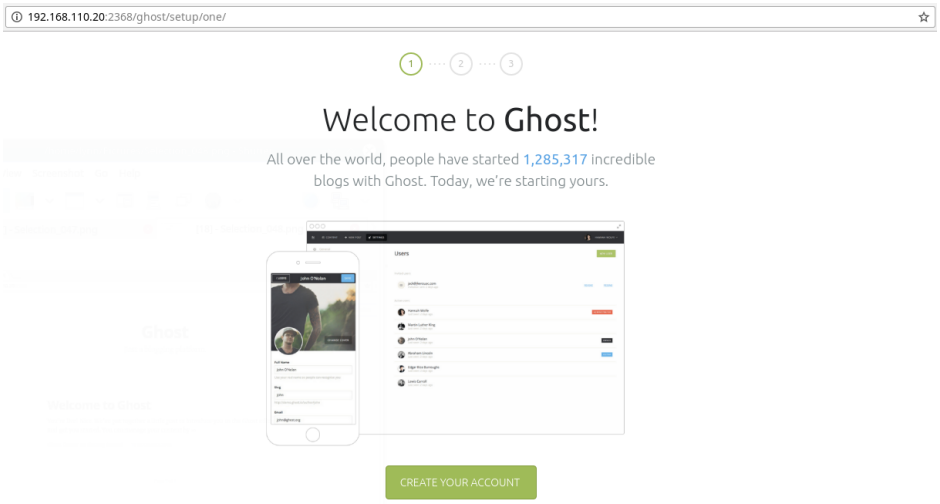
6、访问Ghost博客

- 在浏览器输入http://IP 或 http://域名 即可访问Ghost：



注：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；

- 需要对博客进行编辑修改，可在浏览器输入：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。打开 **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**。

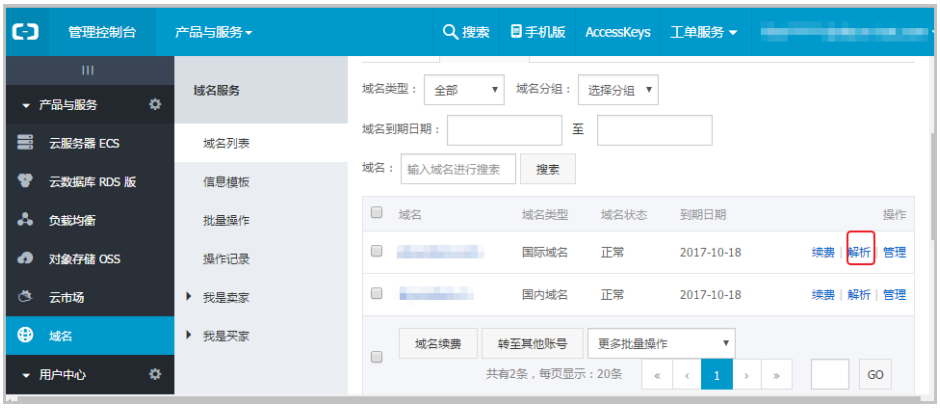
步骤 6：配置域名解析

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！

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 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 创建ECS实例 和 添加安全组规则。

	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
	入方向	允许	自定义TCP	80/80	地址段访问	0.0.0.0/0	1
	入方向	允许	自定义TCP	3306/3306	地址段访问	0.0.0.0/0	1

操作步骤

使用云服务器ECS搭建Magento网站的操作步骤如下：

步骤1. 安装配置LAMP平台
步骤2. 创建数据库
步骤3. 安装配置Composer
步骤4. 安装配置Magento
步骤5. 添加cron作业

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。

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。

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

启动HTTP和MySQL服务并设置开机自启动。

```
# systemctl start httpd
# systemctl enable httpd
# systemctl start mysqld
# systemctl enable mysqld
```

编辑Apache配置文件：

i. 运行命令vim /etc/httpd/conf/httpd.conf。

ii. 按 i 键进入编辑模式。
iii. 做以下修改：
i. 在 Include conf.modules.d/*.conf 之后添加 LoadModule rewrite_module modules/mod_rewrite.so。
ii. 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
```

iv. 按 Esc 键退出编辑，并输入 :wq 保存并退出。

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
```

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
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!
```

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
```

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
```

编辑配置文件/etc/php.ini：

i. 运行命令 vim /etc/php.ini。
ii. 按 i 进入编辑模式。
iii. 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。
```

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。

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)
```

运行 exit 退出MySQL。

（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。

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
```

配置Composer全局使用。

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

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。

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
```

（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'
```

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 ..
```

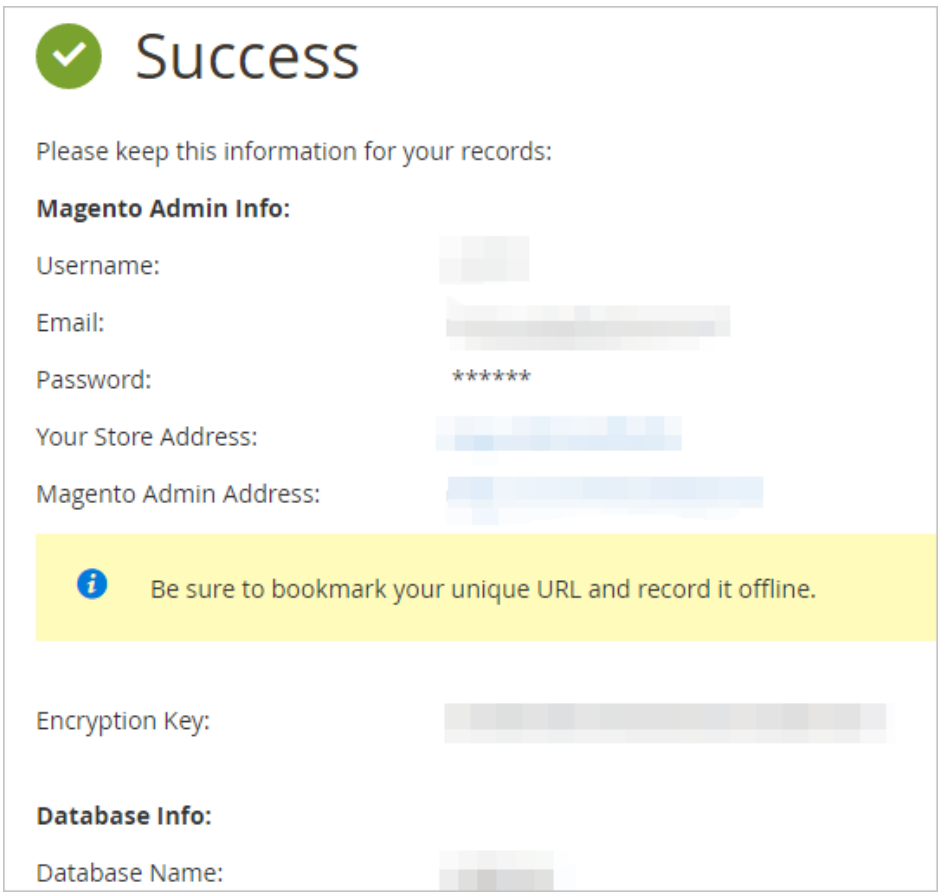
设置Magento文件适当的权限。

```
# 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
```

运行 composer install 安装Magento。

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。

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

运行 crontab -u apache -e 设置cron运行调度工作。

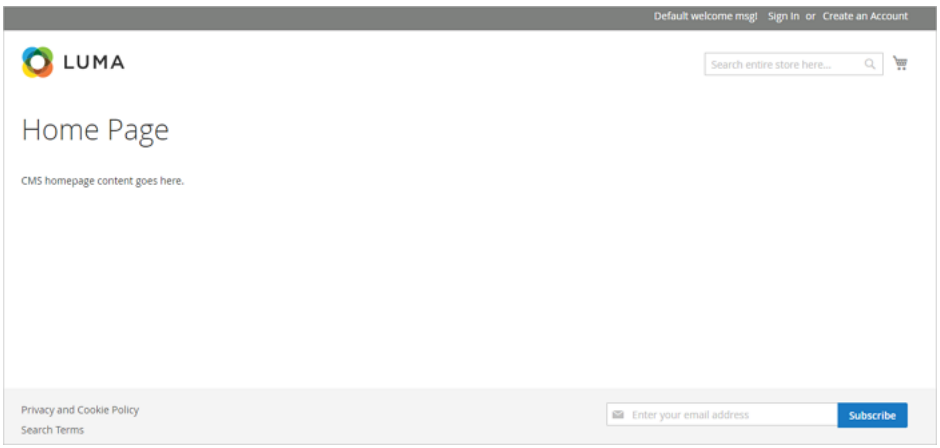
添加以下内容。

```
*/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
```

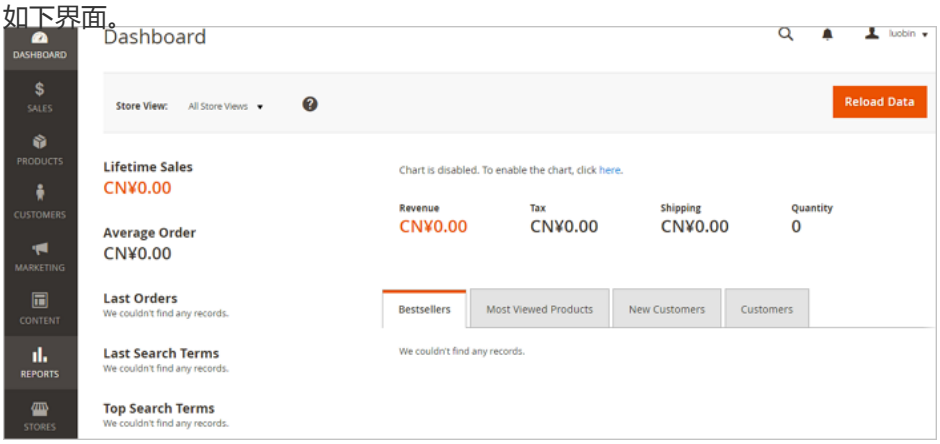
关于Magento上使用cron作业，请参见 Magento官方文档。

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。

部署Node.js项目（CentOS）

Node.js 是一个基于 Chrome V8 引擎的 JavaScript 运行环境，用来方便地搭建快速的易于扩展的网络应用。Node.js 使用了一个事件驱动、非阻塞式 I/O 的模型，使其轻量又高效，非常适合运行在分布式设备的数据密集型的实时应用。Node.js 的包管理器 npm，是全球最大的开源库生态系统。典型的应用场景包括：

- 实时应用：如在线聊天，实时通知推送等等（如socket.io）
- 分布式应用：通过高效的并行I/O使用已有的数据
- 工具类应用：海量的工具，小到前端压缩部署（如grunt），大到桌面图形界面应用程序
- 游戏类应用：游戏领域对实时和并发有很高的要求（如网易的pomelo框架）
- 利用稳定接口提升Web渲染能力
- 前后端编程语言环境统一：前端开发人员可以非常快速地切入到服务器端的开发（如著名的纯Javascript全栈式MEAN架构）

适用对象

本文档介绍如何在阿里云CentOS系统的云服务器ECS实例上，安装Nodejs并部署项目。

准备工作

部署之前，请做如下准备工作：

- 购买ECS实例
- 您的实例运行的镜像是CentOS7.2
- 您的实例可以连接公网
- 本地已经安装用于连接 Linux 实例的工具，如 PuTTY。

基本流程

使用云服务器ECS安装Nodejs并部署项目的操作步骤如下：

1. 购买ECS实例，并连接实例。
2. 选择以下任一种方法部署Node.js环境：
    - 使用二进制文件。
    - 使用NVM安装多版本。
3. 部署测试项目。

操作步骤

步骤 1：创建ECS实例

创建ECS实例。选择操作系统为公共镜像CentOS7.2。使用root用户登录Linux实例。

步骤2：部署Node.js环境

使用以下任一种方法部署Node.js环境。

使用二进制文件安装

该部署过程使用的安装包是已编译好的二进制文件，解压之后，在bin文件夹中就已存在node和npm，无需手工编译。

安装步骤：

wget命令下载Node.js安装包。该安装包是编译好的文件，解压之后，在bin文件夹中就已存在node和npm，无需重复编译。

```
wget https://nodejs.org/dist/v6.9.5/node-v6.9.5-linux-x64.tar.xz
```

解压文件。

```
tar xvf node-v6.9.5-linux-x64.tar.xz
```

创建软链接，使node和npm命令全局有效。通过创建软链接的方法，使得在任意目录下都可以直接使用node和npm命令：

```
ln -s /root/node-v6.9.5-linux-x64/bin/node /usr/local/bin/node
ln -s /root/node-v6.9.5-linux-x64/bin/npm /usr/local/bin/npm
```

查看node、npm版本。

```
node -v
npm -v
```

至此，Node.js环境已安装完毕。软件默认安装在/root/node-v6.9.5-linux-x64/目录下。如果需要将该软件安装到其他目录（如：/opt/node/）下，请进行如下操作：

```
mkdir -p /opt/node/
mv /root/node-v6.9.5-linux-x64/* /opt/node/
rm -f /usr/local/bin/node
rm -f /usr/local/bin/npm
ln -s /opt/node/bin/node /usr/local/bin/node
ln -s /opt/node/bin/npm /usr/local/bin/npm
```

使用NVM安装多版本

NVM（Node version manager）是Node.js的版本管理软件，使用户可以轻松在Node.js各个版本间进行切换。适用于长期做 node 开发的人员或有快速更新node版本、快速切换node版本这一需求的用户。

安装步骤：

直接使用git将源码克隆到本地的~/.nvm目录下，并检查最新版本。

```
yum install git
git clone https://github.com/cnpm/nvm.git ~/.nvm && cd ~/.nvm && git checkout `git describe --abbrev=0 --tags`
```

激活NVM。

```
echo ". ~/.nvm/nvm.sh" >> /etc/profile
source /etc/profile
```

列出Node.js的所有版本。

```
nvm list-remote
```

安装多个Node.js版本。

```
nvm install v6.9.5
nvm install v7.4.0
```

运行 nvm ls 查看已安装Node.js版本，当前使用的版本为v6.9.5。返回结果如下所示。

```
[root@iZXXXXZ .nvm]# nvm ls
v6.9.5
-> v7.4.0
system
stable -> 7.4 (-> v7.4.0) (default)
unstable -> 6.9 (-> v6.9.5) (default)
```

运行 nvm use v7.4.0 切换Node.js版本至v7.4.0。返回结果如下所示。

```
[root@iZXXXXZ .nvm]# nvm use v7.4.0
Now using node v7.4.0
```

NVM的更多操作请参考帮助文档：

```
nvm help
```

步骤3：部署测试项目

1. 新建项目文件example.js。

```
cd ~
touch example.js
```

使用vim编辑器打开项目文件example.js。

```
yum install vim
```

```
vim example.js
```

输入 i，进入编辑模式，将以下项目文件内容粘贴到文件中。使用Esc按钮，退出编辑模式，输入 :wq，回车，保存文件内容并退出。

项目文件内容：

```
const http = require('http');
const hostname = '0.0.0.0';
const port = 3000;
const server = http.createServer((req, res) => {
res.statusCode = 200;
res.setHeader('Content-Type', 'text/plain');
res.end('Hello World\n');
});
server.listen(port, hostname, () => {
console.log(`Server running at http://${hostname}:${port}/`);
});
```

> **说明**：
> 项目文件内容中的3000为端口号，可以自行定义。

运行项目。

```
node ~/example.js
```

> **说明**：
> 您也可以使用命令 node ~/example.js & 将项目置于后台运行。

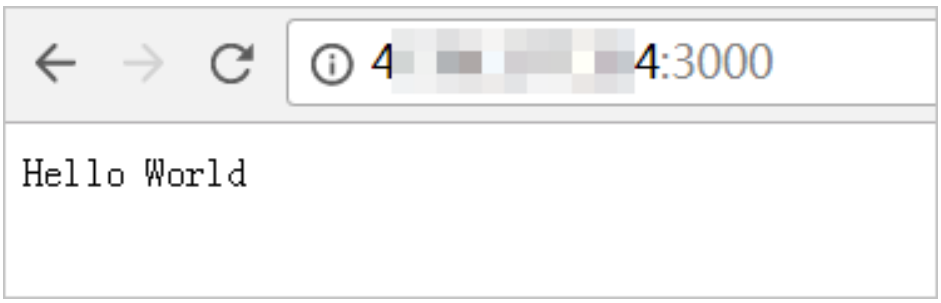
使用命令查看项目端口是否存在。

```
netstat -tpln
```

登录ECS管理控制台，并在安全组中 添加安全组规则 放行端口（如本示例中为TCP 3000端口）。

（可选）如果您的实例中开启了防火墙，必须添加端口的入站规则（如本示例中为TCP 3000端口）。

在本地机器的浏览器中输入 http://实例公网IP地址:端口号 访问项目。



相关链接

用户可通过云中沙箱平台体验上述文档中的操作，**点击此处。**

更多开源软件尽在云市场，**点击此处。**

Drupal建站教程（CentOS7）

Drupal是使用PHP语言编写的开源内容管理框架（CMF），它由内容管理系统（CMS）和PHP开发框架（Framework）共同构成。它用于构造提供多种功能和服务的动态网站，能支持从个人博客到大型社区等各种不同应用的网站项目。本文主要说明如何在阿里云ECS上搭建Drupal电子商务网站。

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。

基本流程

使用云服务器 ECS 搭建 Drupal 网站的操作步骤如下：

1. 选购ECS 实例
2. 构建Web运行环境
3. 安装Drupal

步骤 1：选购ECS实例

对于个人使用的小型网站，**选购一台云服务器ECS实例**可以满足需求，后续您可以根据实际使用情况考虑配置升级或者架构调优变更。

步骤2：构建web环境

在阿里云服务器上构建Web运行环境有3种方式

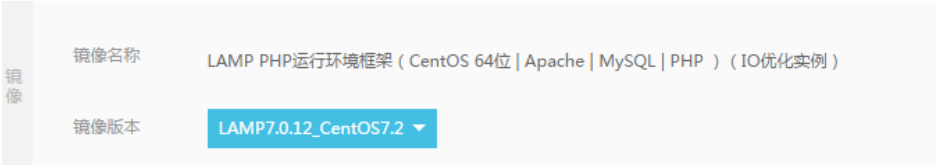
- 镜像部署
- 一键安装包部署
- 手动部署（源码编译安装/YUM安装）

一般推荐镜像部署，适合新手使用，更加快捷方便，一键安装包部署以及手动部署适合对运维知识有基本了解的用户，可以满足用户个性化部署的要求。本文档基于镜像部署的方式，搭建 Drupal 网站。

即在创建ECS实例时，镜像栏指定镜像市场。

单击从镜像市场选择，搜索框输入LAMP进行筛选，本文选择了匹配到的第一个镜像。

单击购买后，选择LAMP7.0.12_CentOS7.2版本。



更多镜像环境用户可在云市场基础环境中搜索筛选。

本文环境软件明细：CentOS 7.2 | Apache 2.4.25 | MySQL 5.7.17 | PHP 7.1.1 |Drupal8.1.1

说明：这是写文档时使用的软件版本，您下载的版本可能与此不同。

步骤 3：安装 Drupal

1、下载Drupal安装包。

```
# wget  http://ftp.drupal.org/files/projects/drupal-8.1.1.zip
```

2、解压到网站根目录。

```
# unzip drupal-8.1.1.zip
# mv drupal-8.1.1/* /var/www/html/
```

3、下载中文翻译包。

```
# cd /var/www/html/
# wget -P profiles/standard/translations http://ftp.drupal.org/files/translations/8.x/drupal/drupal-8.26.zh-hans.po
```

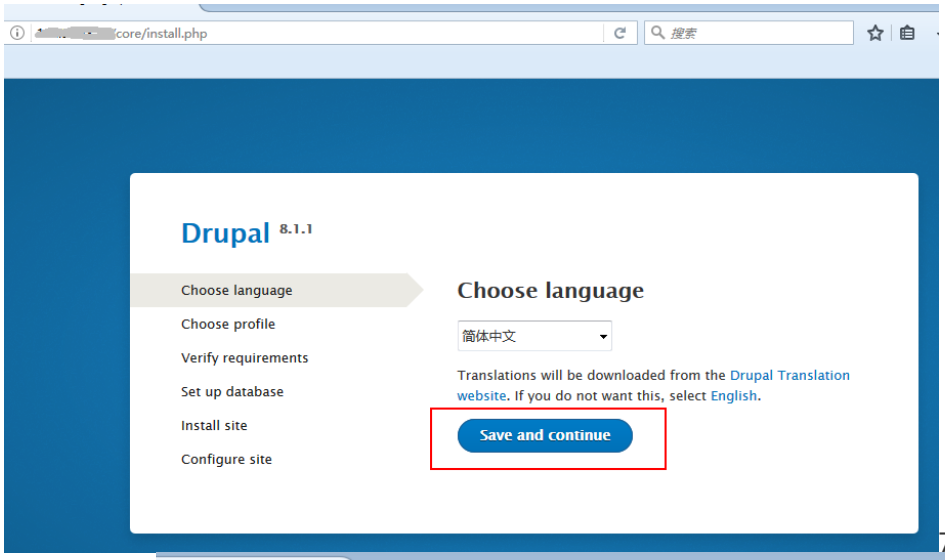
4、修改sites目录属主属组。

```
# chown -R apache:apache /var/www/html/sites
```

5、重启Apache服务。

```
# /etc/init.d/httpd restart
```

6、浏览器访问ECS服务器的公网IP/index.php ，进入到Drupal安装界面。选择安装语言，单击Save and



continue。7、选择标准安装

方式，单击保存并继续。8、填写数据库信息，单击保存并继续。

说明：用户登录mysql数据库后，可使用以下命令自定义用户名密码。

- DBNAME:数据库名称；
- UAERNAME：数据库用户；
- IP：本机可直接填localhost或者127.0.0.1
- YOURPASSWORD：数据库密码；

```
mysql> CREATE DATABASE DBNAME；
mysql> CREATE USER UAERNAME；
mysql> GRANT ALL PRIVILEGES ON *.* TO 'UAERNAME'@'IP' IDENTIFIED BY 'YOURPASSWORD' WITH GRANT OPTION;
mysql> FLUSH PRIVILEGES;
```

9、自动安装完成后进入网站设置界面，填写站点信息，单击保存并继续。

10、安装完成，后续可以根据需要对网站进行个性化设置。

镜像部署Windows环境

您可以根据业务需要，选择下列任意一种方式部署云服务器 ECS 实例的使用环境：

- **镜像部署**
- **手动部署**

下表列出了两种部署方式的特点。一般推荐镜像部署。如果您需要个性化定制部署，建议使用手动部署。

对比项	镜像部署	手动部署
部署所需时间	3-5分钟，快速部署上云	1-2天。选择适合的操作系统、中间件、数据库、各类软件、插

		件、脚本，再进行安装和配置
专业性 IOPS	由运维过万级用户的优质服务商提供	依赖开发人员的开发水平
个性化	支持主流应用场景	可满足个性化的部署要求
安全性	经过严格安全审核，集成最稳定安全的版本	依赖开发人员的开发水平
售后服务	专业售后工程师团队支持	依赖运维人员的经验，或由外包团队支持

注意： 本文档只介绍通用的操作步骤。一般镜像软件安装包都包含了操作指南，请阅读镜像操作指南进行具体的安装和配置。

阿里云的云市场提供了丰富的镜像资源。镜像集成了操作系统和应用程序。在创建实例时，您可以选择包含了应用环境的镜像，创建后无需再部署环境。

注意：云服务器 ECS 不支持虚拟化软件（如 KVM、Xen、VMware 等）的安装部署。

操作步骤

说明：本节介绍的方法适用于已经购买实例、但想使用镜像重新部署环境的用户。您也可以在创建实例的时候就选择镜像，请参考创建实例。

如果您想使用镜像市场的镜像来替换当前实例的操作系统，可以通过本节介绍的更换系统盘的方法来实现。

更换系统盘的时候，**数据盘**的数据则不会受到影响。因此建议您将系统盘的个人数据备份到数据盘中，或采用其他方式进行备份。

更换系统盘后，IP 地址不会改变。

如果您购买的实例已经开始运行，但是您想使用镜像市场中的镜像重新部署环境，操作步骤如下：

登录云服务器管理控制台

定位到需要重新部署环境的实例。

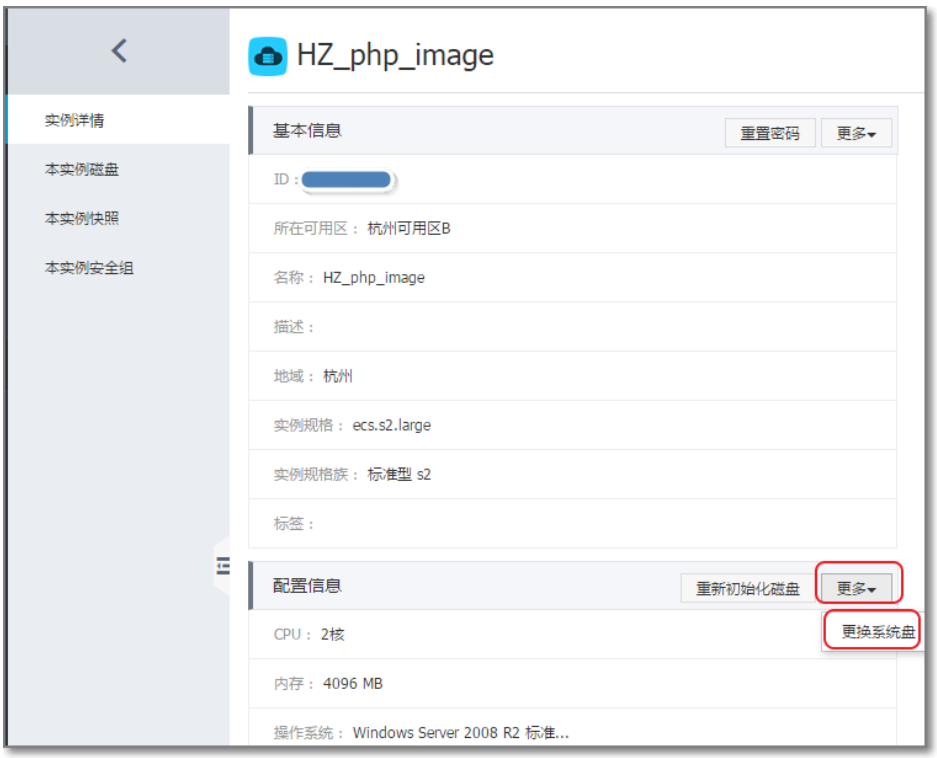
如果该实例刚刚创建，可以直接停止实例。如果实例已经运行了一段时间，您想保留其中的数据，请在操作前将数据备份到数据盘中。
注意：在更换镜像后，系统盘的数据会全部被清空，服务器的自动备份的快照可能会全部清空（取决于您的设置，请参见 自动快照随磁盘释放）。因此务必做好数据备份工作。

停止实例。

确认停止实例。

实例停止后，单击实例名称，或者右侧的 **管理**。

在左侧的 **配置信息** 中，单击 **更多** > **更换系统盘**。



请认真阅读提示信息。然后单击 **确定，更换系统盘。**

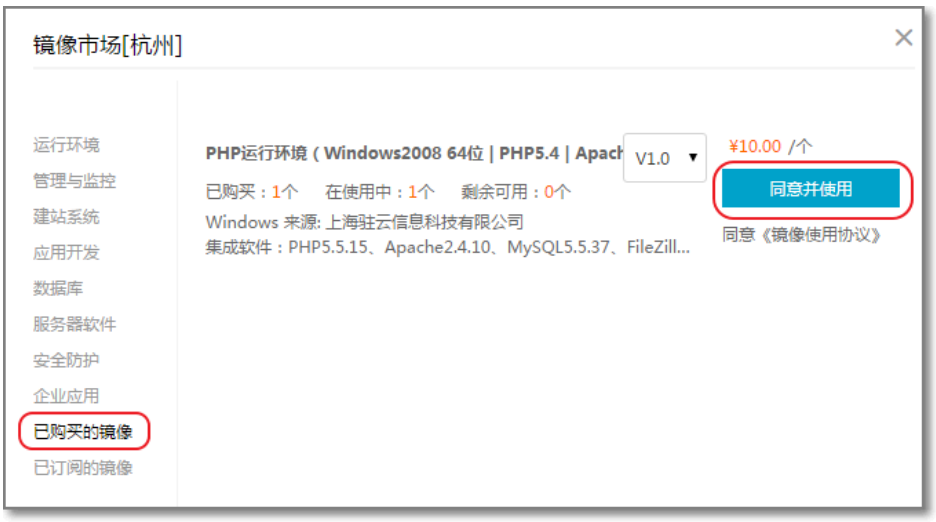
单击 **镜像市场**，然后单击 **从镜像市场选择（含操作系统）。**

镜像市场列表的左侧是镜像的分类。您可以根据分类，选择想使用的镜像。找到需要的镜像后，单击镜像右下方的 **同意并使用**。

注意在左侧最下方，有两个按钮：**已购买的镜像**和**已订阅的镜像**。如果您已经购买过镜像，可以直接单击**已购买的镜像**，选择镜像。

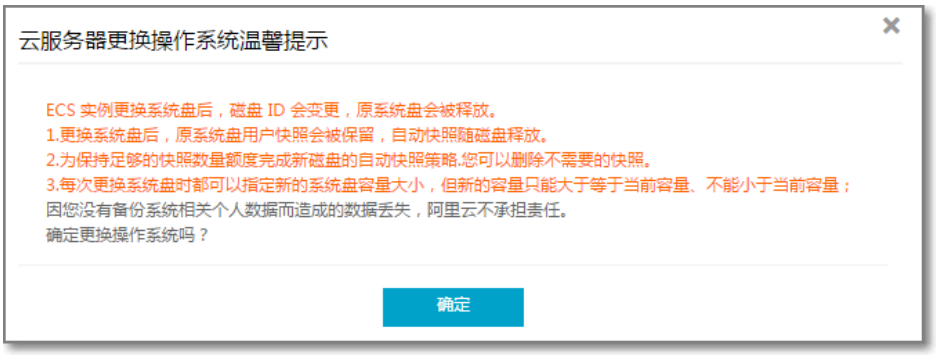
下图是选择已经购买的镜像的示例。单击 **同意并使用**。

注意：在此页面，不要单击镜像链接，否则会直接引导您到购买镜像页面，引起误解。



继续选择系统盘，输入登录密码，然后单击 **去支付**。

您会看到更换操作系统的提示。单击 **确定**。



您成功使用镜像部署了环境。现在可以启动、并登录实例，开始使用您的环境了。

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。

注意：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

在浏览器中打开阿里云的**云市场**。

搜索 **阿里云windows一键安装web环境**，然后购买该软件。

登录阿里云管理控制台。打开 **产品与服务** > **云市场**。

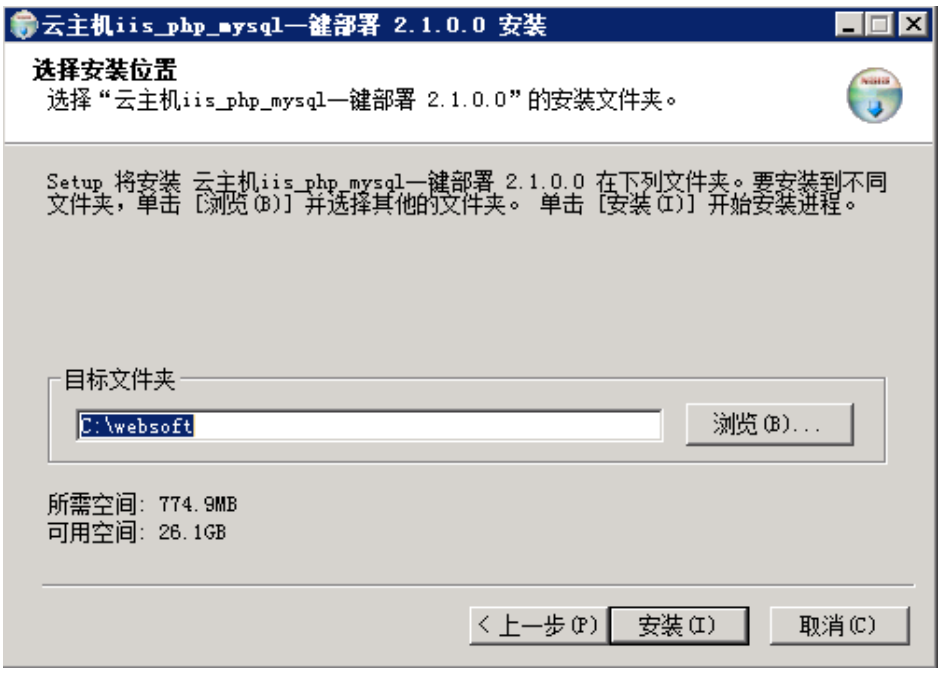
单击 **已购买的服务**。在 **阿里云Windows一键安装Web环境** 的右侧，单击 **下载**，一键下载安装包。

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。

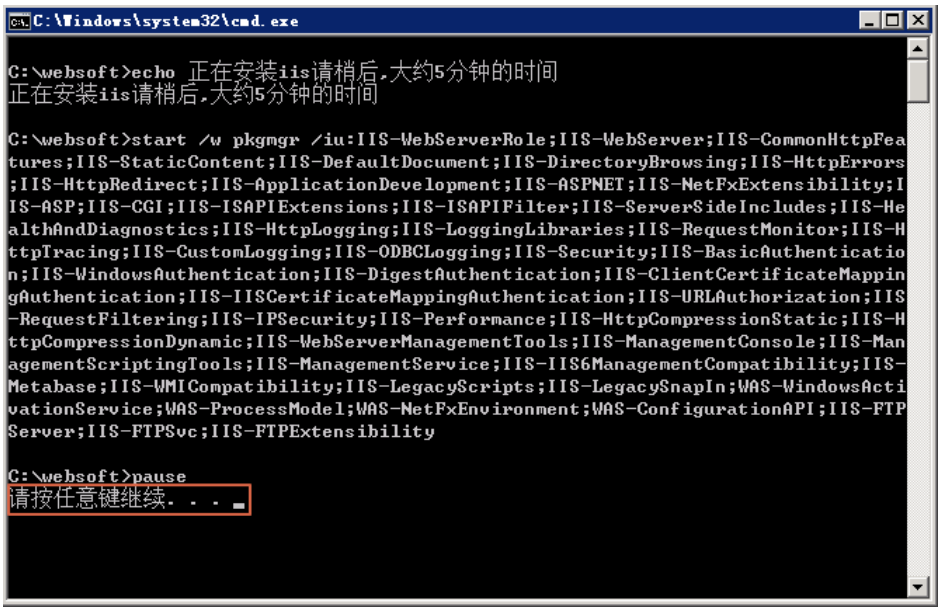
安装包启动后，单击 **下一步**。

指定安装目录，默认使用 **C:\websoft** 。然后单击 **安装**。



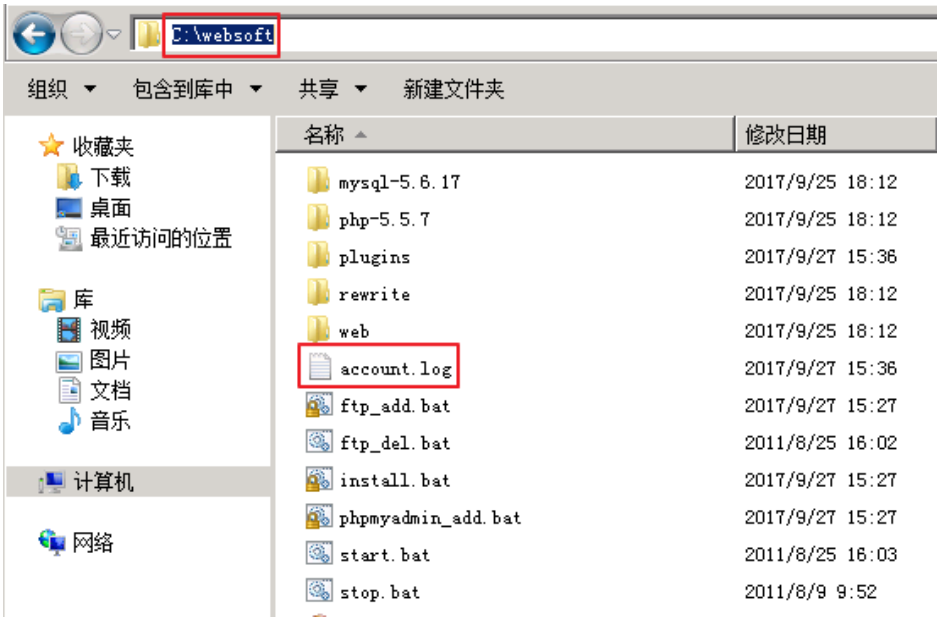
系统会依次自动安装 IIS 组件（不包括 FTP 组件）、PHP 环境、重定向 Rewrite、MySQL、phpwind。

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：



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。

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**。



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。

开始安装 FTP 服务。

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。

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

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。

安装全部完毕。按任意键关闭窗口。

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。若上述操作流程无法正常操作，可参考Windows一键Web安装包使用指南。

ECS上搭建Docker(CentOS7)

本文讲述 Docker 在 CentOS 系统上的部署过程。关于 Ubuntu 系统下安装 Docker，具体实践请参考docker实践文档。

适用对象

适用于熟悉Linux操作系统，刚开始使用阿里云ECS的开发者。

主要内容

- 部署docker
- docker基本用法
- 镜像制作

部署Docker

本文主要说明手动安装docker的操作步骤，您也可以选择在云市场购买相应镜像，一键部署云服务器。

本文实践操作系统版本为CentOS 7.2 64 3.10.0-514.6.2.el7.x86_64。

> Docker要求64位的系统且内核版本至少为3.10

添加yum源。

```
# yum install epel-release –y
# yum clean all
# yum list
```

安装并运行Docker。

```
# yum install docker-io –y
# systemctl start docker
```

检查安装结果。

```
 # docker info
```

出现以下xin说明信息则表明安装成功。

```
Security Options: seccomp
Kernel Version: 3.10.0-514.6.2.el7.x86_64
Operating System: CentOS Linux 7 (Core)
OSType: linux
Architecture: x86_64
Number of Docker Hooks: 2
CPUs: 1
Total Memory: 991.2 MiB
Name: iZuf60ssd5lunjdllnf9d8Z
ID: KJJ5:AJ2R:LR36:62MP:3L3P:2OYK:IMNR:4WIZ:K4WE:SES7:WBXV:7MOW
Docker Root Dir: /var/lib/docker
Debug Mode (client): false
Debug Mode (server): false
Registry: https://index.docker.io/v1/
Insecure Registries:
 127.0.0.0/8
Registries: docker.io (secure)
```

Docker基本用法

Docker守护进程管理。

```
# systemctl start docker     #运行Docker守护进程
# systemctl stop docker #停止Docker守护进程
# systemctl restart docker #重启Docker守护进程
```

镜像管理。本文使用的是来自阿里云仓库的Apache镜像。

```
```

```
# docker pull registry.cn-hangzhou.aliyuncs.com/lxepoo/apache-php5
```

修改标签，由于阿里云仓库镜像的镜像名称很长，可以修改镜像标签以便记忆区分。

```
# docker tag  registry.cn-hangzhou.aliyuncs.com/lxepoo/apache-php5:latest aliweb:v1
```

查看已有镜像。

```
# docker images
```

强制删除镜像。

```
# docker  rmi  –f  registry.cn-hangzhou.aliyuncs.com/lxepoo/apache-php5
```

容器管理。

e121d5f99e1e是执行docker images命令查询到的IMAGE ID，使用docker run命令进入容器。

```
# docker run –ti e121d5f99e1e /bin/bash
```

使用exit可以退出当前容器。

run命令加上–d参数可以在后台运行容器，—name指定容器命名为apache。

```
# docker run -d --name apache e121d5f99e1e
```

进入后台运行的容器。

```
# docker exec -ti apache /bin/bash
```

将容器做成镜像。

```
# docker commit containerID/containerName  newImageName:tag
```

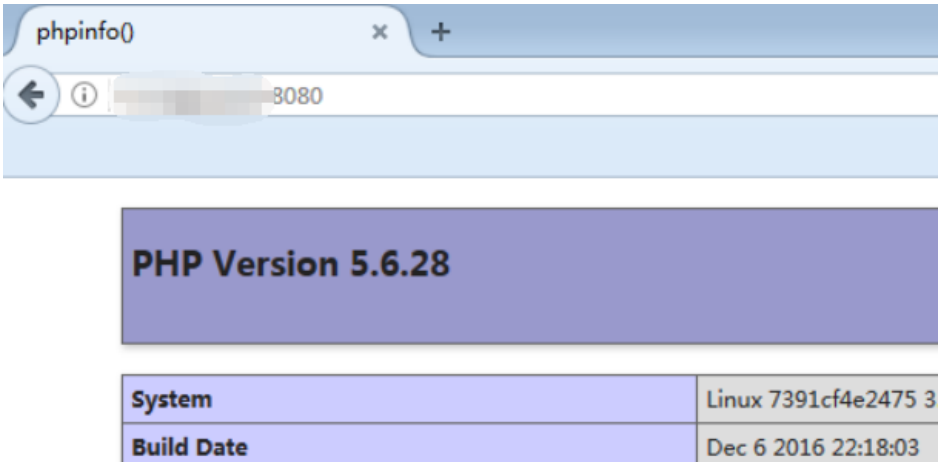
为了方便测试和恢复，先将源镜像运行起来后再做一个命名简单的镜像做测试。

```
# docker commit 4c8066cd8c01 apachephp:v1
```

运行容器并将宿主机的8080端口映射到容器里去。

```
# docker run -d -p 8080:80 apachephp:v1
```

在浏览器输入宿主机ip加8080端口访问测试，出现以下内容则说明运行成功。



镜像制作

准备dockerfile内容。

```
# vim Dockerfile
FROM apachephp:v1 #声明基础镜像来源
MAINTAINER DTSTACK #声明镜像拥有者
RUN mkdir /dtstact #RUN后面接容器运行前需要执行的命令，由于Dockerfile文件不能超过127行，因此当命令较多时建议写到脚本中执行
ENTRYPOINT ping www.aliyun.com #开机启动命令，此处最后一个命令需要是可在前台持续执行的命令，否则容器后台运行时会因为命令执行完而退出。
```

构建镜像。

```
docker build -t webcentos:v1 .  # . 是Dockerfile文件的路径，不能忽略
docker images #查看是否创建成功
docker run –d webcentos:v1 #后台运行容器
docker ps #查看当前运行中的容器
docker ps –a #查看所有容器，包括未运行中的
docker logs CONTAINER ID/IMAGE #如未查看到刚才运行的容器，则用容器id或者名字查看启动日志排错
docker commit fb2844b6c070 dtstackweb:v1 #commit 后接容器id 和构建新镜像的名称和版本号。
docker images #列出本地（已下载的和本地创建的）镜像
docker push #将镜像推送至远程仓库，默认为 Docker Hub
```

将镜像推送到registry。

```
docker login --username=dtstack_plus registry.cn-shanghai.aliyuncs.com #执行后输入镜像仓库密码
docker tag [ImageId] registry.cn-shanghai.aliyuncs.com/dtstack123/test:[镜像版本号]
docker push registry.cn-shanghai.aliyuncs.com/dtstack123/test:[镜像版本号]
```

在镜像仓库能查看到镜像版本信息则说明push成功。

其中[ImageId]，[镜像版本号]请您根据自己的镜像信息进行填写。

部署Linux主机管理系统WDCP

WDlinux Control Panel（简称 wdCP），是一套通过 Web 控制和管理服务器的 Linux 服务器管理系统以及虚拟主机管理系统。在 wdCP 的后台里，您可以更方便地使用 Linux 系统作为我们的网站服务器系统，并对 Linux 服务器进行管理。阿里云的云市场提供了丰富的镜像资源。镜像集成了操作系统和应用程序。创建实例时，您可以选择包含了应用环境的镜像，创建后无需再部署环境。

注意：云服务器 ECS 不支持虚拟化软件（如 KVM、Xen、VMware 等）的安装部署。

根据业务需要，您可以选择下列任意一种方式部署 ECS 实例：

镜像部署

手动部署

通常情况下，推荐您使用镜像部署。如果您需要个性化定制部署，建议使用手动部署。下表列出了两种部署方式的特点。

对比项	镜像部署	手动部署
部署所需时间	3-5 分钟快速部署上云。	1-2 天，选择适合的操作系统、中间件、数据库、各类软件、插件、脚本，再进行安装和配置。
专业性	由运维过万级用户的云市场优质服务商提供。	依赖开发人员的开发水平。
个性化	支持主流应用场景。	可满足个性化部署需求。
安全性	经过严格审核，集成最稳定安全的版本。	依赖开发人员水平。
售后服务	专业售后工程师团队支持。	依赖运维人员经验，或由外包团队支持。

本文档只介绍通用的操作步骤。一般镜像软件安装包都包含了操作指南，请阅读镜像操作指南进行具体的安装和配置。阿里云市场提供了种类丰富的镜像软件，点击查看。

操作步骤

本节介绍的方法适用于已经购买实例，但想使用镜像重新部署环境的用户。此外，您也可以在创建实例的时候就选择镜像，请参考创建实例。

如果您想使用镜像市场的镜像来替换当前实例的操作系统，可以通过本节介绍的 **更换系统盘的方法** 来实现。

更换系统盘的时候，数据盘的数据则不会受到影响。建议将系统盘的个人数据备份到数据盘中，或采用其他方式进行备份。更换系统盘后，IP 地址不会改变。

已购买的实例部署镜像

如果您购买的实例已经开始运行，但是您想使用镜像市场中的镜像重新部署环境，操作步骤如下：

登录云服务器管理控制台。

找到需要重新部署环境的实例。

若实例已经运行了一段时间，您想保留其中的数据，请提前创建快照，将数据备份到数据盘中。若该实例刚刚创建，可以直接停止实例。

注意：更换镜像后，系统盘的数据会全部被清空，服务器自动备份的快照也可能会被删除（取决于您的设置，请参见自动快照随磁盘释放）。请务必提前做好数据备份工作，提前创建快照。

在弹出的 **停止实例** 框中，选择 **停止**，单击 **确认**。

实例停止后，单击实例名称，或者在实例的右侧选择 **更多** > **更换系统盘**。

在弹出的 **更换系统盘** 框中，单击 **确定，更换系统盘**。

单击 **镜像市场**，然后单击 **从镜像市场选择（含操作系统）**。

在镜像市场列表的左侧，根据分类选择待使用的镜像，并在镜像的右下方单击 **同意并使用**。

在弹出的 **云服务器更换操作系统温馨提示** 框中，单击 **确定**。

未购买的实例部署镜像

如果您还未购买ECS实例，可以在创建实例的同时直接部署镜像。

登录云服务器管理控制台。

进入 **创建实例** 页面，根据您的需求选择 **计费方式**、**地域**、**网络**、**实例配置**、**带宽**、**存储**、**购买量**，并在镜像处选择 **镜像市场**。

根据关键词的搜索结果，单击 **购买**。

使用镜像部署了 WDCP 主机管理系统后，您可以启动并登录实例，开始使用您的环境。使用方法请参考服务商提供的使用手册。

说明：详情请参考镜像链接 https://market.aliyun.com/products/53398003/cmjj015205.html 。

部署RabbitMQ

RabbitMQ 是一个开源的 AMQP 实现，服务器端用 Erlang 语言编写，支持多种客户端，如：Python、Ruby、.NET、Java、JMS、C、PHP、ActionScript、XMPP、STOMP 等，支持 AJAX。用于在分布式系统中存储转发消息，在易用性、扩展性、高可用性等方面表现不俗。

部署方式

在阿里云服务器下部署 RabbitMQ 提供两种部署方式：

- RabbitMQ 镜像部署
- 手动部署（源码编译安装）

一般推荐镜像部署适合新手使用更加快捷方便，安装包部署以及手动部署适合对 Linux 命令有基本了解的用户，可以满足用户个性化部署的要求。本教程主要介绍镜像和手工部署的方式。

镜像部署

单击 RabbitMQ 环境（CentOS7.3 Erlang19.3）进入镜像详情页。

单击 **立即购买**，按提示步骤购买 ECS 实例。

登录 ECS 管理控制台。

在左边导航栏里，单击 **实例**，进入 ECS 实例列表页。

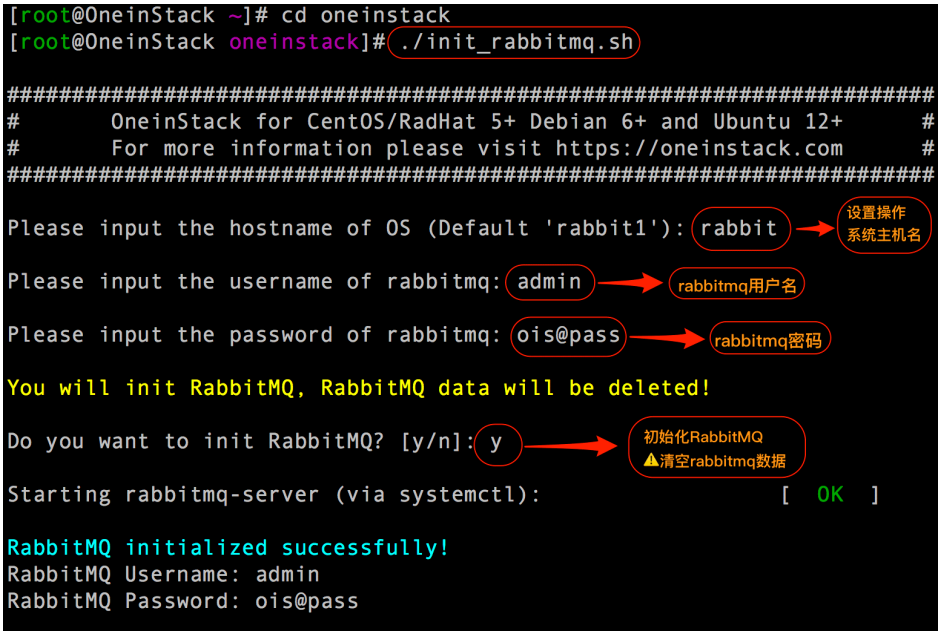
选择所购 ECS 实例所在的地域，并找到所购 ECS 实例，在 **IP 地址** 列获取该实例的公网 IP 地址。

在浏览器地址栏中输入公网 IP 地址，下载操作文档。

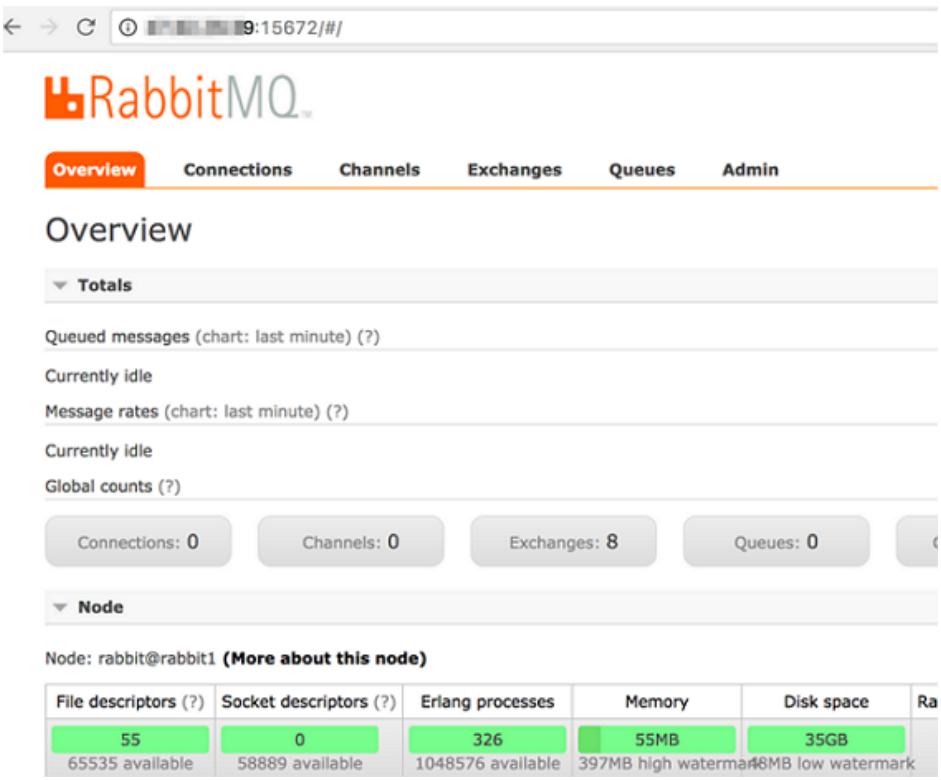
使用 putty 登录 Linux 服务器，请参考连接Linux实例；忘记 root 密码，请参考重置实例密码。

初始化 rabbitmq。

```
cd /root/oneinstack
./init_rabbitmq.sh
```



进入管理页面，浏览器访问 http://公网IP:15672 。



手工部署

- 系统平台：CentOS 7.3
- rabbitmq版本：rabbitmq-server -3.6.9
- erlang版本：erlang19.3
- JDK版本：JDK1.8.0_121

前提准备

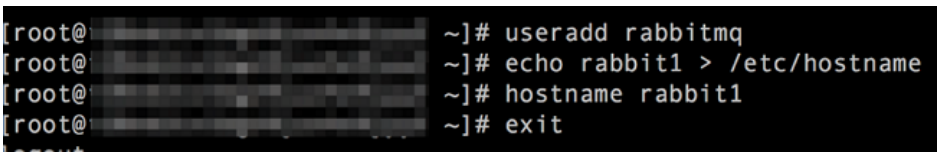
创建一般用户 rabbitmq，运行 rabbitmq。

```
useradd rabbitmq
```

设置 Linux 主机名。

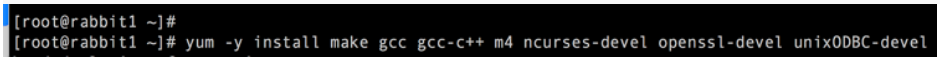
CentOS 7 修改 /etc/hostname，CentOS 6 修改 /etc/sysconfig/network，下面以 CentOS 7 为例：

```
echo rabbit1 > /etc/hostname
hostname rabbit1
exit #退出重新登录
```



安装依赖包

```
yum -y install make gcc gcc-c++ m4 ncurses-devel openssl-devel unixODBC-devel
```



源代码下载

- 下载 Erlang：wget http://erlang.org/download/otp_src_19.3.tar.gz。
- 下载 rabbitmq：wget https://www.rabbitmq.com/releases/rabbitmq-server/v3.6.9/rabbitmq-server-generic-unix-3.6.9.tar.xz

安装 Erlang

```
tar xzf otp_src_19.3.tar.gz #解压
cd otp_src_19.3
./configure --prefix=/usr/local/erlang --enable-shared-zlib --with-ssl --enable-threads --enable-smp-support --enable-kernel-poll --enable-hipe --without-javac
make && make install
```

解压 RabbitMQ

解压 rabbitmq-server-generic-unix-3.6.9.tar.xz 。

```
tar xvJf rabbitmq-server-generic-unix-3.6.9.tar.xz
mv rabbitmq_server-3.6.9 /usr/local/rabbitmq
```

rabbitmq 环境变量配置。

```
sed -i 's@^ERL_DIR=.*@ERL_DIR=/usr/local/erlang/bin/@' /usr/local/rabbitmq/sbin/rabbitmq-defaults
sed -i 's@^LOG_BASE=.*@LOG_BASE=/usr/local/rabbitmq/var/log/rabbitmq@' /usr/local/rabbitmq/sbin/rabbitmq-defaults
mkdir -p /usr/local/rabbitmq/var/{lib,log}/rabbitmq
```

一般用户（rabbitmq）运行 RabbitMQ。

```
wget http://pkgs.fedoraproject.org/cgit/rpms/rabbitmq-server.git/plain/rabbitmq-script-wrapper
sed -i 's@cd /var/lib/rabbitmq@cd /usr/local/rabbitmq/var/lib/rabbitmq@g' rabbitmq-script-wrapper #更改rabbitmq数据存储目录
sed -i 's@/usr/lib/rabbitmq/bin/@/usr/local/rabbitmq/sbin/@g' rabbitmq-script-wrapper
chmod +x rabbitmq-script-wrapper
cp rabbitmq-script-wrapper /usr/sbin/rabbitmqctl
cp rabbitmq-script-wrapper /usr/sbin/rabbitmq-server
cp rabbitmq-script-wrapper /usr/sbin/rabbitmq-plugins
chown -R rabbitmq.rabbitmq /usr/local/rabbitmq/var
```

rabbitmq 日志割接。

```
cat >> /etc/logrotate.d/rabbitmq-server << EOF
/usr/local/rabbitmq/var/log/rabbitmq/*.log {
weekly
missingok
rotate 20
compress
delaycompress
notifempty
sharedscripts
postrotate
/sbin/service rabbitmq-server rotate-logs > /dev/null
endscript
}
EOF
```

rabbitmq 启动脚本。

```
vi /etc/init.d/rabbitmq-server
#!/bin/sh
#
# rabbitmq-server RabbitMQ broker
#
# chkconfig: - 80 05
# description: Enable AMQP service provided by RabbitMQ
#
### BEGIN INIT INFO
# Provides: rabbitmq-server
# Required-Start: $remote_fs $network
# Required-Stop: $remote_fs $network
# Description: RabbitMQ broker
# Short-Description: Enable AMQP service provided by RabbitMQ broker
### END INIT INFO
# Source function library.
. /etc/init.d/functions
PATH=/sbin:/usr/sbin:/bin:/usr/bin:/usr/local/erlang/bin
NAME=rabbitmq-server
DAEMON=/usr/sbin/${NAME}
CONTROL=/usr/sbin/rabbitmqctl
DESC=rabbitmq-server
USER=rabbitmq
ROTATE_SUFFIX=
```

```
INIT_LOG_DIR=/usr/local/rabbitmq/var/log/rabbitmq
PID_FILE=/var/run/rabbitmq/pid
START_PROG="daemon"
LOCK_FILE=/var/lock/subsys/$NAME
test -x $DAEMON || exit 0
test -x $CONTROL || exit 0
RETVAL=0
set -e
[ -f /etc/default/${NAME} ] && . /etc/default/${NAME}
[ -f /etc/sysconfig/${NAME} ] && . /etc/sysconfig/${NAME}
ensure_pid_dir () {
PID_DIR=`dirname ${PID_FILE}`
if [ ! -d ${PID_DIR} ] ; then
mkdir -p ${PID_DIR}
chown -R ${USER}:${USER} ${PID_DIR}
chmod 755 ${PID_DIR}
fi
}
remove_pid () {
rm -f ${PID_FILE}
rmdir `dirname ${PID_FILE}` || :
}
start_rabbitmq () {
status_rabbitmq quiet
if [ $RETVAL = 0 ] ; then
echo RabbitMQ is currently running
else
RETVAL=0
# RABBIT_NOFILES_LIMIT from /etc/sysconfig/rabbitmq-server is not handled
# automatically
if [ "$RABBITMQ_NOFILES_LIMIT" ]; then
ulimit -n $RABBITMQ_NOFILES_LIMIT
fi
ensure_pid_dir
set +e
RABBITMQ_PID_FILE=$PID_FILE $START_PROG $DAEMON \
> "${INIT_LOG_DIR}/startup_log" \
2> "${INIT_LOG_DIR}/startup_err" \
0<&- &
$CONTROL wait $PID_FILE >/dev/null 2>&1
RETVAL=$?
set -e
case "$RETVAL" in
0)
echo SUCCESS
if [ -n "$LOCK_FILE" ] ; then
touch $LOCK_FILE
fi
;;
*)
remove_pid
echo FAILED - check ${INIT_LOG_DIR}/startup_\{log, _err\}
RETVAL=1
;;
esac
fi
```

```
}
stop_rabbitmq () {
status_rabbitmq quiet
if [ $RETVAL = 0 ] ; then
set +e
$CONTROL stop ${PID_FILE} > ${INIT_LOG_DIR}/shutdown_log 2> ${INIT_LOG_DIR}/shutdown_err
RETVAL=$?
set -e
if [ $RETVAL = 0 ] ; then
remove_pid
if [ -n "$LOCK_FILE" ] ; then
rm -f $LOCK_FILE
fi
else
echo FAILED - check ${INIT_LOG_DIR}/shutdown_log, _err
fi
else
echo RabbitMQ is not running
RETVAL=0
fi
}
status_rabbitmq() {
set +e
if [ "$1" != "quiet" ] ; then
$CONTROL status 2>&1
else
$CONTROL status > /dev/null 2>&1
fi
if [ $? != 0 ] ; then
RETVAL=3
fi
set -e
}
rotate_logs_rabbitmq() {
set +e
$CONTROL rotate_logs ${ROTATE_SUFFIX}
if [ $? != 0 ] ; then
RETVAL=1
fi
set -e
}
restart_running_rabbitmq () {
status_rabbitmq quiet
if [ $RETVAL = 0 ] ; then
restart_rabbitmq
else
echo RabbitMQ is not runnning
RETVAL=0
fi
}
restart_rabbitmq() {
stop_rabbitmq
start_rabbitmq
}
case "$1" in
start)
```

```
  echo -n "Starting $DESC: "
  start_rabbitmq
  echo "$NAME."
  ;;
  stop)
  echo -n "Stopping $DESC: "
  stop_rabbitmq
  echo "$NAME."
  ;;
  status)
  status_rabbitmq
  ;;
  rotate-logs)
  echo -n "Rotating log files for $DESC: "
  rotate_logs_rabbitmq
  ;;
  force-reload|reload|restart)
  echo -n "Restarting $DESC: "
  restart_rabbitmq
  echo "$NAME."
  ;;
  try-restart)
  echo -n "Restarting $DESC: "
  restart_running_rabbitmq
  echo "$NAME."
  ;;
  *)
  echo "Usage: $0 {start|stop|status|rotate-logs|restart|condrestart|try-restart|reload|force-reload}" >&2
  RETVAL=1
  ;;
  esac
  exit $RETVAL
```

保存后，添加执行权限，并设置自启动。

```
chmod +x /etc/init.d/rabbitmq-server
chkconfig --add rabbitmq-server
chkconfig rabbitmq-server on
```

修改 rabbitmq.config 。

特别注意默认用户名密码，请自行修改 default_user，default_pass，loopback_users。

```
cat > /usr/local/rabbitmq/etc/rabbitmq/rabbitmq.config << EOF
[
{rabbit, [
{tcp_listeners,[{"0.0.0.0",5672}]},
{tcp_listen_options, [binary, {packet,raw},
{reuseaddr,true},
{backlog,128},
{nodelay,true},
{exit_on_close,false},
```

```
{keepalive,true}]},
{default_vhost, <<"/">>},
{default_user, <<"guest">>},
{default_pass, <<"guest">>},
{loopback_users, ["guest"]},
{default_permissions, [<<".*">>, <<".*">>, <<".*">>]}
]}
].
EOF
```

开启 rabbitmq manager。

```
cat > /usr/local/rabbitmq/etc/rabbitmq/enabled_plugins << EOF
[rabbitmq_management].
EOF
```

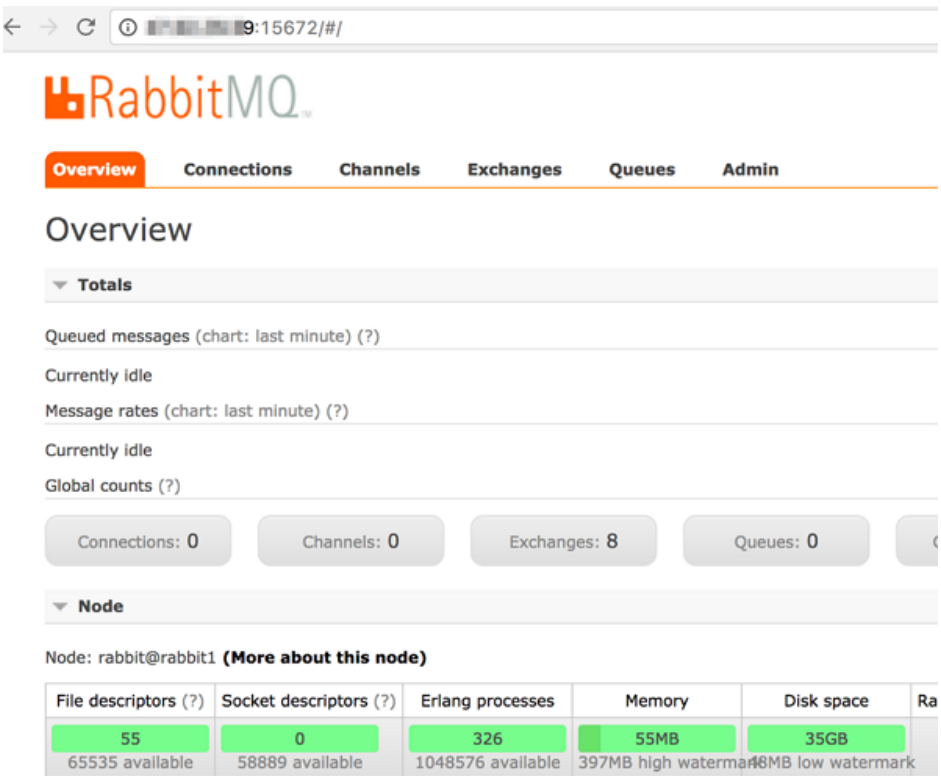
启动 rabbitmq 。

```
service rabbitmq-server start
```

```
[root@rabbit1 ~]# service rabbitmq-server start
Starting rabbitmq-server (via systemctl):                  [  OK  ]
[root@rabbit1 ~]# ps -ef | grep rabbitmq
rabbitmq 18655     1  0 22:26 ?        00:00:00 /usr/local/erlang/lib/erlang/erts-8.3/bin/epmd -daemon
root     19522     1  0 22:27 ?        00:00:00 /bin/sh /etc/rc.d/init.d/rabbitmq-server start
root     19524 19522  0 22:27 ?        00:00:00 /bin/bash -c ulimit -S -c 0 >/dev/null 2>&1 ; /usr/sbin/rabbitmq-ser
root     19526 19524  0 22:27 ?        00:00:00 /bin/sh /usr/sbin/rabbitmq-server
root     19544 19526  0 22:27 ?        00:00:00 su rabbitmq -s /bin/sh -c /usr/local/rabbitmq/sbin/rabbitmq-server
rabbitmq 19547 19544  0 22:27 ?        00:00:00 /bin/sh -e /usr/local/rabbitmq/sbin/rabbitmq-server
rabbitmq 19749 19547 24 22:27 ?        00:00:01 /usr/local/erlang/lib/erlang/erts-8.3/bin/beam -W w -A 64 -P 1048576
stbt db -zdbbl 32000 -K true -B i -- -root /usr/local/erlang/lib/erlang -progname erl -- -home /home/rabbitmq -- -pa
abbitmq/ebin -noshell -noinput -s rabbit boot -sname rabbit@rabbit1 -boot start_sasl -config /usr/local/rabbitmq/etc
bitmq -kernel inet_default_connect_options [{nodelay,true}] -sasl errlog_type error -sasl sasl_error_logger false -r
ogger {file,"/usr/local/rabbitmq/var/log/rabbitmq/rabbit@rabbit1.log"} -rabbit sasl_error_logger {file,"/usr/local/r
og/rabbitmq/rabbit@rabbit1-sasl.log"} -rabbit enabled_plugins_file "/usr/local/rabbitmq/etc/rabbitmq/enabled_plugins
gins_dir "/usr/local/rabbitmq/plugins" -rabbit plugins_expand_dir "/usr/local/rabbitmq/var/lib/rabbitmq/mnesia/rabbi
gins-expand" -os_mon start_cpu_sup false -os_mon start_disksup false -os_mon start_memsup false -mnesia dir "/usr/lo
var/lib/rabbitmq/mnesia/rabbit@rabbit1" -kernel inet_dist_listen_min 25672 -kernel inet_dist_listen_max 25672
rabbitmq 19861 19749  0 22:27 ?        00:00:00 erl_child_setup 65535
rabbitmq 19869 19861  0 22:27 ?        00:00:00 inet_gethost 4
rabbitmq 19870 19869  0 22:27 ?        00:00:00 inet_gethost 4
```

进入管理页面。

浏览器访问 http://公网IP:15672。



更多开源软件尽在云市场：https://market.aliyun.com/software 。

GitLab的安装及使用

前言

GitLab是利用 Ruby on Rails 一个开源的版本管理系统，实现一个自托管的Git项目仓库，可通过Web界面进行访问公开的或者私人项目。

它拥有与Github类似的功能，能够浏览源代码，管理缺陷和注释。可以管理团队对仓库的访问，它非常易于浏览提交过的版本并提供一个文件历史库。

团队成员可以利用内置的简单聊天程序(Wall)进行交流。

它还提供一个代码片段收集功能可以轻松实现代码复用，便于日后有需要的时候进行查找。

Git的家族成员

Git：是一种版本控制系统，是一个命令，是一种工具。

Gitlib：是用于实现Git功能的开发库。

Github：是一个基于Git实现的在线代码托管仓库，包含一个网站界面，向互联网开放。

GitLab：是一个基于Git实现的在线代码仓库托管软件，你可以用gitlab自己搭建一个类似于Github一样的系统，一般用于在企业、学校等内部网络搭建git私服。

Gitlab的服务构成

Nginx：静态web服务器。

gitlab-shell：用于处理Git命令和修改authorized keys列表。

gitlab-workhorse:轻量级的反向代理服务器。

logrotate：日志文件管理工具。

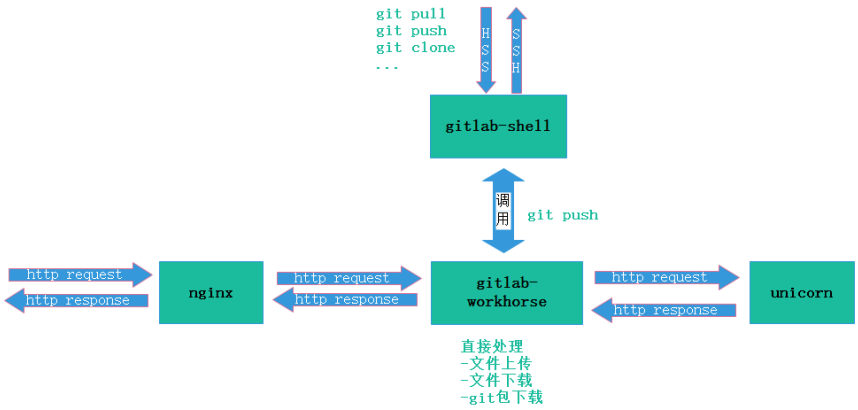
postgresql：数据库。

redis：缓存数据库。

sidekiq：用于在后台执行队列任务（异步执行）。

unicorn：An HTTP server for Rack applications，GitLab Rails应用是托管在这个服务器上面的。

GitLab工作流程



GitLab Shell

GitLab Shell有两个作用：为GitLab处理Git命令、修改authorized keys列表。

当通过SSH访问GitLab Server时，GitLab Shell会：

限制执行预定义好的Git命令（git push, git pull, git annex）

调用GitLab Rails API 检查权限

执行pre-receive钩子（在GitLab企业版中叫做Git钩子）

执行你请求的动作 处理GitLab的post-receive动作

处理自定义的post-receive动作

当通过http(s)访问GitLab Server时，工作流程取决于你是从Git仓库拉取(pull)代码还是向git仓库推送(push)代码。

如果你是从Git仓库拉取(pull)代码，GitLab Rails应用会全权负责处理用户鉴权和执行Git命令的工作；

如果你是向Git仓库推送(push)代码，GitLab Rails应用既不会进行用户鉴权也不会执行Git命令，它会把以下工作交由GitLab Shell进行处理：

```
调用GitLab Rails API 检查权限
执行pre-receive钩子（在GitLab企业版中叫做Git钩子）
执行你请求的动作
处理GitLab的post-receive动作
处理自定义的post-receive动作
```

GitLab Workhorse

GitLab Workhorse是一个敏捷的反向代理。它会处理一些大的HTTP请求，比如文件上传、文件下载、Git push/pull和Git包下载。其它请求会反向代理到GitLab Rails应用，即反向代理给后端的unicorn。

Gitlab环境部署

ECS配置要求：内存2G以上

方法一：镜像部署

镜像名称：GitLab代码管理（Centos 64位 | GitLab）| 镜像帮助文档

进入镜像详情页面，单击立即购买，按提示步骤购买 ECS 实例。

购买完成之后，登录”ECS 管理控制台”,在左边导航栏里，单击”实例”，进入 ECS 实例列表页,选择所购 ECS 实例所在的地域，并找到所购 ECS 实例，在”IP 地址”列获取该实例的公网 IP 地址。

注意：镜像部署好后默认是禁止远端访问的，所以直接访问ECS服务器的公网IP是不能访问到GitLab的登录界面的，请先运行/alidata目录下的gitlab_opennet.sh脚本，开启远程访问，然后再通过浏览器访问公网IP来访问GitLab的主页。

方法二：手动部署：

1、配置yum源

```
vim /etc/yum.repos.d/gitlab-ce.repo
```

复制以下内容：

```
[gitlab-ce]
name=gitlab-ce
baseurl=http://mirrors.tuna.tsinghua.edu.cn/gitlab-ce/yum/el6
Repo_gpgcheck=0
Enabled=1
Gpgkey=https://packages.gitlab.com/gpg.key
```

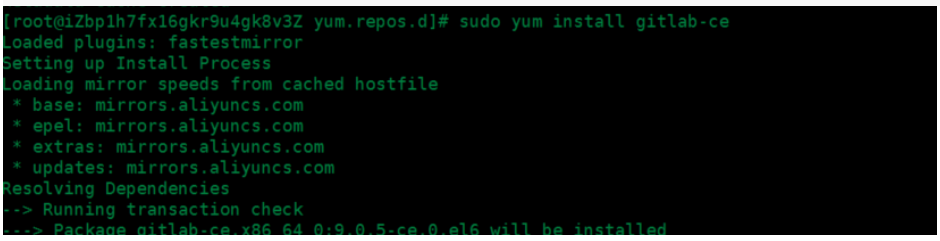
2、更新本地yum缓存

```
sudo yum makecache
```

3、安装GitLab社区版

```
sudo yum install gitlab-ce       #自动安装最新版

sudo yum install gitlab-ce-x.x.x #安装指定版本
```



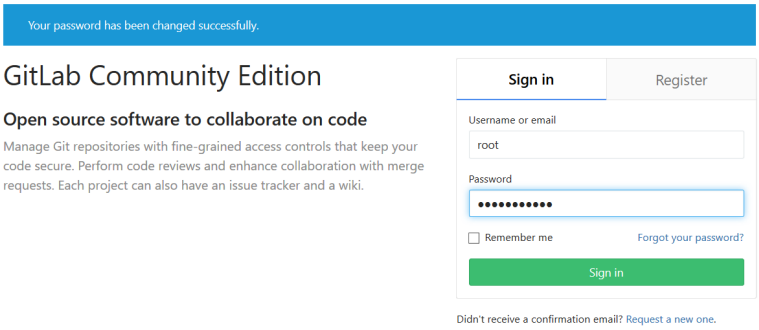
GitLab常用命令：

```
sudo gitlab-ctl start    # 启动所有 gitlab 组件；
sudo gitlab-ctl stop # 停止所有 gitlab 组件；
sudo gitlab-ctl restart # 重启所有 gitlab 组件；
sudo gitlab-ctl status # 查看服务状态；
sudo gitlab-ctl reconfigure # 启动服务；
sudo vim /etc/gitlab/gitlab.rb # 修改默认的配置文件；
gitlab-rake gitlab:check SANITIZE=true --trace # 检查gitlab；
sudo gitlab-ctl tail # 查看日志；
```

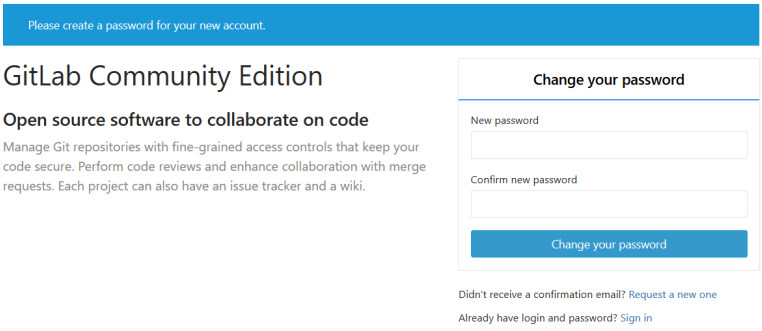
GitLab使用

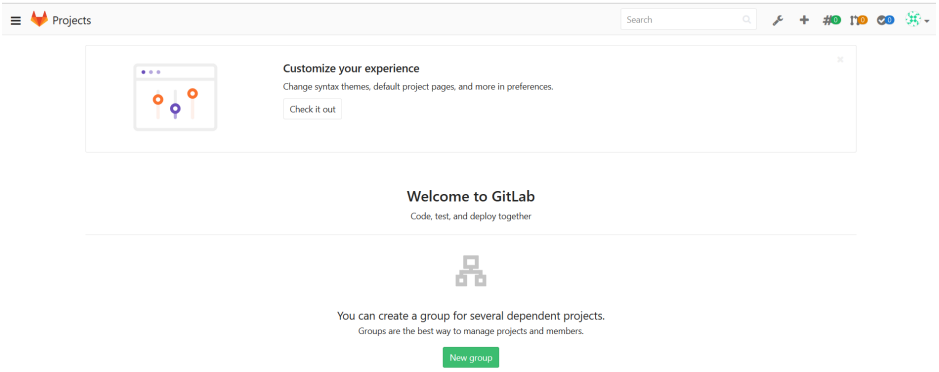
登录GitLab

1、在浏览器的地址栏中输入ECS服务器的公网IP即可登录GitLab的界面，第一次登录使用的用户名和密码为root 和 5iveL!fe。



2、首次登录会强制用户修改密码。密码修改成功后，输入新密码进行登录。





创建Project

1、安装Git工具linux：安装Git，使用自带的源安装。

```
yum install git
```

```
[root@iZbp1h7fx16gkr9u4gk8v3Z ~]# yum install git
```

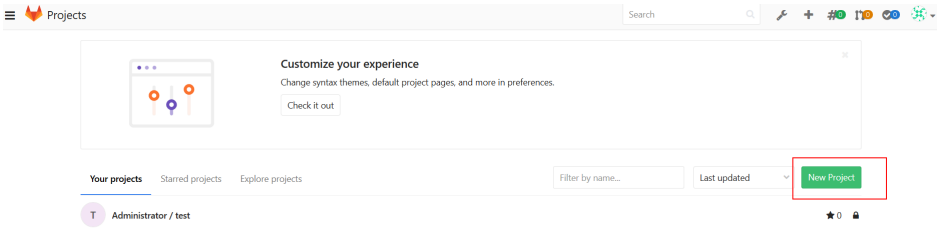
2、生成密钥文件

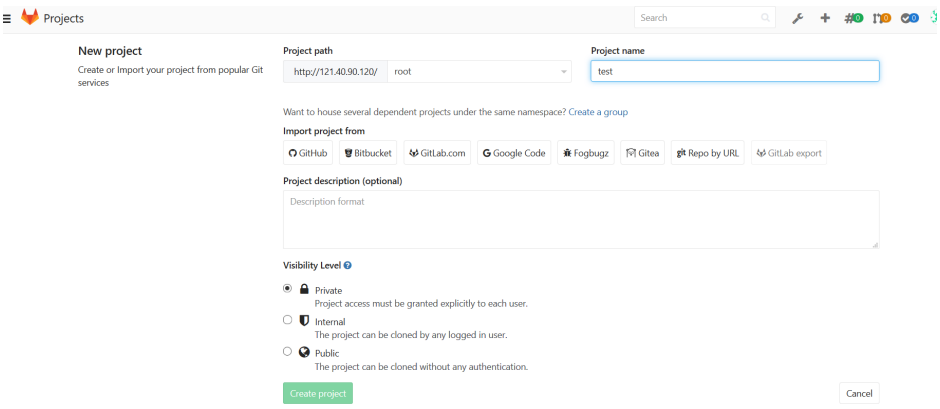
使用ssh-keygen生成密钥文件.ssh/id_rsa.pub。

```
[root@iZbp1h7fx16gkr9u4gk8v3Z ~]# ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa):
Created directory '/root/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /root/.ssh/id_rsa.
Your public key has been saved in /root/.ssh/id_rsa.pub.
The key fingerprint is:
34:ac:e7:de:34:0d:7b:b6:69:72:1d:00:3e:50:04:4c root@iZbp1h7fx16gkr9u4gk8v3Z
The key's randomart image is:
+--[ RSA 2048]----+
|       oE+o      |
|       .o .      |
|        +o .     |
|       o .o .    |
|      . S .. .   |
|       o   +  .  |
|        . + +. . |
|       . o.+oo.  |
|        . .+o    |
+-----------------+
[root@iZbp1h7fx16gkr9u4gk8v3Z ~]#
```

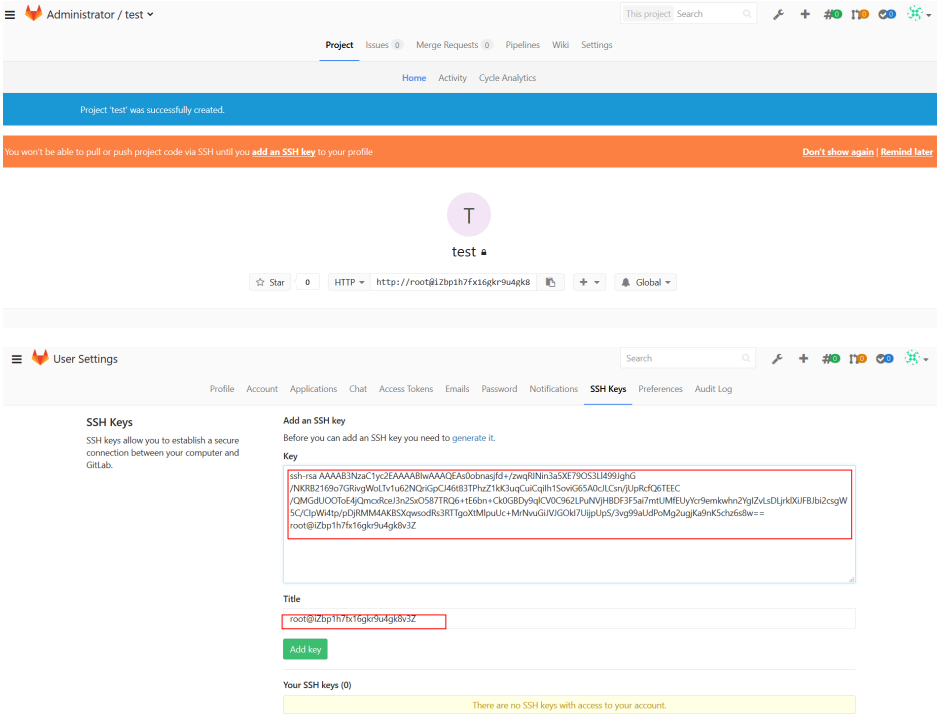
```
[root@iZbp1h7fx16gkr9u4gk8v3Z ~]# cat .ssh/id_rsa.pub
ssh-rsa AAAAB3NzaC1yc2EAAAABIwAAAQEAs0obnasjfd+/zwqRINin3a5XE79OS3Ll499JghG/NKRB2169o7GRivg
WoLTv1u62NQriGpCJ46t83TPhzZ1kK3uqCuiCqiIh1SoviG65A0cJLCsn/jUpRcfQ6TEEC/QMGdUOOToE4jQmcxRceJ
3n2SxO587TRQ6+tE6bn+Ck0GBDy9qlCV0C962LPuNVjHBDF3F5ai7mtUMfEUyYcr9emkwhn2YgIZvLsDLjrklXiJFBJ
bi2csgW5C/CIpWi4tp/pDjRMM4AKBSXqwsodRs3RTTgoXtMlpuUc+MrNvuGiJVJGOkI7UijpUpS/3vg99aUdPoMg2ug
jKa9nK5chz6s8w== root@iZbp1h7fx16gkr9u4gk8v3Z
[root@iZbp1h7fx16gkr9u4gk8v3Z ~]#
```

3.在GitLab的主页中新建一个Project

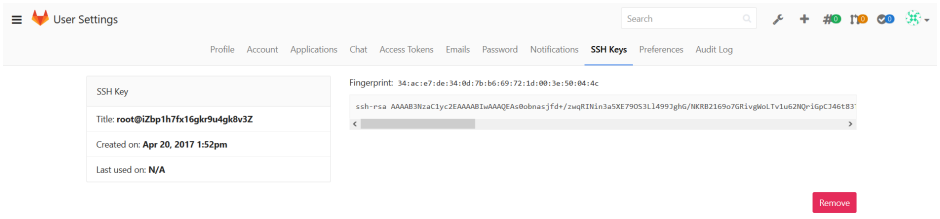




4.添加ssh key导入步骤2中生成的密钥文件内容：



ssh key添加完成：



项目地址，该地址在进行clone操作时需要用到:

简单配置

1、配置使用Git仓库的人员姓名

```
git config --global user.name "上海驻云"
```

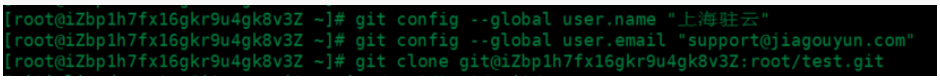
2、配置使用Git仓库的人员email，填写自己的公司邮箱

```
```

```
git config --global user.email "support@jiagouyun.com"
```

3、克隆项目，在本地生成同名目录，并且目录中会有所有的项目文件

```
git clone git@iZbp1h7fx16gkr9u4gk8v3Z:root/test.git
```



上传文件

1、进入到项目目录

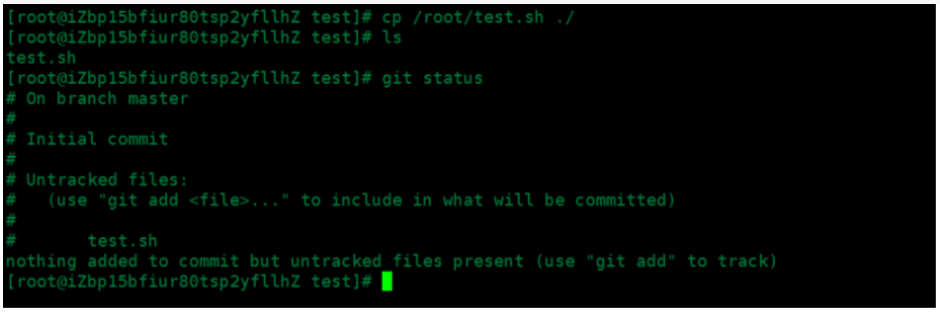
```
cd test/
```

2、创建需要上传到GitLab中的目标文件

```
echo “test” > /root/test.sh
```

3、将目标文件或者目录拷贝到项目目录下

```
cp /root/test.sh ./
```



4、将test.sh文件加入到索引中

```
git add test.sh
```

5、将test.sh提交到本地仓库

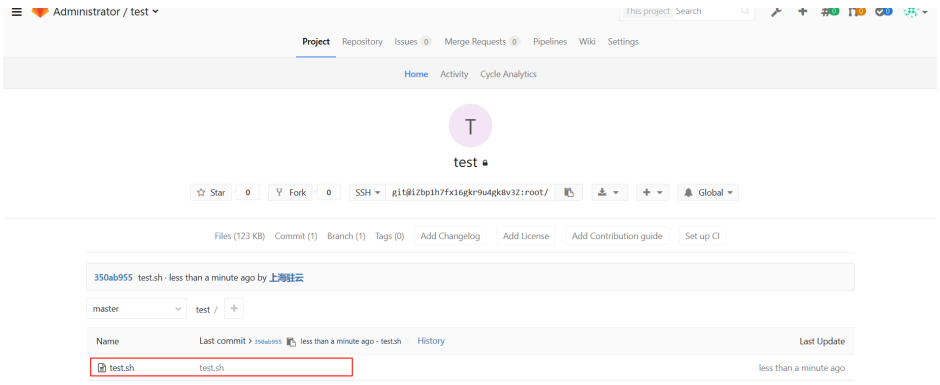
```
git commit -m “test.sh”
```

6、将文件同步到GitLab服务器上

```
git push -u origin master
```

```
[root@iZbp1h7fx16gkr9u4gk8v3Z test]# git commit -m "test.sh"
[master (root-commit) 350ab95] test.sh
 1 files changed, 1 insertions(+), 0 deletions(-)
 create mode 100644 test.sh
[root@iZbp1h7fx16gkr9u4gk8v3Z test]# git push -u origin master
Counting objects: 3, done.
Writing objects: 100% (3/3), 228 bytes, done.
Total 3 (delta 0), reused 0 (delta 0)
To git@iZbp1h7fx16gkr9u4gk8v3Z:root/test.git
 * [new branch]      master -> master
Branch master set up to track remote branch master from origin.
[root@iZbp1h7fx16gkr9u4gk8v3Z test]#
```

7、在网页中查看上传的test.sh文件已经同步到GitLab中



相关链接

更多开源软件尽在云市场：https://market.aliyun.com/software

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：

	SSH	HTTP	MySQL
	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。

运行命令关闭防火墙。

```
systemctl stop firewalld.service
```

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

运行命令安装相关依赖包。

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

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
```

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
```

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
```

依次运行以下命令编译安装Apache。

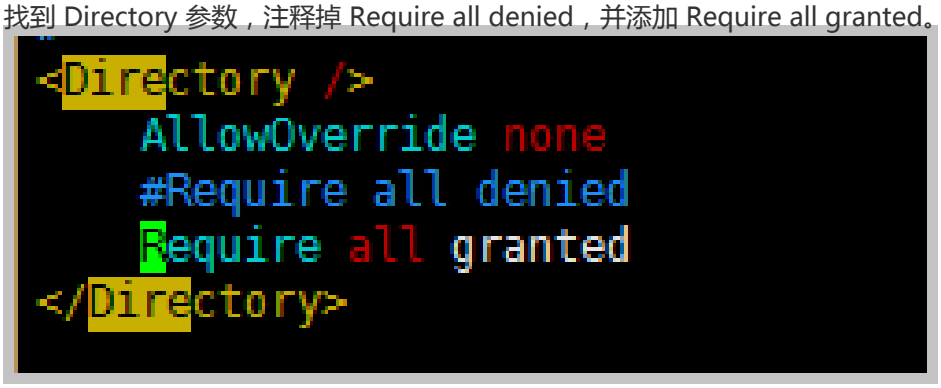
```
cd /usr/local/src/
wget http://zy-res.oss-cn-hangzhou.aliyuncs.com/apache/httpd-2.4.23.tar.gz
tar zxvf httpd-2.4.23.tar.gz
cd httpd-2.4.23
./configure \
--prefix=/usr/local/apache --sysconfdir=/etc/httpd \
--enable-so --enable-cgi --enable-rewrite \
--with-zlib --with-pcre=/usr/local/pcre \
--with-apr=/usr/local/apr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

修改httpd.conf配置文件参数：

i. 运行 cd /etc/httpd/ 切换到/etc/httpd/目录。

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

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



iv. 找到 ServerName 参数，添加 ServerName localhost:80。

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

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

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。

设置开机自启动。

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

ii. 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
~
```

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

设置环境变量。

i. 运行 vi /root/.bash_profile 打开文件，按 i 进入编辑模式。
ii. 将 PATH=$PATH:$HOME/bin 修改为
PATH=$PATH:$HOME/bin:/usr/local/apache/bin。
iii. 按 Esc 键退出编辑模式，输入 :wq 保存并关闭文件。
iv. 运行 source /root/.bash_profile 重新执行文件。

步骤2. 编译安装MySQL

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，则运行以下任一个命令删除。

```
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 #卸载不成功时使用此命令强制卸载
```

卸载后，再用 rpm -qa|grep mariadb 或者 rpm -qa|grep mysql 查看结果。

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/
```

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
```

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
```

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

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。

设置环境变量。

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

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。

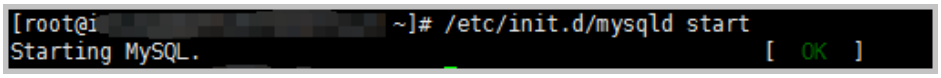
按 Esc 键退出编辑模式，输入 :wq 保存并关闭文件。

运行 source /root/.bash_profile 重新执行文件。

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。



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
```

测试登录MySQL数据库。

```
mysql -uroot -p                    #-p和密码之间无空格
```

```
[root@i      ~]# mysql -uroot -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 20
Server version: 5.7.17 MySQL Community Server (GPL)

Copyright (c) 2000, 2016, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql>
```

运行 \q 退出MySQL。

步骤3. 编译安装 PHP

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
```

运行命令复制配置文件。

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

编辑Apache配置文件 httpd.conf，以Apache支持PHP。

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

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。

iv. 按 Esc 键退出编辑模式，输入 :wq 保存并关闭文件。

重启Apache服务：

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

测试是否能够正常解析PHP。

i. 依次运行以下命令，找开index.php文件。

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

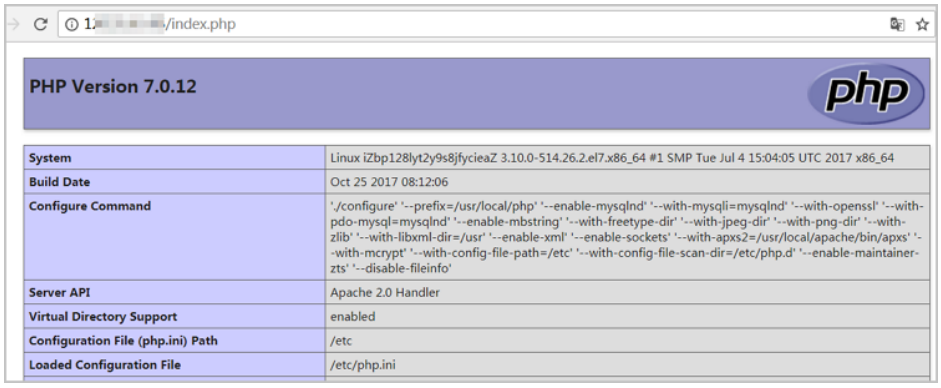
ii. 按 i 键进入编辑模式，并添加以下内容。

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

iii. 按 Esc 键退出编辑模式，并输入 :wq 保存并关闭文件。
iv. 重启Apache服务：

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

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**。

PostgreSQL 本地Slave搭建步骤

PostgreSQL 被业界誉为**最先进的开源数据库**，目前阿里云数据库 PostgreSQL 版具有 NoSQL 兼容，高效查询，插件化管理，安全稳定的特性。本文档介绍使用阿里云 ECS 搭建 PostgreSQL 主从架构的操作步骤。

适用对象

适用于熟悉 ECS，熟悉 Linux 系统，熟悉 PostgreSQL 的阿里云用户。

前提条件

已在安全组中添加规则放行 5432 端口。

基本流程

使用阿里云 ECS 搭建 PostgreSQL 主从架构的操作步骤如下：

1. 选购 ECS 实例。
2. 主节点安装配置。
3. 从节点安装配置。
4. 检测验证。

步骤 1：选购 ECS 实例

搭建主从复制架构，需要选购2台专有网络类型的云服务器ECS实例，建议不分配公网IP，可按需购买弹性公网IP绑定至对应ECS实例，进行配置操作。后续使用您可以根据实际情况考虑配置升级或者架构调优变更。

步骤 2：安装 PostgreSQL

在阿里云服务器上安装 PostgreSQL 有 2 种方式：

- 镜像部署
- 手动部署（源码编译安装/YUM安装）

本文档基于yum部署的方式，安装postgresql；您也可以在云市场基础环境中搜索筛选，使用镜像部署，更加快捷方便。

本文环境软件明细：CentOS 7.2 |PostgreSQL (9.5.6)

步骤 3：PostgreSQL 主节点配置

主节点上执行以下命令安装 PostgreSQL。

```
# yum update -y
# yum install https://download.postgresql.org/pub/repos/yum/9.5/redhat/rhel-7-x86_64/pgdg-centos95-9.5-2.noarch.rpm -y
# yum install postgresql95-server postgresql95-contrib -y
# /usr/pgsql-9.5/bin/postgresql95-setup initdb
# systemctl enable postgresql-9.5.service
# systemctl start postgresql-9.5.service
```

主节点上创建进行主从复制的数据库账号，并设置密码及登录和备份权限。

```
# su - postgres
# psql
postgres=# CREATE ROLE replica login replication encrypted password 'replica';
CREATE ROLE
postgres=# SELECT usename from pg_user ;
usename
----------
postgres
replica
(2 rows)

postgres=# SELECT rolname from pg_roles ;
rolname
----------
postgres
replica
(2 rows)
```

修改pg_hba.conf，设置replica用户白名单。

```
# vim /var/lib/pgsql/9.5/data/pg_hba.conf
```

在IPv4 local connections段添加下面两行内容

host all all 192.168.1.0/24 md5 允许VPC网段中md5密码认证连接 host replication replica 192.168.1.0/24 md5 允许用户从replication数据库进行数据同步

修改postgresql.conf。

```
# vim /var/lib/pgsql/9.5/data/postgresql.conf
```

设置以下参数

- wal_level = hot_standby 启用热备模式
- synchronous_commit = on 开启同步复制
- max_wal_senders = 32 同步最大的进程数量
- wal_sender_timeout = 60s 流复制主机发送数据的超时时间
- max_connections = 100 最大连接数，从库的max_connections必须要大于主库的

重启服务

```
# systemctl restart postgresql-9.5.service
```

步骤 4：PostgreSQL从节点配置

安装postgres。

```
 # yum update -y
# yum install https://download.postgresql.org/pub/repos/yum/9.5/redhat/rhel-7- x86_64/pgdg-centos95-9.5-2.noarch.rpm -y
# yum install postgresql95-server postgresql95-contrib -y
```

使用pg_basebackup基础备份的工具制定备份目录。

```
 # pg_basebackup -D /var/lib/pgsql/9.5/data -h 主节点IP -p 5432 -U replica -X stream -P
Password:
30075/30075 kB (100%), 1/1 tablespace
```

添加并修改recovery.conf。

```
  # cp /usr/pgsql-9.5/share/recovery.conf.sample /var/lib/pgsql/9.5/data/recovery.conf
 # vim /var/lib/pgsql/9.5/data/recovery.conf
```

设置以下参数。

standby_mode = on # 声明此节点为从库

primary_conninfo = ‘host=主节点IP port=5432 user=replica password=replica’ # 对应主库的连接信息

recovery_target_timeline = ‘latest’ # 流复制同步到最新的数据

修改postgresql.conf。

```
  # vim /var/lib/pgsql/9.5/data/postgresql.conf
```

设置以下参数。

```
- max_connections = 1000            # 最大连接数，从节点需设置比主节点大
- hot_standby = on # 开启热备
- max_standby_streaming_delay = 30s # 数据流备份的最大延迟时间
- wal_receiver_status_interval = 1s # 从节点向主节点报告自身状态的最长间隔时间
- hot_standby_feedback = on # 如果有错误的数据复制向主进行反馈
```

修改数据目录属组属主。

```
# chown -R postgres.postgres /var/lib/pgsql/9.5/data
```

启动服务，设置开机自启。

```
 # systemctl start postgresql-9.5.service
# systemctl enable postgresql-9.5.service
```

步骤 5：检测验证

主节点中可查看到sender进程。

```
 # ps aux |grep sender
postgres 2916 0.0 0.3 340388 3220 ? Ss 15:38 0:00 postgres: wal sender process replica
192.168.1.222(49640) streaming 0/F01C1A8
```

从节点中可查看到receiver进程。

```
# ps aux |grep receiver
postgres 23284 0.0 0.3 387100 3444 ? Ss 16:04 0:00 postgres: wal receiver process streaming 0/F01C1A8
```

主库中可查看到从库状态。

```
 replication=# select * from pg_stat_replication;
pid | usesysid | usename | application_name | client_addr | client_hostname | client_port | backend_start |
backend_xmin | state | sent_location | write_locati
on | flush_location | replay_location | sync_priority | sync_state
------+----------+---------+------------------+---------------+-----------------+------------- +---------------
-----------------+--------------+-----------+---------------+-------------
---+----------------+-----------------+---------------+------------
2916 | 16393 | replica | walreceiver | 192.168.1.222 | | 49640 | 2017-05-02 15:38:06.188988+08 | 1836 |
streaming | 0/F01C0C8 | 0/F01C0C8
| 0/F01C0C8 | 0/F01C0C8 | 0 | async
(1 rows)
```

搭建和使用SVN

Subversion(SVN) 是一个开源的版本控制系统, 也就是说 Subversion 管理着随时间改变的数据。

这些数据放置在一个中央资料档案库(repository) 中。这个档案库很像一个普通的文件服务器, 不过它会记住每一次文件的变动。这样您就可以把档案恢复到旧的版本, 或是浏览文件的变动历史。

SVN 的一些概念

- repository（源代码库）：源代码统一存放的地方
- Checkout（提取）：当您手上没有源代码时，您需要从repository checkout一份源代码
- Commit（提交）：如果您已经修改了代码，您需要Commit到repository
- Update（更新）：当您已经Checkout了一份源代码，Update一下，您就可以与Repository上的源代码同步，您手上的代码就会有最新的变更

日常开发过程其实就是这样的（假设您已经Checkout并且已经工作了几天）：Update（获得最新的代码）—> 作出自己的修改并调试成功 —> Commit（大家就可以看到您的修改了）。

如果您与同事同时修改了同一个文件，SVN可以合并你们的改动，实际上SVN管理源代码是以行为单位的，就是说你们只要不是修改了同一行程序，SVN都会自动合并两种修改。如果是同一行，SVN会提示文件Confict（冲突），需要手动确认。

安装SVN

您可以采用以下任一种方法安装SVN。

使用SVN版本控制镜像

您可以在云市场购买使用 SVN版本控制镜像 的ECS实例。

创建了实例后，按以下步骤操作：

登录 ECS管理控制台。

在左侧导航栏里，单击 **实例**。

选择地域。

找到新创建的ECS实例，在 **IP地址** 列获取实例的公网IP地址。

手动安装SVN

本文以CentOS 7.2 64位系统为例，说明如何在CentOS 7.2上安装SVN。

远程连接Linux实例。

运行以下命令安装SVN。

```
yum install subversion
```

运行以下命令查看SVN版本。

```
svnserve --version
```

```
[root@iZb                Z conf]# svnserve --version
svnserve, version 1.7.14 (r1542130)
   compiled Nov 20 2015, 19:25:09

Copyright (C) 2013 The Apache Software Foundation.
This software consists of contributions made by many people; see the NOTICE
file for more information.
Subversion is open source software, see http://subversion.apache.org/

The following repository back-end (FS) modules are available:

* fs_base : Module for working with a Berkeley DB repository.
* fs_fs : Module for working with a plain file (FSFS) repository.

Cyrus SASL authentication is available.
```

按以下步骤创建版本库：

i. 运行以下命令创建目录。

```
mkdir /var/svn
```

ii. 依次运行以下命令创建版本库。

```
cd /var/svn
svnadmin create /var/svn/svnrepos
```

iii. 依次运行以下命令查看自动生成的版本库文件。

```
cd svnrepos
ls
```

```
[root@iZb                beZ svnrepos]# ls
conf  db  format  hooks  locks  README.txt
```

Subversion目录说明：

i. db目录：所有版本控制的数据存放文件。
ii. hooks目录：放置hook脚本文件的目录。

iii. locks目录：用来追踪存取文件库的客户端。
iv. format文件：是一个文本文件，里面只放了一个整数，表示当前文件库配置的版本号。
v. conf目录：是这个仓库的配置文件（仓库的用户访问账号、权限等）。

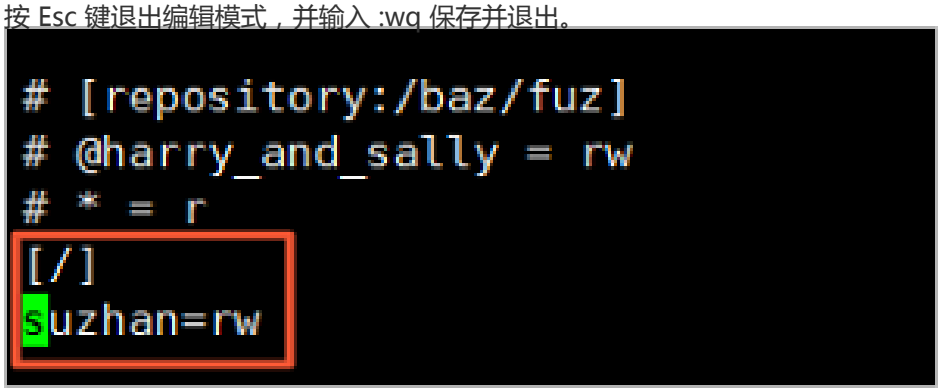
iv. 运行命令cd conf/进入conf目录（该SVN版本库配置文件）。
返回结果如下：
i. authz：是权限控制文件。
ii. passwd：是账号密码文件。
iii. svnserve.conf：SVN服务配置文件。

v. 按以下步骤设置账号密码：
a. 运行 vi passwd。
b. 按 i 键进入编辑模式。
c. 在 [users] 块中添加用户账号和密码，格式：账号=密码，比如示例中的suzhan = redhat（注意等号两端要有一个空格）。
d. 按 Esc 键退出编辑模式，并输入 :wq 保存并退出。

vi. 按以下步骤设置权限：
a. 运行 vi authz。
b. 按 i 键进入编辑模式。
c. 在末尾添加如下代码（其中，r表示读，w表示写）：

```
[/]
suzhan=rw
```

d. 按 Esc 键退出编辑模式，并输入 :wq 保存并退出。

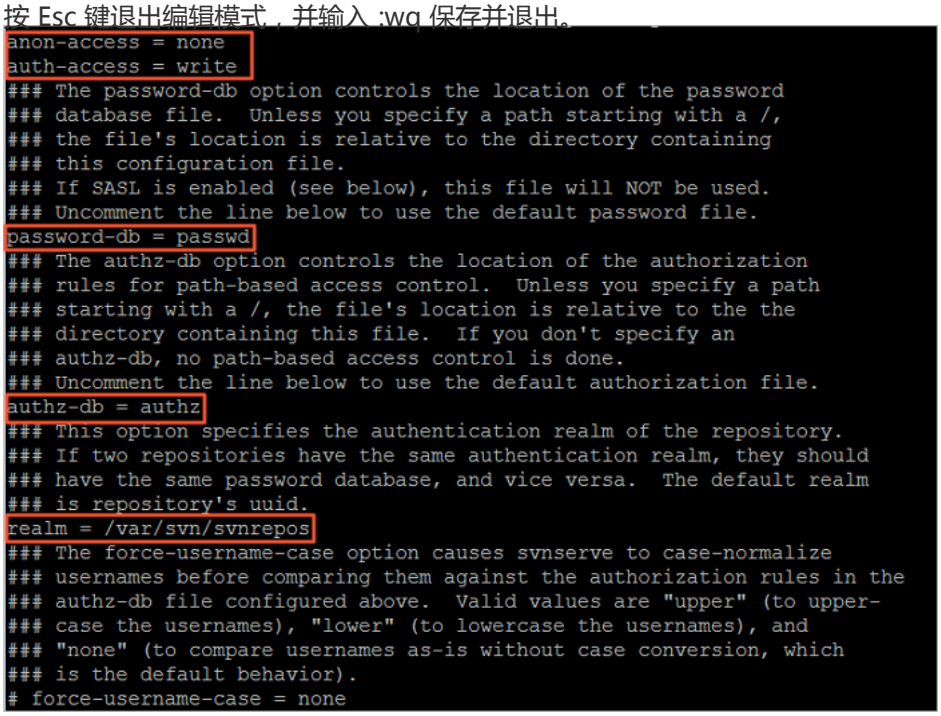


vii. 按以下步骤修改svnserve.conf文件。
a. 运行命令 vi svnserve.conf。
b. 按 i 键进入编辑模式。

c. 打开以下几个注释（注意每行不能以空格开始，等号两端要有一个空格）：

```
anon-access = read #匿名用户可读，您也可以设置 anon-access = none，不允许匿名用户访问。设置为 none，可以使日志日期正常显示
auth-access = write #授权用户可写
password-db = passwd #使用哪个文件作为账号文件
authz-db = authz #使用哪个文件作为权限文件
realm = /var/svn/svnrepos #认证空间名，版本库所在目录
```

d. 按 Esc 键退出编辑模式，并输入 :wq 保存并退出。

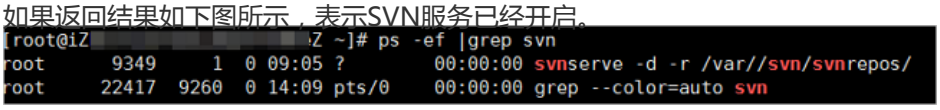


viii. 运行以下命令启动SVN版本库。

```
svnserve -d -r /var/svn/svnrepos
```

ix. 运行命令 ps -ef |grep svn 查看SVN服务是否开启。

如果返回结果如下图所示，表示SVN服务已经开启。



注意：

运行以下命令停止SVN命令。

```
killall svnserve
```

添加安全组规则

SVN服务的默认端口为TCP 3690。您需要登录 **ECS管理控制台**，**添加安全组规则** 放行TCP 3690端口。

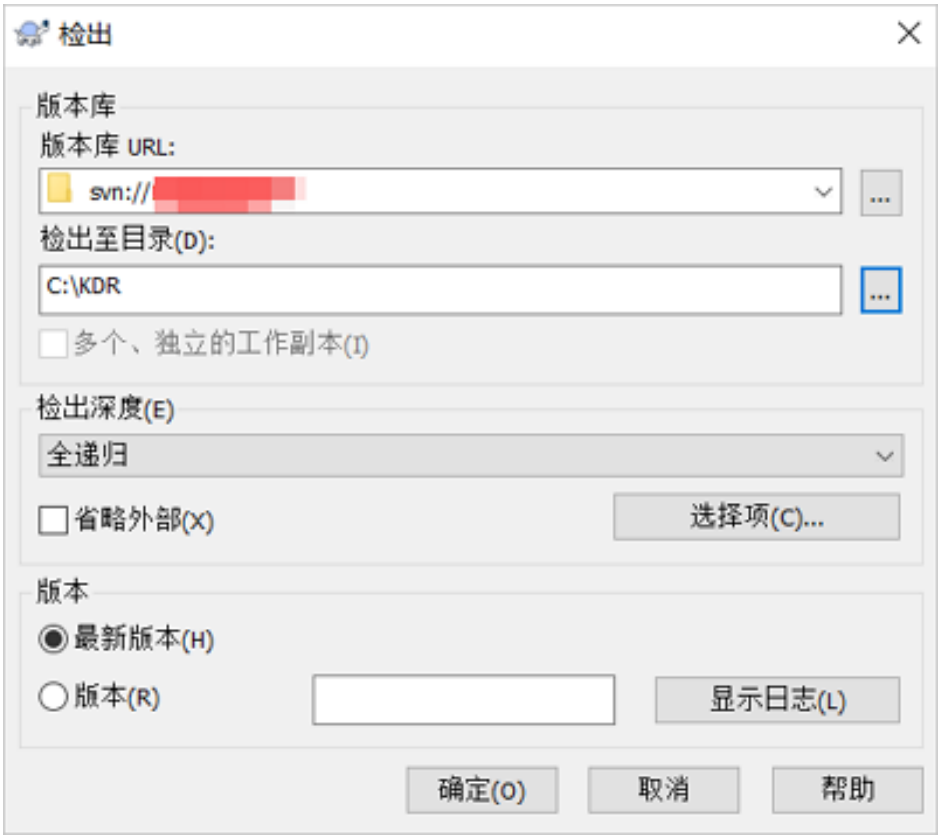
在Windows上测试

这部分说明如何从本地（Windows操作系统）访问ECS实例上安装的SVN服务。

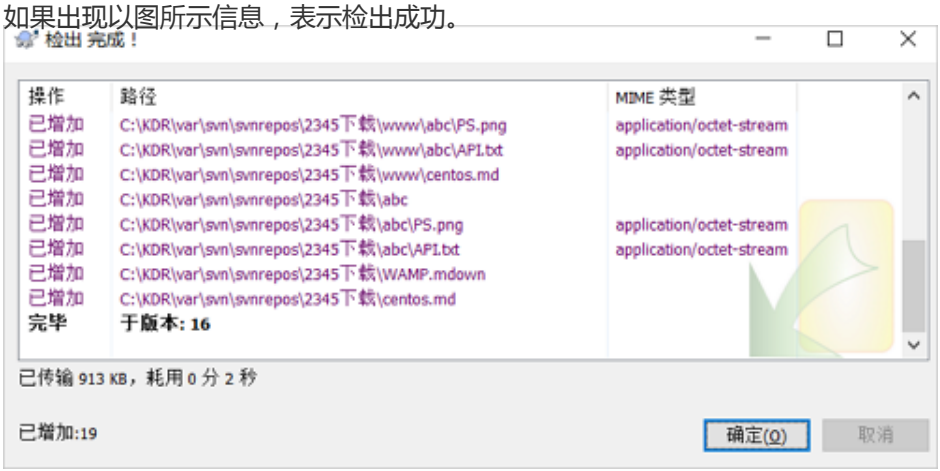
在本地机器上安装 TortoiseSVN客户端。

在您的本地项目文件夹（如示例中的C:\KDR），右键空白处弹出菜单，选择 **SVN检出**。

指定资源库URL，格式为 svn://实例公网IP地址/资源库名；指定 **检出至目录**（如本示例中的 C:\KDR）；再单击 **确定**。



如果出现以图所示信息，表示检出成功。



注意：
第一次登录需要输入密码，一切以passwd文件里面的账户密码为主。

修改并提交项目

将项目下载到本地机器后，您可以在添加文件、修改文件、删除文件等。

提交修改

按以下步骤提交修改：

在项目文件空白处单击右键，选择 **SVN提交**。

输入本次提交的版本更新信息（所作修改的注释）、勾选要提交的操作内容，点击 **确定**，即可把本机项目提交到SVN服务器资源库，覆盖掉资源库项目从而实现更新。

注意：
如果发生提交冲突，即两人都提交修改，后提交者由于版本落后会提交失败。这时可以先备份自己的项目，从服务端下载最新的项目后，再将自己的项目覆盖到本地项目文件夹，最后SVN提交即可成功提交。
假设您刚刚删掉了一个文件，这里就会显示如下截图所示信息。

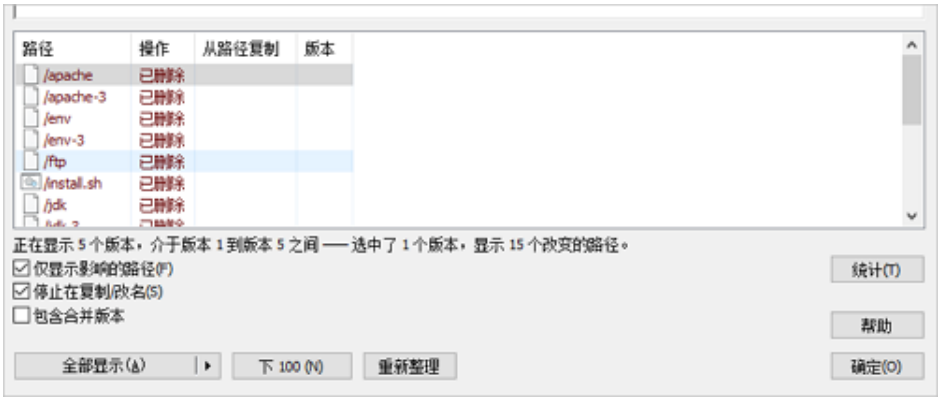
获取更新

如果别人修改了SVN服务端资源库上的项目，您想下载最新的项目，则在本机项目文件空白处单击右键，选择**SVN更新**，即可自动完成下载，并会提示所作的更新有哪些。

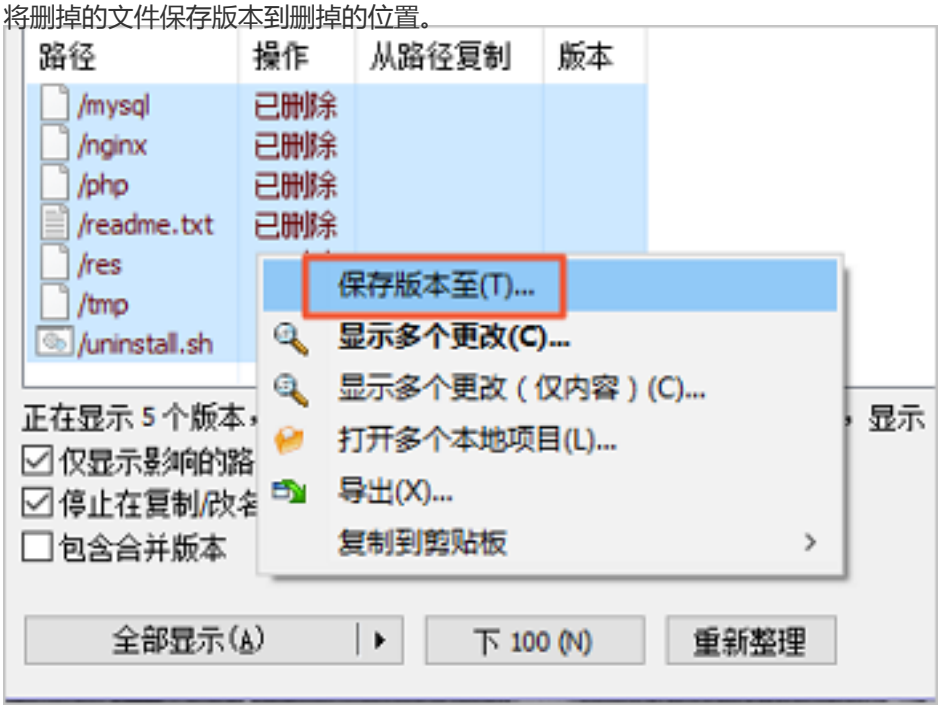
> **注意**：
> 在原项目文件夹内选择SVN更新，会自动覆盖原有内容。我们建议您先备份，再更新，防止自己本来的项目内容丢失。

SVN还原

1. 打开一个文件夹，右键检出数据。
2. 删掉数据。
3. 根据您是否已经提交修改采取不同的操作：
    - 未提交时，右键单击空白处，选择 **TortoiseSVN** > **SVN 还原**。
    - 已提交时，系统库里的数据也会得到同步，系统也会把它存的数据删掉。此时，您需要采取以下方法还原数据：
        a. 查看日志，确认删除了哪些文件。



b. 将删掉的文件保存版本到删掉的位置。



4. 打开原文件夹，选择 **SVN提交**，系统库里的数据就和这个文件同步了。

相关链接

更多开源软件尽在云市场：https://market.aliyun.com/software

快速搭建 ThinkPHP 框架

ThinkPHP 是一款免费开源的，快速、简单的面向对象的轻量级 PHP 开发框架，遵循 Apache2 开源协议发布，是为了敏捷 Web 应用开发和简化企业应用开发而诞生的。

适用对象

本文档介绍如何使用云市场的 **ThinkPHP 框架（含智慧云虚机面板）** 快速搭建 ThinkPHP 框架。适用于正在学习 PHP 或者已经基于 ThinkPHP 框架研发的开发者。

基本流程

1. 购买 ThinkPHP 框架镜像。
2. 上传您的程序。
3. 切换 PHP 脚本适应您的程序。
4. 开启 pathinfo。
5. 绑定域名。

1. 购买 ThinkPHP 框架镜像

单击 ThinkPHP 框架（含智慧云虚机面板） 进入镜像详情页。

单击 **立即购买**，按提示步骤根据您的实际业务需求购买 ECS 实例。

登录 ECS 管理控制台。

在左边导航栏里，单击 **实例**，进入 ECS 实例列表页。

选择所购 ECS 实例所在的地域，并找到所购 ECS 实例，在 **IP 地址** 列获取该实例的公网 IP 地址。

在浏览器地址栏中输入公网 IP 地址。屏幕上会显示提示页面。

在提示页面上单击 **获取权限** 按钮，下载权限文档 zhcloud-readme.doc。

权限文档中包含了智慧云虚机面板权限、FTP 权限和 MySQL 数据库权限，请保存好。

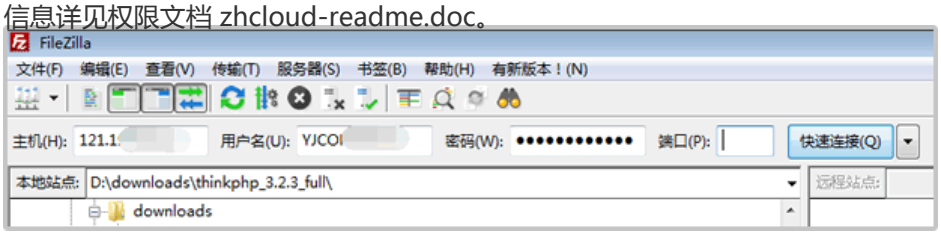
2. 上传您的程序

如果您已经用 ThinkPHP 框架写好了自己的应用程序，您可以通过 FTP 上传您的程序。

下载 FTP 工具。我们这里以 FileZilla FTP工具为例。下载地址为：https://www.filezilla.cn/download/client。

下载 FileZilla 后，双击 filezilla.exe，开始按软件提示安装 FileZilla FTP。

启动 FileZilla FTP，在 **主机**、**用户名** 和 **密码** 处分别输入 FTP IP 地址、FTP 账号和 FTP 密码，相关信息详见权限文档 zhcloud-readme.doc。



单击 **快速连接**，开始连接 FTP。

将您已经写好的应用程序拉到右边区域即可实现上传。

3. 切换 PHP 脚本适应您的程序

由于 PHP 的版本不同所支持的 PHP 函数也不尽相同。若您的程序对 PHP 版本有严格的要求，您可以通过 **脚**

本切换 来切换到您需要的 PHP 版本。如果没有严格要求，这一步就可以略过。

登录 **智慧云虚机面板** 。登录信息参见权限文档 zhcloud-readme.doc：

- host url 是指 **智慧云虚机面板** 的登录地址；
- host account 是指 **智慧云虚机面板** 的登录账号；
- host password 是指 **智慧云虚机面板** 的登录密码。

登录之后，如图所示。

单击 **脚本切换** ，选择您需要的 PHP 版本，单击 **确定** 。

4. 开启 pathinfo

使用 ThinkPHP 框架写的程序一般会用到 pathinfo。若您确实需要开启 pathinfo，请按如下操作。

登录 智慧云虚机面板。

单击 PATH_INFO，选择您的站点，单击开启按钮。

5. 绑定域名

如果您的实例公网 IP 地址已经完成了**域名备案**，您可以在智慧云虚机面板上绑定您的域名。

登录 智慧云虚机面板。

单击 **域名绑定**，输入您的域名即可绑定。
若您想禁止 IP 访问，删除含有 IP 地址的这条记录即可。

常见问题

301 重定向

登录 智慧云虚机面板。

找到 **自定义伪静态** > **自定义**，写入 301 重定向的 Nginx 规则，单击 **保存**。

下面以域名 yjcom.com 为例写 301 重定向 Nginx 规则。

- 方法 A：不使用www.yjcom.com域名访问网站时都 301 重定向到www.yjcom.com。

```
if ($host != 'www.yjcom.com' ) {
rewrite ^/(.*)$ http://www.yjcom.com/$1 permanent;
}
```

方法 B：使用yjcom.com域名访问网站时才 301 重定向到www.yjcom.com。

```
if ($host = 'yjcom.com' ) {
rewrite ^/(.*)$ http://www.yjcom.com/$1 permanent;
}
```

注意：实际使用时，将以上代码中的域名替换为您自己的域名。

更多开源软件尽在云市场：https://market.aliyun.com/software。

快速使用AMH建站

AMH 是一套通过 Web 控制和管理服务器的 Linux 服务器管理系统以及虚拟主机管理系统。使用阿里云的云服务器 ECS 安装 AMH 可以快速地搭建出任意 PHP 网站。阿里云云市场包含大量的镜像资源，您只需简单购买所需的镜像环境就可快速搭建出应用环境。

准备工作

提前创建好 ECS 实例，详情请参考**创建实例**。

镜像部署

说明：这里的镜像部署只针对还未购买 ECS 服务器的用户。

操作步骤

登录**阿里云云市场**，搜索关键字 **AMH4.2**。

选择 **PHP运行环境**，单击 **立即购买。**

配置 ECS 服务器，同时也会把 AMH 镜像环境安装进去，最后单击 **立即购买**，完成支付。

手动部署

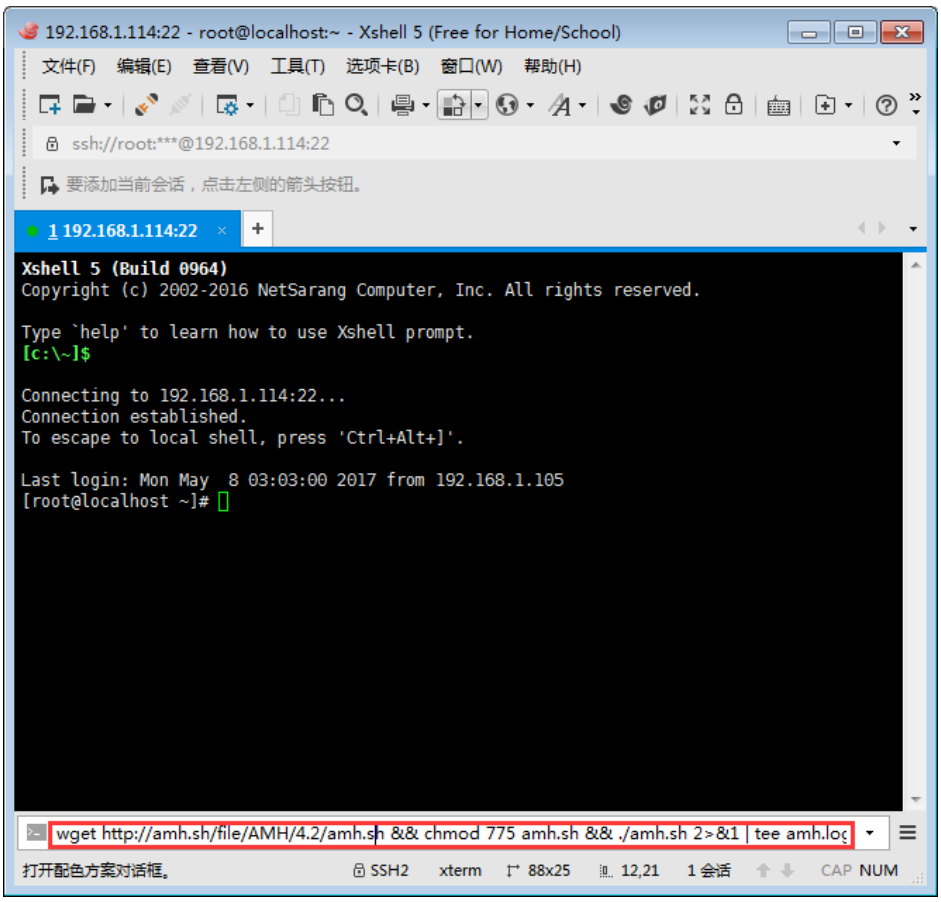
AMH 4.2 编译安装

AMH 4.2 为独立的一套 LNMP/Nginx 虚拟主机面板，安装请使用纯净系统。

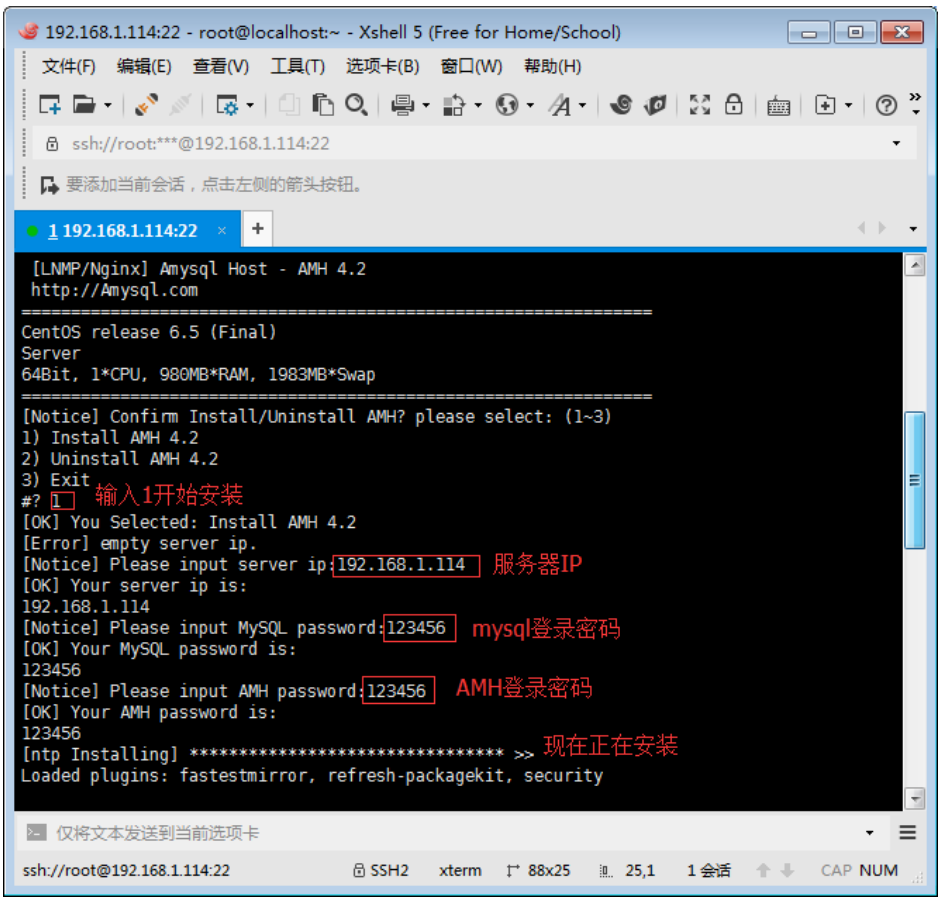
使用 root 账号登录 Linux 服务器。

执行 amh 安装脚本。

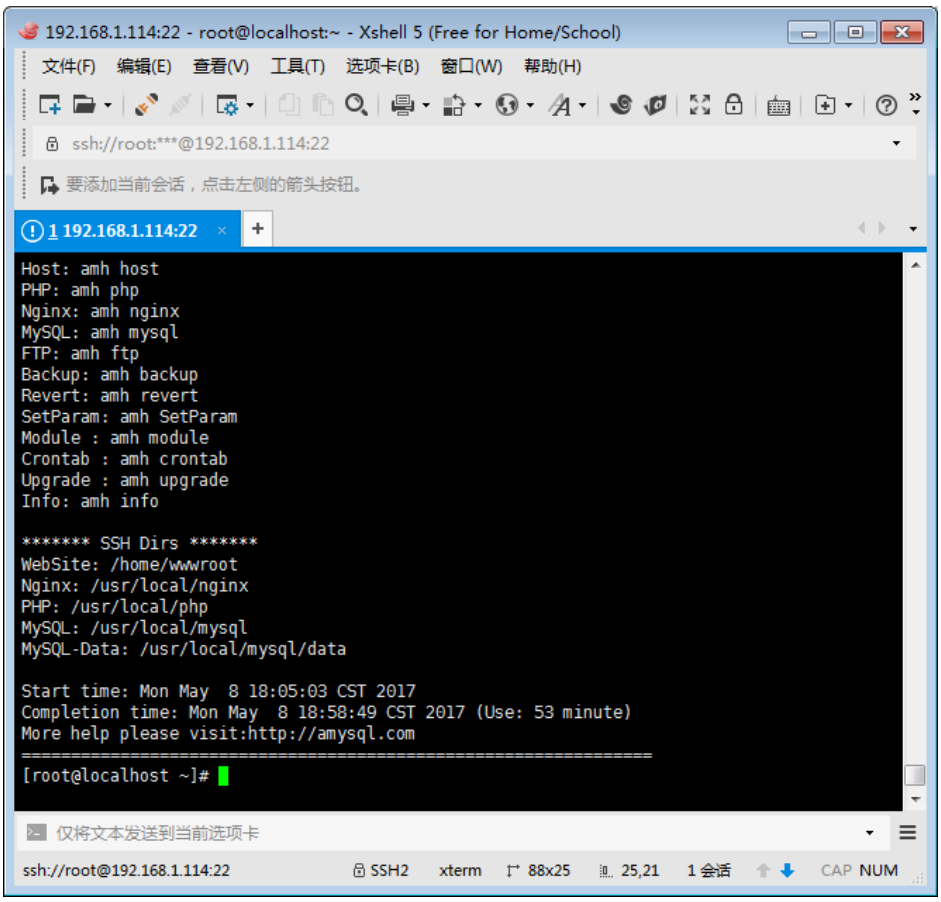
```
wget http://amh.sh/file/AMH/4.2/amh.sh && chmod 775 amh.sh && ./amh.sh 2>&1 | tee amh.log
```



根据提示输入选择 1 ~ 3 选项。其中，1 代表安装 amh，2 代表卸载 amh，3 代表退出不做操作。输入 1 回车，接着输入 MySQL 和 AMH 的登录密码后进入安装流程。最后如看到安装成功提示，说明系统已安装完成。



成功安装后，如有必要请删除日志文件 amh.log，若安装失败需协助安装请将错误日志反馈我们。
进入AMH web端管理，默认账号为 admin。



说明：安装成功需花费 1 小时左右。

AMH 使用流程

进入 AMH 后台登录界面。

说明：输入 ECS 公网 ip:8888。

使用镜像安装。默认帐号：admin 密码：cldera.com。

登录成功后，您可以看到很多功能，如下图所示。

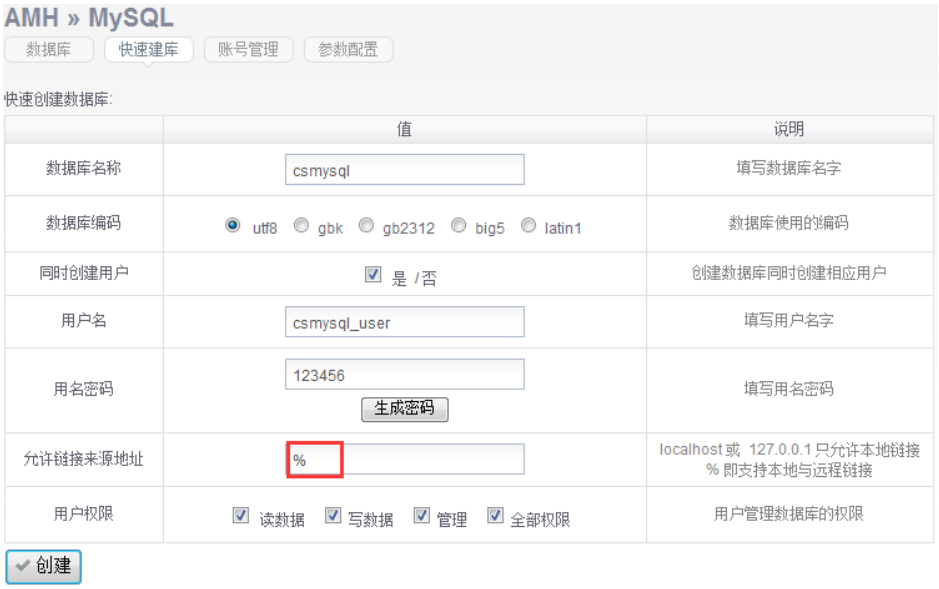
在顶端导航栏中，单击 **虚拟主机**，开始创建空间。

设置 **主标识域名** 和 **绑定域名** （其它默认），单击 **保存**。

在顶端导航栏中，选择 **MySQL** > **快速建库**，继续创建 PHP 网站所需的 MySQL 数据库。

按下图所示完成数据库的创建，其中数据库编码一般选择 UTF8 即可。

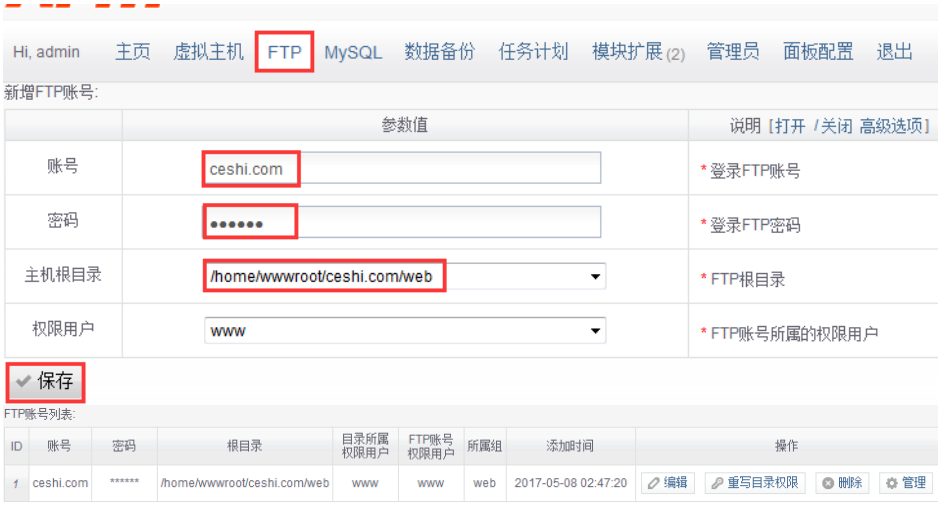
另一种创建是把 **localhost** 换成 **%**，这样可以远程管理 MYSQL。



AMH 搭建网站准备工作完成后，可以通过 dedecms 系统安装默认网站。

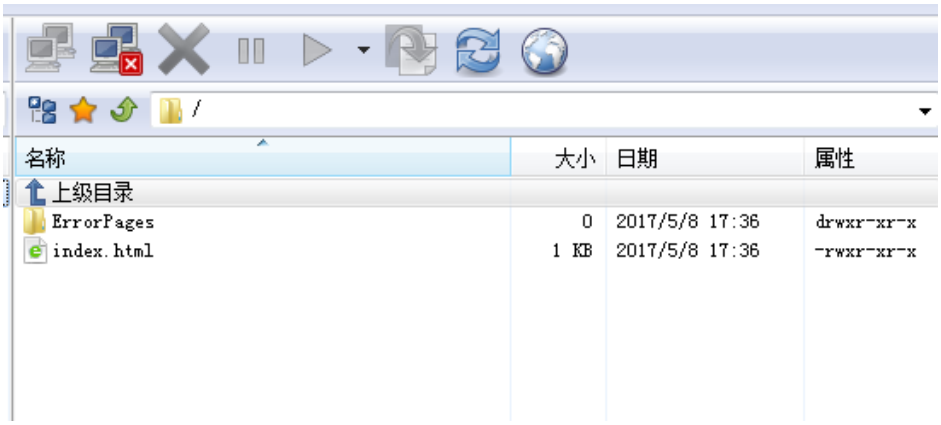
去官网下载 dedecms 系统上传到**空间根目录**。

在导航栏中选择 **FTP**，新增 FTP 账号。【先使用AMH创建个FTP 绑定到之前的空间】

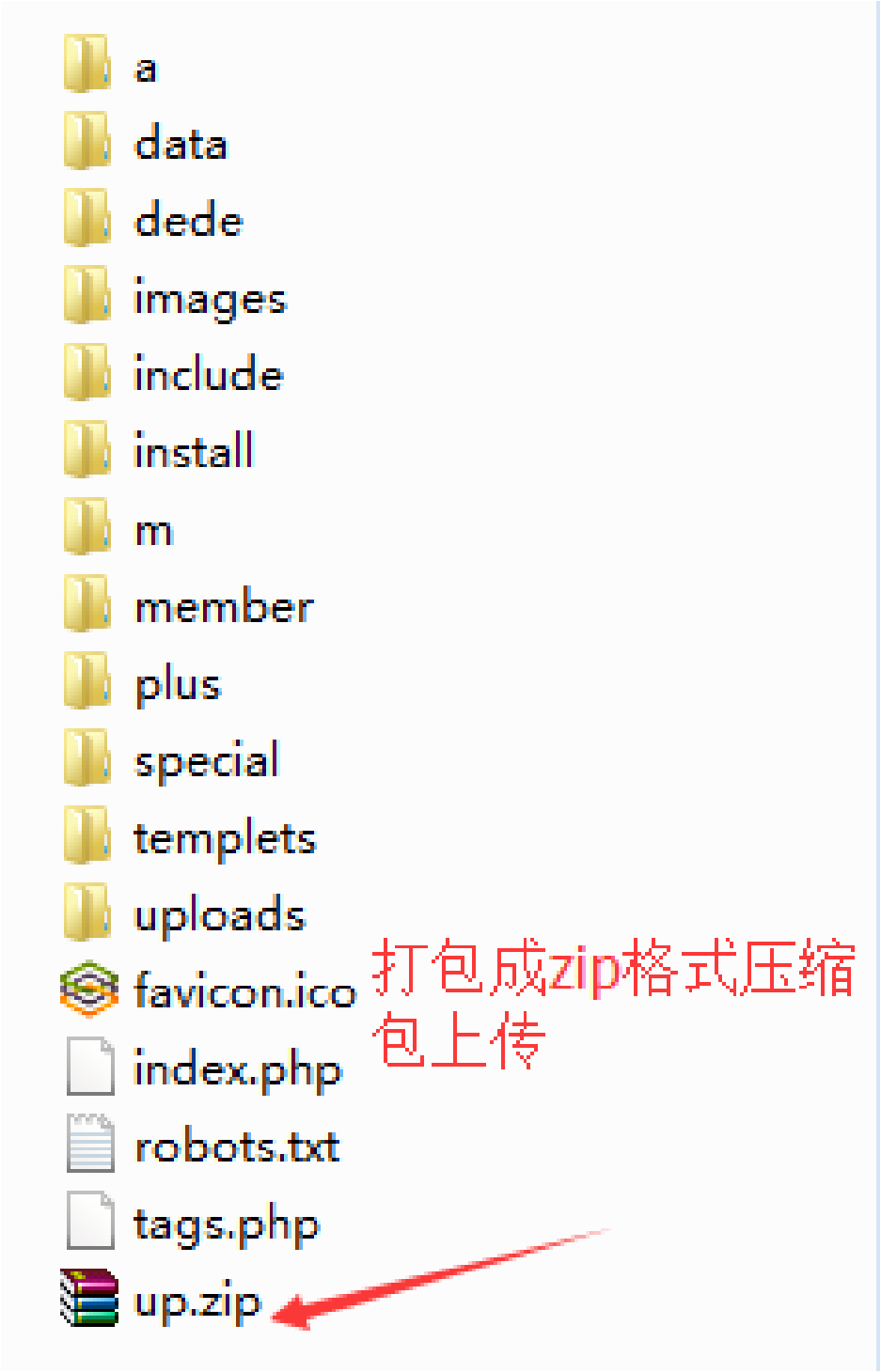


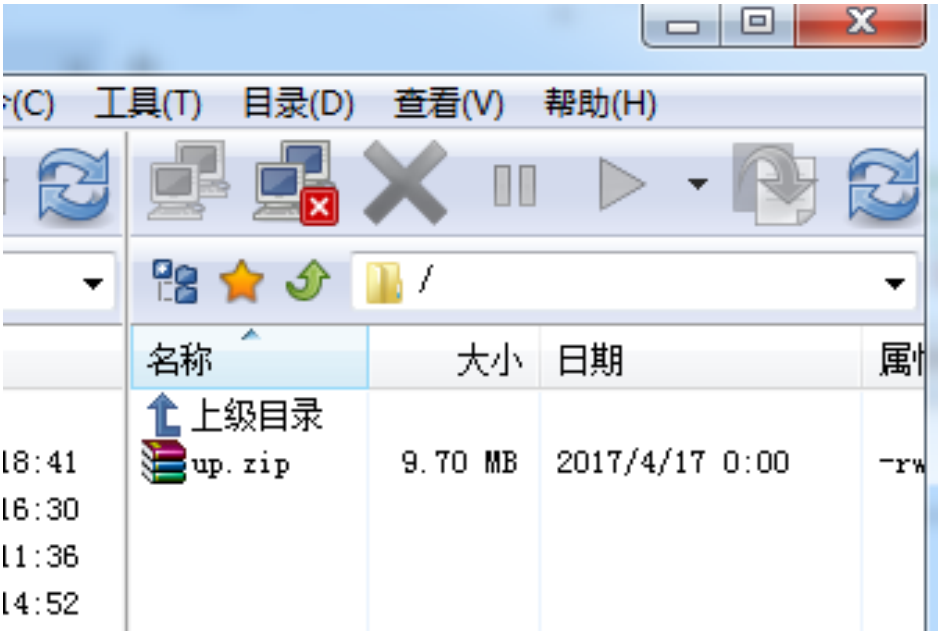
说明：该 FTP 需绑定到之前已创建的空间中。

登录 FTP。登录后，FTP 中有 2 个默认主页文件，您可以直接删除。



上传 dedecms 系统。


a
data
dede
images
include
install
m
member
plus
special
templets
uploads
favicon.ico
index.php
robots.txt
tags.php
up.zip
打包成zip格式压缩包上传

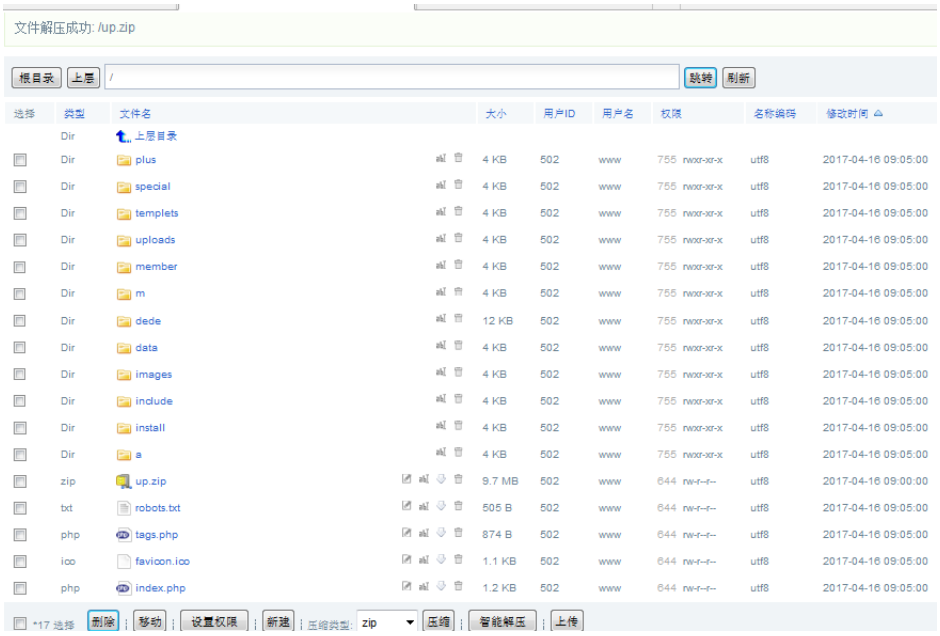


选择 FTP 创建的帐号，单击 **管理**。

登录 FTP，对已上传的文件进行解压。

勾选需要解压的文件， 单击 **智能解压**。

解压完成后，开始安装网站。



在浏览器中输入之前绑定的域名。该域名需要先解析到服务器。

系统环境要求必须满足下列所有条件，否则系统或系统部份功能将无法使用。

需开启的变量或函数	要求	实际状态及建议
allow_url_fopen	On	[√]On (不符合要求将导致采集、远程资料本地化等功能无法应用)
safe_mode	Off	[√]Off (本系统不支持在非win主机的安全模式下运行)
GD 支持	On	[√]On (不支持将导致与图片相关的大多数功能无法使用或引发警告)
MySQL 支持	On	[√]On (不支持无法使用本系统)

目录权限检测

系统要求必须满足下列所有的目录权限全部可读写的需求才能使用，其它应用目录可安装后在管理后台检测。

目录名	读取权限	写入权限
/	[√]读	[√]写
/plus/*	[√]读	[√]写
/dede/*	[√]读	[√]写
/data/*	[√]读	[√]写
/a/*	[√]读	[√]写
/install	[√]读	[√]写
/special	[√]读	[√]写
/uploads/*	[√]读	[√]写

后 退 继 续

数据库类型： MySQL ▼ 一般为MySQL，SQLite仅用于开发调试不建议生产中使用

数据库主机： localhost 一般为localhost

数据库用户： csmysql_user

数据库密码： 123456 信息正确

数据表前缀： dede_ 如无特殊需要,请不要修改

数据库名称： csmysql 数据库已经存在，系统将覆盖数据库

数据库编码： UTF8 仅对4.1+以上版本的MySql选择

管理员初始密码

用户名： admin

只能用'0-9'、'a-z'、'A-Z'、'.'、'@'、'_'、'-'、'!'以内范围的字符

密 码： admin

Cookie加密码： w6WDekqeyNsF0DihLvlcXTKvW\

网站设置

网站名称： 我的网站

管理员邮箱： admin@dedecms.com

网站网址： http://www.ceshi.com

说明：完成操作后，您可以快速使用 AMH 建站。与其它 PHP 系统的安装使用都是相同的。

在ECS上部署数据库

数据库是依照某种数据模型组织起来并存放二级存储器中的数据集合。这种数据集合具有如下特点：尽可能不重复，以最优方式为某个特定组织的多种应用服务，其数据结构独立于使用它的应用程序，对数据的增、删、改和检索由统一软件进行管理和控制.

阿里云有提供相应的高可用数据库架构RDS，但由于RDS具有一定的限制条件，可能无法满足部分生产环境的要求，例如需要使用Oracle数据库、需要使用SQL Server报表服务等，在这种情况下，我们需要考虑在ECS上搭建数据库的方式.

本文档介绍如何在云服务器ECS实例搭建常用数据库（Oracle、MySQL、SQL Server）。

常用数据库简介

常用数据库包含以下三种：Oracle、MySQL、SQL Server。

Oracle

Oracle可以支持多种不同的硬件和操作系统平台，从台式机到大型和超级计算机，为各种硬件结构提供高度的可伸缩性，支持对称多处理器、群集多处理器、大规模处理器等，并提供广泛的国际语言支持。

Oracle是一个多用户系统，能自动从批处理或在线环境的系统故障中恢复运行。系统提供了一个完整的软件开发工具Developer2000，包括交互式应用程序生成器、报表打印软件、字处理软件以及集中式数据字典，用户可以利用这些工具生成自己的应用程序。

Oracle以二维表的形式表示数据，并提供了SQL(结构式查询语言)，可完成数据查询、操作、定义和控制等基本数据库管理功能。

Oracle具有很好的可移植性，通过它的通信功能，微型计算机上的程序可以同小型乃至大型计算机上的Oracle，并且能相互传递数据。

Oracle属于大型数据库系统，主要适用于大、中小型应用系统，或作为客户机/服务器系统中服务器端的数据库系统。

MySQL

MySQL是一种开放源代码的关系型数据库管理系统（RDBMS），MySQL数据库系统使用最常用的数据库管理语言—结构化查询语言（SQL）进行数据库管理。MySQL数据库也是可以跨平台使用的（如linux和Windows）。

SQL Server

SQL Server是美国Microsoft公司推出的一种关系型数据库系统，是一个可扩展的、高性能的、为分布式客户机/服务器计算所设计的数据库管理系统，实现了与WindowsNT的有机结合，提供了基于事务的企业级信息管理系统方案，SQL Server 2016以前的版本只支持在Windows上运行，不支持在Linux上运行。

在ECS（Windows系统）上部署Oracle数据库

企业中在Windows上部署Oracle数据库的方式是先部署一台Windows系统的机器，然后在Windows系统上安装Oracle软件。这种部署方式具有耗时长、部署复杂、易出错等缺陷。在阿里云平台上，可通过自带的镜像市场实现一键部署Windows系统的Oracle数据库，完美解决耗时长、部署易出错的缺陷。

操作步骤

登录云服务器管理控制台。

单击左侧导航中的 **云服务器>创建实例。**在创建实例的页面上，定位到镜像，单击 **镜像市场。**

单击镜像市场的 **从镜像市场选择（含操作系统）。**

在镜像市场的页面，选择 **数据库。**

在操作系统选择，选择主流使用的Windows Server 2012，架构选择64位系统。

在下方查看到具有Windows2012 x64 oracle11g11.1.0.4企业版，单击购买。

进入到购买页面，单击购买即可。

购买完成配置后，如需要正常使用，还需要在ECS所属的安全组配置入方向的开放1521、1158端口，安全组添加安全组规则操作。

在ECS（Linux系统）上部署Oracle数据库

在阿里云上自带的镜像市场还包含Linux系统的Oracle数据库，可通过购买实现一键部署Linux系统的Oracle数据库，节省大量的敲击代码的时间。

操作步骤

登录云服务器管理控制台。

单击左侧导航中的 **云服务器>创建实例**。在创建实例的页面上，定位到镜像，单击**镜像市场**。

单击镜像市场的 **从镜像市场选择（含操作系统）**。

在镜像市场的页面，在搜索框中输入Oracle。

列出了相应的Oracel数据库的版本，选择相应的版本进行购买。

进入到购买页面，单击购买即可。

购买完成配置后，如需要正常使用，还需要在ecs的所属的安全组配置入方向的开放1521、1158端口，安全组添加安全组规则操作。

在ECS（Windows系统）上部署SQL Server数据库

企业中还会用到微软SQL Server数据库，因目前SQL Server 2016之前的版本只支持在Windows上运行安装，所以本文档只介绍在Windows系统的ECS实例上部署SQL Server数据库的方法。

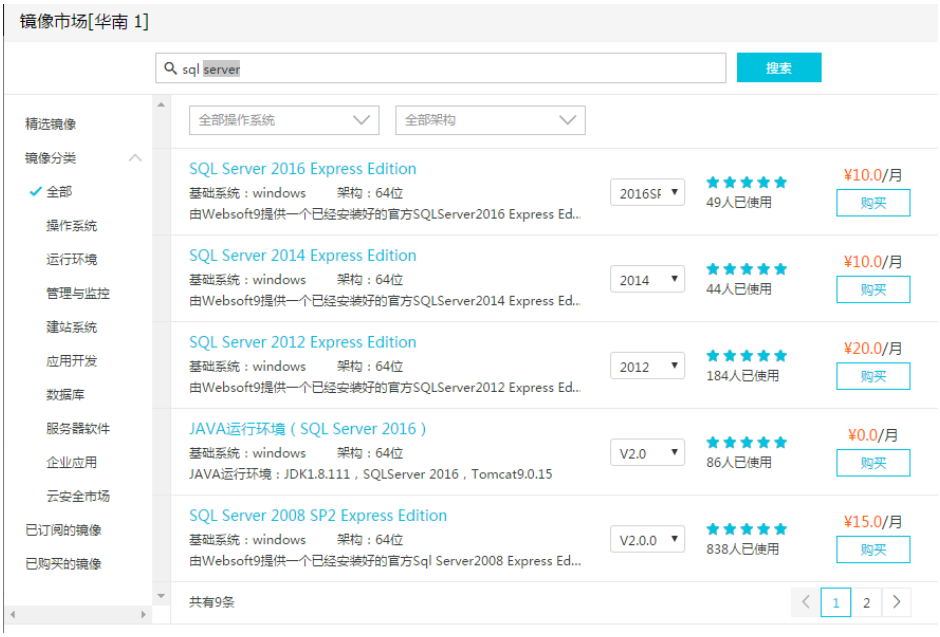
操作步骤

登录云服务器管理控制台。

单击左侧导航中的 **云服务器>创建实例。**在创建实例的页面上，定位到镜像，单击 **镜像市场。**

单击镜像市场的 **从镜像市场选择（含操作系统）。**

在镜像市场的页面，在搜索镜像框中输入SQL Server。



选择需要的相应版本，单击购买，进入到购买页面，单击购买即可。

购买完成配置后，如需要正常使用，还需要在ecs的所属的安全组配置入方向的开放1433、1434端口，安全组添加安全组规则操作。

在ECS（Linux系统）上部署MySQL数据库

MySQL数据库在企业中经常被用到，阿里云除了有RDS云数据库产品支持MySQL外，在云镜像市场中还有已完成安装MySQL数据库的Linux系统，可借助云镜像市场实现便捷、快速地部署MySQL数据库。

操作步骤

登录云服务器管理控制台。

单击左侧导航中的 **云服务器>创建实例。**在创建实例的页面上，定位到镜像，单击 **镜像市场。**

单击镜像市场的 **从镜像市场选择（含操作系统）。**

在镜像市场的页面，选择数据库，在搜索框中输入MySQL。

选中相应的版本及规格，单击购买，进入到购买页面，单击购买即可。

安装后，如需要正常使用，还需要在ecs的所属的安全组配置入方向的开放3306端口，安全组添加安全组规则操作。

在ECS（Windows系统）上部署MySQL数据库

目前在云市场上暂未包含有Windows系统的MySQL数据库的镜像，所以需要手动部署MySQL数据库。

操作步骤

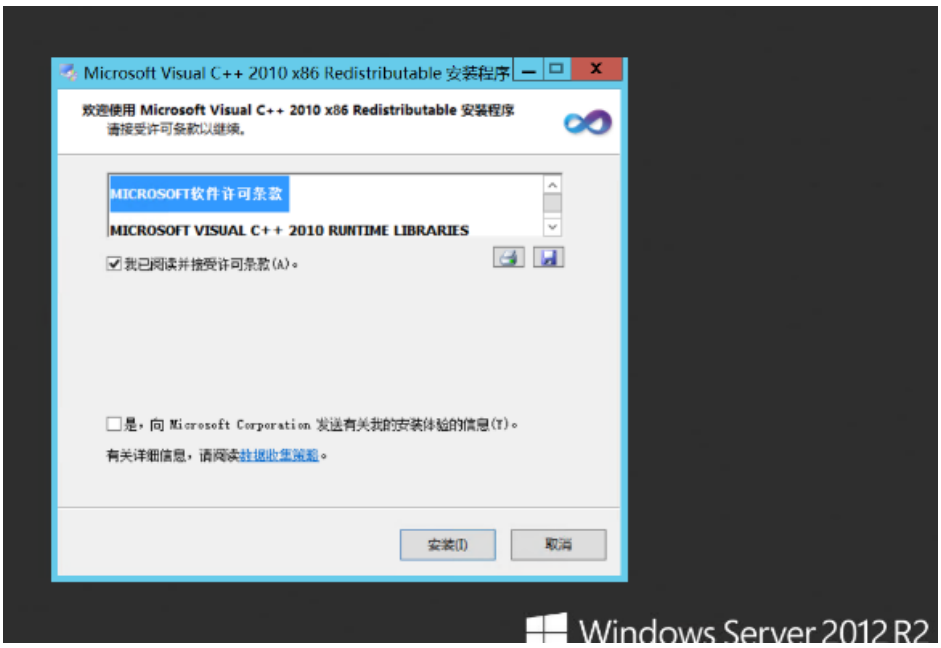
登录云服务器管理控制台，购买相应的Windows Server实例，可参考购买Windows实例。

购买成功后，进行相应的系统层面配置，远程登录ECS实例。

进入MySQL官网下载MySQL的安装包。

安装MySQL之前，需要先下载插件vcredist_x86.exe。

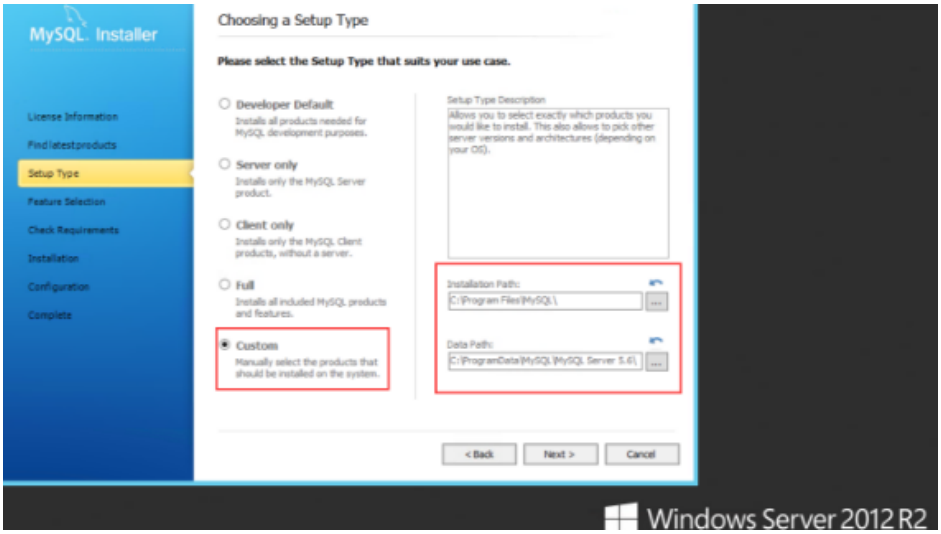
安装vcredist_x86.exe插件。



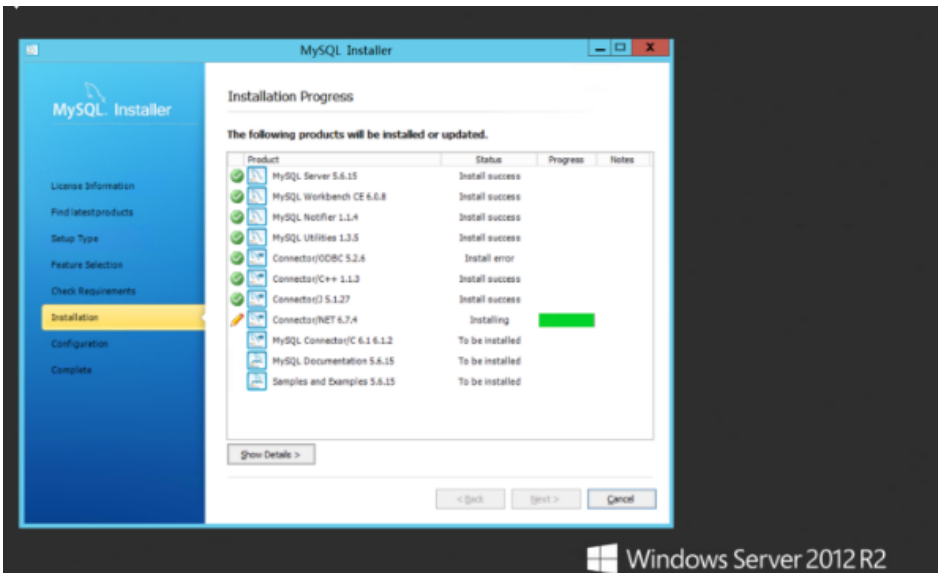
下载完成后，打开**mysql-installer-community-5.6.15.0.msi**进行MySQL安装。

选择第一项**Install MySQL Products**。

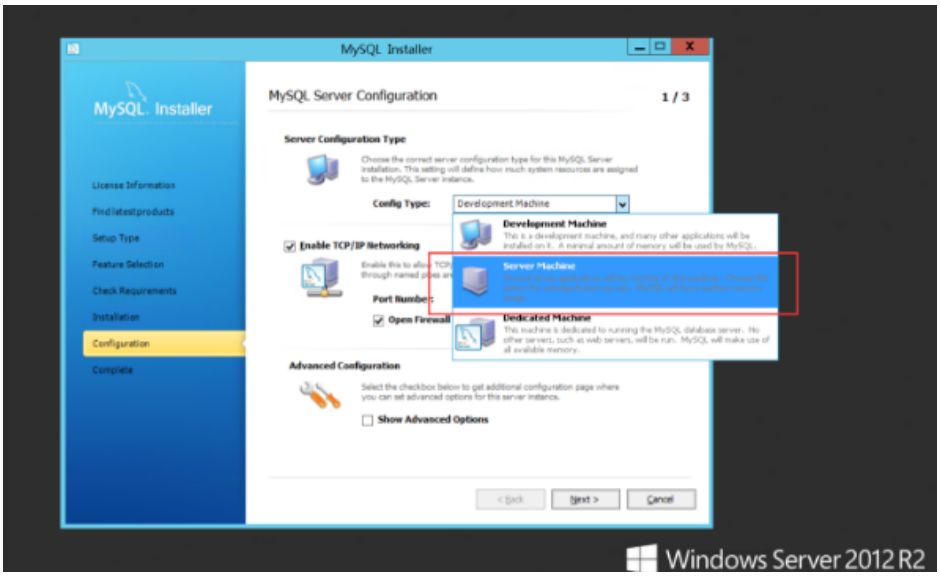
勾选接受协议和跳过检测更新，单击下一步，单击**Custom**，也就是自定义安装，右边是选择MySQL的安装位置和数据库位置，下图操作案例选择的是默认路径，单击**NEXT**。



保持默认点解**NEXT**， 单击**Execute**，开始执行安装。



单击**NEXT**至配置页面，选择**Server Machine**。



保持默认**NEXT**输入管理员root的密码，直至最后完成安装；安装完成后会在页面出现MySQL的管理命令控制台。

安装后，如需要正常使用，还需要在ECS的所属的安全组配置入方向的开放3306端口，安全组添加安全组规则操作。

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。

注意：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.点击登录云服务器管理控制台。如果尚未注册，单击 免费注册。

2.定位到 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。

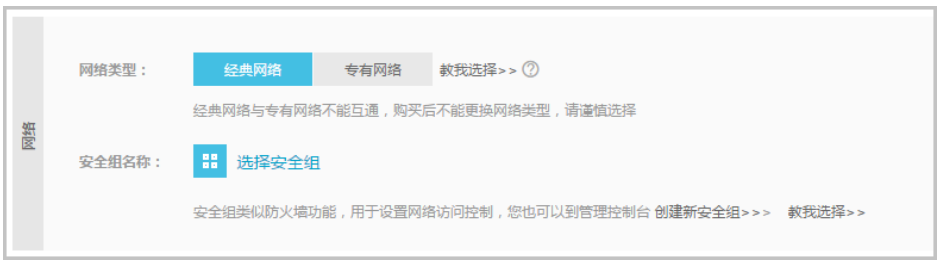
例如，如果您的网站访问者都分布在北京地区，则可以选择 华北 2。

注意：

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。

7.1按固定带宽付费。

7.2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）。

1、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。

2、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。

3、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。

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。

4、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。

5、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
```

(注意：cat 后面要有空格)

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
```

注意：-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
```

备注1: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。

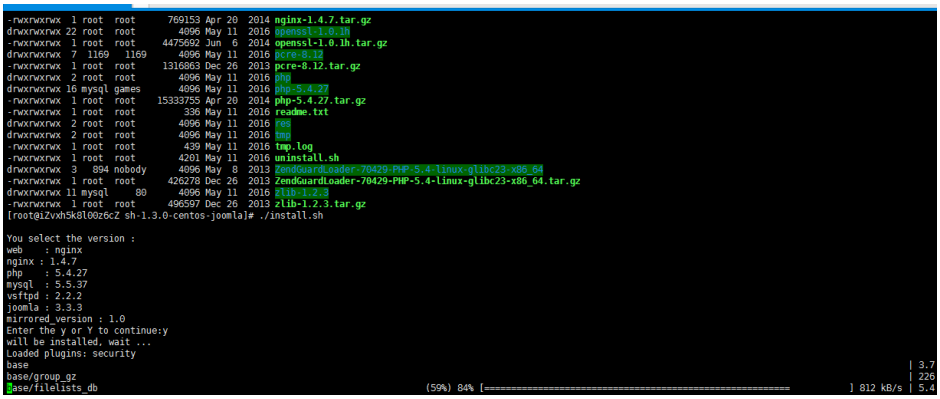
备注2:如果不小心删除了 /root/sh-1.3.0-centos-joomla,可以解压缩 /root/sh-1.3.0-centos-joomla.zip 参考一下命令:

```
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
```

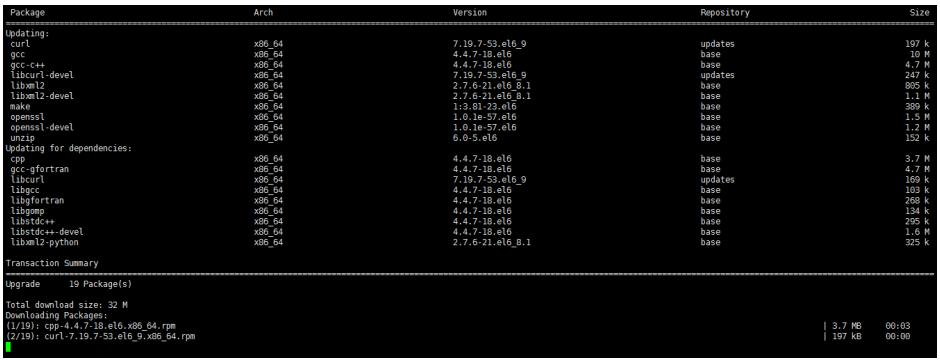
在 centos6.5 系统中自行安装

/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

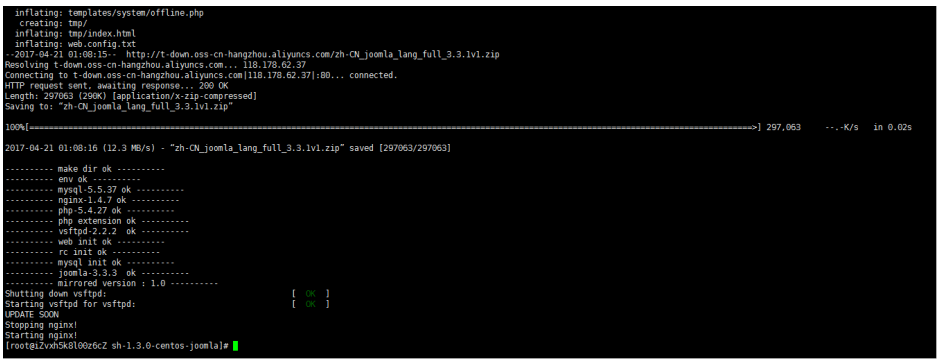
根据提示输入 y。

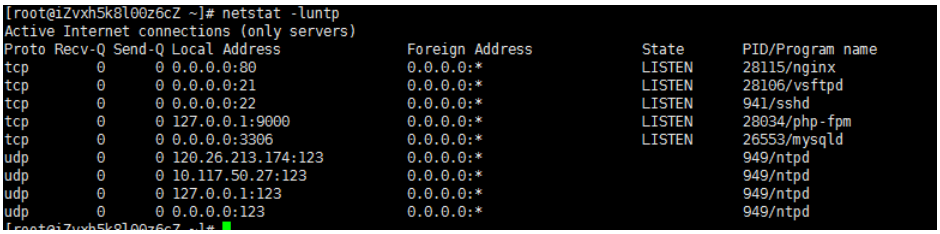


持续安装中。



安装结束出现以下界面。





80、21、9000、3306 等端口都已开启。

配置外网访问

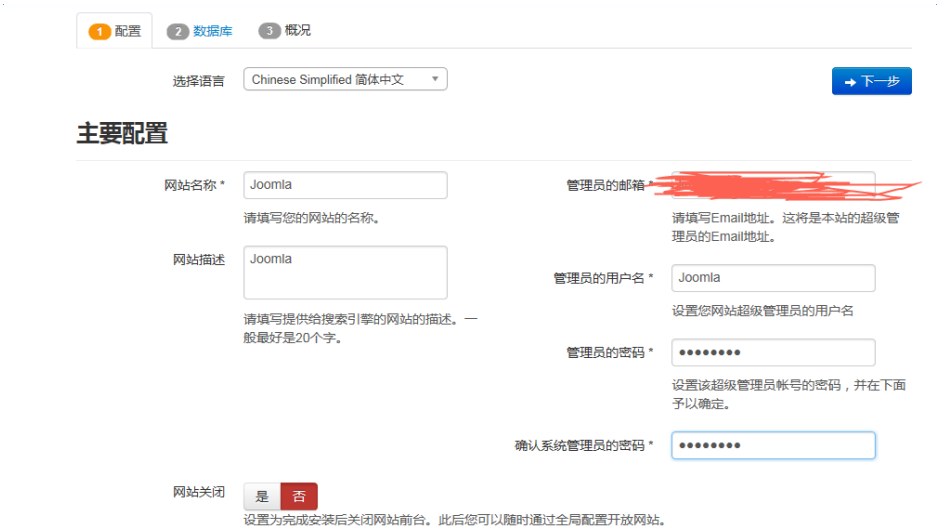
在镜像环境中处于安全考虑，joomla 默认页面只允许 127.0.0.1 访问，/root/ 目录下提供了一个 joomla_opennet.sh 的脚本。用户运行之后，可通过外网访问 joomla 的默认设置页面。

运行脚本文件：

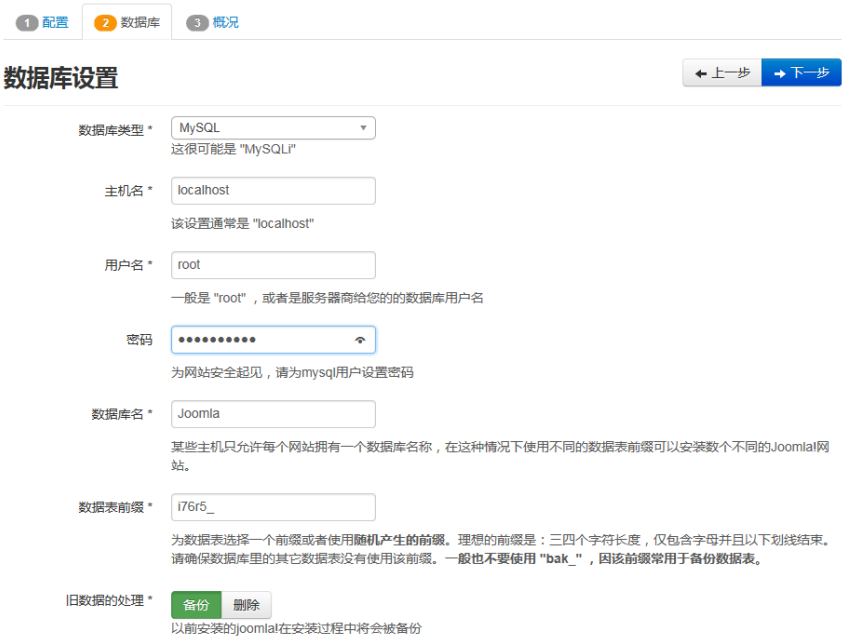
```
/root/joomla_opennet.sh
```

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，并填写相关内容点击下一步。



选择mysql数据库，填写相关权限后，点击下一步。

查看相关配置是否符合，确认完毕点击安装。

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/
```

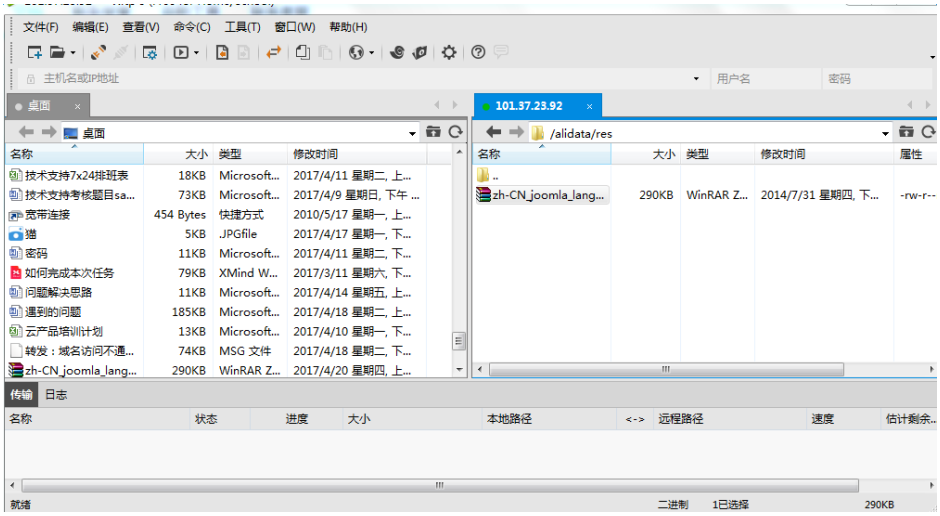
至此，joomla 搭建完成。

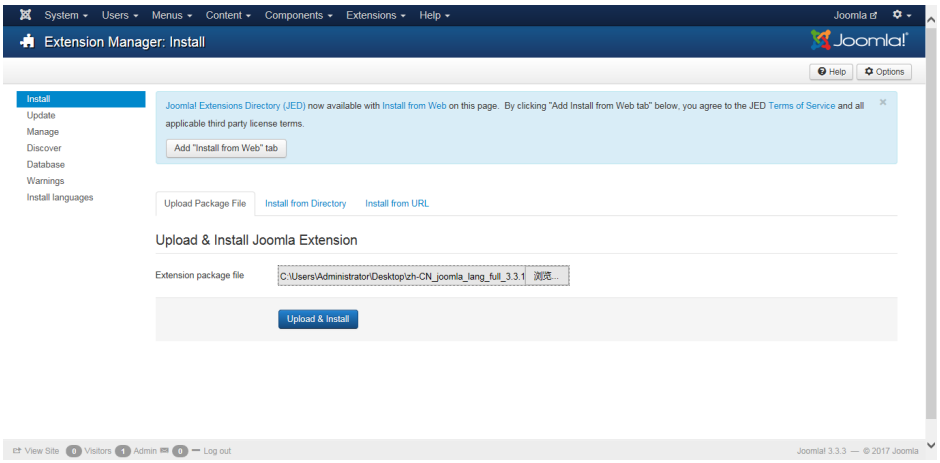
访问前端网站 http://ip;访问后台管理 http://ip/administrator。

关于 Joomla 支持中文。

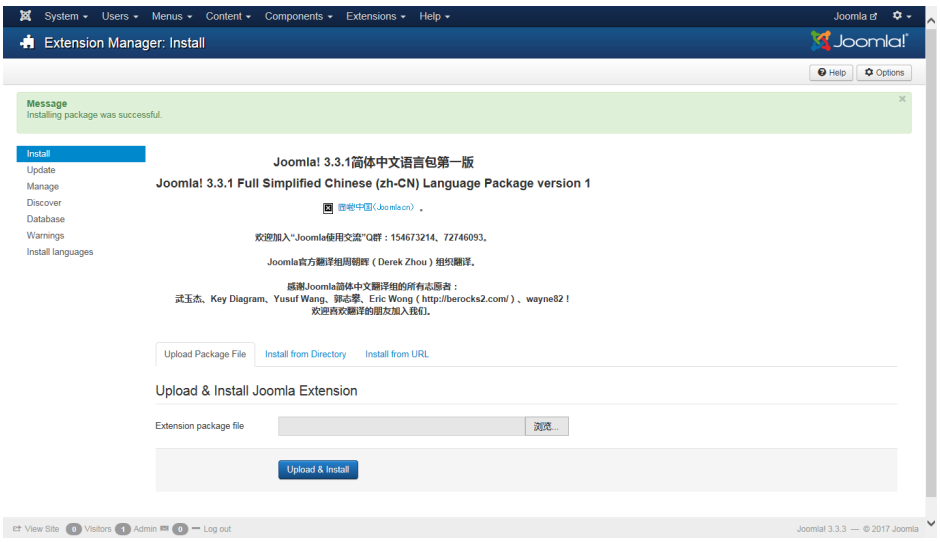
Joomla 安装完成之后默认前台后台都是英文界面，中文语言需要手动安装。登陆 Joomla 之后在 Extensions (扩展) —— Extension Manager (扩展管理)

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。

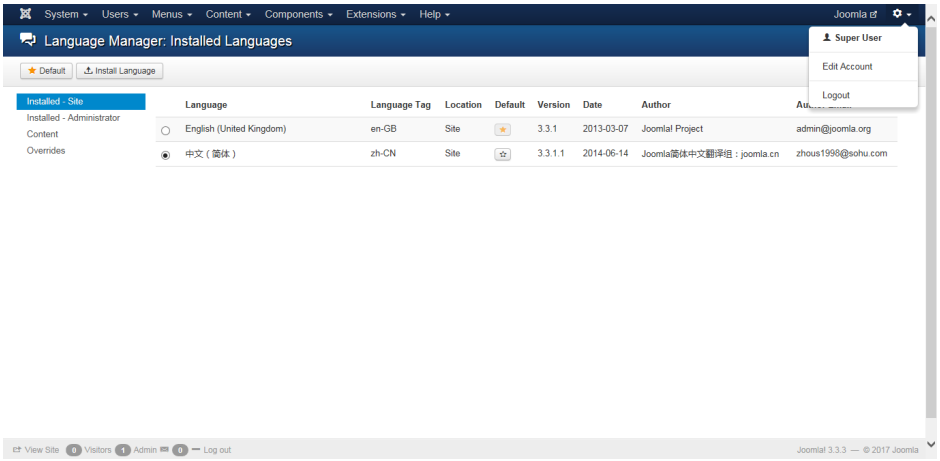




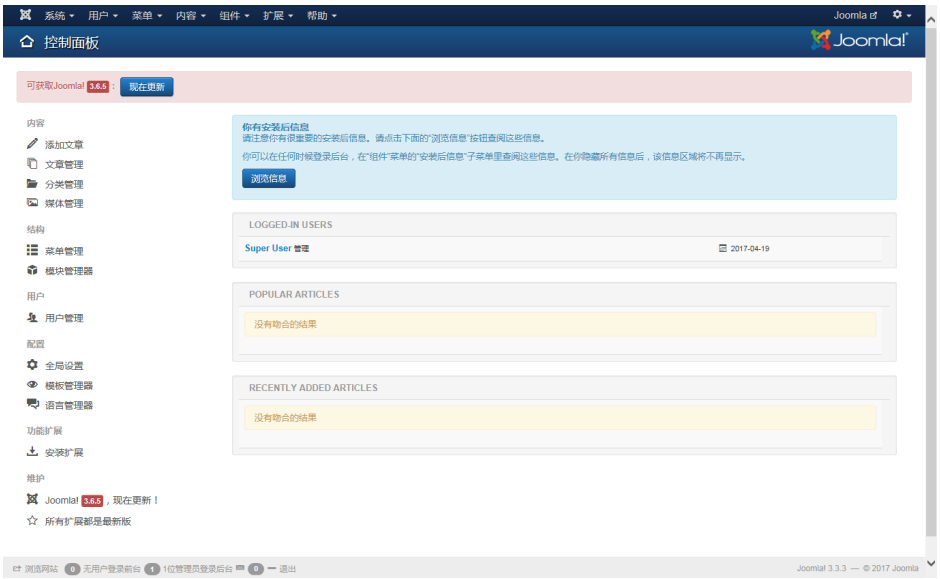
点击 “Update” & “Install” 上传。



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



登陆后就能进入中文界面了。



快速搭建 Moodle 课程管理系统

Moodle 是一个开源课程管理系统，采用 PHP + MySQL 方式运行的自由开源软件，遵循 GNU 公共许可协议。世界各地教育工作者越来越喜欢使用 Moodle 为学生建立网上动态网站。Moodle 平台界面简单、精巧，您可以根据需要随时调整界面，增减内容。

本文档介绍如何使用云市场的 **moodle 网络教学平台（Centos 7.0 64位）** 快速搭建 Moodle 课程管理系统。

适用对象

适用于要搭建 Moodle 课程管理系统的用户。

操作流程

1. 创建使用 Moodle 网络教学平台镜像的 ECS 实例。
2. 远程连接 ECS 实例并查看权限。
3. 安装 Moodle 课程管理系统。
4. （**可选**）在服务器里绑定域名。

前提条件

- 您已经拥有一个阿里云账号。
- 如果您希望用户通过域名访问您的站点，您应该已经有一个已备案的域名。如果域名没有备案，您购买 **ECS 实例** 后到阿里云备案中心备案。备案地址为：https://beian.aliyun.com 。
- 您已经明白**远程连接 Linux 实例的方法**。这里假设您本地使用的是 Windows 操作系统，并使用 PuTTy.exe 远程连接 Linux ECS 实例。这里是 PuTTY 的下载地址：https://www.chiark.greenend.org.uk/~sgtatham/putty/latest.html 。

创建 ECS 实例

在云市场中搜索 moodle 网络教学平台（Centos 7.0 64位），并进入镜像详情页。

在镜像详情页上，单击 **立即购买**，按提示步骤购买 ECS 实例。需要注意以下配置：

- **网络类型**：应选择 **经典网络**。
- **带宽**：不能为 0 Mbps。
- **付费方式**：如果您的站点需要备案，应选择 **包月套餐**。
- 其他配置您可以按需选择。

登录 云服务器 ECS 管理控制台。

在左边导航栏里，单击 **实例**，进入 ECS 列表页。

选择所购 ECS 实例所在的地域，找到所购 ECS 实例，开始管理实例。您需要做以下操作：

- 在 **IP 地址** 列获取该实例的公网 IP 地址。
- 确认 ECS 实例所在的安全组中已经添加了以下几个安全组规则：
  - SSH 远程连接 ECS 实例：

网卡类型	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
公网	入方向	允许	SSH(22)	22/22	地址段访问	0.0.0.0/0	1

  - HTTP 访问 Web 服务器：

网卡类型	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
公网	入方向	允许	HTTP(80)	80/80	地址段访问	0.0.0.0/0	1

  - 允许使用 FTP 上传或下载文件：

网卡类型	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
公网	入方向	允许	自定义 TCP	20/21	地址段访问	0.0.0.0/0	1

  - 允许使用 MySQL：

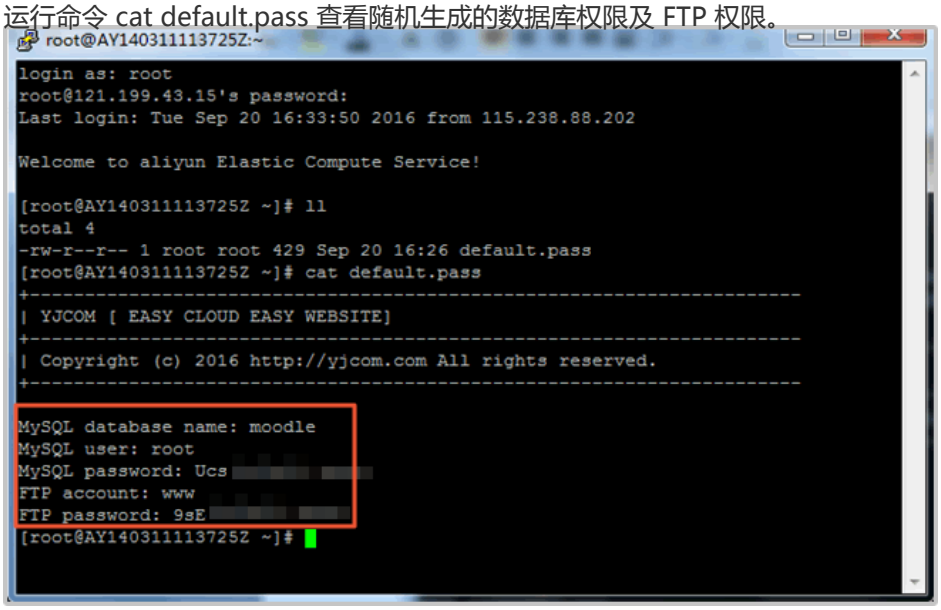
网卡类型	规则方向	授权策略	协议类型	端口范围	授权类型	授权对象	优先级
公网	入方向	允许	自定义 TCP	3306/3306	地址段访问	0.0.0.0/0	1

- 使用 重置实例密码 功能设置 ECS 实例的登录密码。

远程连接 ECS 实例并查看权限

远程连接到 ECS 实例。

运行命令 cat default.pass 查看随机生成的数据库权限及 FTP 权限。



安装 Moodle 系统

安装 Moodle 系统需要知道以下 2 个地址：

- 数据库的管理地址为：http://您的公网 IP 地址/phpmyadmin/。
- Moodle 的安装地址为：http://您的公网 IP 地址/install.php。

按以下步骤安装 Moodle 系统：

在浏览器地址栏里，输入 Moodle 的安装地址。

选择您想要的语言，选择后单击 **向后**。本示例中，选择 **简体中文**。

在**确认路径**页面上，保持所有默认目录不变。单击 **向后**。

在 **选择数据库驱动** 页面上，采用默认类型。单击 **向后**。

设置数据库。其中：

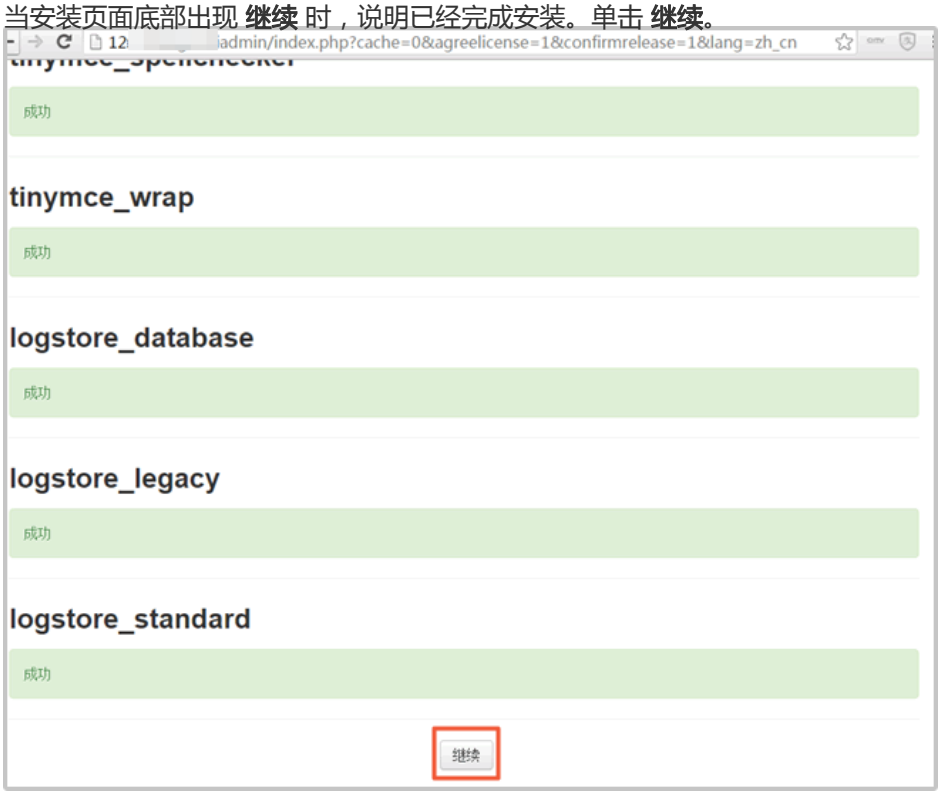
- **数据库主机**：只能填 127.0.0.1。
- **数据库名**、**数据用户名**和**数据库密码**采用 远程登录 ECS 实例并查看权限 里查得的 MySQL 权限信息。
- **数据库服务端口**：填写 3306。

确认所有信息后，单击 **向后**。

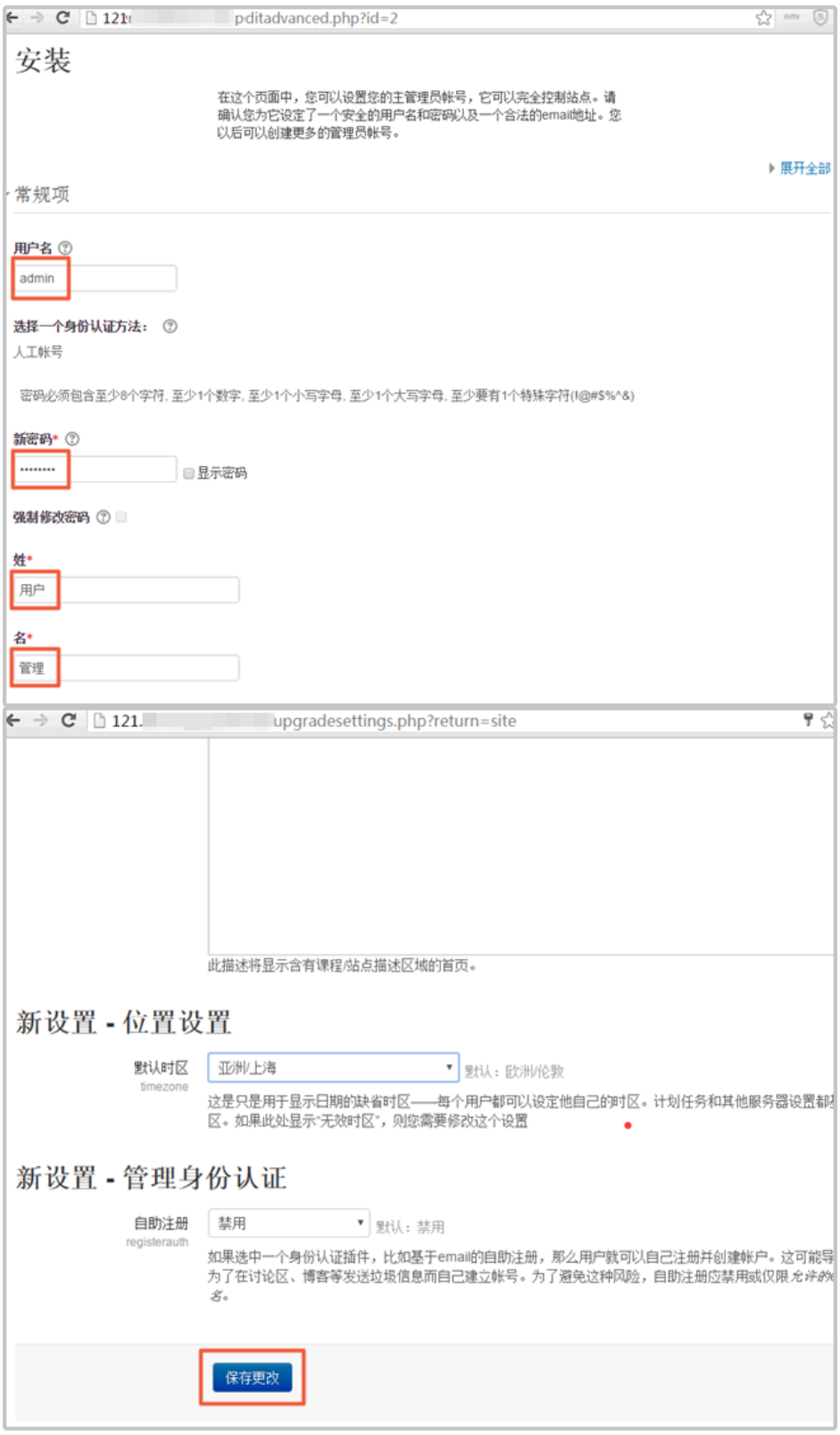
阅读并确认了解版权声明。单击 **继续**。

下图中显示的是安装 Moodle 需要的一些组件，都已经部署好了。单击 **继续** 就可以安装系统。

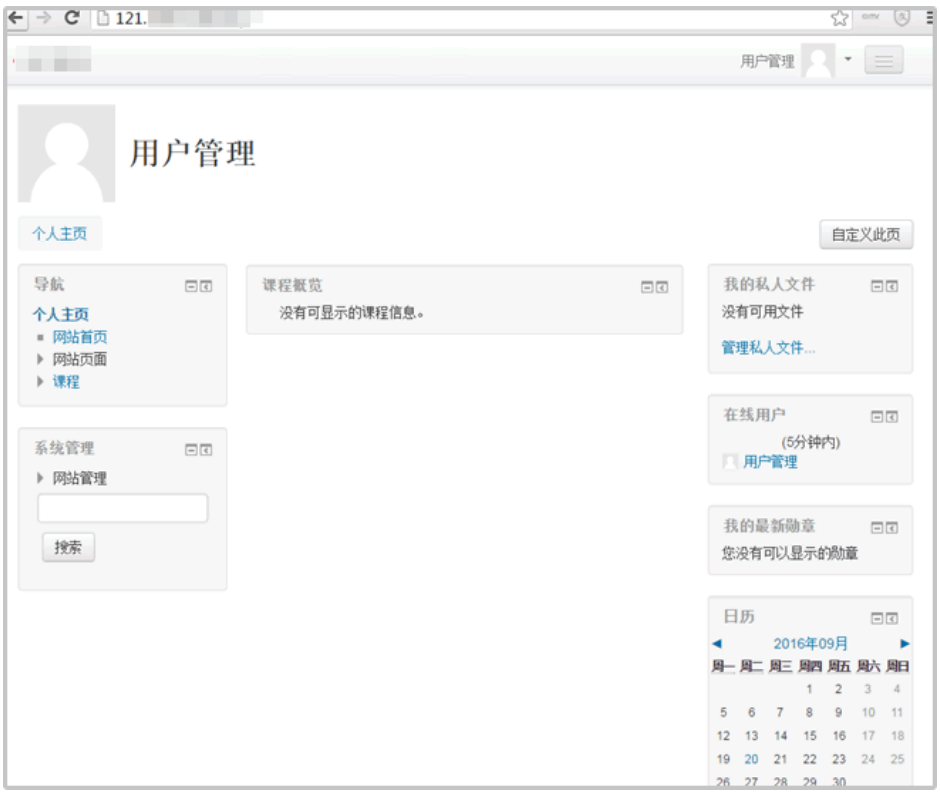
当安装页面底部出现 **继续** 时，说明已经完成安装。单击 **继续**。



按要求设置 Moodle 系统的登录信息后，单击 **保存更改**。



安装完成，自动进入管理后台首页。



绑定域名

用户可以使用公网 IP 地址访问您的站点。如果您希望用户使用域名访问您的站点，您应先将您的域名解析到公网 IP 地址。

如果需要在服务器里绑定域名，按以下步骤操作：

> 远程连接 Linux 实例后，输入命令 vim /etc/httpd/conf/httpd.conf 打开配置文件，找到 Servername 选项。
>
> 将localhost改为 www.yourdomain.com 即可。其中，www.yourdomain.com 必须替换为您自己的域名。

更多开源软件尽在云市场：https://market.aliyun.com/software。

快速搭建 phpwind 论坛系统

phpwind 是采用 PHP + MySQL 方式运行的开源社区程序。轻架构，高效率简易开发，帮助您快速搭建并轻松管理社区站点。phpwind 提供了 2 款完全不同的版本，分别是拥有成熟功能、海量插件支撑的 phpwind 稳

定版（v8.7.1）和注重轻社区、高效、易开发的 phpwind 先进版（v9.0.1）。

本文档介绍如何使用云市场的 **PHPWind论坛系统（含智慧云虚机面板）** 快速搭建论坛，包括：

- 安装并使用 phpwind 先进版
- 安装并使用 phpwind 稳定版
- 获取商业授权

适用对象

适用于要搭建论坛的 **经典网络** 用户。

安装并使用 phpwind 先进版

前提条件

您应该已经拥有已经备案的域名。如果没有备案，购买 **ECS 实例** 后，您应到阿里云备案中心备案，备案地址为：https://beian.aliyun.com。

操作步骤

安装并使用 phpwind 先进版包括以下几个步骤：

1. 购买安装了 **PHPWind论坛系统（含智慧云虚机面板）** 镜像的 ECS 实例
2. 登录 phpwind 论坛
3. 绑定域名

安装 PHPWind论坛系统（含智慧云虚机面板） 镜像

单击 PHPWind论坛系统（含智慧云虚机面板） 进入镜像详情页。

单击 **立即购买**，按提示步骤购买 ECS 实例，其中，**网络类型** 选择**经典网络**。

登录 phpwind 论坛

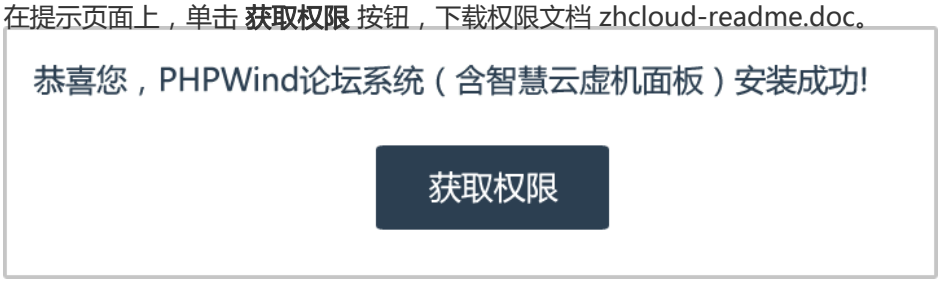
登录 ECS 管理控制台。

在左边导航栏里，单击 **实例**，进入 ECS 实例列表页。

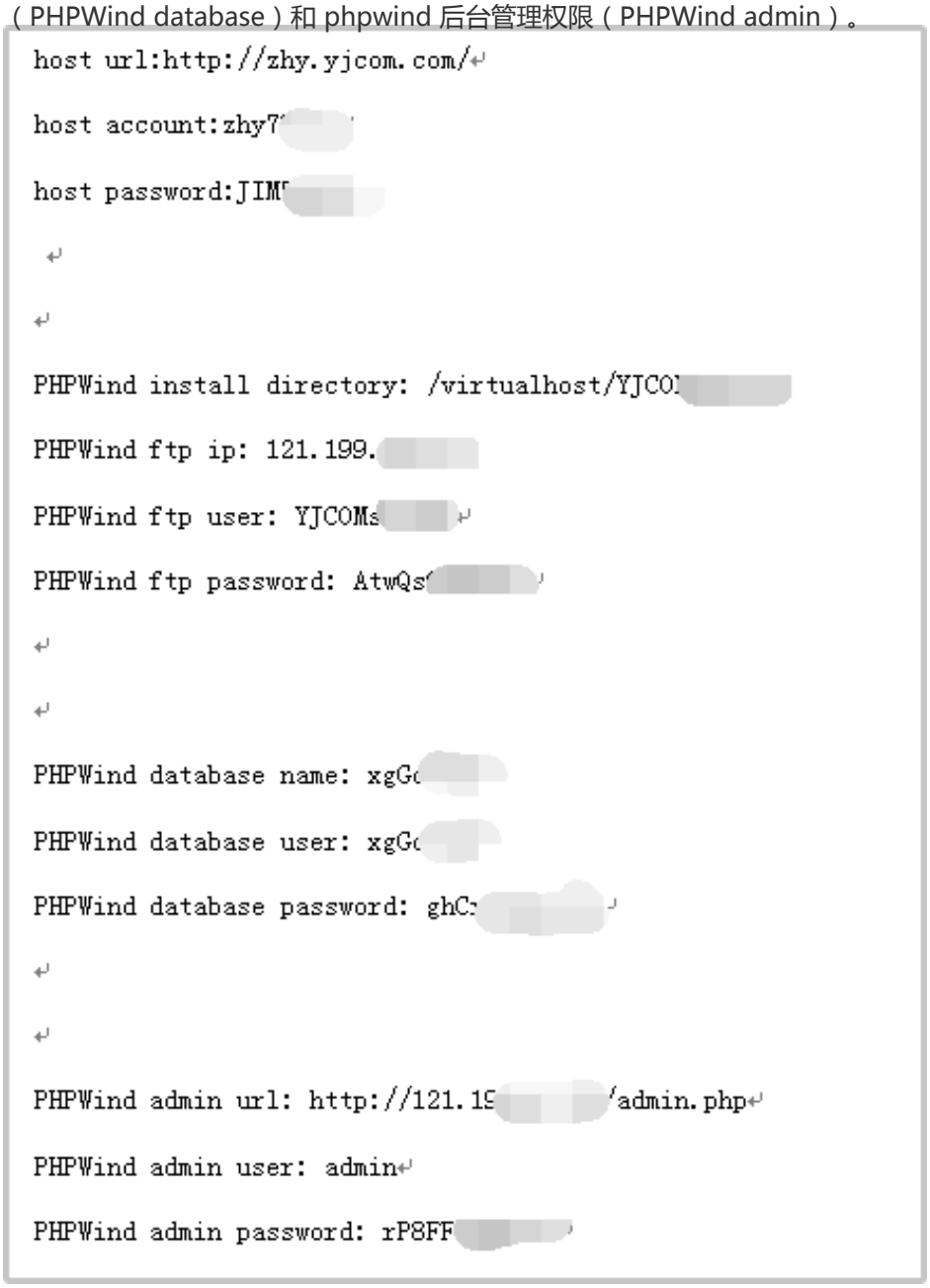
选择所购 ECS 实例所在的地域，并找到所购 ECS 实例，在 **IP 地址** 列获取该实例的公网 IP 地址。

在浏览器地址栏中，输入公网 IP 地址。屏幕上会显示提示页面。

在提示页面上，单击 **获取权限** 按钮，下载权限文档 zhcloud-readme.doc。



权限文档中包含了智慧云虚机面板权限（host)、FTP 权限（PHPWind ftp)、 MySQL 数据库权限（PHPWind database）和 phpwind 后台管理权限（PHPWind admin）。



在浏览器地址栏里，输入 http://实例公网 IP 地址/admin.php，进入 phpwind 的登录页面。

在 phpwind 登录页面上，输入在权限文档 zhcloud-readme.doc 中获取的 phpwind 后台管理的用户名和密码，再单击 **登录**。

登录 phpwind 后台，您就可以管理 phpwind 论坛了。

绑定域名

登录 智慧云虚机面板 。登录信息参见权限文档 zhcloud-readme.doc：

- host url 是指 **智慧云虚机面板** 的登录地址；
- host account 是指 **智慧云虚机面板** 的登录账号；
- host password 是指 **智慧云虚机面板** 的登录密码。

登录之后，如图所示。

单击 **域名绑定**，输入您的域名即可绑定。若您想禁止 IP 访问，删除含有 IP 地址的这条记录即可。

安装并使用 phpwind 稳定版

如果您想要安装 phpwind 稳定版（v8.7.1），按下面步骤操作。如果不想要安装 phpwind 稳定版（v8.7.1），可以略过这部分内容。

1. 新增网站
2. 使用 FTP 连接到新建的站点
3. 购买并下载安装phpwind 稳定版（v8.7.1）

前提条件

您应已经安装了 FTP 工具。在这里，我们以 FileZilla FTP 为例（下载地址为：https://www.filezilla.cn/download/client）。

您应该已经拥有已经备案的域名。如果没有备案，购买 **ECS 实例** 后，您应到阿里云备案中心备案，备案地址为：https://beian.aliyun.com。

新增网站

登录 智慧云虚机面板。

单击 **增加网站** ，进入新增站点的页面。

指定一个 FTP 账号密码，选择站点目录，单击 **下一步**。

启用JSP 选择 **否**，**启用PHP** 选择 **是**，**PHP版本** 选择 **PHP5.4** 版本。

类型 选择 **mysql**，指定数据库 **名称** 及 数据库 **密码**，单击 **下一步**。

输入一个 **域名**，单击 **执行创建**。

至此，您已经成功创建了一个新的站点。您可以查看新站点 FTP 和数据库的权限：

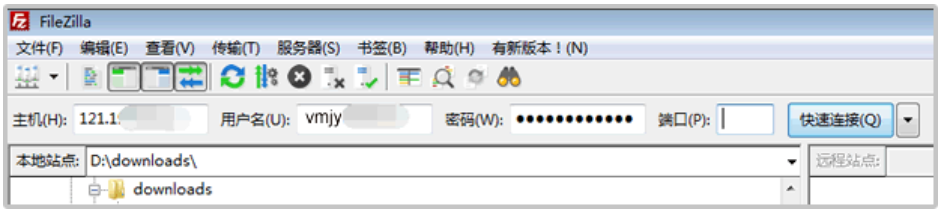
查看 FTP 的权限：进入面板首页，单击 **FTP 账号**，单击 **查看密码**，可以查看到新站点的 FTP 账号和 FTP 密码。

查看数据库权限：在 智慧云虚机面板 的 **数据库** 可以查看到。

使用 FTP 连接到新建的站点

启动 FileZilla FTP。

输入 ECS 实例的公网 IP 地址、FTP 账号、FTP 密码，单击 **快速链接**。



购买并下载安装 phpwind 稳定版（v8.7.1）

单击 phpwind 8.7.1安装包（UTF8） 进入程序下载详情页。确认信息后，单击 **立即购买**。

确认订单后，单击 **确认开通**。

开通成功后，单击 **管理控制台**。

在 **已购买的服务** 里，找到 phpwind 8.7.1安装包（UTF8），单击 **管理**。

在 **应用管理信息** 中，单击下载地址。

下载并解压安装包。

在 FTP 中，打开 phpwind 8.7.1安装包（UTF8）的 upload 目录，将 upload 目录下的程序上传到站点根目录下。

在浏览器中访问 **新增站点** 里绑定的域名。

在弹出对话框中，单击 **接受**。

单击 **下一步**。

输入数据库权限，单击 **下一步**。

完成安装，进入 phpwind 论坛首页。

获取商业授权

如果您的 phpwind 论坛需要商业授权，请按下列步骤操作。获取商业授权后，您可以合理合法地商业使用 phpwind 论坛程序。

如果不需要商业授权，可以省略这部分操作。

单击 phpwind商业授权（含软件包），进入phpwind商业授权详情页。

单击 **立即购买**，按步骤付 1 元购买。

登录 ECS 管理控制台。

进入 **云市场** > **已购买服务**，找到 phpwind 商业授权，单击 **管理**。

在 **应用管理信息** 中，单击管理地址。

单击 **点击这里下载**，得到验证文件 verify.html，通过 FTP 工具上传至站点根目录，再单击 **立即授权**。

输入您的站点域名，单击 **确定**，就完成商业授权了。

常见问题

301 重定向

登录 智慧云虚机面板。

找到 **自定义伪静态** > **自定义**，写入 301 重定向的 Nginx 规则，单击 **保存**。

下面以域名 yjcom.com 为例写 301 重定向 Nginx 规则。

方法 A：不使用www.yjcom.com域名访问网站时都 301 重定向到www.yjcom.com。

```
if ($host !=  'www.yjcom.com'  ) {
rewrite ^/(.*)$ http://www.yjcom.com/$1 permanent;
}
```

方法 B：使用yjcom.com域名访问网站时才 301 重定向到www.yjcom.com。

```
if ($host =  'yjcom.com'  ) {
rewrite ^/(.*)$ http://www.yjcom.com/$1 permanent;
}
```

注意：实际使用时，将以上代码中的域名替换为您自己的域名。

使用智慧云虚机面板需要对公网开放哪些端口？

- 确保您 ECS 实例所在的安全组已经对公网开放如下端口：21 端口、80 端口、3306 端口、1777 端口、8081 端口。
- 确保您使用的安全软件没有封掉 1777 端口。

更多开源软件尽在云市场：https://market.aliyun.com/software。

ECS搭建Microsoft SharePoint 2016

Microsoft SharePoint是Microsoft SharePoint Portal Server的简称。SharePoint Portal Server是一个门户站点，使得企业能够开发出智能的门户站点，这个站点能够无缝连接到用户、团队和知识，因此人们能够更好地利用业务流程中的相关信息，更有效地开展工作。SharePoint Portal Server 提供了一个企业的业务解决方案，它利用了企业应用程序集成功能，以及灵活的部署选项和管理工具，将来自不同系统的信息集成到一个解决方案中。本文主要说明如何在阿里云ECS上搭建Microsoft SharePoint 2016。使用的操作系统为Windows Server 2012 R2 DataCenter。

适用对象

适用于熟悉ECS，熟悉Windows Server系统的用户

基本流程

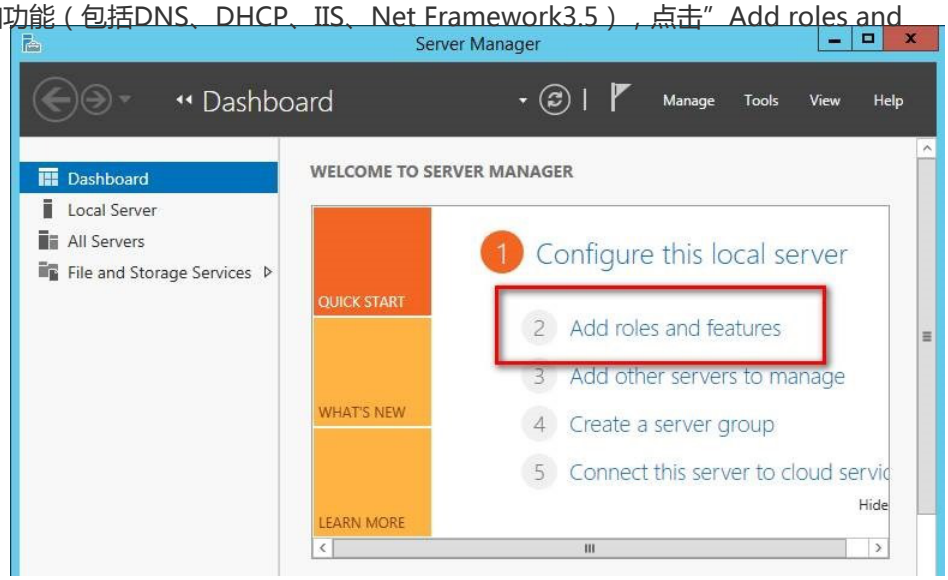
1. 添加AD、DHCP、DNS、IIS服务
2. 安装数据库SQL Server 2014
3. 安装SharePoint 2016
4. 配置SharePoint 2016

步骤一：添加AD、DHCP、DNS、IIS服务

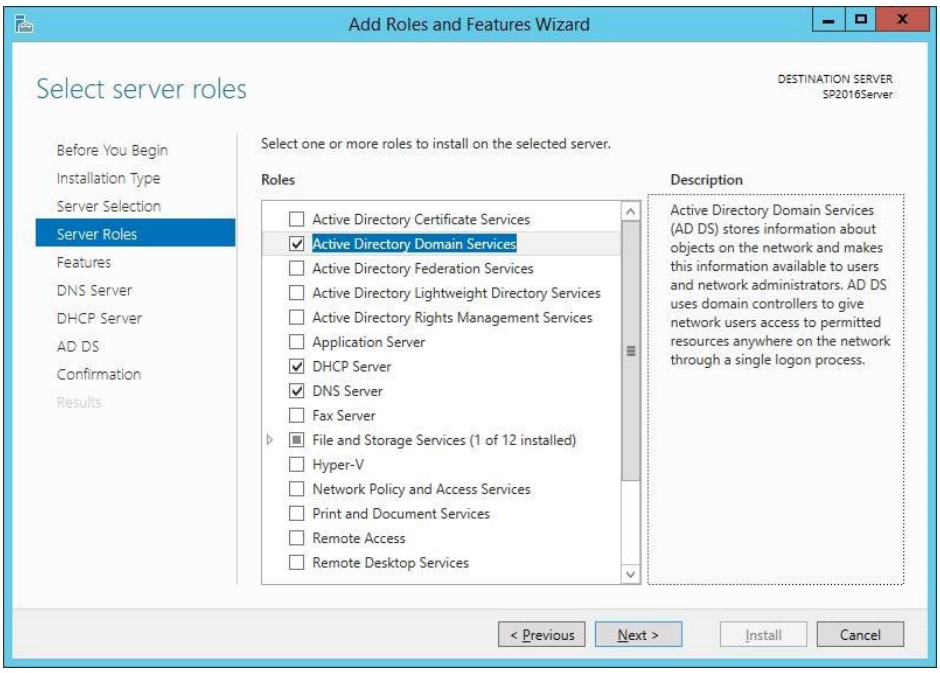
1、采购一台4核8G的云服务器，本实验环境操作系统版本为 Windows Server 2012 R2 数据中心版64位。

2、关闭IE增强的安全设置。

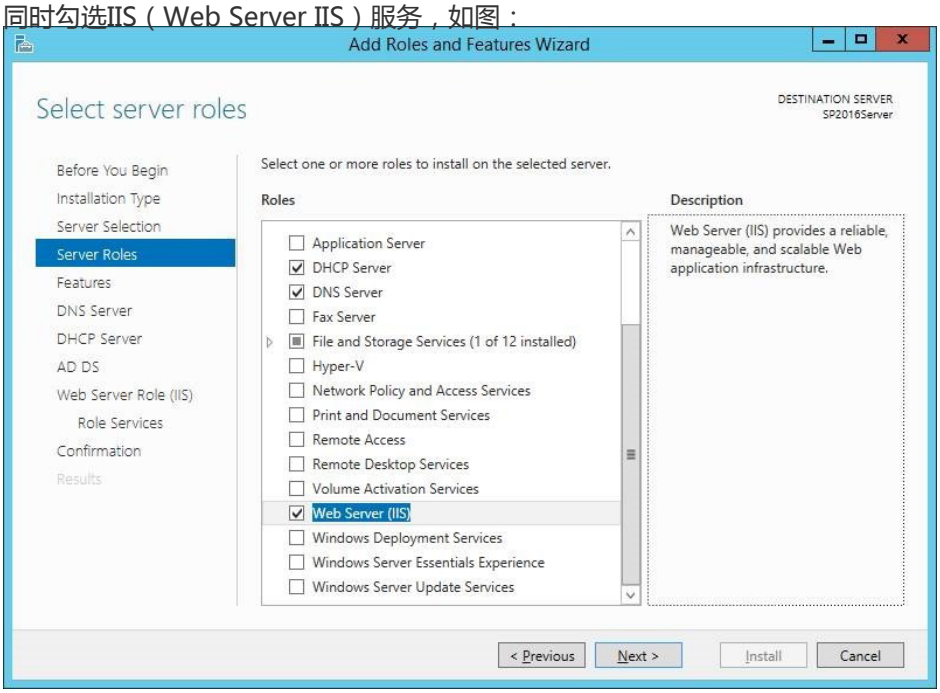
3、添加新的角色和功能（包括DNS、DHCP、IIS、Net Framework3.5），点击”Add roles and features”，如图：



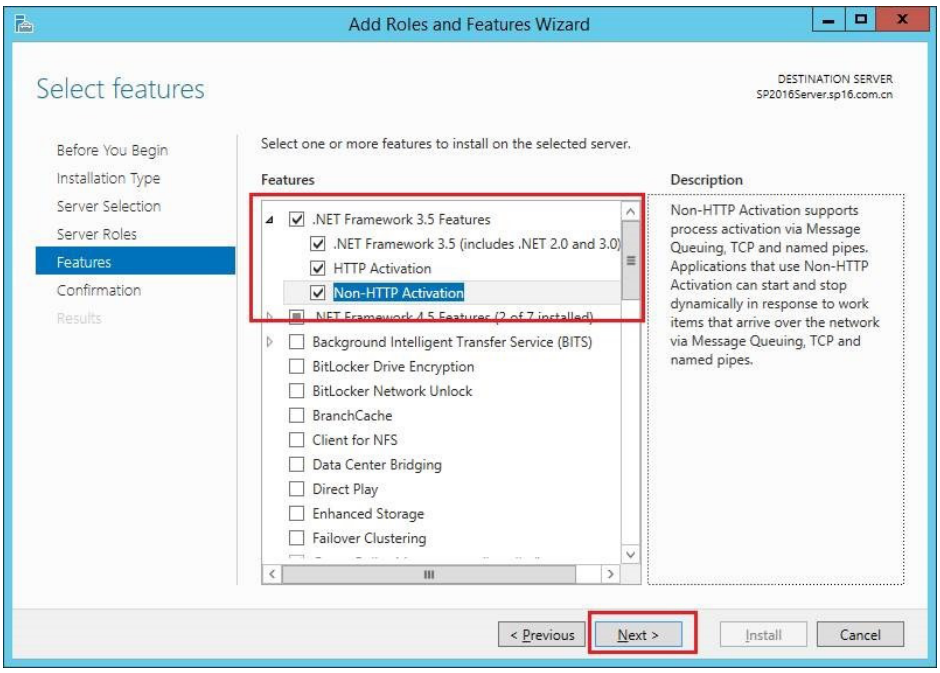
添加AD、DHCP、DNS服务，在这里勾选就可以，然后点击下一步，如图：



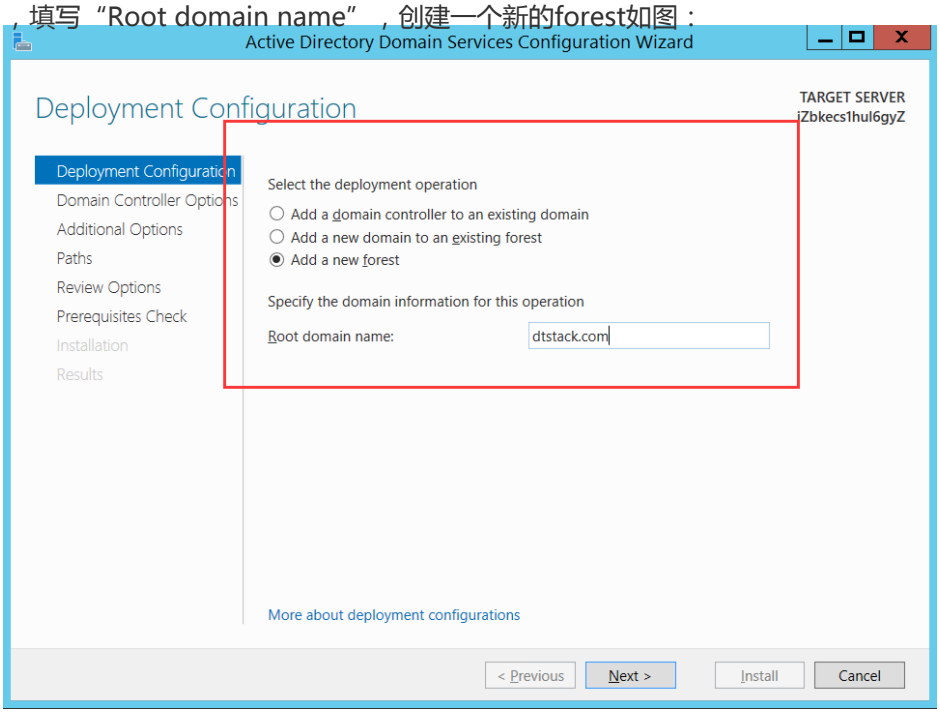
同时勾选IIS（Web Server IIS）服务，如图：



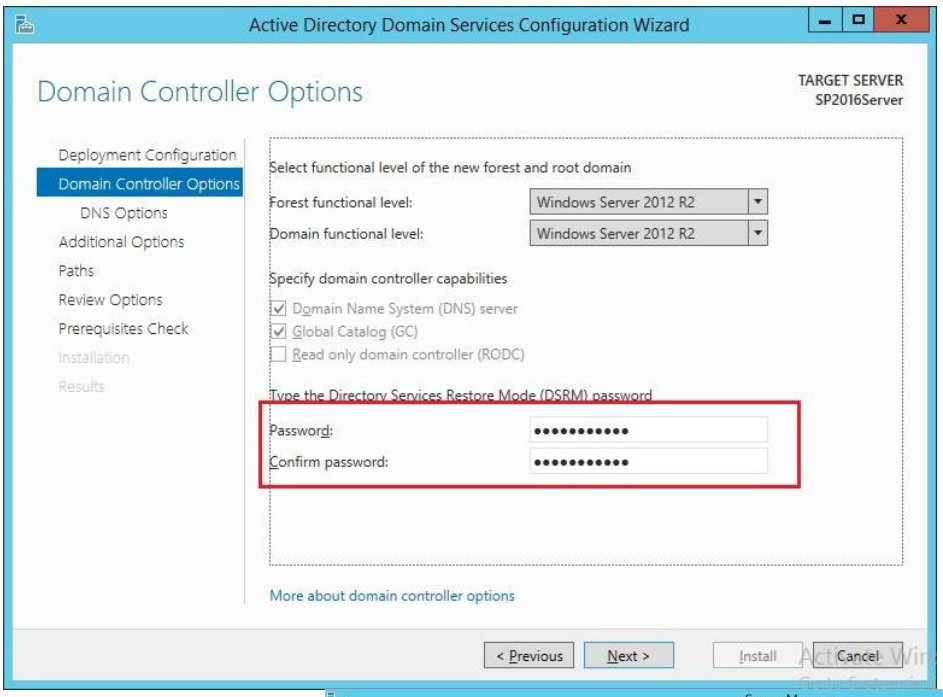
4、在Features的选项卡中，勾选Net Framework 3.5功能，如图：

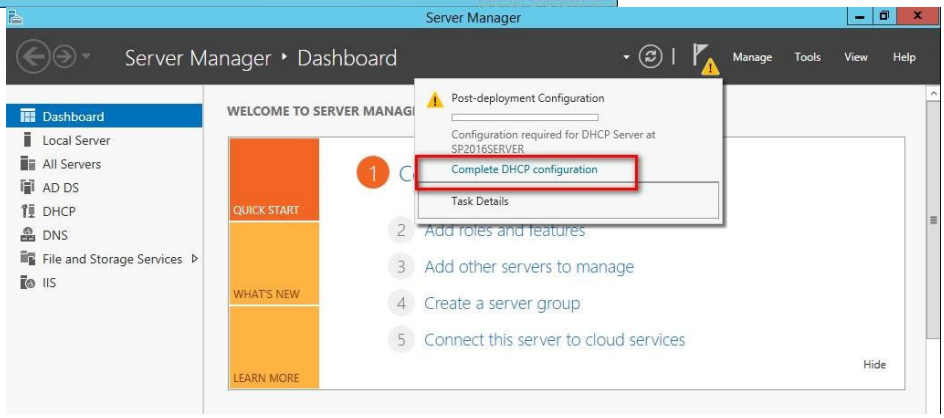


单击下一步，直至完成。6、安装完成，配置一下AD服务，点击红框里面的链接，因为没有已经存在的域环境，填写“Root domain name”，创建一个新的forest如图：



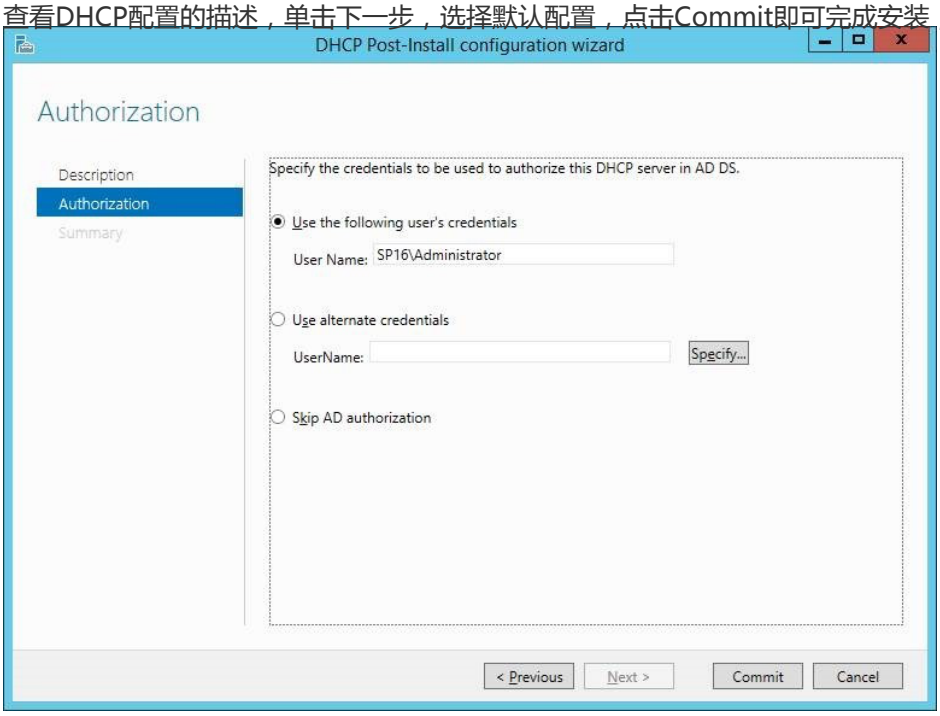
7、填写密码，完成后单击下一步直至完成。





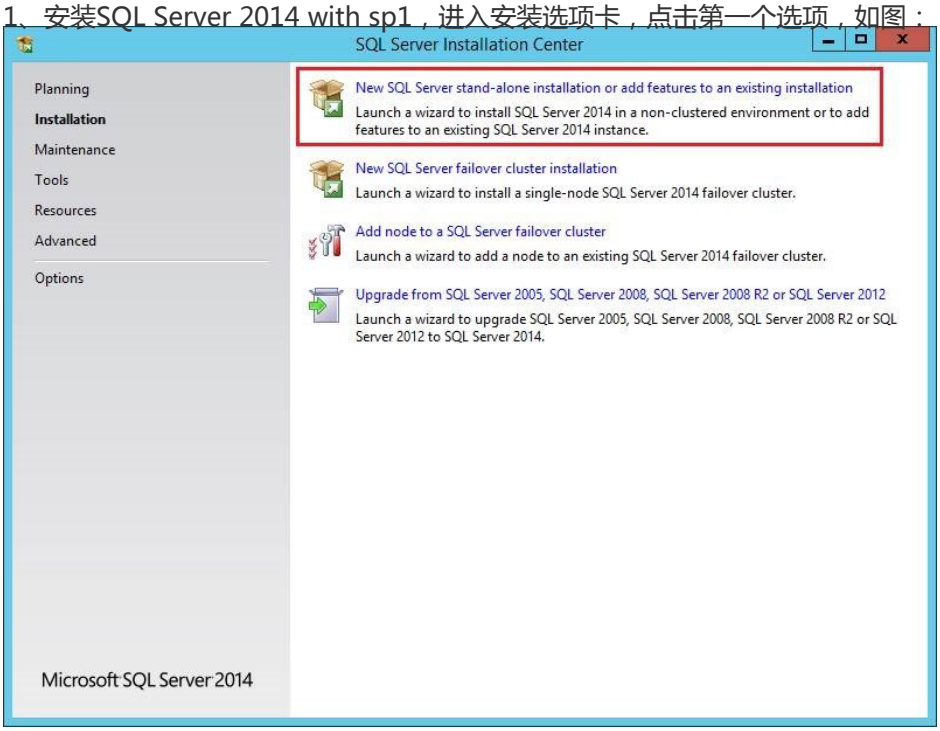
8、配置DHCP功能，如图：

查看DHCP配置的描述，单击下一步，选择默认配置，点击Commit即可完成安装，如图：

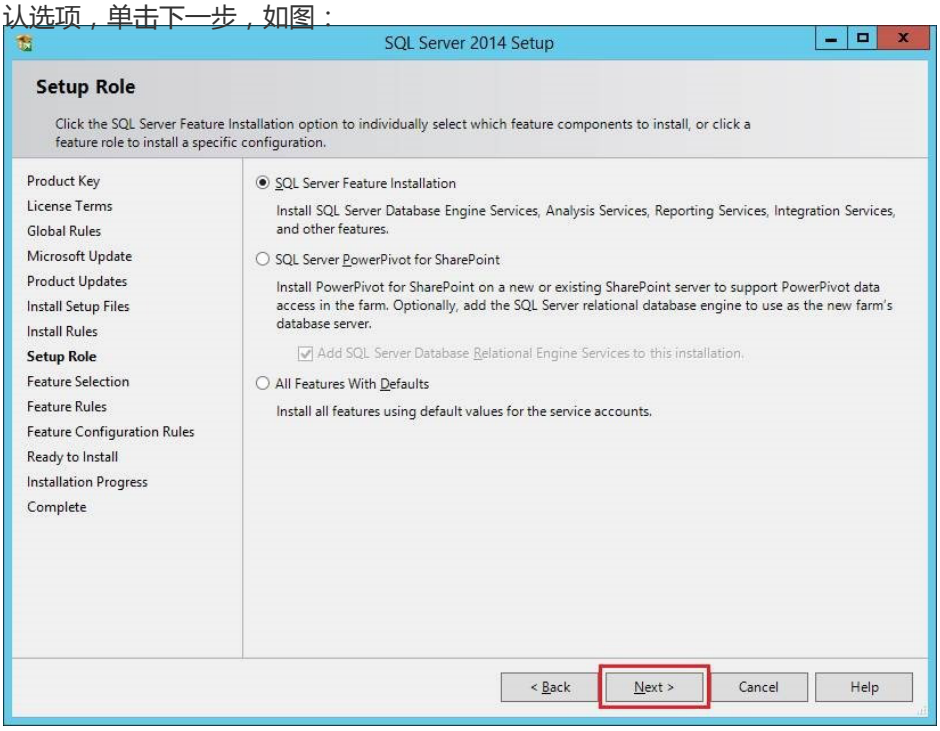


步骤二：安装SQL Server

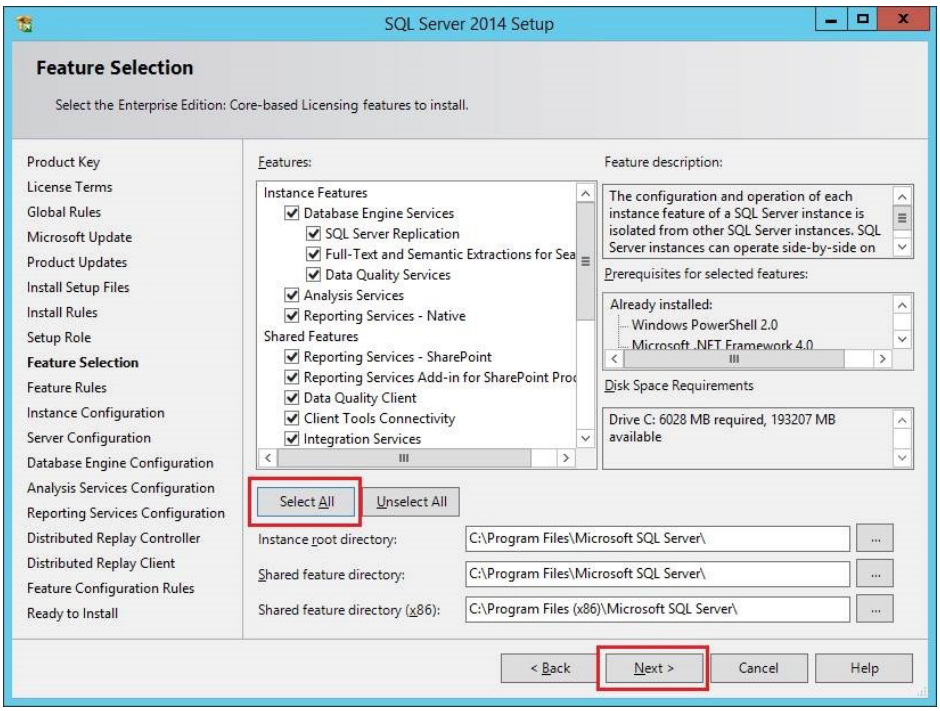
1、安装SQL Server 2014 with sp1，进入安装选项卡，点击第一个选项，如图：



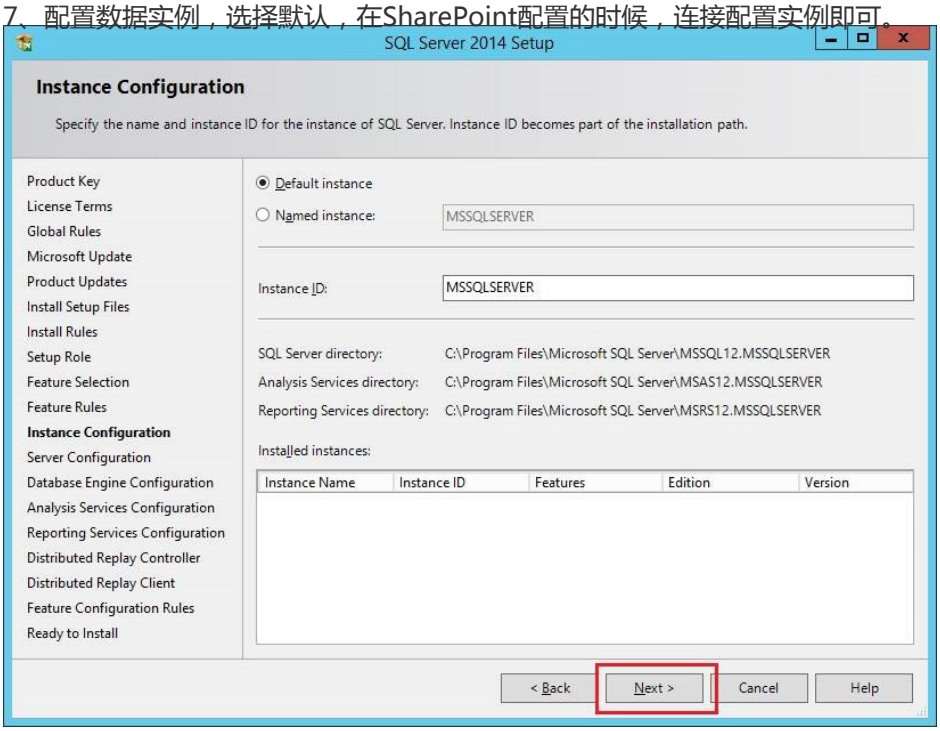
2、输入产品秘钥，单击下一步。3、接受License，单击下一步。4、完成安装检查，单击下一步。5、选择默认选项，单击下一步，如图：



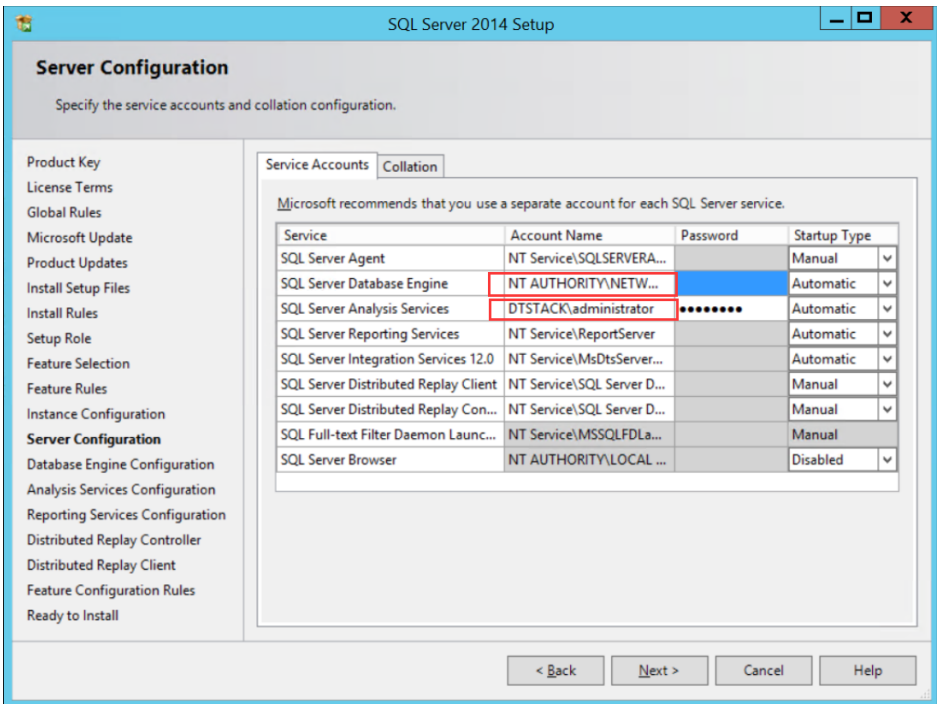
6、勾选全部功能，单击下一步。



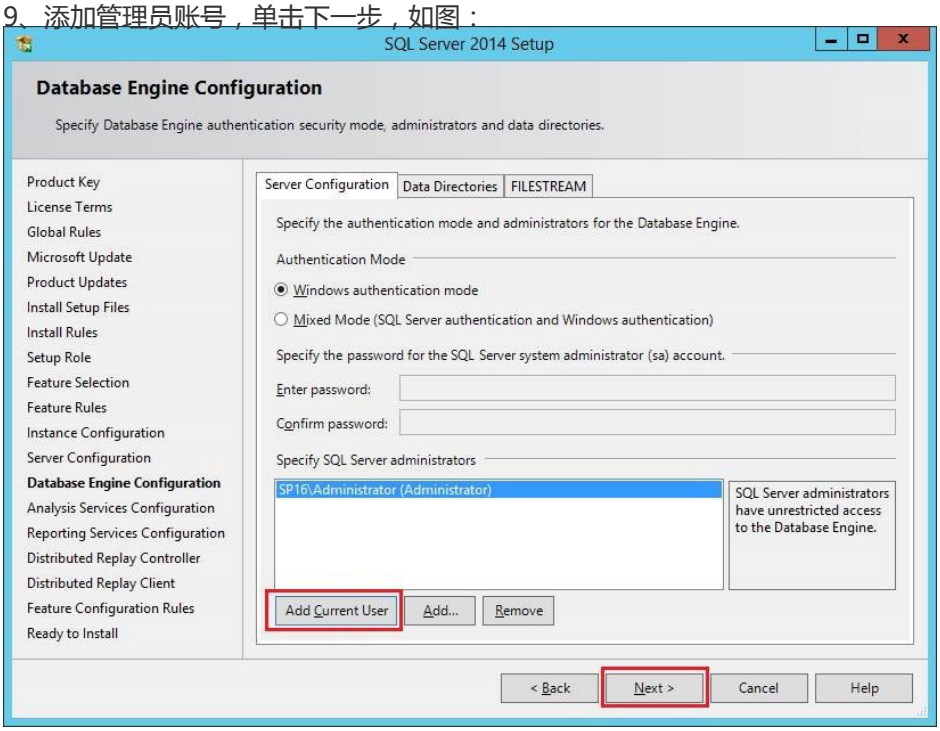
7、配置数据实例，选择默认，在SharePoint配置的时候，连接配置实例即可。



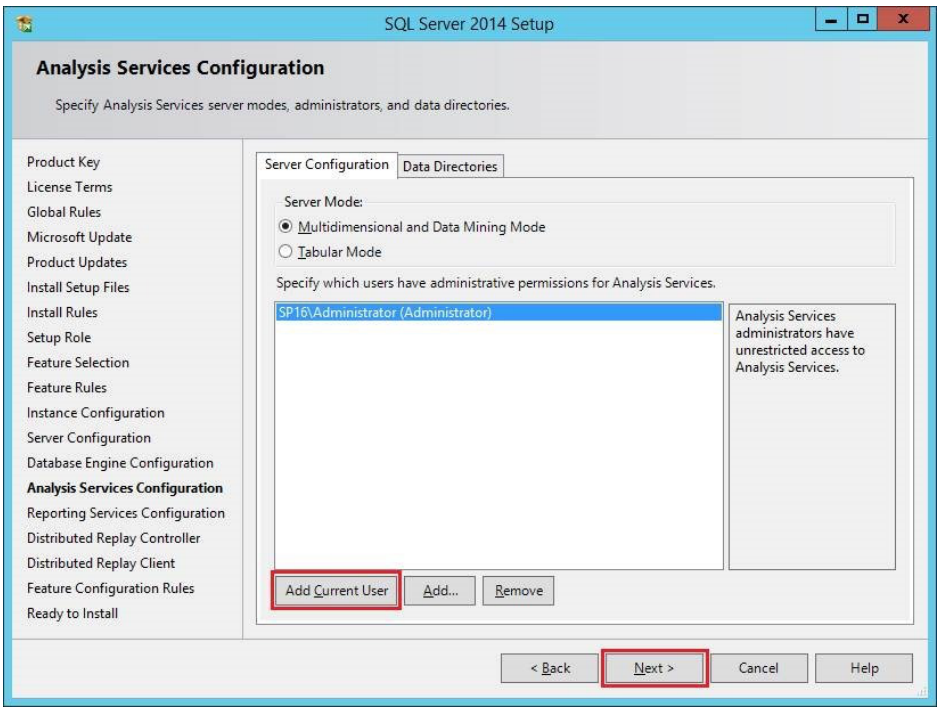
8、配置SQL Server Database Engine服务和SQL Server Analysis Services服务账号和密码。



9、添加管理员账号，单击下一步，如图：



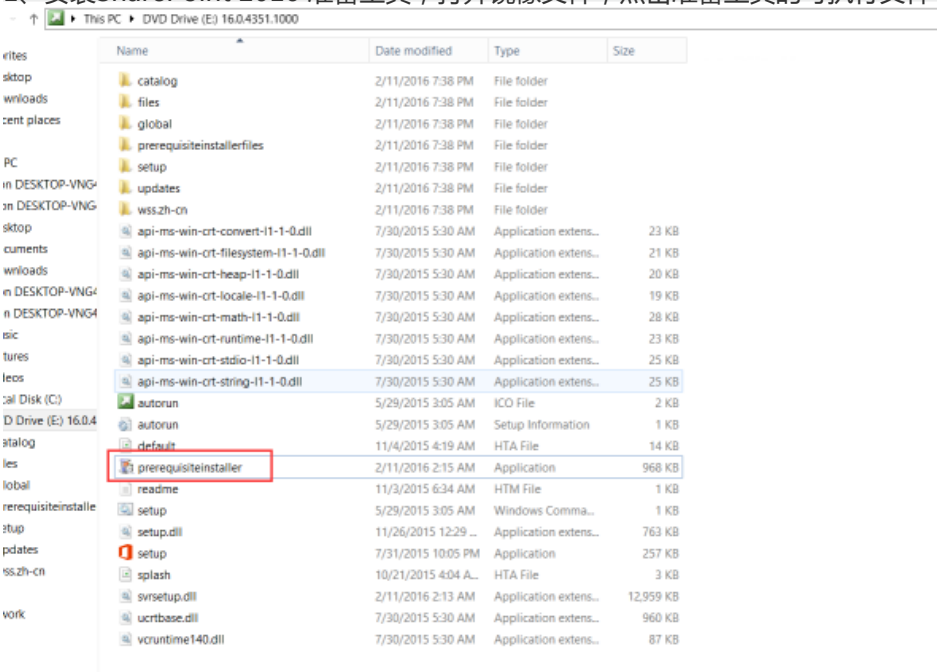
10、继续添加当前账号，单击下一步，如图：



接着单击下一步，直至安装完成。

步骤三：安装SharePoint 2016

1、安装SharePoint 2016 准备工具，打开镜像文件，点击准备工具的可执行文件，如图：

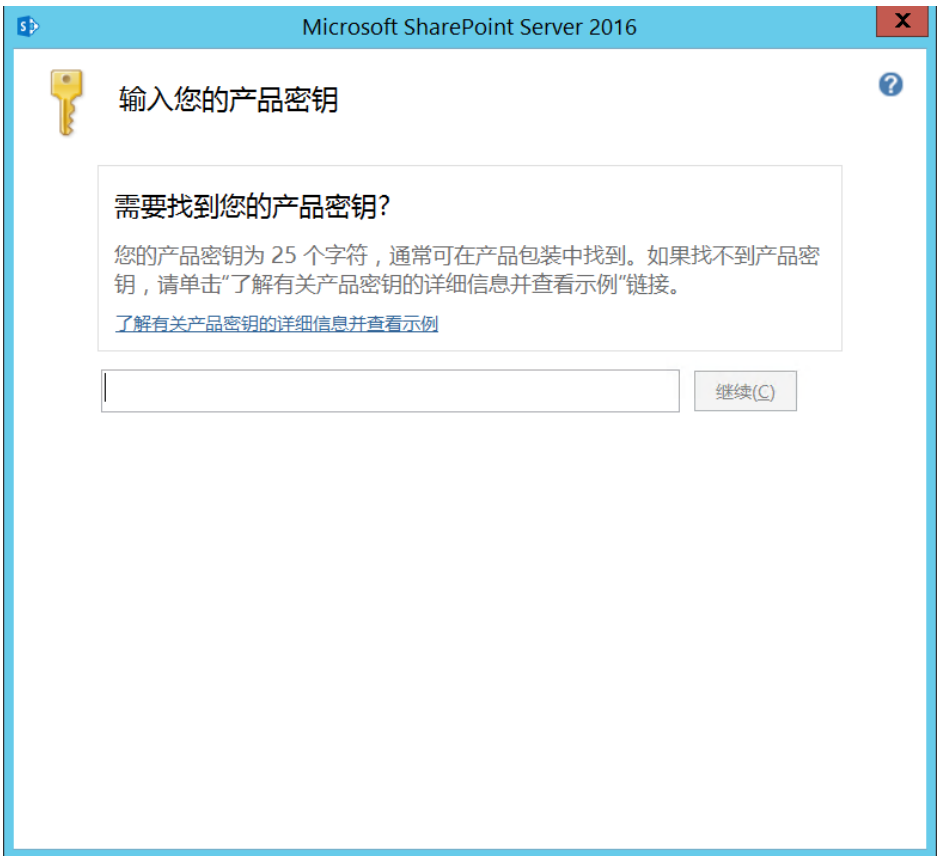


2、准备工具的向导，需要安装以下一系列软件

- 应用程序服务器角色、Web 服务器(IIS)角色
- Microsoft SQL Server 2012 Native Client
- Microsoft ODBC Driver 11 for SQL Server
- Microsoft Sync Framework Runtime v1.0 SP1 (x64)

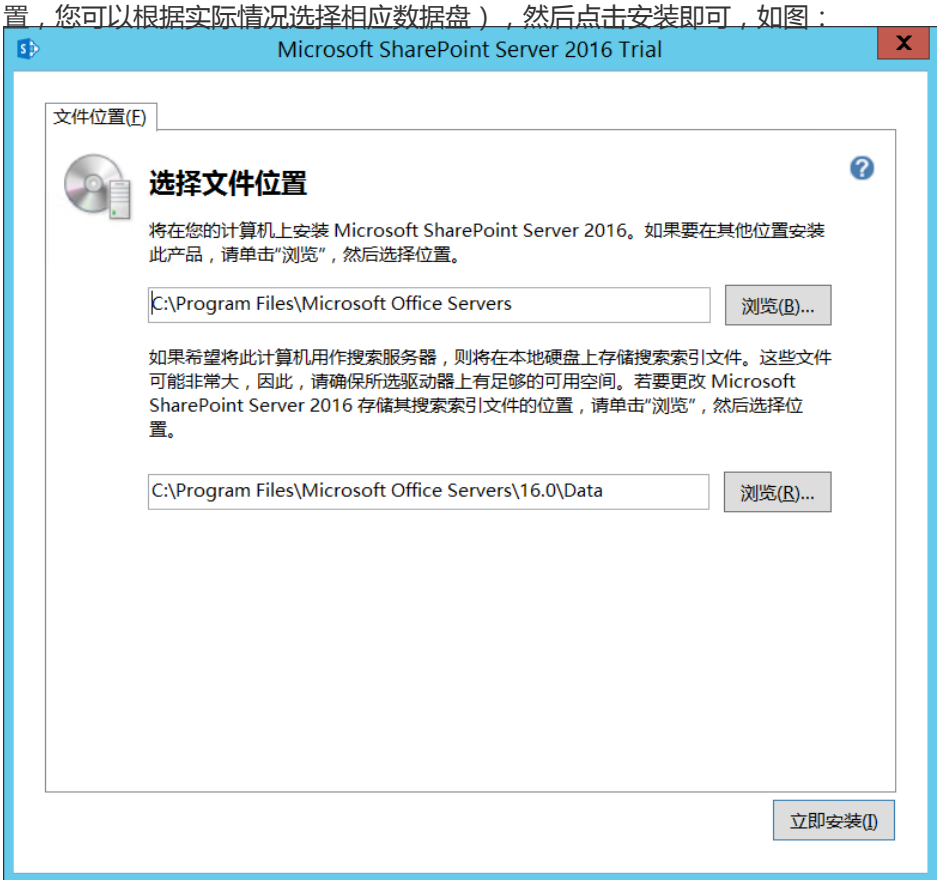
- Windows Server AppFabric
- Microsoft Identity Extensions
- Microsoft Information Protection and Control Client 2.1
- Microsoft WCF Data Services 5.6
- Microsoft .NET Framework 4.6
- Microsoft AppFabric 1.1 for Windows Server 累积更新包 1 (KB2671763)
- Visual Studio 2012 Visual C++ 的可再发行组件包
- Visual C++ Visual Studio 2015 的可再发行程序包

3、接受License，准备安装必备组件，直至安装完毕准备工具，可以安装SharePoint server 2016，（点击目



录下的Setup.exe）如图：

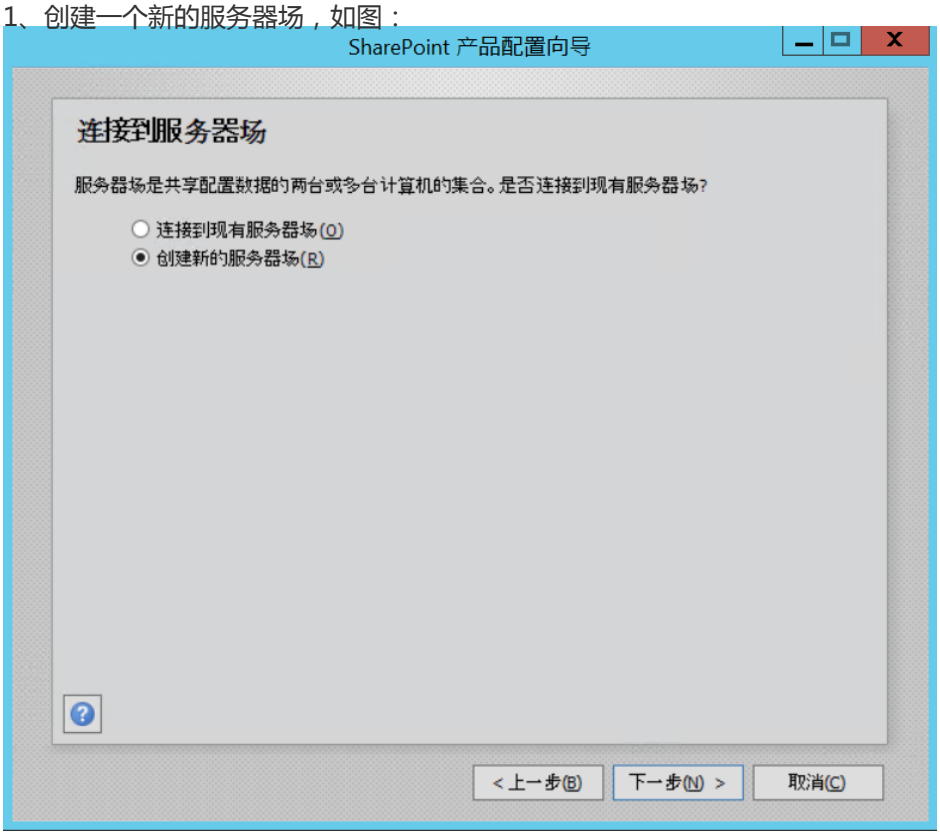
180天试用版Key：NQGJR-63HC8-XCRQH-MYVCH-3J3QR4、接受许可条款，选择安转目录（此处是默认设置，您可以根据实际情况选择相应数据盘），然后点击安装即可，如图：



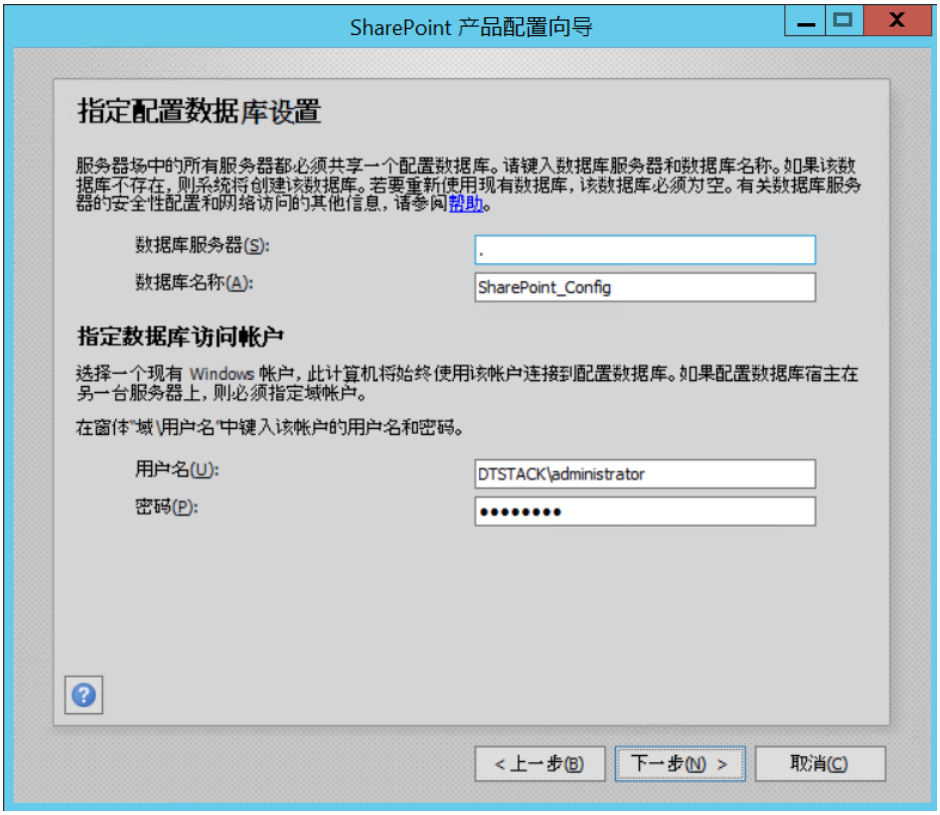
5、安装完成，提示我们关闭安装向导的同时，勾选立即运行SharePoint产品配置向导。

步骤四：配置SharePoint 2016

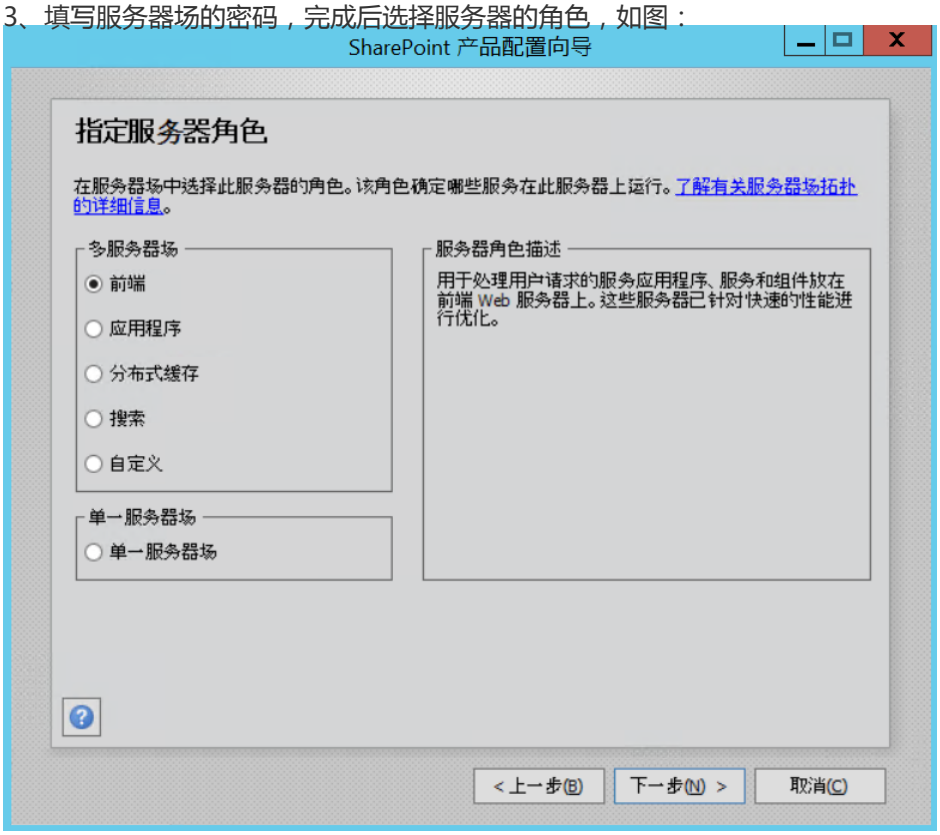
1、创建一个新的服务器场，如图：



2、配置数据库设置，由于数据库在本机，所以数据库服务器直接填写本机。



3、填写服务器场的密码，完成后选择服务器的角色，如图：



4、修改管理中心默认的端口号4466为10000，您可以根据实际情况修改

5、查看配置，如图：

6、配置成功后，可以打开SharePoint的管理中心，如图：

阿里云安装SharePoint 2016

系统环境

1.基础配置

Windows Server 2012

4C 8G （请根据真实环境设计架构以及购买服务器配置）

2.软件环境

SQL server 2012 express

Sharepoint 2016

AD

DNS

IIS

3.必备组件

Net Framework 3.5 (安装SQL server 需要.net Framework 3.5 支持)

使用“添加角色和功能”安装可能会安装失败，参考链接：
https://help.aliyun.com/knowledge_detail/38203.html

Sharepoint 必备组件可以参考微软官方文档 Sharepoint 安装时会提示安装依赖组件，如果依赖组件安装和安装失败，会导致 Sharepoint 无法安装

安装

1.搭建AD参考链接：https://help.aliyun.com/document_detail/52565.html

注意：客户端加域前记得修改SID，此篇文章只是为了介绍 Sharepoint 如何安装，使用的是一台服务器，所以将所有角色和功能都安装在一台服务器上（生产环境中千万不要将 SQL 和 AD以及 sharepoint 应用服务器服务器搭建在一起）

2.安装SQL SERVER 2012 Express

SQL SERVER 采用默认安装即可，由于是测试环境我这里使用的是express版本

注意：

a.express 版本访问默认不支持 tcp/ip 协议，需要手动开启。

b.express 版本默认（可能）没有管理控制台，需要单独安装 SQL 管理工具。

c.生成系统建议使用企业版数据库，express 版本相对企业版缺少部分功能。

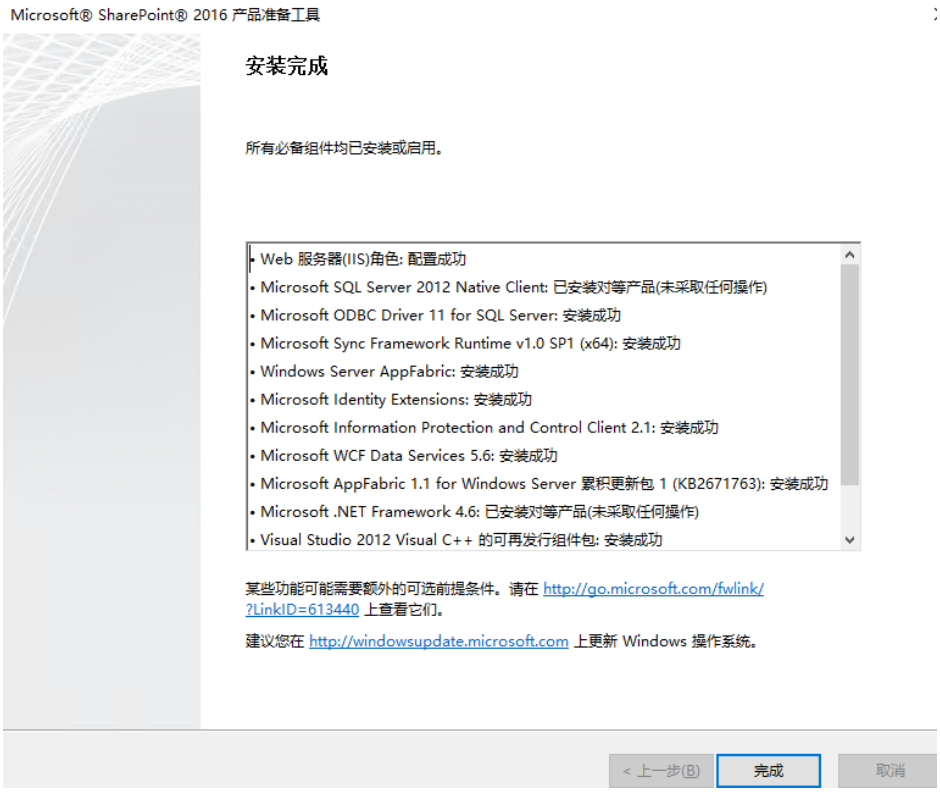
3.安装Sharepoint 2016

![1](http://docs-aliyun.cn-hangzhou.oss.aliyun-inc.com/assets/pic/53749/cn_zh/1495313483611/1.png)

安装 SharePoint 必备组件

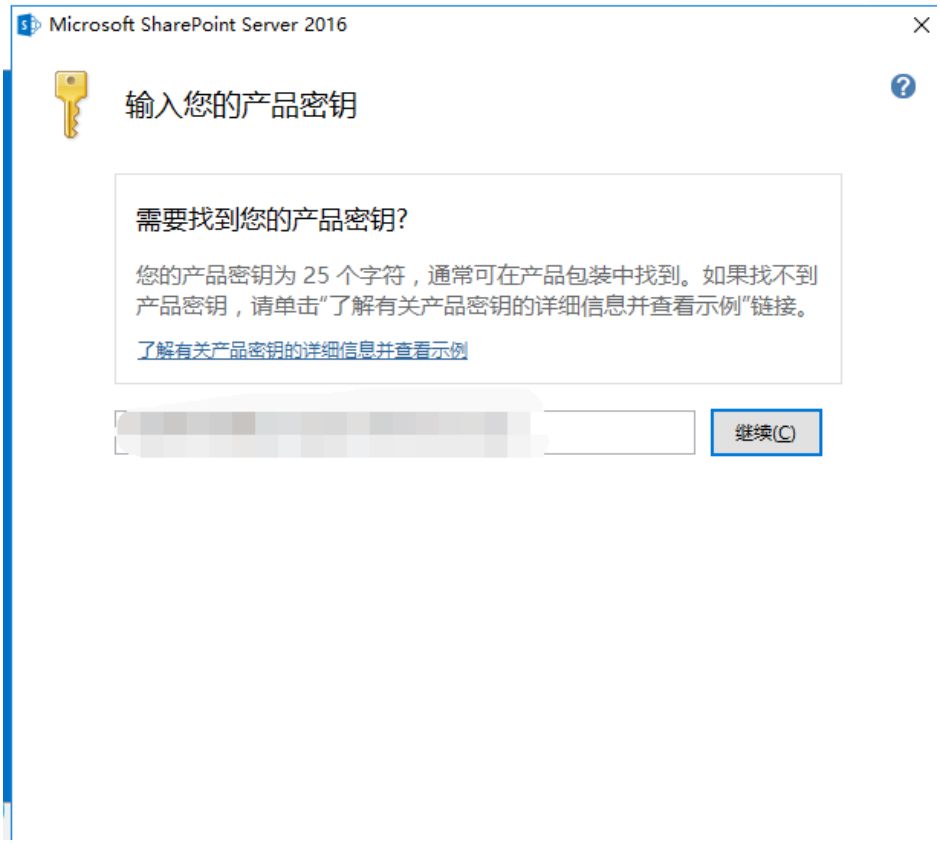
注：使用自动安装工具需要服务器能访问外网，如果访问不了，需要手动下载组件，然后使用命令安装，具体可以查看微软官方文档

必备组件安装成功。

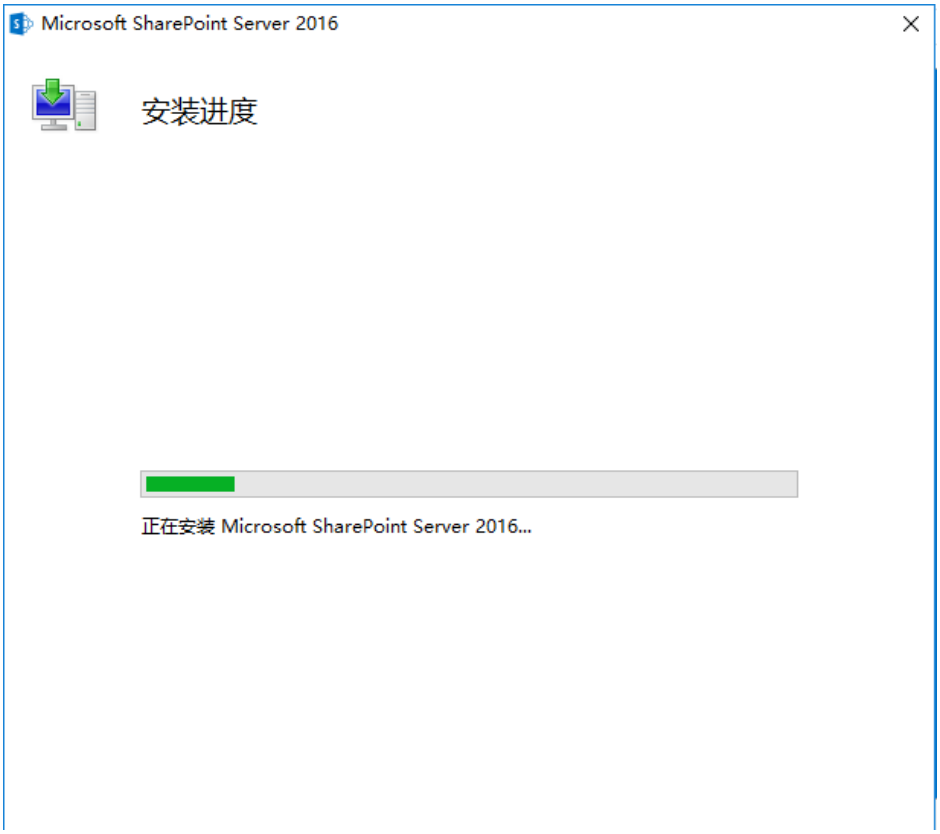


所有组件安装完成后，重启服务器，然后开始安装 Sharepoint。

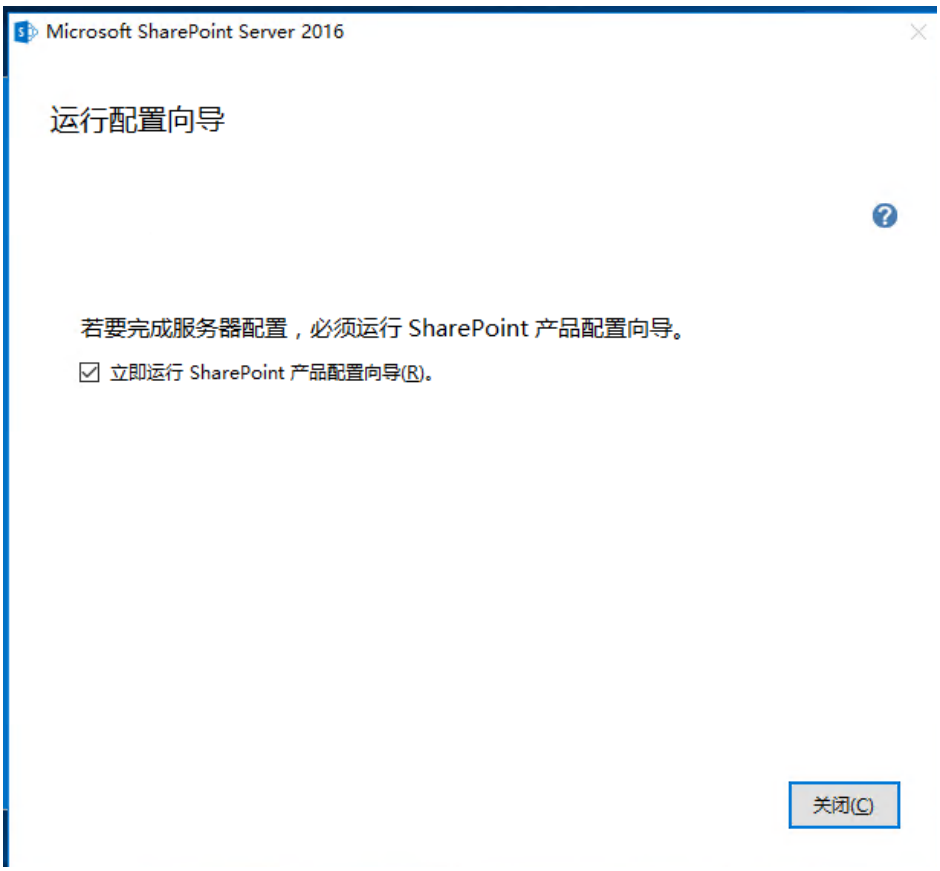
运行SharePoint2016安装向导，输入产品 Key。



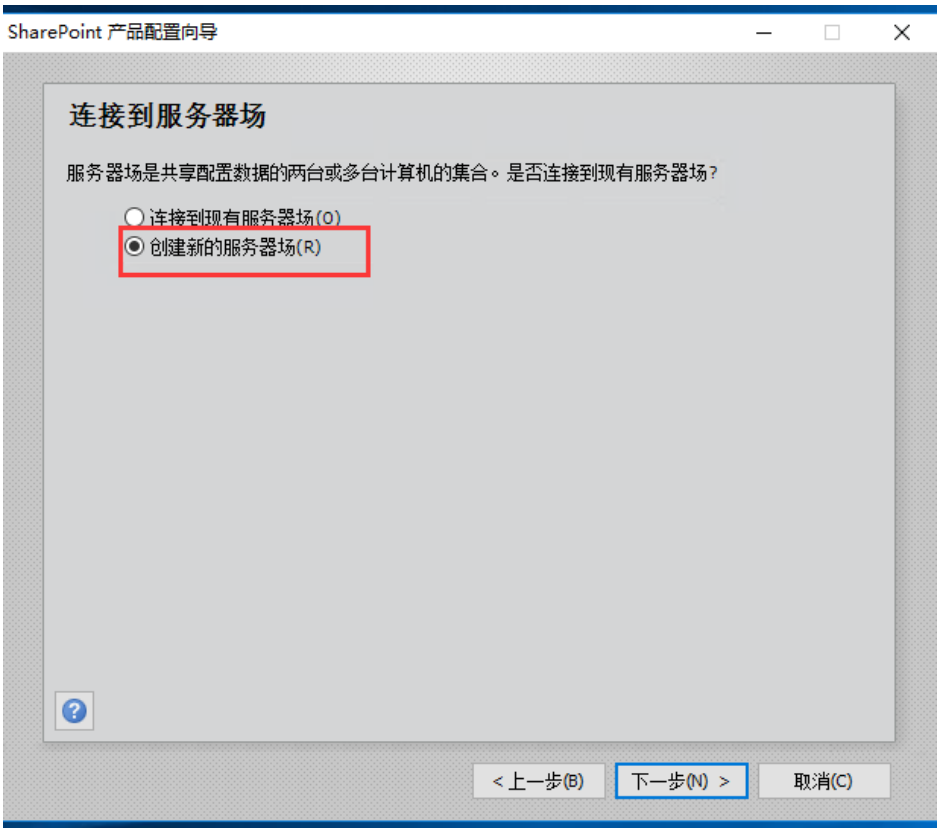
开始安装 ShrePoint 2016。



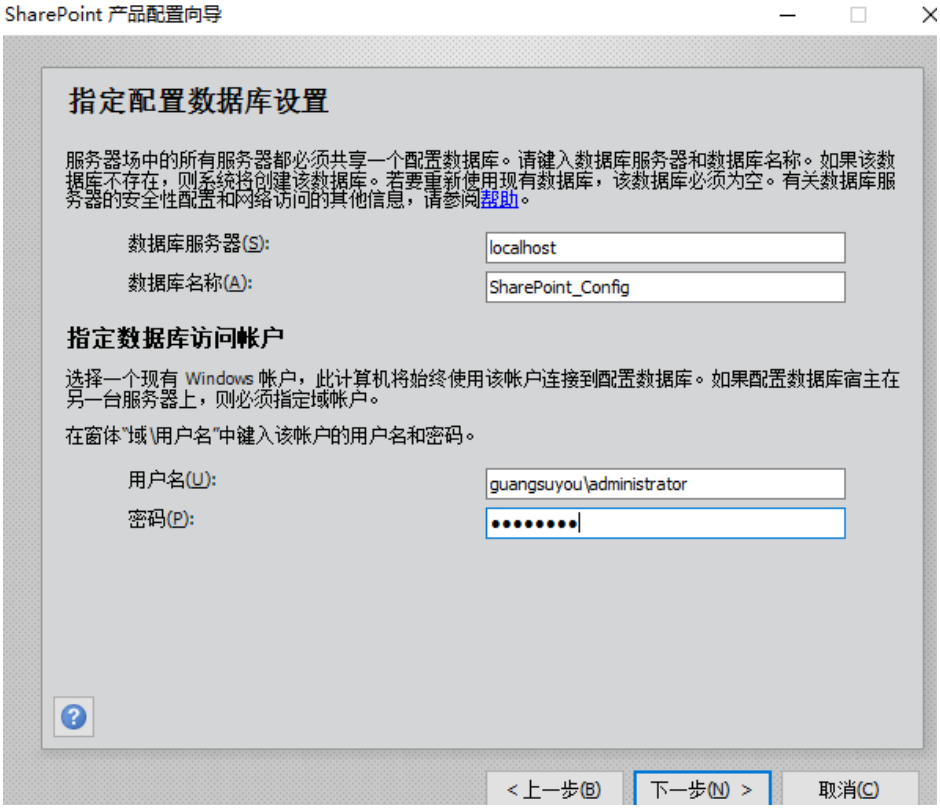
运行 SharePoint 配置向导。


Microsoft SharePoint Server 2016
运行配置向导
若要完成服务器配置，必须运行 SharePoint 产品配置向导。
立即运行 SharePoint 产品配置向导(R)。
关闭(C)

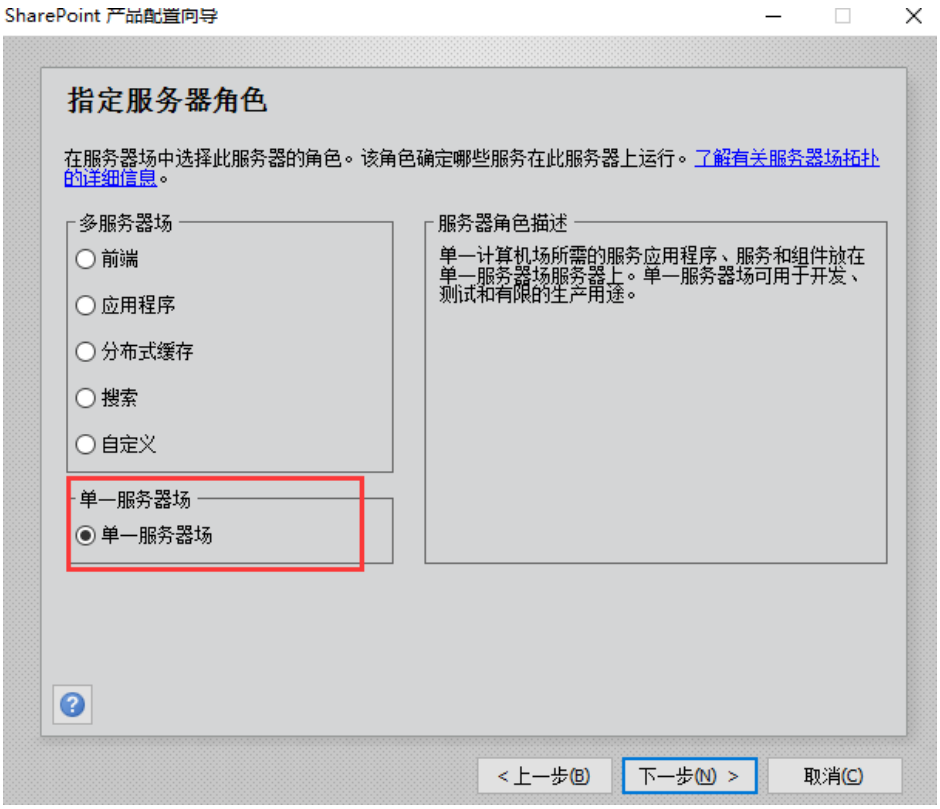
选择创建新的服务器场。


SharePoint 产品配置向导
连接到服务器场
服务器场是共享配置数据的两台或多台计算机的集合。是否连接到现有服务器场?
连接到现有服务器场(O)
创建新的服务器场(R)
< 上一步(B)
下一步(N) >
取消(C)

指定配置数据库设置。

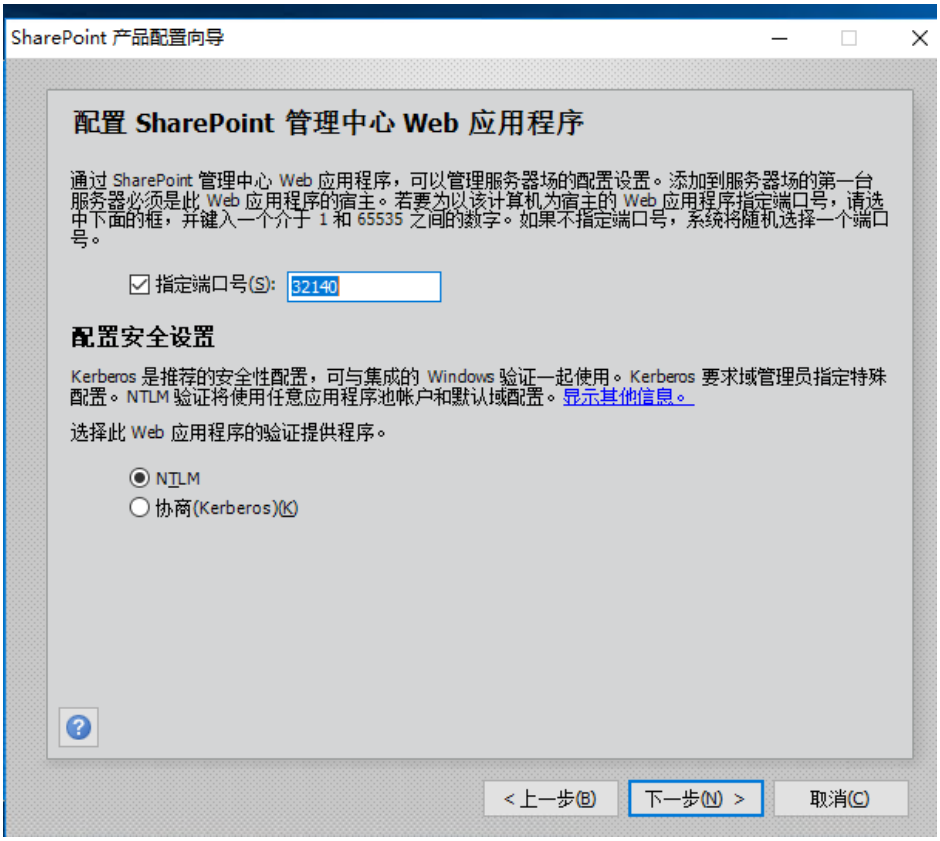


指定服务器角色。

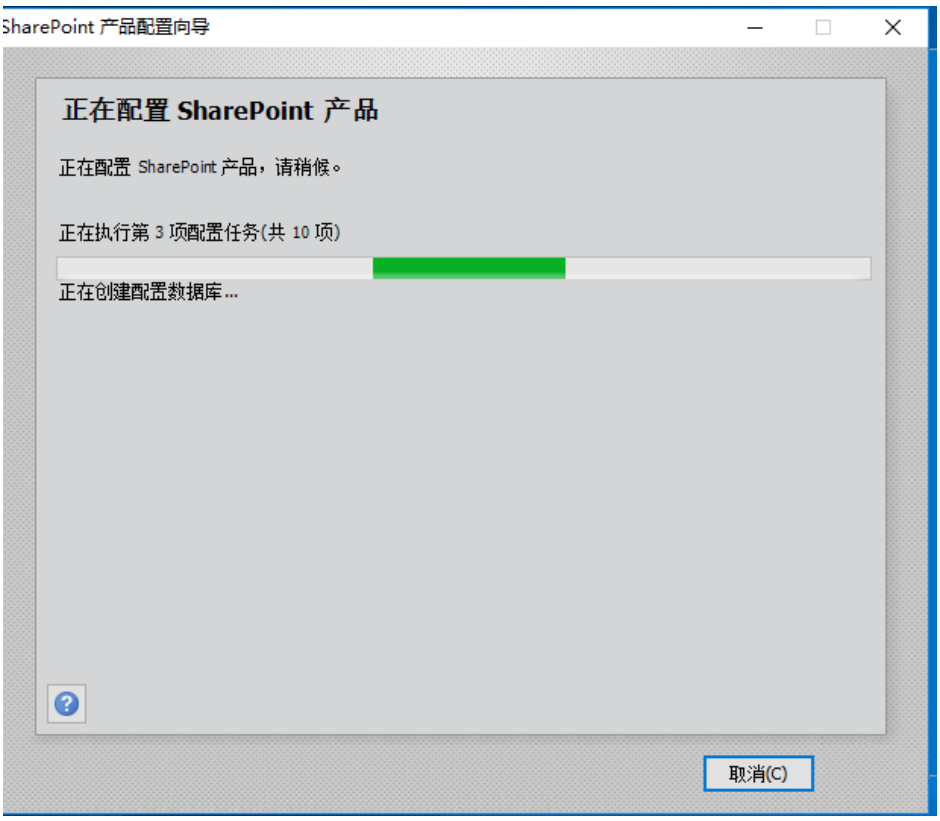


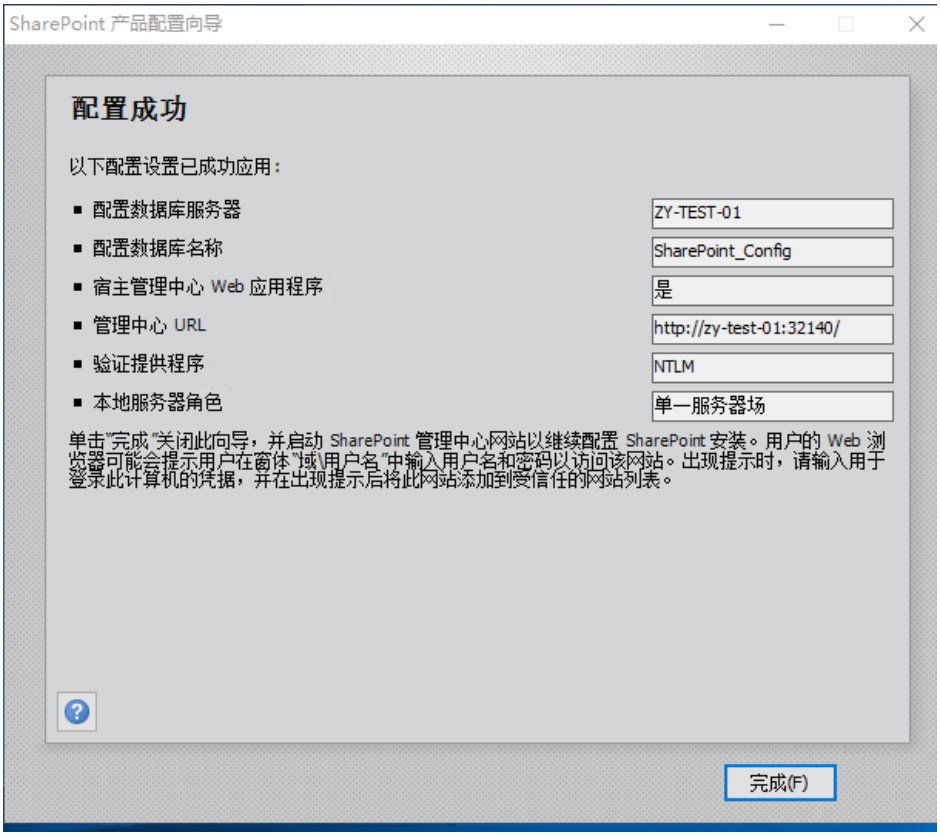
注： 我这里单台服务器做测试，因此选择单一服务器场。

指定 SharePoint 管理中心端口以及安全设置。



完成配置向导并开始安装。





到这里就全部安装完成，后面可以通过管理中心配置服务器场，配置服务器场时选择开启自己需要的服务，否则会造成不必要的内存开支。

搭建FTP站点

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

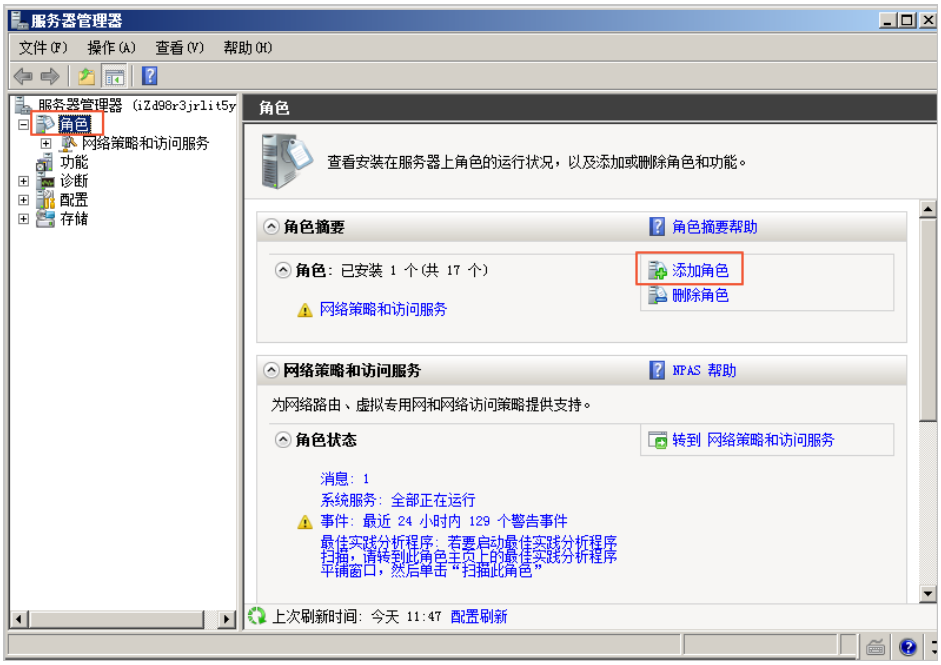
步骤一： 添加 IIS 以及 FTP 服务角色

在创建 FTP 站点前，首先需要安装 IIS 及 FTP 服务。

远程连接并登录到 Windows 实例。

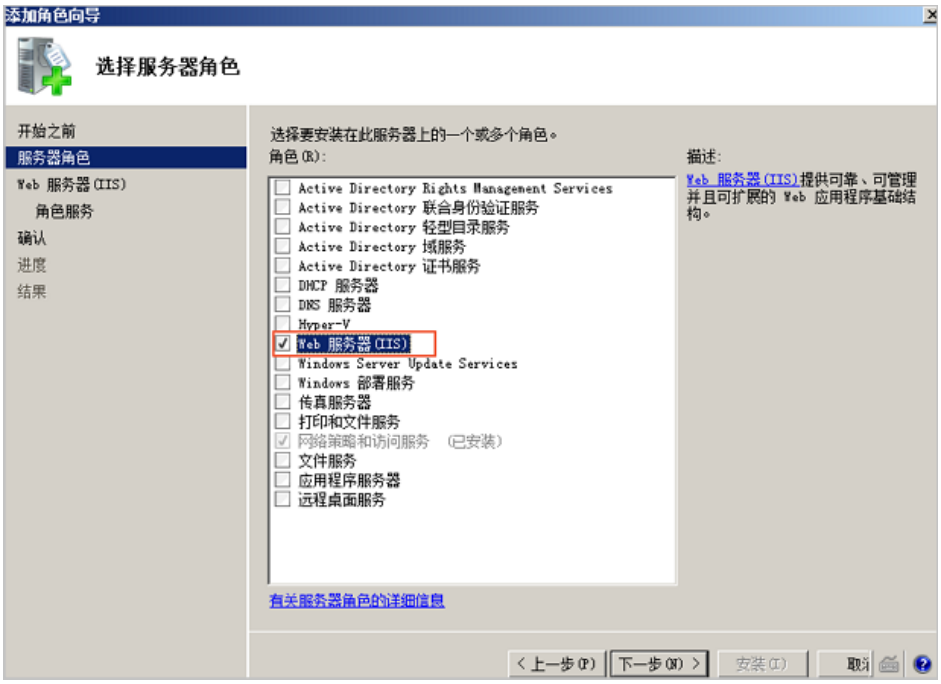
选择 **开始** > **所有程序** > **管理工具** > **服务管理器**。

单击 **角色**，然后单击 **添加角色**。

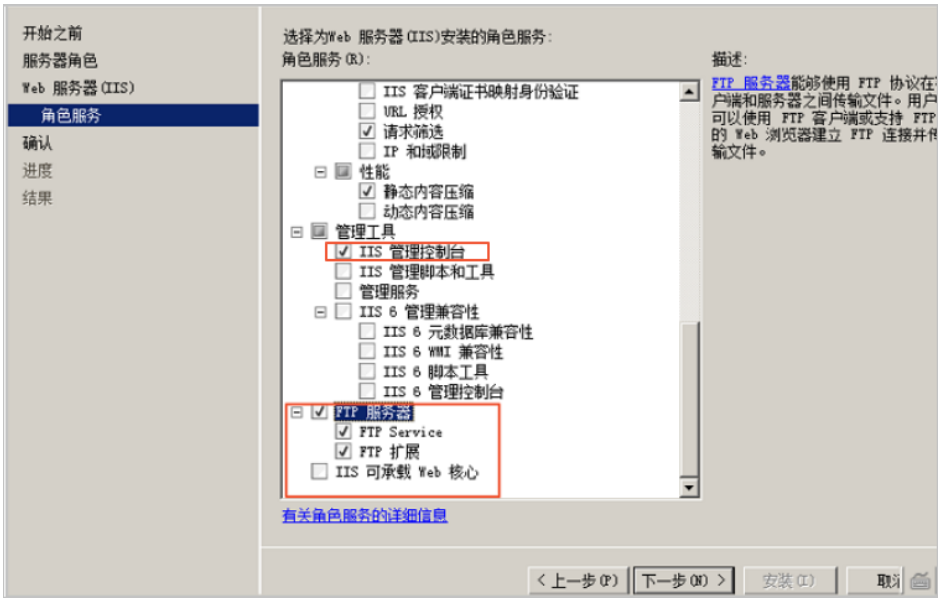


在弹出的对话框中，选择 **下一步**。

选择 **Web 服务器（IIS）**，然后单击 **下一步**。



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



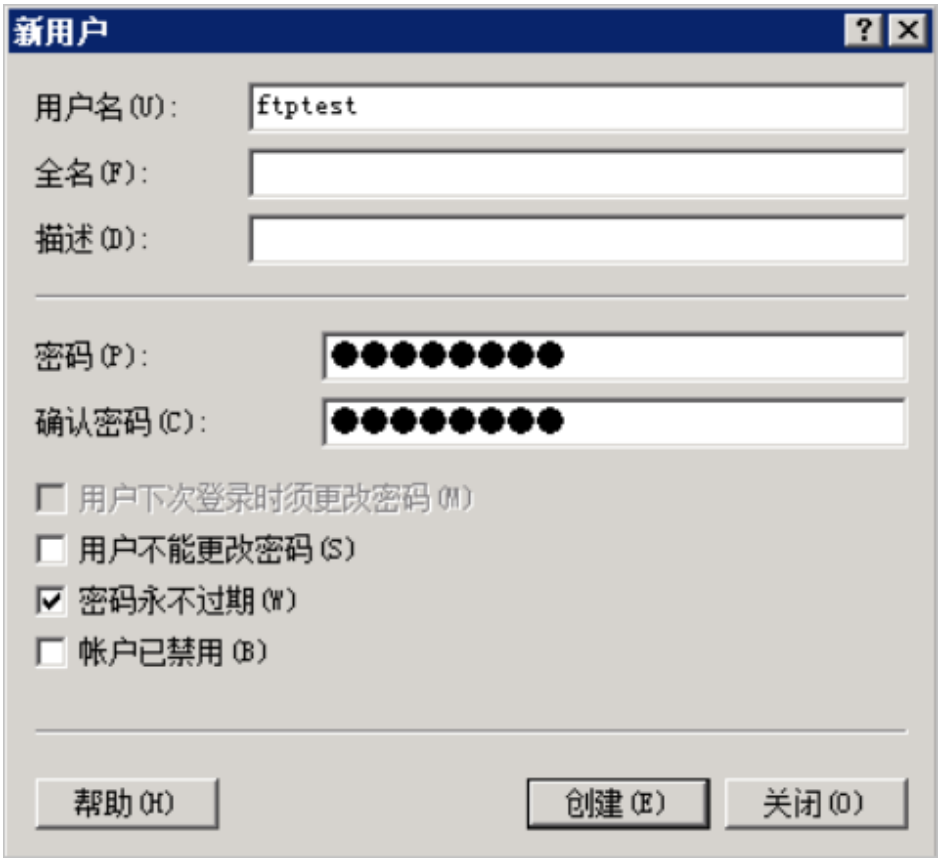
步骤二： 创建 FTP 用户名及密码

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。

选择 **开始** > **管理工具** > **服务器管理器**。

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。

注意：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。

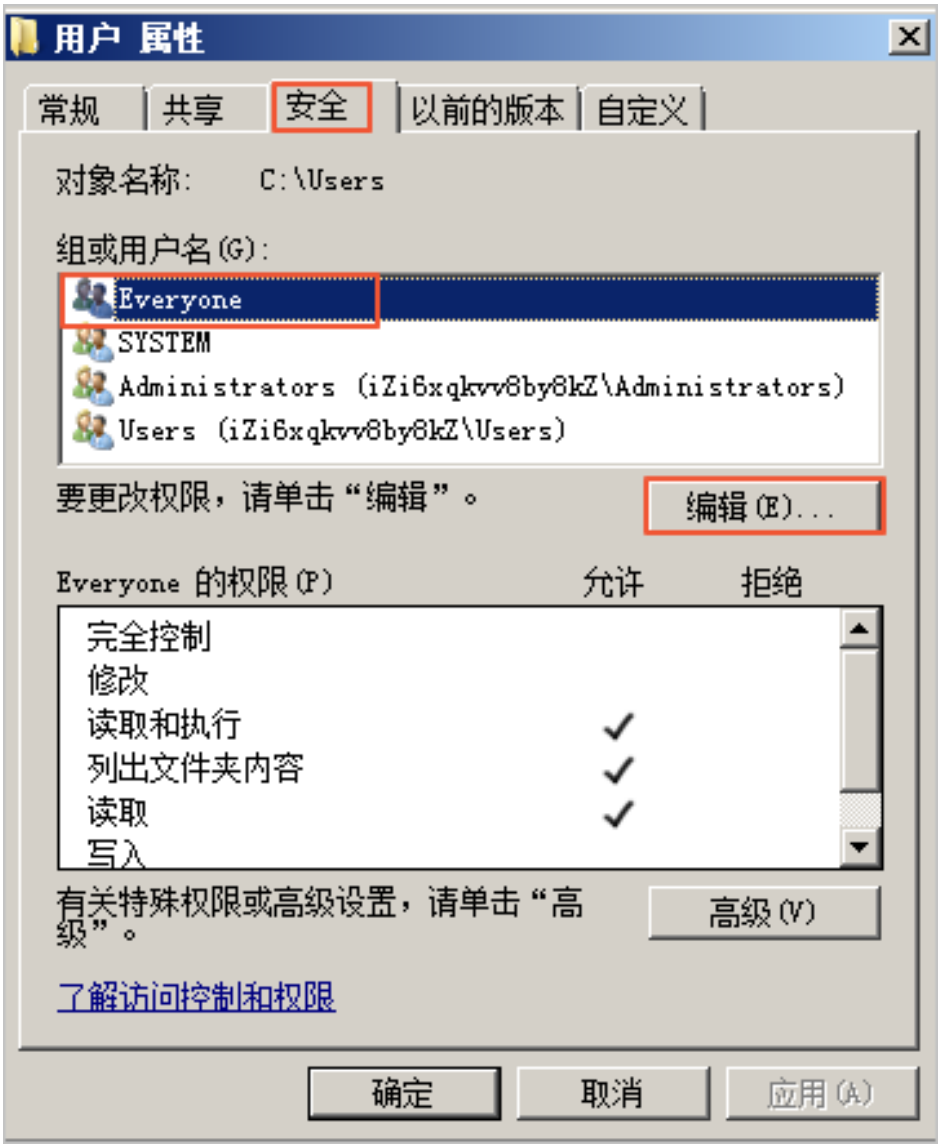


步骤三： 设置共享文件的权限

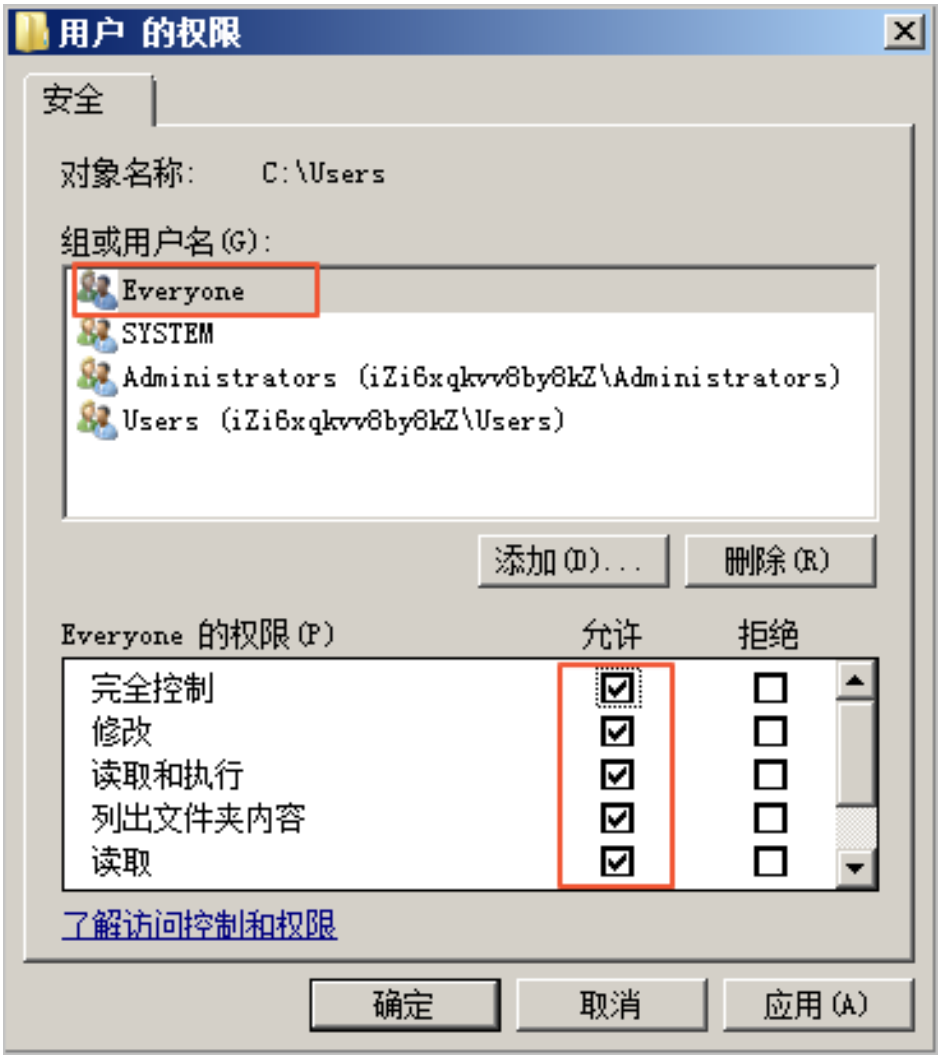
您需要为在 FTP 站点共享给用户的文件夹设置访问以及修改等权限。

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**。

单击 **安全**，选择 **Everyone**，然后选择 **编辑**。



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。

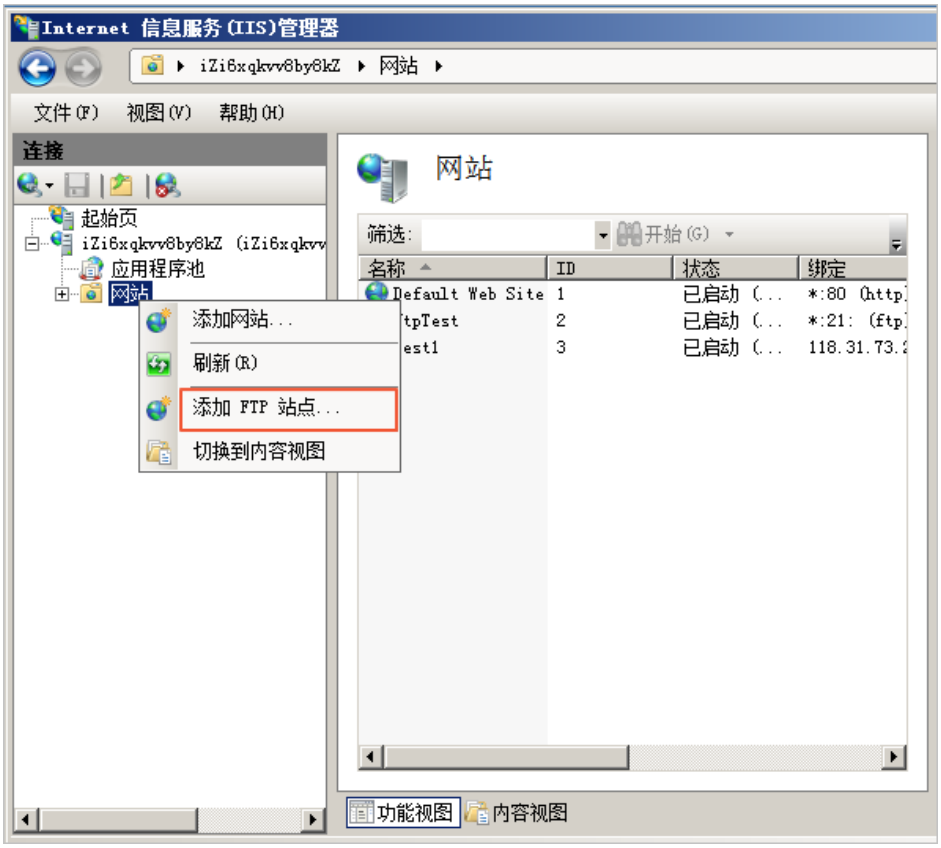


步骤四： 添加及设置 FTP 站点

安装 FTP，设置好共享文件夹权限后，您需要创建 FTP 站点。

选择 **开始** > **所有程序** > **管理工具** > **Internet 信息服务（IIS）管理器**。

右键单击 **网站**，选择 **添加 FTP 站点**。



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**。

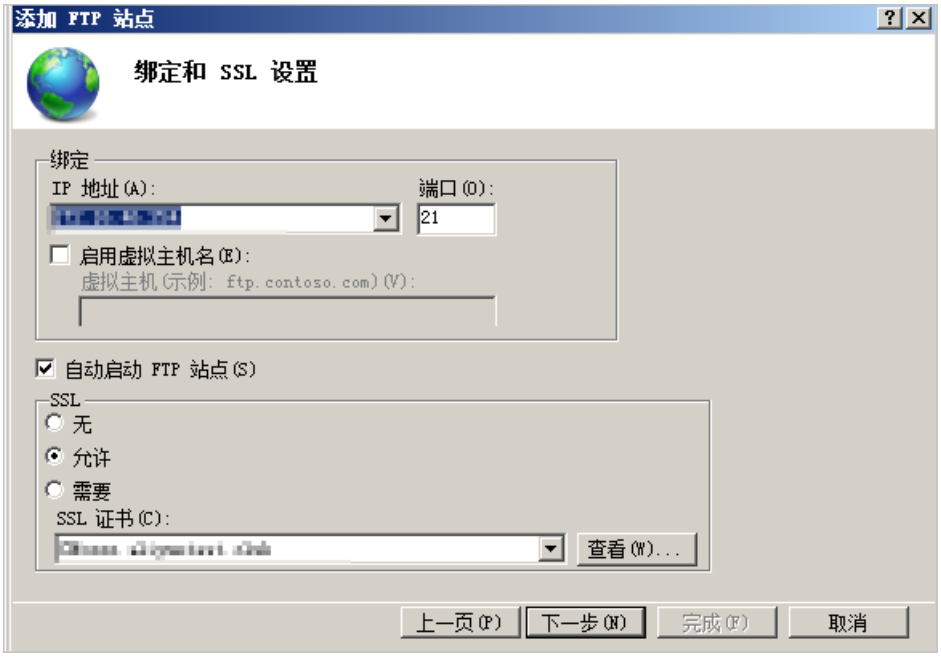
IP 地址默认选择 **全部未分配**。端口号可自行设置，FTP 默认端口号为 21。

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**。



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。

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。

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_FTP**。

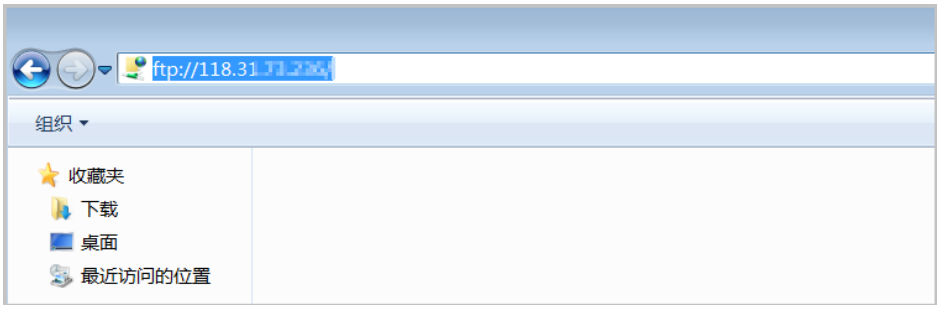
服务器防火墙默认放行 TCP 21 端口用于 FTP 服务。如果选用其他端口，您需要在防火墙中添加一条放行此端口的入站规则。具体方法参见 **设置 ECS 实例远程连接防火墙** 中的**添加端口规则**章节。其他防火墙设置参见 **微软官方文档**。

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。

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

远程连接并登录到 Linux 实例。

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 ~]#
```

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。

运行以下命令设置开机自启动。

```
systemctl enable vsftpd.service
```

运行以下命令启动 FTP 服务。

```
systemctl start vsftpd.service
```

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。

修改/etc/vsftpd/vsftpd.conf：

运行vim /etc/vsftpd/vsftpd.conf。

按 i 键进入编辑模式。

将写权限修改为write_enable=YES。

将匿名上传权限修改为anon_upload_enable=YES。

按 Esc 键退出编辑模式，然后输入 :wq 保存并退出文件。

```
anonymous_enable=YES
#
# 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
#
# Uncomment this to allow the anonymous FTP user to upload files. This only
# has an effect if the above global write enable is activated. Also, you will
# obviously need to create a directory writable by the FTP user.
# When SELinux is enforcing check for SE bool allow_ftpd_anon_write, allow_ftpd_
anon_upload_enable=YES
                                                                      29,1
```

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
```

```
[root@iZbp1g1kolvxh5k8l00z6cZ ~]# chmod o+w /var/ftp/pub/
[root@iZbp1g1kolvxh5k8l00z6cZ ~]# systemctl restart vsftpd.service
[root@iZbp1g1kolvxh5k8l00z6cZ ~]#
```

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：

运行以下命令创建 ftptest 用户。

```
useradd ftptest
```

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 ~]#
```

修改/etc/vsftpd/vsftpd.conf：

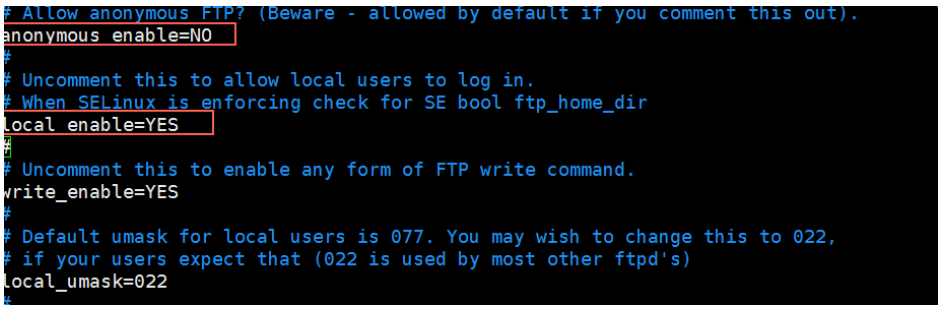
运行vim /etc/vsftpd/vsftpd.conf。

按键 **i** 进入编辑模式。

将是否允许匿名登录 FTP 的参数修改为anonymous enable=NO。

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

按键 **Esc** 退出编辑模式，然后按键：wq保存并退出文件。



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
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

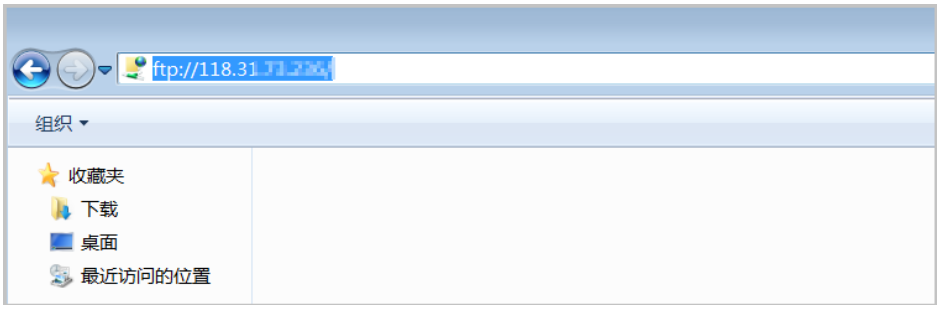
步骤三： 设置安全组

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**，具体配置可以参考 **安全组规则的典型应用_FTP**。

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** 对 FTP 服务进行安全加固。

搭建FTP站点