

弹性云服务器

最佳实践

文档版本 01
发布日期 2023-11-13



版权所有 © 华为技术有限公司 2024。保留一切权利。

非经本公司书面许可，任何单位和个人不得擅自摘抄、复制本文档内容的部分或全部，并不得以任何形式传播。

商标声明



HUAWEI和其他华为商标均为华为技术有限公司的商标。

本文档提及的其他所有商标或注册商标，由各自的所有人拥有。

注意

您购买的产品、服务或特性等应受华为公司商业合同和条款的约束，本文档中描述的全部或部分产品、服务或特性可能不在您的购买或使用范围之内。除非合同另有约定，华为公司对本文档内容不做任何明示或暗示的声明或保证。

由于产品版本升级或其他原因，本文档内容会不定期进行更新。除非另有约定，本文档仅作为使用指导，本文档中的所有陈述、信息和建议不构成任何明示或暗示的担保。

安全声明

漏洞处理流程

华为公司对产品漏洞管理的规定以“漏洞处理流程”为准，该流程的详细内容请参见如下网址：

<https://www.huawei.com/cn/psirt/vul-response-process>

如企业客户须获取漏洞信息，请参见如下网址：

<https://securitybulletin.huawei.com/enterprise/cn/security-advisory>

目录

1 弹性云服务器配置最佳实践	1
2 最佳实践-主机迁移	4
3 跨账号跨区域迁移云服务器	7
4 Linux 云服务器 SSH 登录的安全加固	16
5 使用 VNC Viewer 连接 Linux 云服务器	21
6 搭建 Keepalived Nginx 高可用 Web 集群	27
7 最佳实践-自助建站汇总	37
8 搭建 Discuz 论坛网站	40
8.1 方案概述.....	40
8.2 购买服务.....	43
8.3 搭建网站.....	47
8.4 配置特性.....	54
8.5 访问网站.....	62
9 基于 WordPress 搭建个人网站 (Linux)	63
10 快速构建 FTP 站点 (Windows)	73
11 快速构建 FTP 站点 (Linux)	90
12 基于 Tomcat 构建 Java web 环境 (CentOS 7.4)	95
13 手工搭建 Magento 电子商务网站 (Linux)	102
14 搭建 Microsoft SharePoint Server 2016	114
14.1 购买并登录弹性云服务器.....	114
14.2 添加 AD、DHCP、DNS、IIS 服务.....	116
14.3 安装 SQL Server.....	122
14.4 安装 Sharepoint 2016.....	128
14.5 配置 Sharepoint 2016.....	133
14.6 验证 Sharepoint 2016.....	138
15 手工搭建 LNMP 环境 (CentOS7.2 PHP7.0)	142
16 手工部署 Docker (CentOS 7.5)	149

17 搭建微信公众号后台——收发文本消息.....	153
18 手工部署 GitLab (CentOS7.2)	162
19 手工部署 RabbitMQ (CentOS 7.4)	165
20 手工搭建 Ghost 博客 (Ubuntu 20.04)	170
21 手工部署 Node.js (CentOS 7.2)	177
22 PostgreSQL 本地 Slave 搭建步骤.....	180
23 手工安装宝塔面板 (CentOS7.2)	183
24 在 ECS 上通过内网访问 OBS.....	185
24.1 概述.....	185
24.2 在 Windows ECS 上使用 OBS Browser+通过内网访问 OBS.....	187
24.3 在 Linux ECS 上使用 obsutil 通过内网访问 OBS.....	189
25 通过 Extundelete 恢复弹性云服务器误删除数据 (Linux)	193
26 基于弹性云服务器安装及部署 Jenkins.....	195

1 弹性云服务器配置最佳实践

为了能够更加安全、可靠、灵活、高效的使用弹性云服务器，建议您购买弹性云服务器后执行以下最佳实践。

访问与连接

为了确保弹性云服务器创建成功且状态正常，建议您第一次通过VNC方式远程登录弹性云服务器，若登录成功则表明弹性云服务器资源状态正常。

详细操作，请参见：

- [Windows弹性云服务器管理控制台远程登录（VNC方式）](#)
- [Linux弹性云服务器远程登录（VNC方式）](#)

之后，您就可以根据本地环境以及弹性云服务器是否绑定弹性公网IP，选择合适的登录方式，详细内容请参见[登录弹性云服务器](#)。

系统更新

- 更新镜像源（Linux）
为了及时获取最新的系统更新和软件安装依赖，建议您在使用弹性云服务器之前更新镜像源。
华为云内网源无需连接Internet、且服务稳定，对于使用EulerOS、CentOS、Debian、Ubuntu操作系统的弹性云服务器，建议您更新为华为云内网源，详细操作，请参见[如何使用自动化工具配置华为云镜像源\(x86_64和ARM\)?](#)
- 更新Windows补丁和驱动（Windows）
为了提高弹性云服务器的问题修复能力和性能，建议您定时更新Windows的补丁和驱动程序。
您可以通过设置Windows自动更新，及时发现最新的补丁和驱动版本，提醒您进行更新。

数据存储

- 存储安全
为了实现数据的存储安全性，建议您将操作系统和应用数据分别存储在系统盘和数据盘，一方面保证数据的安全，避免因系统故障导致用户数据丢失；另一方面可根据业务需要扩容或新增磁盘扩展存储容量。

- 扩容磁盘：支持扩容系统盘和数据盘，建议您参考[扩容“正在使用”状态的云硬盘容量](#)扩容磁盘容量。
- 新增磁盘：支持新增数据盘，建议您[新增磁盘](#)，并完成[挂载磁盘](#)和[初始化数据盘](#)后，才可以正常使用。
- 数据加密
为了进一步保护数据的安全性，建议您使用加密的系统盘和数据盘。详细内容，请参见[管理加密云硬盘](#)。

安全管理

- 身份认证
为了实现对资源的安全访问和权限控制，建议您创建IAM用户，并结合企业项目实现对资源的身份认证、权限管理和资源分组管理，详细内容请参见[多运维人员权限设置案例](#)和[多项目管理实践案例](#)。
- 登录鉴权方式
为了保证在丢失密码或密码失效的情况下，可以成功重置密码，建议您在弹性云服务器的购买完成后，优先执行[安装一键式重置密码插件（可选）](#)或[更新一键式重置密码插件（单台操作）](#)操作。
- 访问控制
为了实现对弹性云服务器入方向和出方向的访问进行控制，提高安全性，建议您根据控制的范围设置相应的访问控制策略：
 - 基于ECS实例：建议您使用安全组功能，通过安全组规则实现对ECS的访问控制。详细内容，请参见[安全组配置示例](#)。
 - 基于ECS所在子网：建议您使用网络ACL功能，通过网络ACL规则实现对子网内所有ECS实例的访问控制。详细内容，请参见[网络ACL配置示例](#)。
- 安全防护
为了实现对弹性云服务器的高阶安全防护，除了免费开启的主机安全基础防护外，建议您根据防护需求购买高阶的主机安全防护版本，详细的版本功能、版本差异及购买方式，请参见[版本功能差异](#)和[主机安全快速入门](#)。
- 网络防护
为了防止弹性云服务器绑定的弹性公网IP被攻击，建议您购买华为云的[DDoS防护](#)服务。
- 敏感操作保护
为了保证账号及操作的安全性，建议您开启敏感操作保护功能，在控制台执行关机、重启、删除等敏感操作时，进行身份验证。该功能对账号以及账号下的用户都生效。
详细内容，请参见[敏感操作保护](#)。

备份与恢复

- 数据备份与恢复
为了在发生病毒入侵、人为误删除、软硬件故障等事件时，能够快速恢复数据，建议您定期进行数据备份，备份的具体方案和适用场景，请参见[云备份概述](#)。
备份成功后，可以[使用云服务器备份恢复数据](#)或[使用备份创建镜像](#)的方式恢复数据。
- 高可用容灾业务部署

为了实现较高的业务容灾能力，建议您将云服务器资源部署在同一区域的不同可用区内。可用区的具体内容，请参见[区域和可用区](#)和[步骤一：基础配置](#)。

- 主备应用HA业务部署

为了实现弹性云服务器的主备切换、达到高可用HA的目的，建议您参考[搭建Keepalived Nginx高可用Web集群](#)方案进行部署。

操作维护管理

- 监控管理

为了使用户能够及时获取弹性云服务器的各项性能指标及运行状态，建议您使用云监控服务，执行自动实时监控、告警和通知操作。

详细内容，请参见[云监控服务](#)。

- 关键操作管理

为了能够对弹性云服务器的操作进行查询、审计和回溯，建议您开启云审计服务记录与之相关的操作事件。

更多内容，请参见[云审计服务](#)。

- 日志管理

为了提高弹性云服务器和应用程序的可用性和性能，建议您通过云日志服务收集日志数据，通过对海量日志数据的分析和处理，实现实时决策分析、设备运维管理、用户业务趋势分析等能力。

详细内容，请参见[云日志服务](#)。

2 最佳实践-主机迁移

背景信息

由于公有云“敏捷弹性、安全可靠、简单易用、节省成本”等特点，越来越多的企业将自己的IT应用和负载迁移到公有云，如何方便快捷地将已有的服务器系统迁移到华为云或者从其他的公有云迁移到华为云，有着非常重要的意义。华为云支持把X86物理服务器，或者私有云、公有云平台上的虚拟机迁移到华为云弹性云服务器上。

可以通过以下两种方式进行迁移。

- 主机迁移服务（推荐）
- 镜像导入方式

本文主要介绍如何使用以上两种方式帮助您把服务器上的应用和数据迁移到华为云。

主机迁移服务（推荐）

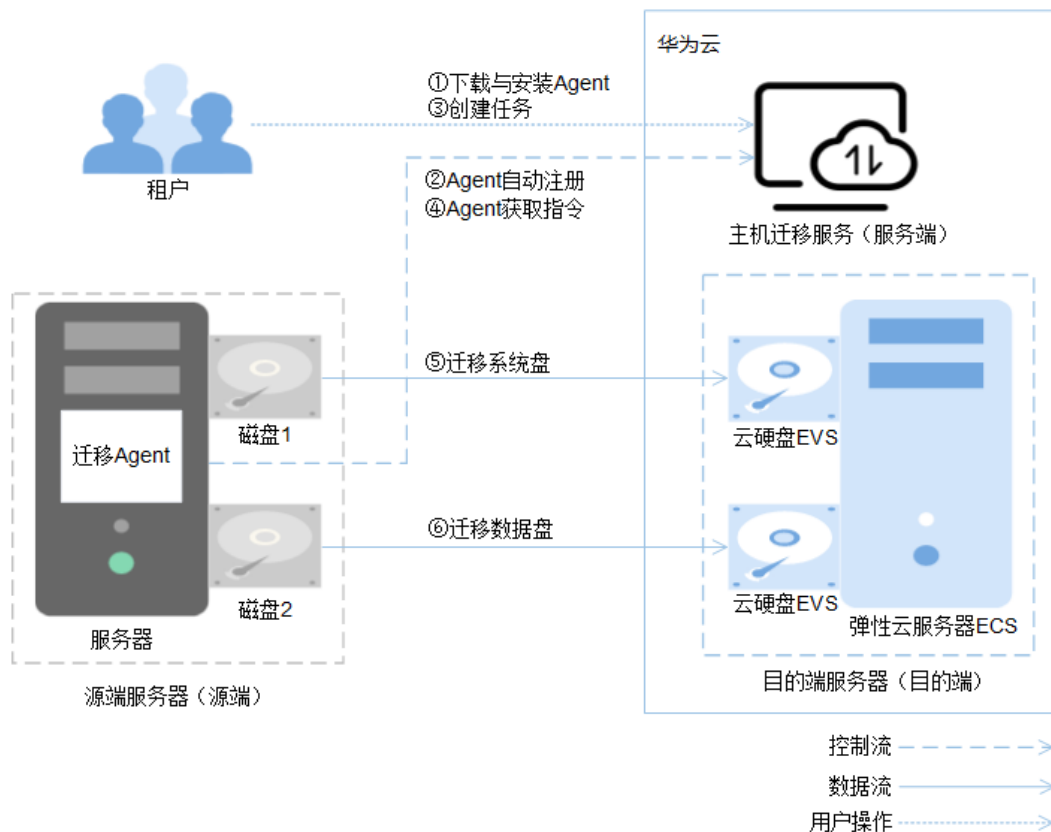
服务简介

主机迁移服务（Server Migration Service）是一种P2V/V2V迁移服务，可以帮您把X86物理服务器，或者私有云、公有云平台上的虚拟机迁移到华为云弹性云服务器上，从而帮助您轻松地把服务器上的应用和数据迁移到华为云。

主机迁移服务支持迁移的源端服务器OS类型请参见[兼容性列表](#)。

使用主机迁移服务时，对于源端服务器的约束与限制请参见[约束与限制](#)。

图 2-1 主机迁移服务工作原理



主机迁移服务的工作原理如下，其中第1步和第3步需要用户操作，其余步骤由主机迁移服务自动完成。

1. 用户在源端服务器上安装迁移Agent，具体操作方法请您参见[安装Agent](#)。
2. 源端服务器上的迁移Agent向主机迁移服务注册自身连接状态并将源端服务器信息上报到主机迁移服务，完成迁移可行性检查。
3. 通过迁移可行性检查后，用户开始创建迁移任务，具体操作方法请您参见[创建迁移任务](#)。
4. 迁移Agent获取并执行主机迁移服务发送的迁移指令。
5. 开始迁移源端服务器系统盘。
6. 开始迁移源端服务器数据盘。

📖 说明

- **源端：**指迁移任务中的源端服务器。
- **目的端：**指迁移任务中的目的端服务器。
- **服务端：**指主机迁移服务。

服务入口

主机迁移服务的操作步骤：[创建迁移任务](#)。

主机迁移服务的详细介绍：[主机迁移服务](#)。

镜像导入方式

1. 制作镜像。例如可通过QEMU制作镜像，可参见[此处](#)获取相应的使用帮助。
2. 创建私有镜像，详细操作可参见[此处](#)。
3. 根据私有镜像创建弹性云服务器，请参见[购买弹性云服务器](#)。

3 跨账号跨区域迁移云服务器

操作场景

本节操作介绍华为云上云服务器的跨账号跨区域迁移，建议采用镜像迁移方式。

服务器迁移的常见场景与常用的迁移方式请参考[迁移的背景知识](#)。

跨账号跨区域迁移的方法请参考[方案介绍](#)。

迁移的背景知识

常见的服务器迁移场景包括物理服务器与云服务器之间、VMware等虚拟化平台与华为云服务器之间、华为云上的云服务器之间（不同账号或地域之间）、其他云平台的云服务器和华为云服务器、线下制作成镜像文件导入华为云，推荐的迁移方式请参考[表 3-1](#)。

表 3-1 常见的服务器迁移场景

迁移场景	适用条件	推荐方式	备注
x86物理服务器 → 华为云服务器 示例：P2V	服务器能够访问公网	主机迁移服务 请参考 最佳实践-主机迁移 。	-
VMware等虚拟化平台的虚拟机 → 华为云服务器	虚拟机能够访问公网	主机迁移服务 请参考 最佳实践-主机迁移 。	-
	虚拟机不能访问公网	1. 先从VMware虚拟化平台导出虚拟机的镜像文件。 2. 然后使用 镜像导入 。	如果镜像文件格式不在vhd、vmdk、qcow2、raw、vhdx、qcow、vdi、qed、zvhd、zvhd2范围，需要先转换格式。

迁移场景	适用条件	推荐方式	备注
华为云上的云服务器（不同账号或地域之间） 示例：账号A在“华北-北京四”区域的云服务器 → 账号B的“华东-上海一”区域。	云服务器能够访问公网	主机迁移服务 请参考 最佳实践-主机迁移 。	-
	云服务器不能访问公网	（推荐） <ul style="list-style-type: none"> 不同账号间迁移：共享镜像 将云服务器的系统盘制作成系统盘镜像、数据盘制作成数据盘镜像，或者直接将云服务器制作成整机镜像，然后共享给其他账号。 不同地域间迁移：跨区域复制镜像 将云服务器的系统盘制作成系统盘镜像、数据盘制作成数据盘镜像，或者直接将云服务器制作成整机镜像，然后复制到其他区域。 不同账号、不同地域间迁移：共享镜像和跨区域复制镜像结合使用。 	只有通过云备份创建的整机镜像，才支持共享。 跨区域复制的镜像大小不能超过128GB。
其他云平台的云服务器 示例：阿里云/腾讯云 → 华为云	云服务器能够访问公网	主机迁移服务 请参考 最佳实践-主机迁移 。	-
	云服务器不能访问公网	<ol style="list-style-type: none"> 先从其他云平台导出镜像。 然后使用镜像导入。 	如果镜像文件格式不在vhd、vmdk、qcow2、raw、vhdx、qcow、vdi、qed、zvhd、zvhd2范围，需要先转换格式。

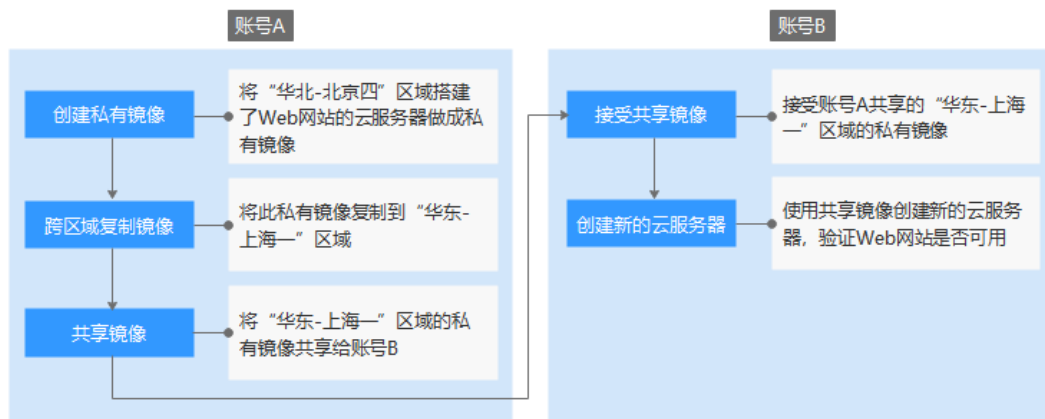
迁移场景	适用条件	推荐方式	备注
安装盘映像 示例：线下制作成 镜像文件导入华为 云	已有所需操作系统的ISO格式的安装盘映像	使用VirtualBox创建镜像并上传到华为云 或 使用控制台的导入ISO文件方式创建镜像	-

方案介绍

跨账号跨区域迁移云服务器的方案为：账号A将区域A的云服务器做成私有镜像，将此私有镜像复制到同账号的区域B，再共享给账号B；账号B接受账号A的共享镜像后，使用该镜像创建新的云服务器。

例如，账号A在“华北-北京四”区域的云服务器上搭建了Web网站，想要将云服务器迁移到账号B的“华东-上海一”，操作流程如下：

图 3-1 操作流程



1. 步骤一：创建私有镜像
2. 步骤二：跨区域复制镜像
3. 步骤三：共享镜像
4. 步骤四：接受共享镜像
5. 步骤五：创建新的云服务器

步骤一：创建私有镜像

账号A将“华北-北京四”区域搭建了Web网站的云服务器做成私有镜像。假设云服务器Web访问地址为：<http://121.36.xxx.xxx/index.html>



1. 账号A登录管理控制台，在左上角切换区域为“华北-北京四”。
2. 选择“服务列表 > 计算 > 弹性云服务器”。
进入弹性云服务器列表页面。
3. 在搭建了Web网站的云服务器（假设为ecs-fe9e）所在行，单击操作列的“更多 > 镜像/磁盘/备份 > 创建镜像”。
进入创建私有镜像页面。
4. 填写如下参数：

图 3-2 创建私有镜像



- 创建方式：整机镜像
由于数据盘包含了业务数据，本例中使用弹性云服务器携带其挂载的数据盘一起创建整机镜像。
- 选择镜像源：云服务器，并选择“ecs-fe9e”
- 选择“云服务器备份存储库”，用于存放备份。如果云服务器已绑定存储库，系统会默认选择已绑定的存储库。
创建的整机镜像及中间产物备份副本会默认存放在备份存储库中，因此必须选择一个云服务器备份存储库。
如果没有可用的存储库，请单击“新建云服务器备份存储库”进行创建，注意“保护类型”需选择“备份”。
- 名称：输入私有镜像名称，如“migrate_test”

- 企业项目：选择默认项目“default”
5. 单击“立即创建”。
 6. 确认无误后，阅读并勾选协议，单击“提交申请”。
 7. 返回私有镜像列表，等待几分钟后，私有镜像创建成功。

图 3-3 查看私有镜像

<input type="checkbox"/>	名称	状态	操作系统类型	操作系统	镜像类型
<input type="checkbox"/>	migrate_test	正常	Linux	CentOS 7.2 64bit	ECS整机镜像(x86)

步骤二：跨区域复制镜像

账号A将**步骤一：创建私有镜像**中创建的私有镜像复制到“华东-上海一”区域。在跨区域复制前，需要先创建IAM委托，详见以下步骤。

1. 创建IAM委托。
 - a. 单击右上方登录的用户名，在下拉列表中选择“统一身份认证”。
 - b. 在左侧导航栏中，单击“委托”。
 - c. 在“委托”页面，单击“创建委托”。
 - d. 在“创建委托”页面，设置如下参数：
 - 委托名称：按需填写，例如“ims_administrator_agency”。

图 3-4 创建委托



图 3-4 展示了创建 IAM 委托的表单配置。表单包含以下字段：

- * 委托名称**：输入框，值为 `ims_administrator_agency`。
- * 委托类型**：包含两个单选按钮。
 - 普通帐号：将帐号内资源的操作权限委托给其他华为云帐号。
 - 云服务：将帐号内资源的操作权限委托给华为云服务。
- * 云服务**：下拉菜单，选择 `镜像服务 IMS`。
- * 持续时间**：下拉菜单，选择 `永久`。
- 描述**：文本输入框，提示为“请输入委托信息。”，当前字符数为 0/255。

底部有两个按钮：**下一步**（红色背景）和 **取消**（白色背景）。

- 委托类型：选择“云服务”。

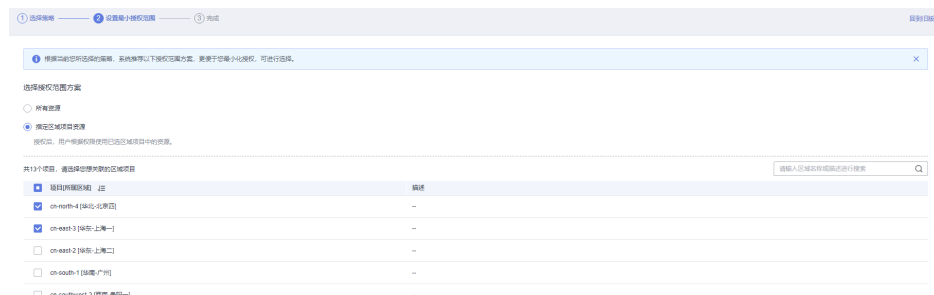
- 云服务：（“委托类型”选择“云服务”时出现此参数项。）单击“选择”，在弹出的“选择云服务”页面选择“镜像服务 IMS”，单击“确定”。
 - 持续时间：选择“永久”。
 - 描述：非必选，可以填写“拥有IMS Administrator权限的委托”。
- e. 单击“下一步”，为委托选择策略。
- f. 勾选“IMS Administrator”策略。

图 3-5 选择策略



- g. 单击“下一步”，选择授权区域。
- h. 在“项目[所属区域]”中选择“华北-北京四”和“华东-上海一”，单击“确定”。

图 3-6 选择区域



注意

配置权限时，请勿选择所有项目，否则委托将不生效。

- i. 单击“完成”，在“基本信息”页签查看创建的委托。

图 3-7 查看委托

委托名称ID	委托对象	委托时长	创建时间	描述	操作
ims_administrator_agency	云服务 镜像服务 IMS	永久	2022/02/25 15:50:34 GMT+08:00	--	授权 修改 删除

2. 选择“服务列表 > 计算 > 镜像服务”，单击“私有镜像”页签。进入私有镜像列表页。
3. 在私有镜像“migrate_test”所在行，单击操作列的“更多 > 复制”。弹出“复制镜像”对话框。
4. 填写如下参数：

图 3-8 复制镜像

复制类型: 本区域内复制 | 跨区域复制

* 名称: copy_cn-north-4_migrate_test

* 目的区域: 华东-上海一

* 目的项目: cn-east-3

* 目的云服务器备份存储库: vault-2398 | 0/100(GB) [新建云服务器备份存储库](#)

只允许选择类型为复制的云服务器备份存储库。

* IAM委托: ims_administrator_agency [查看委托](#) ?

描述: 0/1,024

确定 取消

- 名称: 保持默认值copy_cn-north-4_migrate_test
 - 目的区域: 华东-上海一
 - 目的项目: cn-east-3
 - 目的云服务器备份存储库: vault-2398
 - IAM委托: 选择1中创建好的委托“ims_administrator_agency”。
5. 单击“确定”。
 6. 在控制台左上角切换区域为“华东-上海一”，等待几分钟后，私有镜像复制成功。

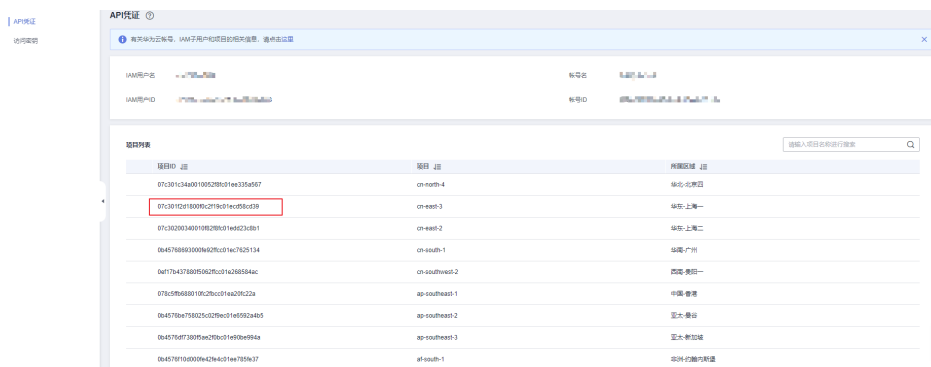
图 3-9 查看私有镜像

<input type="checkbox"/>	名称	状态	操作系统类型	操作系统	镜像类型
<input type="checkbox"/>	copy_cn-north-4_...	正常	Linux	CentOS 7.2 64bit	ECS整机镜像(x86)

步骤三：共享镜像

账号A将“华东-上海一”区域的私有镜像共享给账号B。需要提前在“我的凭证”中获取获取账号B“华东-上海一”区域的项目ID。

图 3-10 获取项目 ID



1. 在“华东-上海一”区域，选择“服务列表 > 计算 > 镜像服务”，单击“私有镜像”页签。
进入私有镜像列表页。
2. 在私有镜像“copy_cn-north-4_migrate_test”所在行，单击操作列的“更多 > 共享”。
弹出“共享镜像”对话框。
3. 在“共享镜像”页签，输入账号B的项目ID，单击“添加”。

图 3-11 共享镜像

共享镜像



4. 单击“确定”。

步骤四：接受共享镜像

账号B接受账号A共享的“华东-上海一”区域的私有镜像。

1. 账号B登录管理控制台，在左上角切换区域为“华东-上海一”。
2. 选择“服务列表 > 计算 > 镜像服务”，单击“共享镜像”页签。
进入镜像列表页面。
3. 在提示信息中，勾选“copy_cn-north-4_migrate_test”，单击“接受”。

图 3-12 接受共享镜像



接受后，该私有镜像将显示在共享镜像列表中。

图 3-13 共享镜像列表

名称	操作系统类型	操作系统	镜像类型
copy_cn-north-4_migrate_test	Linux	CentOS 7.2 64bit	ECS整机镜像(x86)

步骤五：创建新的云服务器

账号B使用共享镜像创建新的云服务器，并验证Web网站是否可用。

1. 在共享镜像“copy_cn-north-4_migrate_test”所在行，单击操作列的“申请服务器”。
进入弹性云服务器购买向导页面。
2. 按需选择计费模式、可用区、规格、网络等参数，镜像保持默认值，按界面提示完成云服务器创建。

图 3-14 选择镜像



3. 返回云服务器列表，等待几分钟，云服务器创建成功。
4. 尝试访问云服务器的Web网站，验证是否可用。
浏览器中输入http://云服务器弹性公网IP/index.html，假设为http://124.70.xxx.xxx/index.html。经验证可以正常访问，表示云服务器迁移成功，任务结束。

图 3-15 验证 Web 网站

124.70.xxx.xxx/index.html

Welcome to nginx!

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to nginx.org.
Commercial support is available at nginx.com.

Thank you for using nginx.

4 Linux 云服务器 SSH 登录的安全加固

Linux云服务器常用的登录方式是SSH，对于密码登录方式创建的云服务器，如何保证登录安全性呢？本文以CentOS 7.6为例，对SSH登录进行安全加固。

表 4-1 弹性云服务器详细信息

参数	取值
名称	ecs-f5a2
操作系统	CentOS 7.6 64bit
弹性公网IP	119.3.xxx.x
登录方式	密码

修改默认端口

1. 通过[SSH密码方式](#)远程登录云服务器。
2. 执行以下命令，修改SSH登录的默认端口，比如修改为“5000”。

vim /etc/ssh/sshd_config

按“i”进入编辑模式，在第17行，将注释符“#”删掉，修改为“Port 5000”。

图 4-1 修改前

```
#  
#Port 22  
#AddressFamily any  
#ListenAddress 0.0.0.0  
#ListenAddress ::
```

图 4-2 修改后

```
Port 5000  
#AddressFamily any  
#ListenAddress 0.0.0.0  
#ListenAddress ::
```

- 按“Esc”，输入:wq保存并退出。

增加防火墙规则：开放指定端口

CentOS 7系列操作系统的默认防火墙是fireware，而不是iptables。因此，如果系统是默认防火墙，则不需要执行本节操作；如果安装过iptables，则要参考本节指导开放SSH登录的5000端口。

- 执行以下命令，检查是否安装了iptables。

service iptables status

- 如果提示如下类似信息，表示未安装iptables，跳过本节，继续[添加安全组规则](#)操作。

```
[root@ecs-~]# service iptables status
Redirecting to /bin/systemctl status iptables.service
Unit iptables.service could not be found.
[root@ecs-~]#
```

- 如果提示如下类似信息，表示安装了iptables，并且为“active”状态。继续执行步骤2。

```
[root@ecs-~]# service iptables status
Redirecting to /bin/systemctl status iptables.service
■ iptables.service - IPv4 firewall with iptables
Loaded: loaded (/usr/lib/systemd/system/iptables.service; disabled; vendor preset: disabled)
Active: active (exited) since Tue 2019-04-16 10:42:53 CST; 3s ago
Process: 23744 ExecStart=/usr/libexec/iptables/iptables.init start (code=exited, status=0/SUCCESS)
Main PID: 23744 (code=exited, status=0/SUCCESS)

Apr 16 10:42:53 ecs- systemd[1]: Starting IPv4 firewall with iptables...
Apr 16 10:42:53 ecs- iptables.init[23744]: iptables: Applying firewall rules: [ OK ]
Apr 16 10:42:53 ecs- systemd[1]: Started IPv4 firewall with iptables.
```

- 执行以下命令增加iptables规则，开放5000端口。

```
iptables -A INPUT -p tcp -m state --state NEW -m tcp --dport 5000 -j ACCEPT
```

- 执行以下命令查看iptables现有规则是否已包含5000端口。

iptables -L -n

```
Chain INPUT (policy ACCEPT)
target     prot opt source                destination
ACCEPT     tcp  --  0.0.0.0/0             0.0.0.0/0           state NEW tcp dpt:5000
ACCEPT     tcp  --  0.0.0.0/0             0.0.0.0/0           state NEW tcp dpt:5000
```

添加安全组规则

安全组中的入方向规则默认开启了22端口，当云服务器的SSH登录端口修改为5000时，需要为安全组新加一条规则。


- 登录管理控制台。
- 选择“计算 > 弹性云服务器”，进入云服务器控制台。
- 单击云服务器名称“ecs-f5a2”进入详情页面。
- 选择“安全组”页签，单击  展开安全组规则详情，单击列表右上角的“更改安全组规则”。
- 添加一条入方向规则，如[图4-3](#)所示。

图 4-3 安全组规则

<input type="checkbox"/>	类型	协议	端口/范围	源地址	描述
<input type="checkbox"/>	IPv4	全部	全部	sg-9341	--
<input type="checkbox"/>	IPv4	TCP	22	0.0.0.0/0	Permit default Linux SSH port.
<input type="checkbox"/>	IPv4	TCP	3389	0.0.0.0/0	Permit default Windows remote desktop port.
<input type="checkbox"/>	IPv4	TCP	5000	0.0.0.0/0	--

关闭密码登录，采用密钥登录

在管理控制台创建一个密钥对，绑定至云服务器，使云服务器登录方式切换为密钥。

1. 登录管理控制台。
2. 选择“计算 > 弹性云服务器”，进入云服务器控制台。
3. 参考[创建密钥对](#)创建一个密钥对，妥善保管好私钥文件。
4. 选择“服务列表 > 安全 > 数据加密服务”，单击左侧导航中的“密钥对管理”。
5. 选择“云服务器列表”页签，在“ecs-f5a2”所在行，单击操作列的“绑定”。按照[图4-5](#)进行配置，单击“确定”。

关闭密码登录，请在绑定密钥对页面勾选“关闭密码登录方式。”或编辑“sshd_config”配置文件，禁止密码登录。

图 4-4 绑定密钥对

绑定密钥对

系统将对服务器进行密钥对配置，执行此操作后可直接使用密钥登录服务器。出于安全考虑，建议关闭服务器的密码登录方式，只使用密钥登录服务器。

ECS名称

IP

状态 运行中

* 新密钥对

* root密码

关闭密码登录方式。[了解详情](#)

我已经阅读并同意《[密钥对管理服务免责声明](#)》。

6. 登录云服务器操作系统，编辑“sshd_config”配置文件，禁止密码登录。

```
vim /etc/ssh/sshd_config
```

按“i”进入编辑模式，在最后几行按下图进行配置。

```
# override default of no subsystems
Subsystem sftp /usr/libexec/openssh/sftp-server

# Example of overriding settings on a per-user basis
#Match User anoncvs
#       X11Forwarding no
#       AllowTcpForwarding no
#       PermitTTY no
#       ForceCommand cvs server
PermitRootLogin yes
UseDNS no
PasswordAuthentication no
```

参数说明：

- PermitRootLogin：是否允许root账户登录，配置为“yes”。
- UseDNS：是否允许DNS解析，配置为“no”。
- PasswordAuthentication：是否允许使用密码登录，配置为“no”。

📖 说明

在步骤5中为云服务器绑定密钥对时，已勾选了“关闭密码登录方式”，此处的“PasswordAuthentication”必为“no”，只需验证即可。

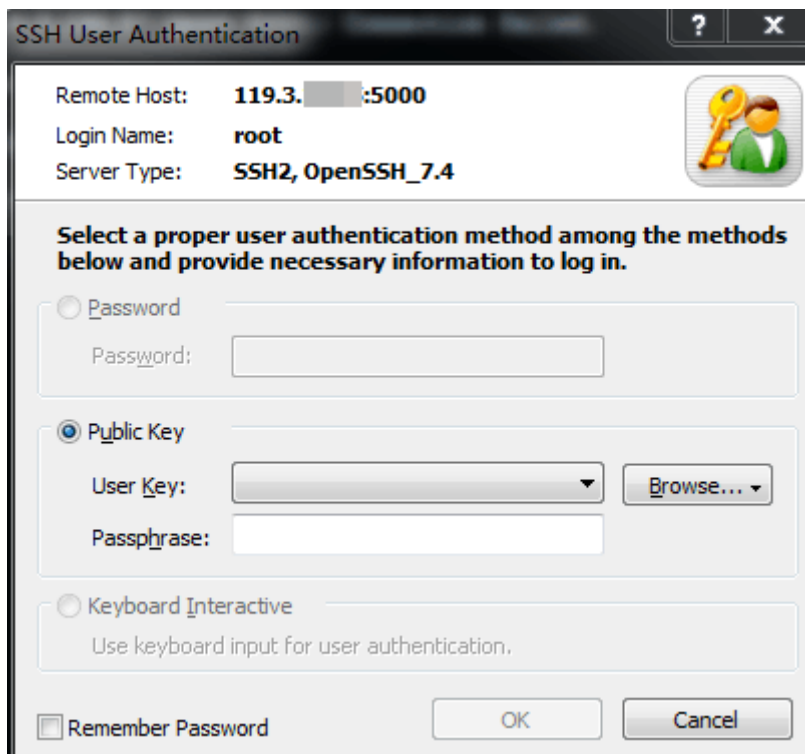
按“Esc”，输入:wq保存并退出。

7. 完成后，重启sshd服务。

systemctl restart sshd

8. 通过Xshell或其他SSH客户端连接云服务器，密码登录会被限制（如图4-5所示，密码输入项置灰），则证明配置成功。

图 4-5 Xshell 登录云服务器



编辑 hosts.allow 和 hosts.deny

“/etc/hosts.allow”和“/etc/hosts.deny”是控制远程访问的文件，通过配置该文件可以允许或者拒绝某个IP或者IP段的客户访问Linux云服务器的某项服务。

比如SSH服务，通常只对管理员开放，那就可以禁用不必要的IP，而只开放管理员可能使用到的IP段。

因为云服务器需要在不同地点登录，建议编辑“/etc/hosts.allow”允许所有IP地址登录，这样不会影响正常使用。

vim /etc/hosts.allow

在最后一行增加“sshd:ALL”。

```
#          either use the tcp_wrappers library or that have been
#          started through a tcp_wrappers-enabled xinetd.
#
#          See 'man 5 hosts_options' and 'man 5 hosts_access'
#          for information on rule syntax.
#          See 'man tcpd' for information on tcp_wrappers
sshd:ALL
```

可以通过一些方法识别云服务器的安全风险，比如检查SSH状态，查看疑似恶意登录的IP，然后在“/etc/hosts.deny”中将这些地址禁止。

5 使用 VNC Viewer 连接 Linux 云服务器

Linux云服务器一般采用SSH连接方式，使用密钥对进行安全地无密码访问。但是SSH连接一般都是字符界面，有时需要使用图形界面进行一些复杂操作。本文以Ubuntu 20.04操作系统为例，介绍如何为云服务器安装VNC Server，以及如何使用VNC Viewer连接云服务器。

准备工作

- 已创建一台Ubuntu 20.04操作系统的云服务器，并且为云服务器绑定弹性公网IP，确保可以连接互联网。
具体操作请参见[购买弹性云服务器](#)、[为弹性云服务器申请和绑定弹性公网IP](#)。
- 已在本地PC安装VNC Viewer客户端。

说明

VNC Viewer下载地址：<https://www.realvnc.com/en/connect/download/viewer/>

安装 VNC Server

默认情况下，Ubuntu 20.04操作系统没有安装图形环境或VNC Server。本示例中，桌面环境使用Xfce，它是一个精简的轻量化桌面，与Gnome和KDE相比更小巧，并且界面美观、对用户友好，适合云服务器远程连接场景下使用。

1. 远程登录云服务器。
账号：root；密码：您在创建云服务器时设置的密码。
2. 执行以下命令，更新软件列表。
sudo apt update
3. 安装Xfce桌面环境。
sudo apt install xfce4 xfce4-goodies
4. 执行完成后，安装TightVNC服务器。
sudo apt install tightvncserver
5. 执行完成后，使用vncserver命令进行配置。
当您第一次运行vncserver时，它会创建一个默认启动脚本。按照提示进行配置：

```
root@ecs-9240- :~# vncserver
You will require a password to access your desktops.
Password:
Verify:
Would you like to enter a view-only password (y/n)? n
xauth: file /root/.Xauthority does not exist

New 'X' desktop is ecs-9240- :1

Creating default startup script /root/.vnc/xstartup
Starting applications specified in /root/.vnc/xstartup
Log file is /root/.vnc/ecs-9240- :1.log
root@ecs-9240- :~#
```

- 密码：长度为6~8个字符。超过8个字符的密码将自动截断。请牢记该密码，VNC Viewer连接云服务器时会使用。
- 验证密码：再次输入密码。
- 是否创建仅查看密码：使用仅查看密码登录的用户将无法使用鼠标或键盘控制云服务器。此处输入“n”。

配置 VNC Server

1. 首先停止第一个虚拟桌面。

```
vncserver -kill :1
```

```
root@ecs-9240- :~# vncserver -kill :1
Killing Xtightvnc process ID 2738
root@ecs-9240- :~#
```

2. 修改xstartup文件。

```
vim ~/.vnc/xstartup
```

按“i”进入编辑模式，添加以下内容：

```
#!/bin/sh
xrdp $HOME/.Xresources
startxfce4 &
```

其中，

- 第一个命令“xrdp \$HOME/.Xresources”告诉VNC的GUI框架读取服务器用户的“.Xresources”文件。用户可以在“.Xresources”中更改图形桌面的某项设置，如终端颜色、光标主题和字体渲染。
- 第二个命令“startxfce4 &”告诉服务器启动Xfce。

```
#!/bin/sh
xrdp $HOME/.Xresources
xsetroot -solid grey
#x-terminal-emulator -geometry 80x24+10+10 -ls -title "$VNCDESKTOP Desktop" &
#x-window-manager &
# Fix to make GNOME work
export XKL_XMODMAP_DISABLE=1
/etc/X11/Xsession
startxfce4 &
```

3. 为了保证VNC服务正常启动，需要配置可执行。

```
sudo chmod +x ~/.vnc/xstartup
```

4. 重启VNC Server。

```
vncserver
```

第二次运行vncserver时，它将创建日志文件。

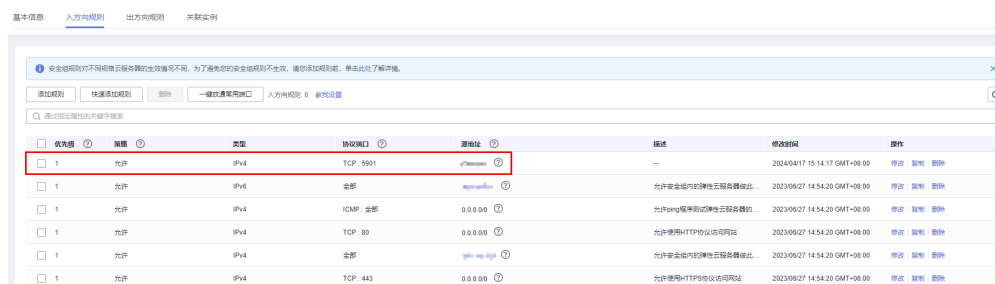
```
root@ecs-9240-:~# vncserver
New 'X' desktop is ecs-9240-:1
Starting applications specified in /root/.vnc/xstartup
Log file is /root/.vnc/ecs-9240-:1.log
root@ecs-9240-:~#
```

回显信息中出现类似这样的一条信息：Log file is /root/.vnc/xxx:1.log。其中，“1”表示当前用户分配的是VNC的第一个虚拟桌面。

云服务器控制台配置

1. 登录云服务器管理控制台。
2. 单击弹性云服务器名称，进入详情页面。
3. 在“安全组”页签，单击“更改安全组规则”，放行5901端口。

图 5-1 更改安全组规则

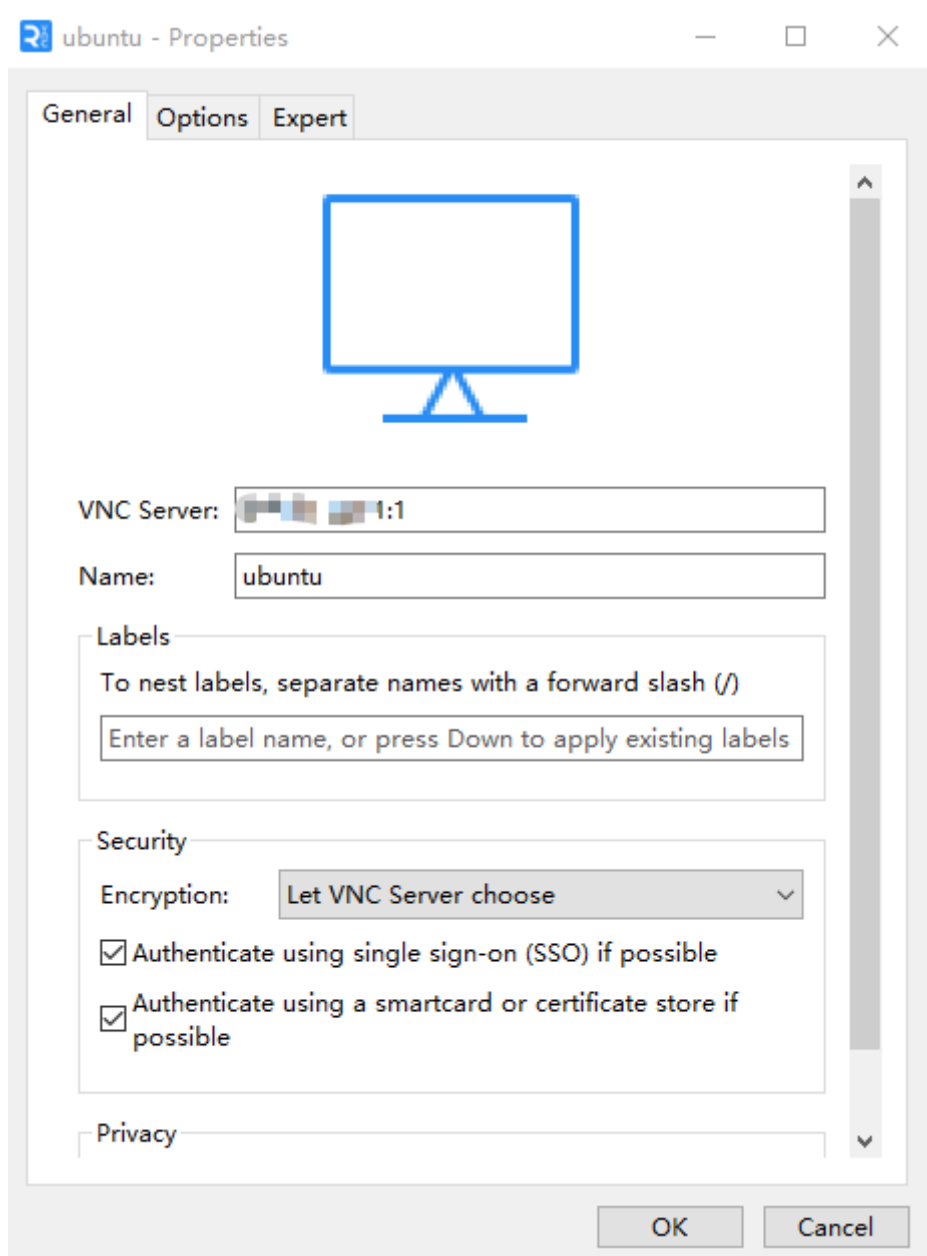


使用 VNC Viewer 连接云服务器

1. 在本地PC打开VNC Viewer客户端，输入云服务器的弹性公网IP.1，设置名称，单击“OK”。

说明

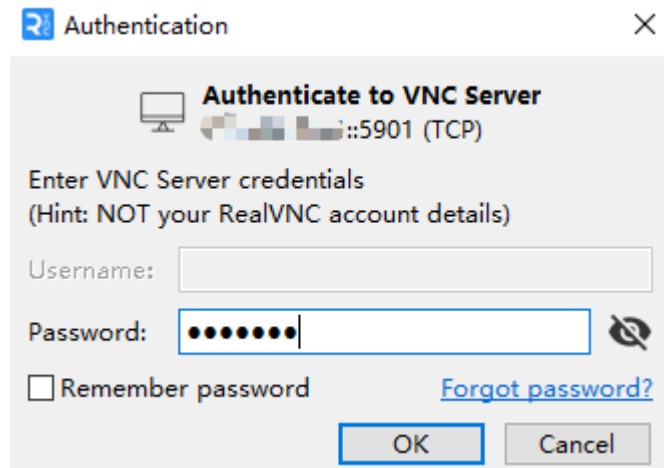
端口号由4中回显的日志文件名决定，如果为“xxx:1.log”，则输入1。



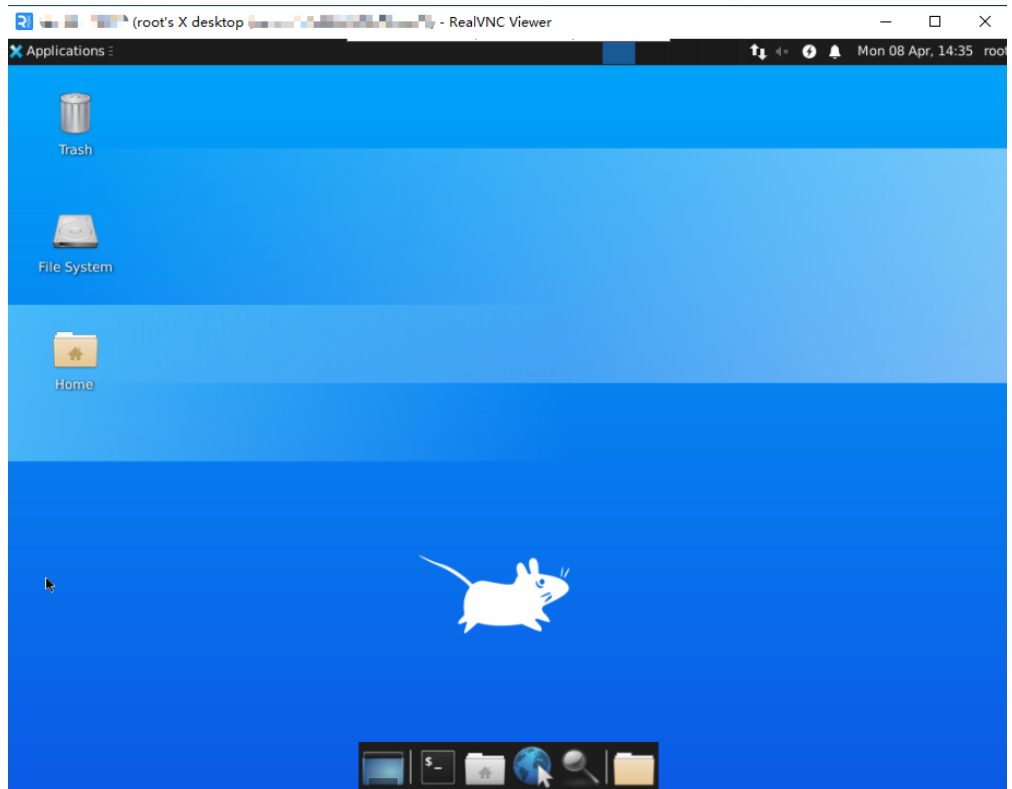
2. 弹出如下界面，直接单击“Continue”。



3. 输入5中设置的密码，单击“OK”。



4. 建立连接后，即可看到Ubuntu 20.04的图形化界面。



6 搭建 Keepalived Nginx 高可用 Web 集群

操作场景

虚拟IP（VIP）主要用于弹性云服务器的主备切换，达到高可用性HA（High Availability）的目的。当主服务器发生故障无法对外提供服务时，动态将虚拟IP切换到备服务器，继续对外提供服务。

本文档以弹性云服务器的CentOS 7.4 64位操作系统为例，实现Keepalived+Nginx高可用WEB集群搭建。

背景知识

WEB集群是由多台Web服务器主机相互联结而形成的一种服务器体系结构，一般结构主要由负载均衡器和后端多个Web节点组成。用户访问请求首先进入负载均衡器，由它根据负载均衡策略将请求分配给后端某个Web应用节点。

本文档采用Nginx实现负载均衡。

网络拓扑

数据规划如下：

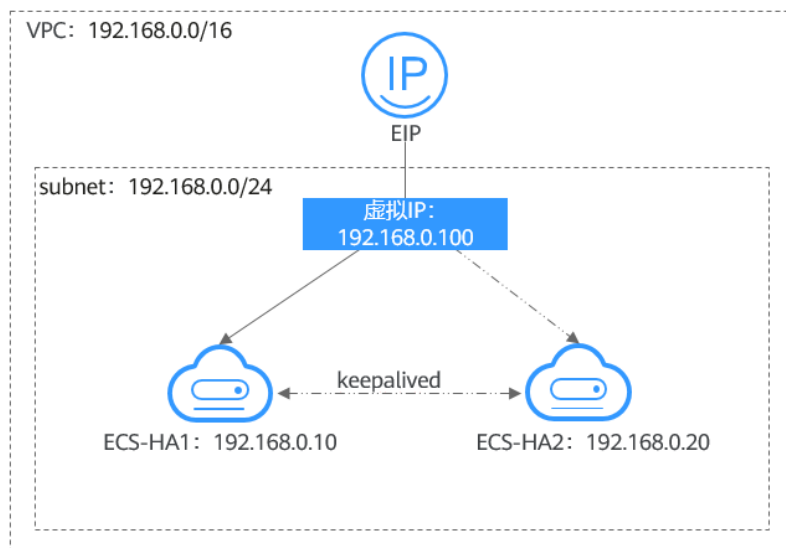
表 6-1 数据规划

序号	产品	数量	规格
1	虚拟私有云（VPC）	1	192.168.0.0/16
	子网（subnet）	1	192.168.0.0/24
2	弹性云服务器（ECS）	2	1vCPUs 1GB CentOS 7.4 64bit
	子网IP（subnet IP）	2	ecs-HA1: 192.168.0.10 ecs-HA2: 192.168.0.20
3	弹性公网IP（EIP）	1	122.xx.xx.189
	虚拟IP（VIP）	1	192.168.0.100

实现方式如下：

- 将2台同子网的弹性云服务器配置Keepalived，一台为主服务器，一台为备份服务器。
- 将这2台弹性云服务器绑定同一个虚拟IP。
- 将虚拟IP与弹性公网IP绑定，从互联网可以访问绑定了该虚拟IP地址的主备云服务器。

图 6-1 组网图



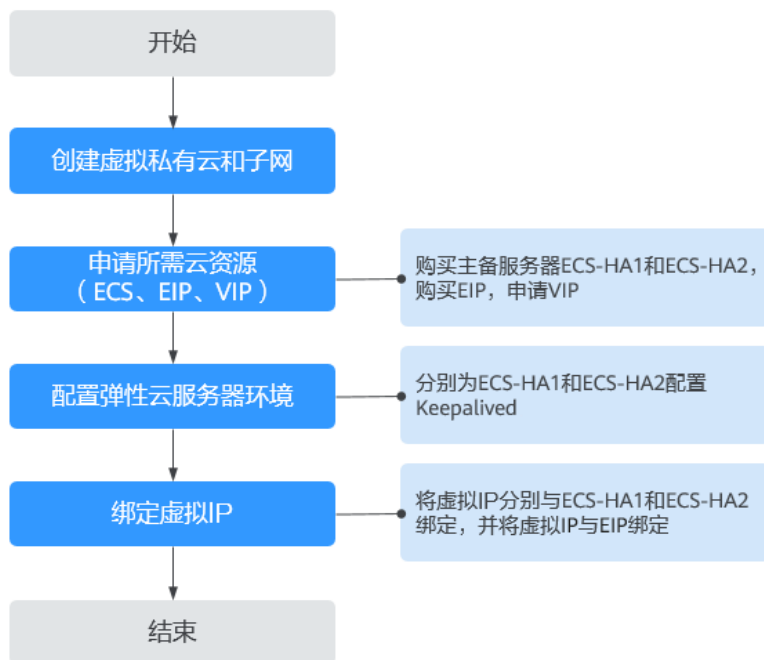
说明

- 实际环境按业务实际需要选择区域购买。
- 所有云资源须在同一区域。

操作步骤

本场景整体操作流程如下：

图 6-2 配置流程



步骤1 创建虚拟私有云和子网

1. 登录管理控制台。
 2. 选择“服务列表 > 网络 > 虚拟私有云”。
 3. 单击“创建虚拟私有云”。
- 根据界面提示配置参数，请参考表6-2。

表 6-2 配置参数

参数	示例
VPC名称	vpc-HA
VPC网段	192.168.0.0/16
子网名称	subnet-HA
子网网段	192.168.0.0/24

4. 单击“立即创建”。

步骤2 申请所需云资源

1. 购买弹性云服务器。
 - a. 登录管理控制台。
 - b. 选择“服务列表 > 计算 > 弹性云服务器”。
 - c. 单击“购买弹性云服务器”。
 - d. 在“购买弹性云服务器”页面，根据界面提示配置参数，请参考表6-1。
 - e. 单击“立即购买”，购买弹性云服务器“ecs-HA1”和“ecs-HA2”。

说明

此示例未选购数据盘，实际环境使用时请按业务需求选购数据盘，并切实考虑两个节点之间业务数据一致性问题。

- f. （可选）配置安全组规则，确保两个ECS内网互通。

此示例中两个ECS位于同一个安全组内，则默认内网互通，不需要执行本操作。

如果您的两个ECS如果位于不同的安全组内，那么需要分别在两个安全组的入方向规则中添加另外一个安全组ID，以确保两个ECS的内网互通，详细说明请参见[不同安全组内的弹性云服务器内网互通](#)。

图 6-3 添加安全组规则



2. 购买弹性公网IP。
 - a. 登录管理控制台。
 - b. 选择“服务列表 > 网络 > 弹性公网IP”。
 - c. 单击“购买弹性公网IP”，根据界面提示配置参数。请参考[表6-1](#)。
3. 申请虚拟IP地址。
 - a. 登录管理控制台。
 - b. 选择“服务列表 > 网络 > 虚拟私有云”。
 - c. 在左侧导航栏选择“子网”。
 - d. 在“子网”列表中，单击子网名称。
 - e. 在“IP地址管理”页签中，单击“申请虚拟IP地址”，根据界面提示配置参数。

步骤3 配置弹性云服务器环境

1. 配置弹性云服务器ecs-HA1环境。
 - a. 绑定弹性公网IP 122.xx.xx.189到ecs-HA1。
 - i. 登录管理控制台。
 - ii. 选择“服务列表 > 计算 > 弹性云服务器”。
 - iii. 在弹性云服务器列表，单击弹性云服务器名称ecs-HA1。
 - iv. 单击“弹性公网IP”页签，并单击“绑定弹性公网IP”。
 - v. 在“绑定弹性公网IP”页面，选择网卡和弹性公网IP后，单击“确定”。

- b. 通过SSH连接ecs-HA1，执行以下命令安装nginx、keepalived软件包及相关依赖包。

```
yum install nginx keepalived -y
```

- c. 执行以下命令，编辑nginx配置文件并保存。

```
vim /etc/nginx/nginx.conf
```

示例如下所示：

```
user root;
worker_processes 1;
#error_log logs/error.log;
#error_log logs/error.log notice;
#error_log logs/error.log info;
#pid logs/nginx.pid;
events {
    worker_connections 1024;
}
http {
    include mime.types;
    default_type application/octet-stream;
    #log_format main '$remote_addr - $remote_user [$time_local] "$request" '
    # '$status $body_bytes_sent "$http_referer" '
    # '"$http_user_agent" "$http_x_forwarded_for"';
    #access_log logs/access.log main;
    sendfile on;
    #tcp_nopush on;
    #keepalive_timeout 0;
    keepalive_timeout 65;
    #gzip on;
    server {
        listen 80;
        server_name localhost;
        #charset koi8-r;
        #access_log logs/host.access.log main;
        location / {
            root html;
            index index.html index.htm;
        }
        #error_page 404 /404.html;
        # redirect server error pages to the static page /50x.html
        error_page 500 502 503 504 /50x.html;
        location = /50x.html {
            root html;
        }
    }
}
```

- d. 执行以下命令，编辑index.html文件内容以演示访问效果并保存。

```
vim /usr/share/nginx/html/index.html
```

示例如下所示：

```
Welcome to ECS-HA1
```

- e. 执行以下命令，设置nginx服务开机自启动并启动服务。

```
systemctl enable nginx
```

```
systemctl start nginx.service
```

- f. Nginx单节点访问验证。

图 6-4 ECS-HA1 访问验证



- g. 执行以下命令，编辑keepalived配置文件并保存。

```
vim /etc/keepalived/keepalived.conf
```

示例如下所示：

```
! Configuration File for keepalived
global_defs {
router_id master-node
}
vrrp_script chk_http_port {
    script "/etc/keepalived/chk_nginx.sh"
    interval 2
    weight -5
    fall 2
    rise 1
}
vrrp_instance VI_1 {
    state MASTER
    interface eth0
    mcast_src_ip 192.168.0.10
    virtual_router_id 51
    priority 101
    advert_int 1
    authentication {
        auth_type PASS
        auth_pass 1111
    }
    unicast_src_ip 192.168.0.10
    virtual_ipaddress {
        192.168.0.100
    }
}
track_script {
    chk_http_port
}
}
```

- h. 执行以下命令，编辑nginx监控脚本并保存。

```
vim /etc/keepalived/chk_nginx.sh
```

示例如下所示：

```
#!/bin/bash
counter=$(ps -C nginx --no-heading|wc -l)
if [ "${counter}" = "0" ]; then
    systemctl start nginx.service
    sleep 2
    counter=$(ps -C nginx --no-heading|wc -l)
    if [ "${counter}" = "0" ]; then
        systemctl stop keepalived.service
    fi
fi
```

```
chmod +x /etc/keepalived/chk_nginx.sh
```

- i. 执行以下命令，设置keepalived服务开机自启动并启动服务。

```
systemctl enable keepalived
```

```
systemctl start keepalived.service
```

2. 配置弹性云服务器ecs-HA2环境。

- a. 将弹性公网IP 122.xx.xx.189从ecs-HA1解绑定。
 - i. 登录管理控制台。
 - ii. 选择“服务列表 > 计算 > 弹性云服务器”。
 - iii. 在弹性云服务器列表，单击弹性云服务器名称ecs-HA1。
 - iv. 单击“弹性公网IP”页签。
 - v. 在弹性公网IP为122.xx.xx.189所在行，单击“解绑”。
- b. 绑定弹性公网IP 122.xx.xx.189到ecs-HA2。
 - i. 登录管理控制台。
 - ii. 选择“服务列表 > 计算 > 弹性云服务器”。
 - iii. 在弹性云服务器列表，单击弹性云服务器名称ecs-HA2。
 - iv. 单击“弹性公网IP”页签。
 - v. 单击“绑定弹性公网IP”
 - vi. 选择网卡和弹性公网IP后，单击“确定”。
- c. 通过ssh连接ecs-HA2，执行以下命令，安装nginx、keepalived软件包及相关依赖包。

yum install nginx keepalived -y

- d. 执行以下命令，编辑nginx配置文件。

```
vim /etc/nginx/nginx.conf
```

内容示例如下：

```
user root;
worker_processes 1;
#error_log logs/error.log;
#error_log logs/error.log notice;
#error_log logs/error.log info;
#pid logs/nginx.pid;
events {
    worker_connections 1024;
}
http {
    include mime.types;
    default_type application/octet-stream;
    #log_format main '$remote_addr - $remote_user [$time_local] "$request" '
    # '$status $body_bytes_sent "$http_referer" '
    # '"$http_user_agent" "$http_x_forwarded_for"';
    #access_log logs/access.log main;
    sendfile on;
    #tcp_nopush on;
    #keepalive_timeout 0;
    keepalive_timeout 65;
    #gzip on;
    server {
        listen 80;
        server_name localhost;
        #charset koi8-r;
        #access_log logs/host.access.log main;
        location / {
            root html;
            index index.html index.htm;
        }
        #error_page 404 /404.html;
        # redirect server error pages to the static page /50x.html
        error_page 500 502 503 504 /50x.html;
        location = /50x.html {
            root html;
        }
    }
}
```

- e. 执行以下命令，编辑index.html文件内容以演示访问效果。

```
vim /usr/share/nginx/html/index.html
```

内容示例如下：

```
Welcome to ECS-HA2
```

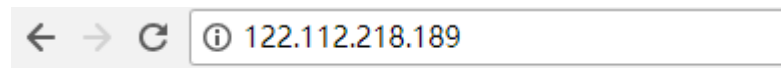
- f. 执行以下命令，设置nginx服务开机自启动并启动服务。

```
systemctl enable nginx
```

```
systemctl start nginx.service
```

- g. Nginx单节点访问测试。

图 6-5 ECS-HA2 验证结果



Welcome to ECS-HA2

- h. 执行以下命令，编辑keepalived配置文件。

```
vim /etc/keepalived/keepalived.conf
```

内容示例如下：

```
! Configuration File for keepalived
global_defs {
router_id master-node
}
vrrp_script chk_http_port {
script "/etc/keepalived/chk_nginx.sh"
interval 2
weight -5
fall 2
rise 1
}
vrrp_instance VI_1 {
state BACKUP
interface eth0
mcast_src_ip 192.168.0.20
virtual_router_id 51
priority 100
advert_int 1
authentication {
auth_type PASS
auth_pass 1111
}
unicast_src_ip 192.168.0.20
virtual_ipaddress {
192.168.0.100
}
track_script {
chk_http_port
}
}
```

- i. 执行以下命令，编辑nginx监控脚本并添加执行权限。

```
vim /etc/keepalived/chk_nginx.sh
```

内容示例如下：

```
#!/bin/bash
counter=$(ps -C nginx --no-heading|wc -l)
if [ "${counter}" = "0" ]; then
systemctl start nginx.service
sleep 2
```

```
counter=$(ps -C nginx --no-heading|wc -l)
if [ "${counter}" = "0" ]; then
    systemctl stop keepalived.service
fi
fi
```

chmod +x /etc/keepalived/chk_nginx.sh

- j. 执行以下命令，设置keepalived服务开机自启动并启动服务。

systemctl enable keepalived

systemctl start keepalived

步骤4 绑定虚拟IP

1. 将弹性公网IP 122.xx.xx.189从ecs-HA2解绑定。
2. 将虚拟IP绑定到弹性云服务器ecs-HA1。
 - a. 登录管理控制台。
 - b. 选择“服务列表 > 网络 > 虚拟私有云”。
 - c. 在左侧导航栏选择“子网”。
 - d. 在“子网”列表中，单击子网名称。
 - e. 选择“IP地址管理”页签，在虚拟IP所在行的操作列下单击“绑定服务器”。
 - f. 在弹出的页面，选择ecs HA1服务器。
3. 参考2将虚拟IP绑定到弹性云服务器ecs-HA2。
4. 将虚拟IP绑定弹性公网IP 122.xx.xx.189。
 - a. 登录管理控制台。
 - b. 选择“服务列表 > 网络 > 虚拟私有云”。
 - c. 在左侧导航栏选择“子网”。
 - d. 在“子网”列表中，单击子网名称。
 - e. 选择“IP地址管理”页签，在虚拟IP所在行的操作列下单击“绑定弹性公网IP”。
 - f. 在弹出的页面，选择弹性公网IP 122.xx.xx.189。
 - g. 单击“确定”。

----结束

验证结果

1. 执行**reboot**分别重启ecs-HA1和ecs-HA2。
2. 通过管理控制台远程登录到ecs-HA1。
3. 执行以下命令，查看虚拟IP是否有绑定到ecs-HA1的eth0网卡上。

ip addr show

如**图6-6**，表示虚拟IP已经绑定到ecs-HA1的eth0网卡上。

图 6-6 查看 ecs-HA1 的虚拟 IP

```
root@ecs-ha1 ~]# ip addr show
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN qlen 1
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP qlen 1000
    link/ether fa:16:3e:a2:c5:72 brd ff:ff:ff:ff:ff:ff
    inet 192.168.0.10/24 brd 192.168.0.255 scope global dynamic eth0
        valid_lft 86066sec preferred_lft 86066sec
    inet 192.168.0.100/32 scope global eth0
        valid_lft forever preferred_lft forever
    inet6 fe80::f816:3eff:fea2:c572/64 scope link
        valid_lft forever preferred_lft forever
```

4. 通过浏览器访问弹性公网IP，验证是否可以访问到ecs-HA1节点上的Web页面。
如图6-7所示，表示访问正常。

图 6-7 访问验证 ecs-HA1



5. 执行以下命令，停止ecs-HA1上的keepalived服务。
systemctl stop keepalived.service
6. 执行以下命令，查看服务器ecs-HA2是否有接管虚拟IP。
ip addr show

图 6-8 查看 ecs-HA2 的虚拟 IP

```
root@ecs-ha2 ~]# ip addr show
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN qlen 1
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP qlen 1000
    link/ether fa:16:3e:79:03:21 brd ff:ff:ff:ff:ff:ff
    inet 192.168.0.20/24 brd 192.168.0.255 scope global dynamic eth0
        valid_lft 04950sec preferred_lft 04950sec
    inet 192.168.0.100/32 scope global eth0
        valid_lft forever preferred_lft forever
    inet6 fe80::f816:3eff:fe79:0321/64 scope link
        valid_lft forever preferred_lft forever
```

7. 通过浏览器访问弹性公网IP，验证是否可以访问到ecs-HA2节点上的Web页面。
如图6-9所示，表示访问正常。

图 6-9 访问验证 ecs-HA2



7 最佳实践-自助建站汇总

简介

为了方便用户搭建网站，本文汇总了华为云常用的网站搭建指导。并提供了指导文档和相关镜像的链接，您可以根据介绍选择需要搭建的网站，并通过链接获取指导文档和镜像，轻松搭建您的网站。

汇总表

表 7-1 网站搭建汇总表

网站搭建方案	搭建方式	操作系统	镜像及所含资源	说明
搭建Discuz论坛网站	手动部署	Linux	公共镜像	Discuz是通用的社区论坛软件系统，基础架构采用流行的web编程组合PHP+MySQL实现。
搭建FTP站点 (Windows)	手动部署	Windows	公共镜像	使用Windows自带的FTP服务搭建FTP站点。
快速构建FTP站点 (Linux)	手动部署	Linux	公共镜像	使用vsftpd软件搭建FTP站点。vsftpd全称是“very secure FTP daemon”，是一款在Linux发行版中使用较多的FTP服务器软件。
基于Tomcat构建Java web环境	手动部署	Linux	公共镜像 <ul style="list-style-type: none">Tomcat 8.5.31JDK 8u171	Tomcat是一个免费的开放源代码的Web应用，是目前比较常用的Web应用程序。可以用来托管常用的Java Web应用程序。

网站搭建方案	搭建方式	操作系统	镜像及所含资源	说明
手工搭建Magento电子商务网站 (Linux)	手动部署	Linux	公共镜像 <ul style="list-style-type: none">MySQL 5.7PHP 7.0Magento 2.1	Magento是一款开源电子商务系统，设计灵活，具有模块化架构体系和丰富的功能，可为大中型站点提供解决方案。
搭建Microsoft SharePoint Server 2016	手动部署	Windows	公共镜像 <ul style="list-style-type: none">Microsoft SQL Server 2014SharePoint Server 2016	Microsoft SharePoint 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。
手工搭建LNMP环境	手动部署	Linux	公共镜像 <ul style="list-style-type: none">Nginx 1.14.0MySQL 5.7PHP 7.0.31	LNMP代表Linux系统下Nginx+MySQL+PHP这种网站服务器架构，Nginx是一个小巧而高效的Linux下的Web服务器软件。
基于WordPress搭建个人网站 (Linux)	手动部署	Linux	公共镜像 <ul style="list-style-type: none">Nginx 1.14.0MySQL 5.7PHP 7.0.31WordPress 4.9.8	使用弹性云服务器的Linux实例手工搭建LNMP平台，并部署WordPress环境。WordPress简称WP，最初是一款博客系统，后逐步演化成一款免费的CMS（内容管理系统/建站系统）。
手工部署Docker (CentOS 7.5)	手动部署	Linux	公共镜像	使用弹性云服务器的Linux实例部署Docker，并提供了Docker常用操作及简单的镜像制作过程。
搭建微信公众号后台——收发文本消息	手动部署	Linux	公共镜像	使用弹性云服务器搭建微信公众号处理后台，使用Python语言编写对应的微信消息处理逻辑代码，接收从微信服务端转发过来的消息，并返回处理结果给最终用户。

网站搭建方案	搭建方式	操作系统	镜像及所含资源	说明
手工部署 GitLab (CentOS 7.2)	手动部署	Linux	公共镜像	使用弹性云服务器的Linux实例手工部署GitLab。GitLab是一个开源的版本管理系统，使用Git作为代码管理工具。
手工部署 RabbitMQ (CentOS 7.4)	手动部署	Linux	公共镜像 <ul style="list-style-type: none">Erlang 8.3RabbitMQ 3.6.9	使用弹性云服务器的Linux实例部署RabbitMQ。RabbitMQ是采用Erlang语言实现AMQP (Advanced Message Queuing Protocol, 高级消息队列协议) 的消息中间件，它最初起源于金融系统，用于在分布式系统中存储转发消息。RabbitMQ凭借其高可靠、易扩展、高可用及丰富的功能特性成为目前非常热门的一款消息中间件。
手工搭建 Ghost博客 (Ubuntu 20.04)	手动部署	Linux	公共镜像 <ul style="list-style-type: none">Nginx 1.14.0MySQL 5.7	Ghost是基于Node.js的开源博客平台，可以为用户提供更加便捷的写作与发布平台，本文指导用户基于华为云弹性云服务器（以Ubuntu 20.04操作系统云服务器为例）部署Ghost博客。
手工部署 Node.js (CentOS 7.2)	手动部署	Linux	公共镜像	使用弹性云服务器的Linux实例部署Node.js。Node.js是一个基于Chrome V8引擎的JavaScript运行环境，用于方便地搭建响应速度快、易于扩展的网络应用。Node.js使用事件驱动，非阻塞I/O模型而得以轻量和高效，非常适合在分布式设备上运行数据密集型的实时应用。

8 搭建 Discuz 论坛网站

8.1 方案概述

应用场景

华为云可以为您提供丰富的解决方案，现以搭建网站业务为例介绍华为云的应用场景。

小型网站一般会部署在单台服务器上，用户对页面的访问，动静态内容的使用，数据库的使用和计算全部是在一台服务器上完成的。当网站业务发展到中型规模时，数据库的访问量剧增，单台服务器配置已不能满足业务要求，此时，可将数据库和网站程序分开部署在不同的服务器上分担性能压力。

以使用华为云搭建某论坛网站为例，在这种场景下有如下需求：

- 将数据节点与业务节点分开部署在不同的服务器上。
- 可针对不同业务量动态调整服务器个数。
- 可自动将流量分发到多台服务器。
- 网站备案。

根据国家规定，如果客户网站所使用的服务器是在中国大陆，就需要进行ICP备案，没有备案的域名不能访问网站。

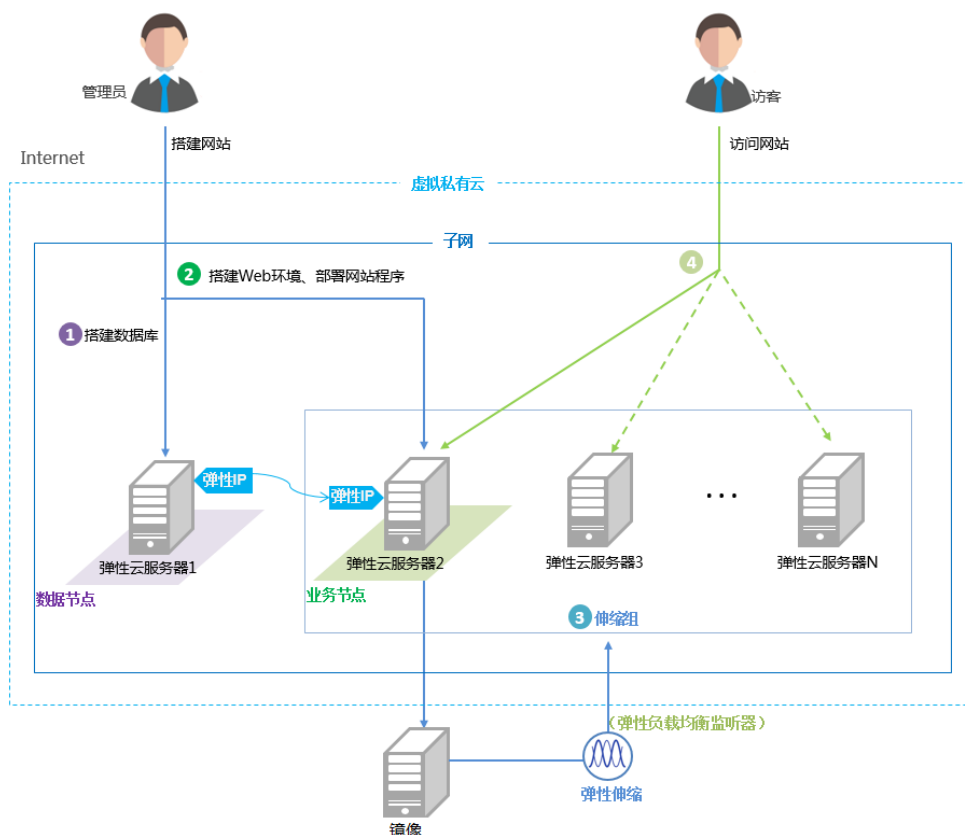
设计方案

针对应用场景的各项需求，使用华为云搭建论坛网站采用[表8-1](#)所示方案。

表 8-1 建站方案及所需服务

需求	华为云方案	服务
将数据节点与业务节点分开部署。	搭建网站： 购买两台弹性云服务器代替传统服务器，分别作为网站的数据节点和基础业务节点。由虚拟私有云为弹性云服务器提供网络资源。购买服务器过程中，用户可以根据实际部署方案的要求，选择是否为云服务器挂载云硬盘作为数据盘。	弹性云服务器 虚拟私有云 云硬盘(可选)
针对不同业务量动态调整服务器个数。	配置特性： 根据业务需求和策略采用弹性伸缩，使用基础业务节点的镜像动态地调整作为业务节点的弹性云服务器实例个数，保证业务平稳健康运行。	弹性伸缩
自动将流量分发到多台服务器。	配置特性： 使用负载均衡将访问流量自动分发到多台业务节点弹性云服务器，扩展应用系统对外的服务能力，实现更高水平的应用程序容错性能。	弹性负载均衡

逻辑架构





1. 为弹性云服务器1绑定弹性公网IP，搭建数据库。
2. 先解绑弹性云服务器1上的弹性公网IP，再将弹性公网IP绑定至弹性云服务器2上，搭建Web环境并部署网站程序。
3. 弹性伸缩可以根据业务量的变化，通过弹性云服务器2的镜像生成弹性伸缩组中的弹性云服务器。弹性伸缩组使用弹性负载均衡监听器。
4. 网站访客通过弹性负载均衡服务的公网IP访问网站。弹性负载均衡服务将访问流量自动分发到多台弹性云服务器。

8.2 购买服务

所需服务

服务参数样例

 虚拟私有云	名称: VPC_DISCUZ VPC网段: 192.168.0.0/16 可用分区: 可用分区1 子网名称: vpc-test 子网网段: 192.168.0.0/24 弹性IP: 安全组: SG-DISCUZ	 弹性负载均衡 (在配置特性阶段申请)	名称: DISCUZ_ELB 类型: 公网 所属VPC: vpc-test 弹性IP: 弹性IP类型: 静态BGP 计费模式: 按带宽计费 公网带宽(Mbit/s): 1Mbit/s
 弹性云服务器1	名称: discuz01 vCPU: 1核 内存: 4G 镜像: CentOS 系统盘: 40G 数据盘: 500G 虚拟私有云: VPC_DISCUZ 安全组: SG-DISCUZ 用户名: root 密码: 私有IP: 192.168.0.26	 弹性云服务器2	名称: discuz02 vCPU: 1核 内存: 4G 镜像: CentOS 系统盘: 40G 数据盘: 100G 虚拟私有云: VPC_DISCUZ 安全组: SG-DISCUZ 用户名: root 密码: 私有IP: 192.168.0.146
 域名注册	名称: discuztest.com		

说明

购买服务及配置特性时，操作步骤中未框选的配置项请保持默认值。

创建虚拟私有云

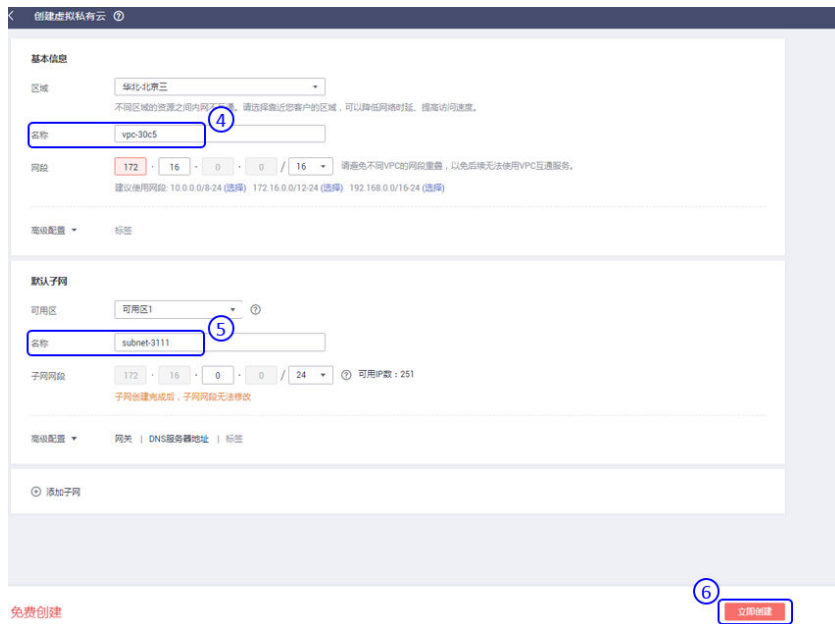
1. 选择“网络 > 虚拟私有云”，单击“创建虚拟私有云”。

图 8-1 创建虚拟私有云



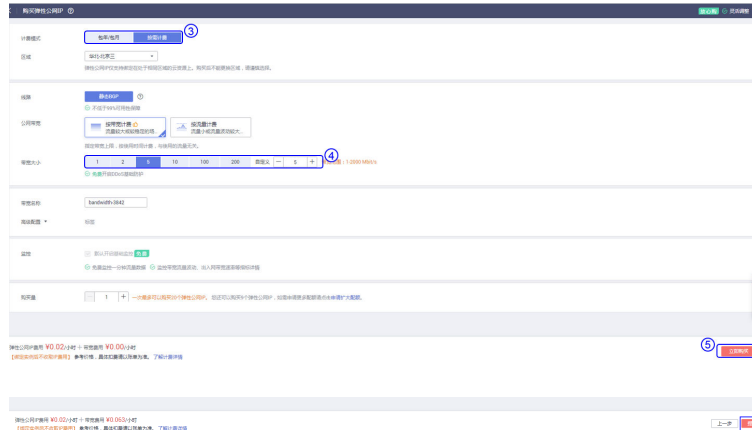
2. 设置相关参数并完成创建。

图 8-2 设置参数

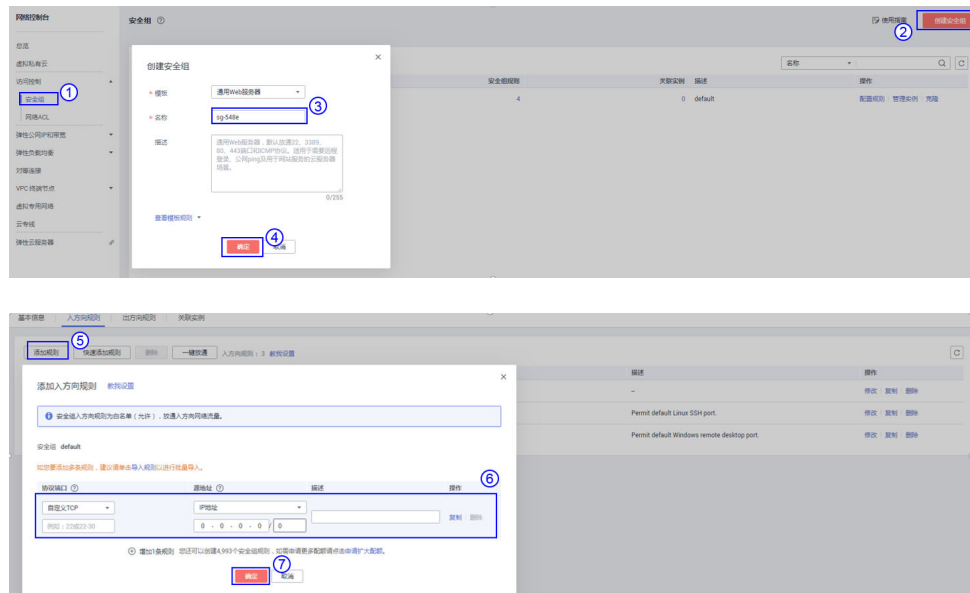


申请弹性公网 IP





创建安全组并添加规则



说明

安全组下默认的规则不能被删除，否则将导致两个服务器彼此通信异常。

购买弹性云服务器

1. 选择“计算 > 弹性云服务器”，单击“购买弹性云服务器”。



2. 设置弹性云服务器参数，提交申请。

图 8-3 基础配置

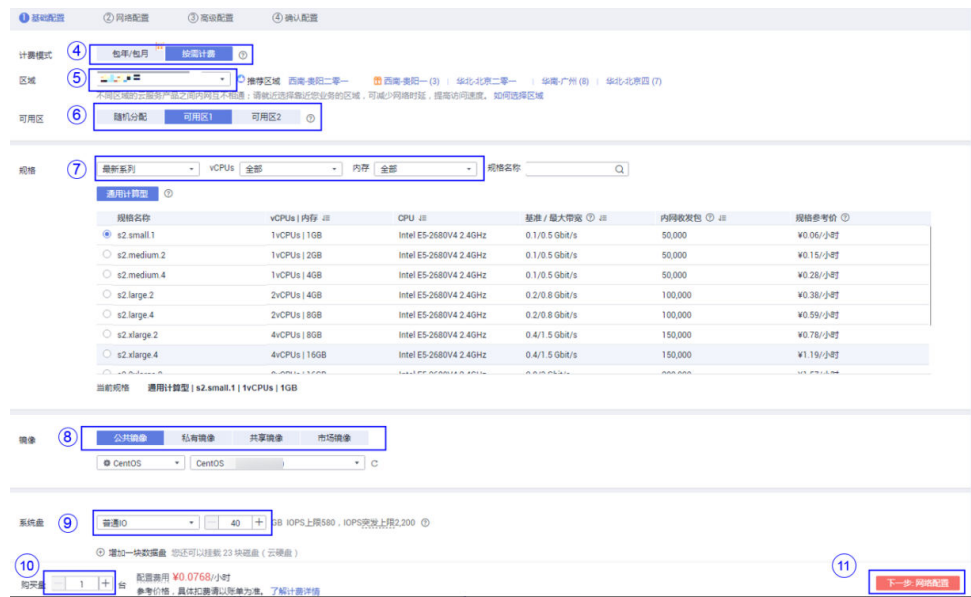


图 8-4 网络配置

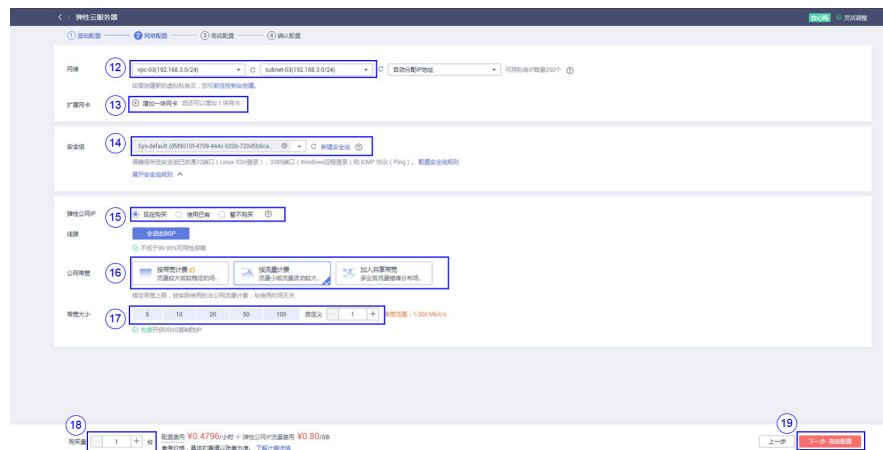


图 8-5 高级配置

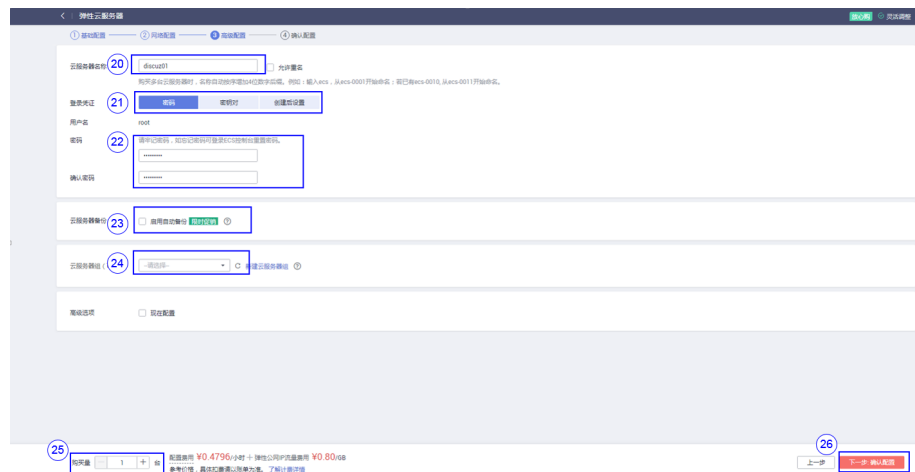
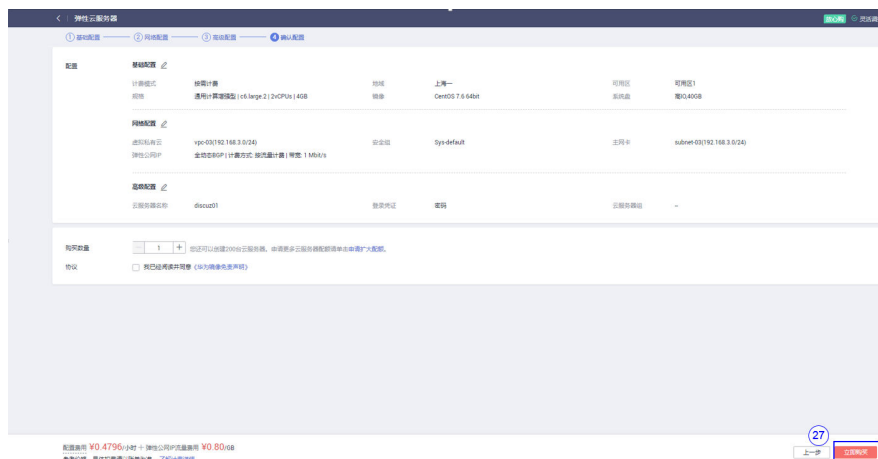


图 8-6 确认配置

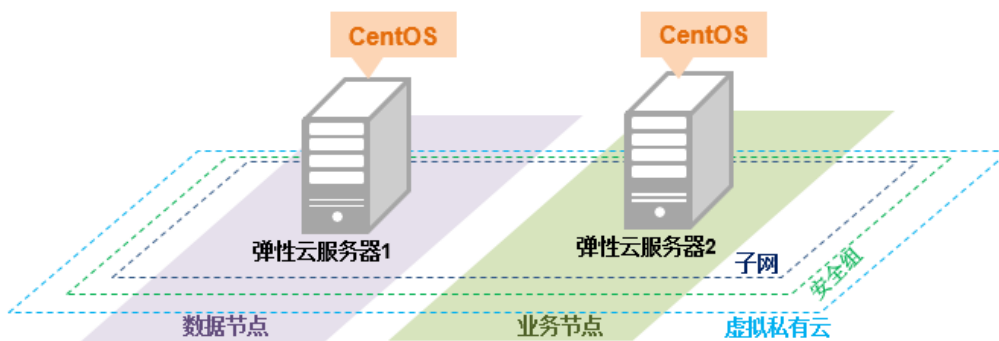


说明

购买时请按照参数样例购买两台弹性云服务器。

8.3 搭建网站

已有服务



服务参数样例

 <p>虚拟私有云</p> <p>可以通过公网IP, 将虚拟私有云连接到公网上。</p> <ul style="list-style-type: none"> 名称: VPC_DISCUZ VPC网段: 192.168.0.0/16 可用分区: 可用分区1 子网名称: vpc-test 子网网段: 192.168.0.0/24 弹性IP: [redacted] 安全组: SG-DISCUZ 	 <p>弹性云服务器1</p> <p>作为数据节点, 搭建数据库。</p> <ul style="list-style-type: none"> 名称: discuz01 vCPU: 1核 内存: 4G 镜像: CentOS 系统盘: 40G 数据盘: 500G 虚拟私有云: VPC_DISCUZ 安全组: SG-DISCUZ 用户名: [redacted] 密码: [redacted] 私有IP: 192.168.0.26 	 <p>弹性云服务器2</p> <p>作为业务节点, 搭建Web环境以及部署网站代码。</p> <ul style="list-style-type: none"> 名称: discuz02 vCPU: 1核 内存: 4G 镜像: CentOS 系统盘: 40G 数据盘: 100G 虚拟私有云: VPC_DISCUZ 安全组: SG-DISCUZ 用户名: [redacted] 密码: [redacted] 私有IP: 192.168.0.146
---	---	--

搭建流程



准备软件

1. WinSCP

WinSCP是在Windows环境下使用SSH的开源图形化SFTP客户端，用于在本地与远程计算机间安全的复制文件。下载地址：[WinSCP](#)

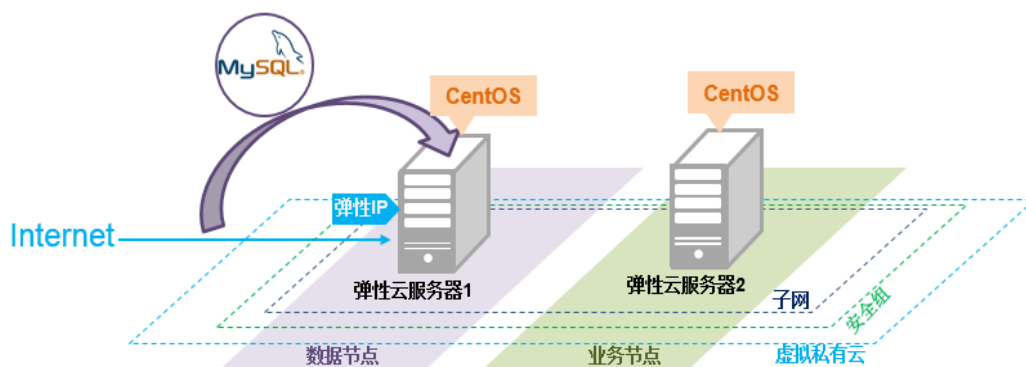
2. Discuz X3.5软件包

请在Discuz官方网站下载Discuz X3.5（简体UTF8）软件包，用于部署网站程序。Discuz相关软件包和资料请[官方网站](#)在获取。

📖 说明

- 以上软件来自第三方网站，仅作参考。建议自行获取需要的版本软件，以应对不同需求。
- 上述软件仅为搭建网站指导示例，若搭建的网站做商业用途请自行准备所需软件。

搭建数据库



安装MySQL

本文档以CentOS 7.2操作系统为例安装MySQL。

1. 远程登录云服务器discuz01，填写用户名和密码。
2. 依次执行以下命令，安装MySQL。

```
wget -i -c http://dev.mysql.com/get/mysql57-community-release-el7-10.noarch.rpm
```

```
yum -y install mysql57-community-release-el7-10.noarch.rpm
```

```
yum -y install mysql-community-server --nogpgcheck
```

配置MySQL

1. 执行以下命令，启动MySQL服务。

```
systemctl start mysqld
```

```
systemctl enable mysqld
```

2. 查看MySQL运行状态。

systemctl status mysqld.service

回显信息如下所示：

```
# systemctl status mysqld.service
● mysqld.service - MySQL Server
  Loaded: loaded (/usr/lib/systemd/system/mysqld.service; enabled; vendor preset: disabled)
  Active: active (running) since Mon 2021-08-23 10:54:55 CST; 7s ago
    Docs: man:mysqld(8)
           http://dev.mysql.com/doc/refman/en/using-systemd.html
  Main PID: 7873 (mysqld)
  CGroup: /system.slice/mysqld.service
          └─7873 /usr/sbin/mysqld --daemonize --pid-file=/var/run/mysqld/mysqld.pid

Aug 23 10:54:49 ecs-adc3-420652-aed6 systemd[1]: Starting MySQL Server...
Aug 23 10:54:55 ecs-adc3-420652-aed6 systemd[1]: Started MySQL Server.
```

3. 执行以下命令，获取安装MySQL时自动设置的root用户密码。

grep 'temporary password' /var/log/mysqld.log

回显如下类似信息。

```
2021-08-16T11:33:37.790533Z 1 [Note] A temporary password is generated for
root@localhost: ;8nPd29lhs,k
```

4. 执行以下命令，并按照回显提示信息进行操作，加固MySQL。

mysql_secure_installation

Securing the MySQL server deployment.

Enter password for user root: #输入上一步骤中获取的安装MySQL时自动设置的root用户密码
The existing password for the user account root has expired. Please set a new password.

New password: #设置新的root用户密码

Re-enter new password: #再次输入密码
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) : N #是否更改root用户密码，输入N

... skipping.
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!
```

5. 执行以下命令，再根据提示输入数据库管理员root账号的密码进入数据库。

```
mysql -u root -p
```

6. 执行以下命令，使用MySQL数据库。

```
use mysql;
```

7. 执行以下命令，查看用户列表。

```
select host,user from user;
```

说明

此命令及以下数据库语句均以分号结尾，请勿忽略。

8. 执行以下命令，刷新用户列表并允许所有IP对数据库进行访问。

```
update user set host='%' where user='root' LIMIT 1;
```

9. 执行以下命令，强制刷新权限。允许同一子网中设置为允许访问的云服务器通过私有IP对MySQL数据库进行访问。

```
flush privileges;
```

10. 执行以下命令，退出数据库。

```
quit
```

11. 执行以下命令，重启MySQL服务。

```
systemctl restart mysqld
```

12. 执行以下命令，设置开机自动启动MySQL服务。

```
systemctl enable mysqld
```

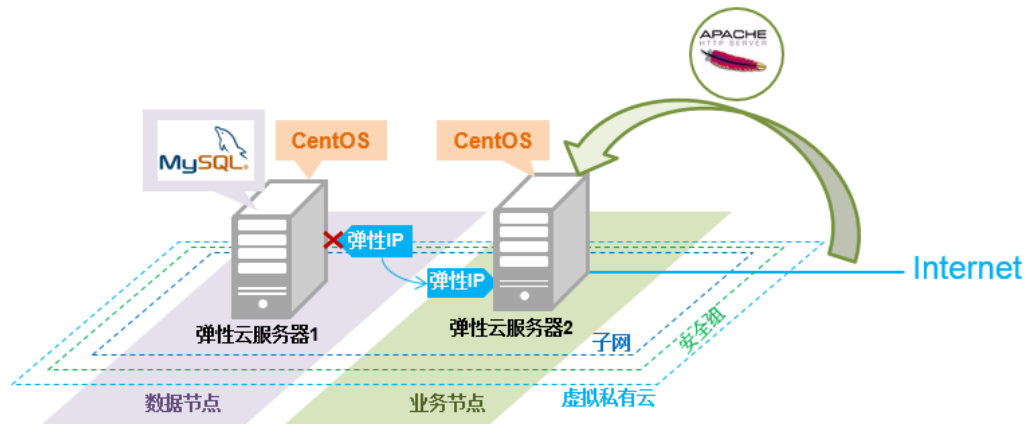
13. 执行以下命令，关闭防火墙。

```
systemctl stop firewalld.service
```

14. 重新查看防火墙状态是否为关闭。

```
systemctl status firewalld
```

搭建 Web 环境



安装Web环境

1. 将弹性公网IP从云服务器discuz01上解绑，并绑定至云服务器discuz02上。





2. 远程登录云服务器discuz02，填写用户名和密码。

3. 依次执行以下命令，安装MySQL。

```
wget -i -c http://dev.mysql.com/get/mysql57-community-release-el7-10.noarch.rpm
```

```
yum -y install mysql57-community-release-el7-10.noarch.rpm
```

```
yum -y install mysql-community-server --nogpgcheck
```

4. 执行以下命令，安装Apache服务器、PHPFastCGI管理器、MySQL客户端和MySQL数据库服务器。

```
yum install -y httpd php php-fpm mysql mysql-server php-mysql
```

回显如下，说明安装成功。

```
Complete!
```

5. 执行以下命令，更新安装Apache服务器、PHPFastCGI管理器、MySQL客户端和MySQL数据库服务器。

```
yum reinstall -y httpd php php-fpm mysql mysql-server php-mysql
```

回显如下，更新安装成功。

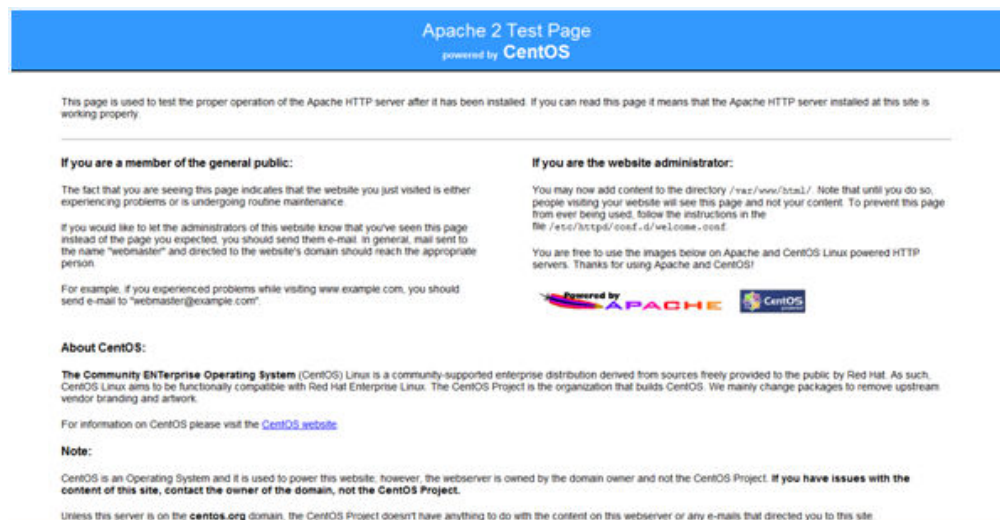
```
Complete!
```

配置Web环境

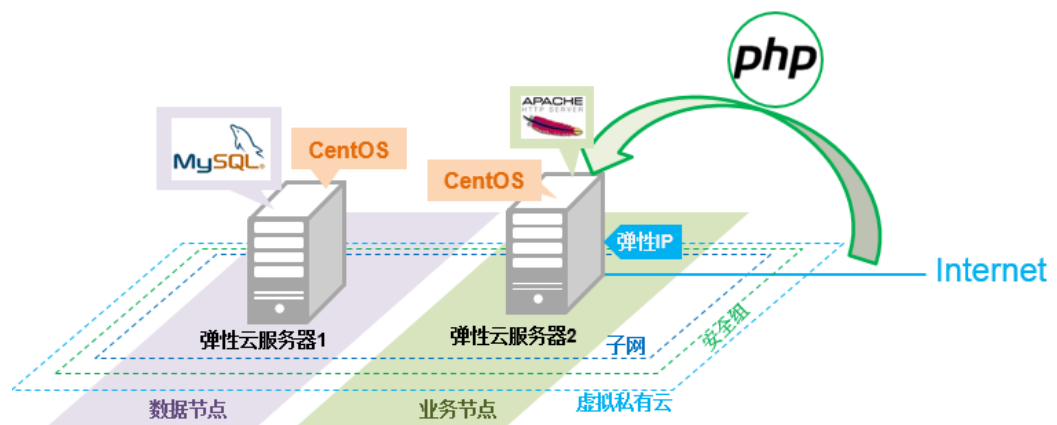
1. 执行以下命令，启动httpd服务。

```
service httpd start
```

2. 执行以下命令，设置开机自动启动httpd服务。
chkconfig httpd on
3. 执行以下命令，启动php-fpm服务。
service php-fpm start
4. 执行以下命令，设置开机自动启动php-fpm服务。
chkconfig php-fpm on
5. 执行以下命令，关闭防火墙。
systemctl stop firewalld.service
6. 执行以下命令，重新查看防火墙状态是否为关闭。
systemctl status firewalld
7. 执行以下命令，启动MySQL服务。
systemctl start mysqld
8. 执行以下命令，设置开机自动启动MySQL服务。
systemctl enable mysqld.service
9. 在浏览器中输入<http://弹性公网IP地址>，即可访问服务器的默认主页。



部署网站代码



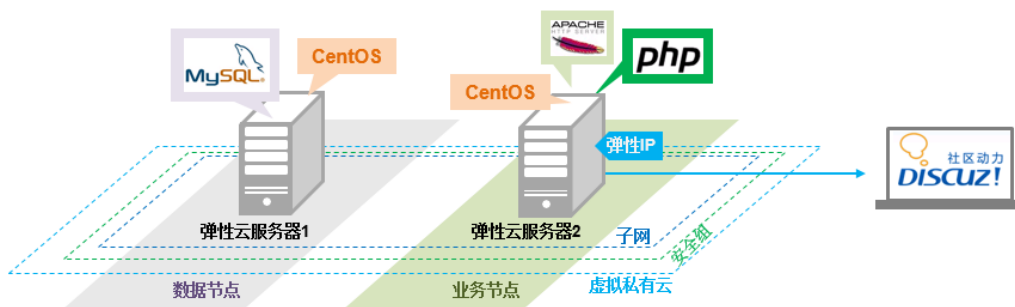
1. 在本地PC上解压缩“Discuz_X3.5_SC_UTF8_20231001.zip”至文件夹“Discuz_X3.5_SC_UTF8_20231001”。
2. 使用WinSCP工具将文件夹“Discuz_X3.5_SC_UTF8_20231001”中的“upload”文件夹中的文件上传至云服务器discuz02的“/var/www/html”路径下。具体操作请参考WinSCP相关使用文档。
3. 默认只有root用户才有写权限，所以需要登录discuz02，执行以下命令将写权限赋予给其他用户。

```
chmod -R 777 /var/www/html
```

4. 在浏览器里输入地址：<http://弹性公网IP地址>进入安装界面。请按照Discuz安装向导进行安装。
 - 数据库服务器地址即为discuz01的私有IP地址。
 - 数据库密码是discuz01配置的数据库管理员root账号的密码。

验证搭建结果

在浏览器中输入：<http://弹性公网IP地址/forum.php> 可登录论坛主页则说明网站搭建成功。



8.4 配置特性

释放弹性公网 IP

用户所能使用的弹性公网IP数目默认为一个，如果需要扩大弹性公网IP的配额则需要另外申请。使用负载均衡功能时，系统会自动分配一个公网IP，该公网IP属于弹性公网IP。为了避免出现弹性公网IP配额不足的情况，建议先释放弹性公网IP再申请弹性负载均衡服务。



创建弹性负载均衡

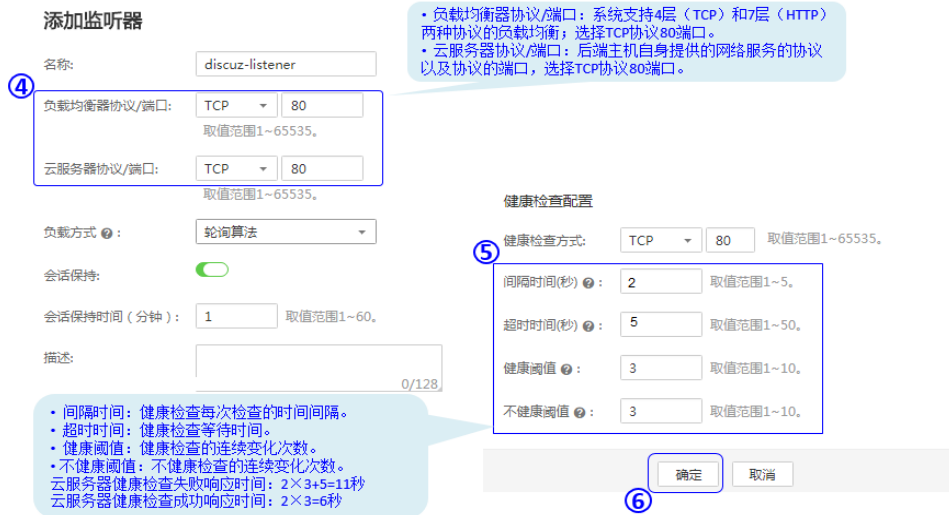
1. 选择“网络 > 弹性负载均衡”，单击“创建负载均衡”。



2. 设置负载均衡器参数，提交申请。



配置弹性负载均衡



制作镜像

1. 选择“计算 > 弹性云服务器”，关闭运行中的弹性云服务器。



2. 制作镜像，设置镜像参数，提交申请。



配置弹性伸缩

1. 选择“计算 > 弹性伸缩”，创建弹性伸缩组及伸缩配置。



* 规格: 全动态BGP 静态BGP BGP(已停售)

* 计费模式: 按带宽计费 按流量计费

11 带宽: 1Mbit/s Mbit/s

登录

* 登录方式: 密钥对 账户密码

12 用户名: root

* 账户密码: 安全等级: 请妥善保管密码，系统无法获取您设置的密码内容。

* 确认密码:

高级配置 [展开](#)

13

14 我已经阅读并同意《华为弹性伸缩服务协议》

15

2. 为弹性伸缩组配置策略。

名称	ID	状态	伸缩配置	当前实例数	期望实例数	最小实例数	最大实例数	操作
as-group-discuz	3eb6...	已启用	as-config-DISCUZ	0	1	1	50	1 查看伸缩策略 停用 更多

2 监控 伸缩实例 伸缩策略 通知 生命周期挂钩

3 您还可以添加10个策略。

添加策略

4 * 策略名称: as-policy-discuz-cpu

* 策略类型: 告警策略 定时策略 周期策略

* 触发告警: 5

执行动作: 个实例

冷却时间 (秒):

6 * 告警名称: as-alarm-cpu

* 触发条件: CPU利用率 最大值 > 70 %

* 监控周期: 5分钟

* 连续出现次数: 3

7

CPU告警策略: 当连续3次监控到CPU利用率超过70%时, 会增加一台弹性云服务器。

添加策略

* 策略名称

* 策略类型 告警策略 定时策略 周期策略

* 触发告警

* 告警名称

* 触发条件 8 %

对于不同的操作系统，监控指标“内存使用率”、“带内网络流出速率”和“带内网络流入速率”是否支持，详细信息请参见《[弹性云服务器用户指南](#)》。

* 监控周期

* 连续出现次数 9

执行动作

冷却时间（秒）

10

3. 增加伸缩实例。

The screenshot shows the Cloud Server Console interface. On the left, the 'Elastic Cloud Servers' section is highlighted with a circled 1. Below it, the 'Start' button is circled with a 3. A table lists two servers: 'discuz02' (status: 关机) and 'discuz01' (status: 运行中). The 'discuz02' row is selected with a circled 2. On the right, a 'Start' dialog box is shown with a circled 4 next to the 'discuz02' row and a circled 4 next to the '确定' button. At the bottom right, the 'Elastic Scaling' button is circled with a 5.

云服务器控制台

① 弹性云服务器

添加伸缩实例前需要开启云服务器。

③ 开机 关机 重启 删除

名称/ID	可用分区	状态
<input checked="" type="checkbox"/> discuz02 905aae76-acbe-473e-8d2d-1	可用区1	① 关机
<input type="checkbox"/> discuz01 be27a01f-ac78-4892-8b31-2	可用区1	运行中

②

开机

④ 确定 取消

⑤ 弹性伸缩



4. 修改伸缩组。



验证配置结果

1. 获取负载均衡服务的弹性公网IP地址。



2. 在浏览器中输入 `http://弹性公网IP地址/forum.php` 可以访问网站则说明网站的特性配置成功。

8.5 访问网站

网站备案

根据国家规定，如果客户网站所使用的服务器是在中国大陆，就需要进行ICP备案，没有备案的域名不能访问网站，具体操作请参见备案中心。

备案受理前置条件如下：

- 有正规注册的域名。
- 确认IP地址为华为公司所属。
- 公司属于非经营性网站性质。

访问方法

访客可以在Internet上通过已备案的域名访问网站。



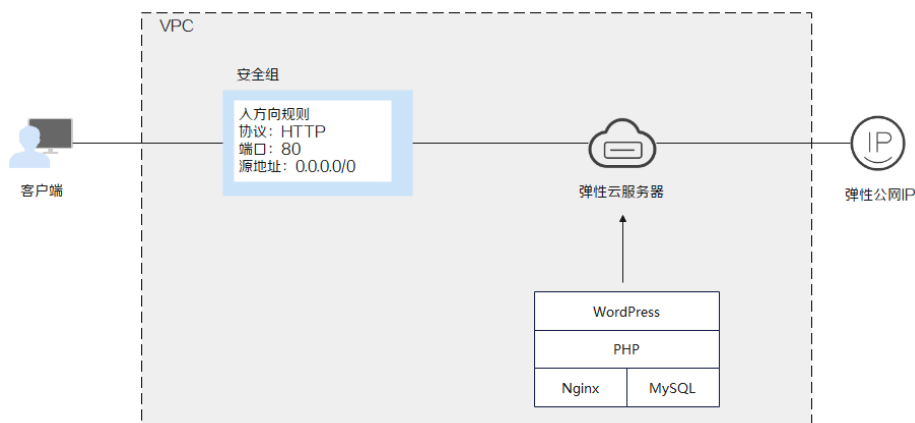
9 基于 WordPress 搭建个人网站 (Linux)

应用场景

本文主要介绍了如何使用弹性云服务器的Linux实例手工搭建LNMP平台，并部署WordPress环境。WordPress简称WP，最初是一款博客系统，后逐步演化成一款免费的CMS（内容管理系统/建站系统）。该指导具体操作以CentOS 7.2 64位操作系统为例。

方案架构

图 9-1 基于 WordPress 搭建个人网站示意图 (Linux)



方案优势

- 快速构建站点，组网架构简单。
- 网站安全易用。

资源和成本规划

表 9-1 资源和成本规划

资源	资源说明	成本说明
虚拟私有云VPC	VPC网段: 192.168.0.0/16	免费
虚拟私有云子网	<ul style="list-style-type: none">可用区: 可用区1子网网段: 192.168.0.0/24	免费
安全组	入方向规则: <ul style="list-style-type: none">协议/应用: HTTP端口: 80源地址: 0.0.0.0/0	免费
弹性云服务器	<ul style="list-style-type: none">计费模式: 包年/包月可用区: 可用区1规格: s6.large.2镜像: CentOS 7.2 64bit系统盘: 40G弹性公网IP: 现在购买线路: 全动态BGP公网带宽: 按流量计费带宽大小: 5 Mbit/s	ECS涉及以下几项费用: <ul style="list-style-type: none">云服务器云硬盘弹性公网IP 具体的计费方式及标准请参考 计费模式概述 。
Nginx	是一个高性能的HTTP和反向代理web服务器。 获取方式: http://nginx.org/packages/centos/7/noarch/RPMS/nginx-release-centos-7-0.el7ngx.noarch.rpm	免费
MySQL	是一款开源的关系数据库软件。 获取方式: http://dev.mysql.com/get/mysql57-community-release-el7-10.noarch.rpm	免费
PHP	是一款开源软件, 用于Web开发。 获取方式: http://rpms.remirepo.net/enterprise/remi-release-7.rpm	免费

资源	资源说明	成本说明
WordPress	是一款开源的博客软件。 获取地址： https://wordpress.org/download/releases/	免费
域名	用于访问搭建的个人网站。	域名的价格以域名注册商所示为准，请参考域名注册商帮助文档。

基于 WordPress 搭建个人网站操作流程

Linux实例基于WordPress手动搭建个人网站的操作步骤如下：

1. [安装Nginx。](#)
2. [安装MySQL。](#)
3. [安装PHP。](#)
4. [测试LNMP环境。](#)
5. [创建数据库。](#)
6. [安装WordPress。](#)
7. [购买域名。](#)
8. [配置域名解析。](#)

实施步骤

安装前准备

- 已购买虚拟私有云和弹性公网IP。
- 如果规划为网站配置域名，需已经购买好相应的域名。
- 弹性云服务器所在安全组添加了如表9-2所示的安全组规则，具体步骤参见[为安全组添加安全组规则](#)。

表 9-2 安全组规则

方向	协议	端口	源地址
入方向	HTTP(80)	80	0.0.0.0/0

- 为了更好的获取和更新系统和软件，建议您更新镜像源为华为云镜像源，详细操作，请参见[如何使用自动化工具配置华为云镜像源\(x86_64和ARM\)?](#)。

操作步骤

步骤1 安装Nginx。

1. 登录弹性云服务器。
2. 执行以下命令，下载对应当前系统版本的Nginx包。

```
wget http://nginx.org/packages/centos/7/noarch/RPMS/nginx-release-centos-7-0.el7ngx.noarch.rpm
```

3. 执行以下命令，建立Nginx的yum仓库。

```
rpm -ivh nginx-release-centos-7-0.el7ngx.noarch.rpm
```

4. 执行以下命令，安装Nginx。

```
yum -y install nginx
```

5. 执行以下命令，启动Nginx并设置开机启动。

```
systemctl start nginx
```

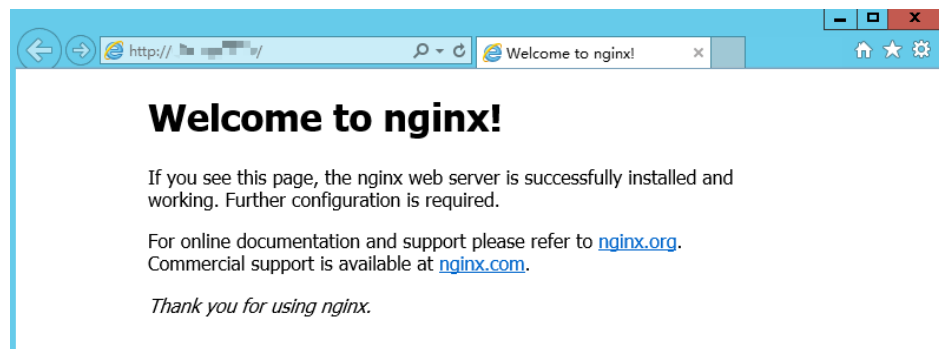
```
systemctl enable nginx
```

6. 查看启动状态。

```
systemctl status nginx.service
```

7. 使用浏览器访问 “http://服务器IP地址”，显示如下页面，说明Nginx安装成功。

图 9-2 测试访问 nginx



步骤2 安装MySQL。

1. 依次执行以下命令，安装MySQL。

```
wget -i -c http://dev.mysql.com/get/mysql57-community-release-el7-10.noarch.rpm
```

```
yum -y install mysql57-community-release-el7-10.noarch.rpm
```

```
yum -y install mysql-community-server --nogpgcheck
```

2. 依次执行以下命令，启动MySQL服务并设置开机自启动。

```
systemctl start mysqld
```

```
systemctl enable mysqld
```

3. 查看MySQL运行状态。

```
systemctl status mysqld.service
```

```
[root@ecs-adc3 ~]# systemctl status mysqld.service
● mysqld.service - MySQL Server
  Loaded: loaded (/usr/lib/systemd/system/mysqld.service; enabled; vendor preset: disabled)
  Active: active (running) since Mon 2021-08-16 19:33:40 CST; 36s ago
  Docs: man:mysqld(8)
        http://dev.mysql.com/doc/refman/en/using-systemd.html
  Main PID: 7916 (mysqld)
  CGroup: /system.slice/mysqld.service
          └─7916 /usr/sbin/mysqld --daemonize --pid-file=/var/run/mysqld/mysqld.pid
```

```
Aug 16 19:33:35 ecs-adc3 systemd[1]: Starting MySQL Server...
```

```
Aug 16 19:33:40 ecs-adc3 systemd[1]: Started MySQL Server.
```

4. 执行以下命令，获取安装MySQL时自动设置的root用户密码。

```
grep 'temporary password' /var/log/mysqld.log
```

回显如下类似信息。

```
2021-08-16T11:33:37.790533Z 1 [Note] A temporary password is generated for root@localhost: ;8nPd29lhs,k
```

5. 执行以下命令，并按照回显提示信息进行操作，加固MySQL。

```
mysql_secure_installation
```

Securing the MySQL server deployment.

Enter password for user root: #输入上一步骤中获取的安装MySQL时自动设置的root用户密码
The existing password for the user account root has expired. Please set a new password.

New password: #设置新的root用户密码

Re-enter new password: #再次输入密码

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) : N #是否更改root用户密码，输入N

... skipping.

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!

步骤3 安装PHP。

1. 执行以下命令，安装epel源。

```
yum install epel-release
```


2. 执行以下命令，安装remi源。
yum install http://rpms.remirepo.net/enterprise/remi-release-7.rpm
3. 执行以下命令，安装Yum源管理工具。
yum install yum-utils
4. 执行以下命令，安装PHP 7.4。
yum -y install php74-php-gd php74-php-pdo php74-php-mbstring php74-php-cli php74-php-fpm php74-php-mysqlnd
5. 执行以下命令，验证PHP的安装版本。
php74 -v
回显如下类似信息：
PHP 7.4.33 (cli) (built: Jun 6 2023 15:55:08) (NTS)
Copyright (c) The PHP Group
Zend Engine v3.4.0, Copyright (c) Zend Technologies
6. 执行以下命令，启动PHP服务并设置开机自启动。
systemctl start php74-php-fpm
systemctl enable php74-php-fpm
7. 修改Nginx配置文件以支持PHP。
 - a. 执行以下命令打开配置文件/etc/nginx/nginx.conf。
vim /etc/nginx/nginx.conf

图 9-3 nginx.conf

```
user nginx;
worker_processes auto;

error_log /var/log/nginx/error.log notice;
pid /var/run/nginx.pid;

events {
    worker_connections 1024;
}

http {
    include /etc/nginx/mime.types;
    default_type application/octet-stream;

    log_format main '$remote_addr - $remote_user [$time_local] "$request" '
        '$status $body_bytes_sent "$http_referer" '
        '"$http_user_agent" "$http_x_forwarded_for"';

    access_log /var/log/nginx/access.log main;

    sendfile on;
    #tcp_nopush on;

    keepalive_timeout 65;

    #gzip on;

    include /etc/nginx/conf.d/*.conf;
}
```

- 从nginx.conf可以看出，配置文件定向至/etc/nginx/conf.d/*.conf。
- b. 输入:quit退出nginx.conf。
 - c. 执行以下命令打开配置文件/etc/nginx/conf.d/default.conf。
vim /etc/nginx/conf.d/default.conf
 - d. 按i键进入编辑模式。

- e. 修改打开的“default.conf”文件。
找到server段落，修改或添加下列配置信息。

```
server {
    listen      80;
    server_name localhost;

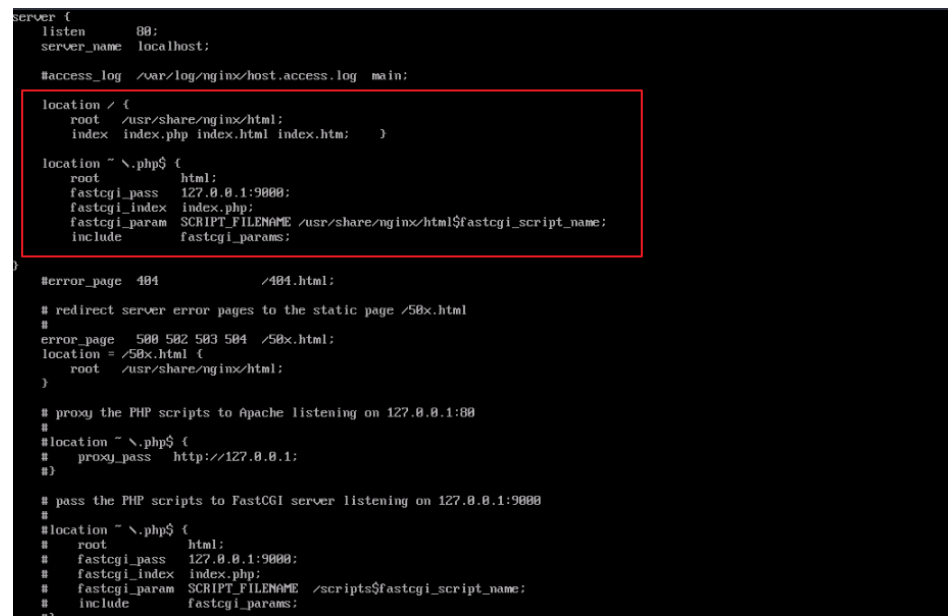
    #access_log /var/log/nginx/host.access.log main;

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

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

如图9-4所示：

图 9-4 修改后截图



```
server {
    listen      80;
    server_name localhost;

    #access_log /var/log/nginx/host.access.log main;

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

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

    #error_page 404              /404.html;

    # redirect server error pages to the static page /50x.html
    #
    error_page 500 502 503 504  /50x.html;
    location = /50x.html {
        root    /usr/share/nginx/html;
    }

    # proxy the PHP scripts to apache listening on 127.0.0.1:80
    #
    location ~ /\.php$ {
        #   proxy_pass http://127.0.0.1;
        #}

    # pass the PHP scripts to FastCGI server listening on 127.0.0.1:9000
    #
    location ~ /\.php$ {
        #   root    html;
        #   fastcgi_pass 127.0.0.1:9000;
        #   fastcgi_index index.php;
        #   fastcgi_param SCRIPT_FILENAME /scripts$fastcgi_script_name;
        #   include    fastcgi_params;
        #}
}
```

- f. 按Esc键退出编辑模式，并输入:wq保存后退出。
8. 执行以下命令，重新载入nginx的配置文件。

service nginx reload

步骤4 浏览器访问测试。

- 在/usr/share/nginx/html目录下创建“info.php”的测试页面。
 - 执行以下命令创建并打开“info.php”的测试文件。
vim /usr/share/nginx/html/info.php
 - 按i键进入编辑模式。
 - 修改打开的“info.php”文件，将如下内容写入文件。

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

- d. 按Esc键退出编辑模式，并输入:wq保存后退出。
2. 使用浏览器访问“http://服务器IP地址/info.php”，显示如下页面，说明环境搭建成功。

PHP Version 7.4.33	
System	Linux ecs-wordpress 3.10.0-1160.53.1.el7.x86_64 #1 SMP Fri Jan 14 13:59:45 UTC 2022 x86_64
Build Date	Jun 6 2023 15:55:08
Build System	Red Hat Enterprise Linux Server release 7.9 (Maipo)
Build Provider	Remi's RPM repository <https://rpms.remirepo.net/> #StandWithUkraine
Server API	FPM/FastCGI
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc/opt/remi/php74
Loaded Configuration File	/etc/opt/remi/php74/php.ini
Scan this dir for additional .ini files	/etc/opt/remi/php74/php.d
Additional .ini files parsed	/etc/opt/remi/php74/php.d/20-bz2.ini, /etc/opt/remi/php74/php.d/20-calendar.ini, /etc/opt/remi/php74/php.d/20-ctype.ini, /etc/opt/remi/php74/php.d/20-curl.ini, /etc/opt/remi/php74/php.d/20-exif.ini, /etc/opt/remi/php74/php.d/20-fileinfo.ini, /etc/opt/remi/php74/php.d/20-ftp.ini, /etc/opt/remi/php74/php.d/20-gd.ini, /etc/opt/remi/php74/php.d/20-gettext.ini, /etc/opt/remi/php74/php.d/20-iconv.ini, /etc/opt/remi/php74/php.d/20-json.ini, /etc/opt/remi/php74/php.d/20-mbstring.ini, /etc/opt/remi/php74/php.d/20-mysqlnd.ini, /etc/opt/remi/php74/php.d/20-pdo.ini, /etc/opt/remi/php74/php.d/20-phar.ini, /etc/opt/remi/php74/php.d/20-sockets.ini, /etc/opt/remi/php74/php.d/20-sqlite3.ini, /etc/opt/remi/php74/php.d/20-tokenizer.ini, /etc/opt/remi/php74/php.d/30-mysqli.ini, /etc/opt/remi/php74/php.d/30-pdo_mysql.ini, /etc/opt/remi/php74/php.d/30-pdo_sqlite.ini
PHP API	20190902
PHP Extension	20190902
Zend Extension	320190902
Zend Extension Build	API320190902,NTS
PHP Extension Build	API20190902,NTS
Debug Build	no
Thread Safety	disabled
Zend Signal Handling	enabled
Zend Memory Manager	enabled
Zend Multibyte Support	provided by mbstring
IPv6 Support	enabled
DTrace Support	available, disabled
Registered PHP Streams	https, ftps, compress.zlib, php, file, glob, data, http, ftp, compress.bzip2, phar
Registered Stream Socket Transports	tcp, udp, unix, udg, ssl, sslv3, tls, tlsv1.0, tlsv1.1, tlsv1.2
Registered Stream Filters	zlib.*, string.rot13, string.toupper, string.tolower, string.strip_tags, convert.*, consumed, dechunk, bzip2.*, convert.iconv.*

步骤5 创建数据库。

1. 执行以下命令，并按照提示信息输入MySQL的root用户，登录到MySQL命令行。
mysql -u root -p
2. 执行以下命令，创建一个新的数据库。
CREATE DATABASE wordpress;
其中，“wordpress”为数据库名，可以自行设置。
3. 执行以下命令，为数据库创建用户并为用户分配数据库的完全访问权限。
GRANT ALL ON wordpress.* TO wordpressuser@localhost IDENTIFIED BY 'xxxxx';
其中，“wordpressuser”为数据库用户名，“xxxxx”为对应的账户密码，可以自行设置。
4. 执行以下命令，退出MySQL命令行。
exit
5. （可选）依次执行以下命令，验证数据库和用户是否已成功创建，并退出MySQL命令行。
mysql -u wordpressuser -p

SHOW DATABASES**exit**其中，“*wordpressuser*”为刚刚创建的数据库用户名。**步骤6** 安装WordPress。

1. 请自行获取WordPress软件包并上传至/usr/share/nginx/html目录。操作方法请参见[本地Windows主机使用WinSCP上传文件到Linux云服务器](#)。

后续操作软件包以“wordpress-6.4.2-zh_CN.tar.gz”为例。

2. 执行以下命令，进入/usr/share/nginx/html/目录，解压缩软件包。

cd /usr/share/nginx/html**tar zxvf wordpress-6.4.2-zh_CN.tar.gz**

解压后生成一个“wordpress”的文件夹。

3. 执行以下命令，进入WordPress安装目录，将wp-config-sample.php文件复制到wp-config.php文件中，并将原先的示例配置文件保留作为备份。

cd /usr/share/nginx/html/wordpress**cp wp-config-sample.php wp-config.php**

4. 执行以下命令，打开并编辑新创建的配置文件。

vim wp-config.php

5. 按i键进入编辑模式。找到文件中MySQL的部分，并将相关配置信息修改为[步骤5](#)中的内容。

图 9-5 修改 MySQL 配置

```
// ** Database settings - You can get this info from your web host ** //  
/** The name of the database for WordPress */  
define( 'DB_NAME', 'wordpress' );  
  
/** Database username */  
define( 'DB_USER', 'wordpressuser' );  
  
/** Database password */  
define( 'DB_PASSWORD', '12345678' );  
  
/** Database hostname */  
define( 'DB_HOST', 'localhost' );  
  
/** Database charset to use in creating database tables. */  
define( 'DB_CHARSET', 'utf8' );  
  
/** The database collate type. Don't change this if in doubt. */  
define( 'DB_COLLATE', '' );
```

6. 按Esc键退出编辑模式，并输入:wq保存后退出。
7. 浏览器访问“http://服务器IP地址/wordpress”进入安装向导。
8. 设置站点标题、管理员账号、密码和邮箱，并单击“安装 WordPress”。

表 9-3 配置参数

参数	说明
站点标题	WordPress网站名称。

参数	说明
用户名	WordPress管理员名称。
密码	可以使用默认的密码或者自定义密码。 请勿重复使用现有密码，并确保将密码保存在安全的位置。
您的电子邮箱地址	接收通知的电子邮件地址。

- 安装完成后，弹出安装成功界面。
- 单击“登录”，或者浏览器访问“http://服务器IP地址/wordpress/wp-login.php”，输入用户名/电子邮箱地址和密码后单击“登录”，进入WordPress。

步骤7 购买域名。

为了便于网站的访问和使用，可以给网站设置一个单独的域名，使用域名访问网站。首先，需要在域名注册商处获得授权的域名。

步骤8 配置域名解析。

配置域名解析后才能使用注册的域名访问网站。具体操作请参见[配置公网域名](#)。

例如，配置的域名为“www.example.com”，配置成功后，可在浏览器地址栏中输入“http://www.example.com”访问部署的网站。

----结束

10 快速构建 FTP 站点 (Windows)

简介

本文介绍了如何使用弹性云服务器的Windows实例搭建FTP站点。该指导具体操作以Windows Server 2012 R2为例。

Windows实例搭建FTP站点具体操作步骤如下：

1. 添加IIS以及FTP服务角色。
2. 创建FTP用户名及密码。
3. 设置共享文件的权限。
4. 添加及设置FTP站点。
5. (可选)配置FTP防火墙支持。
6. 设置安全组及防火墙。
7. 客户端测试。

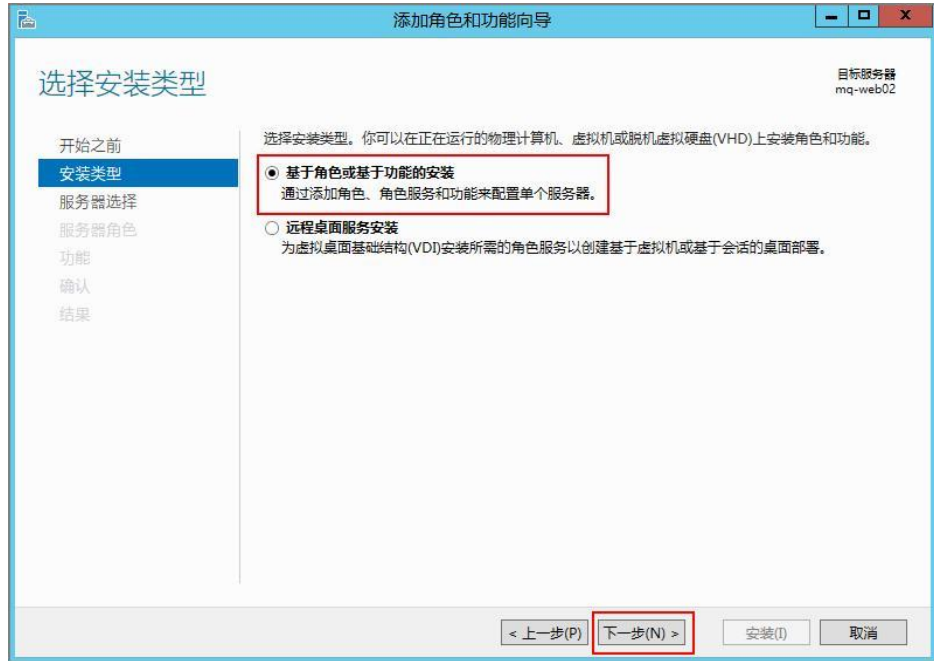
操作步骤

步骤1 添加IIS以及FTP服务角色。

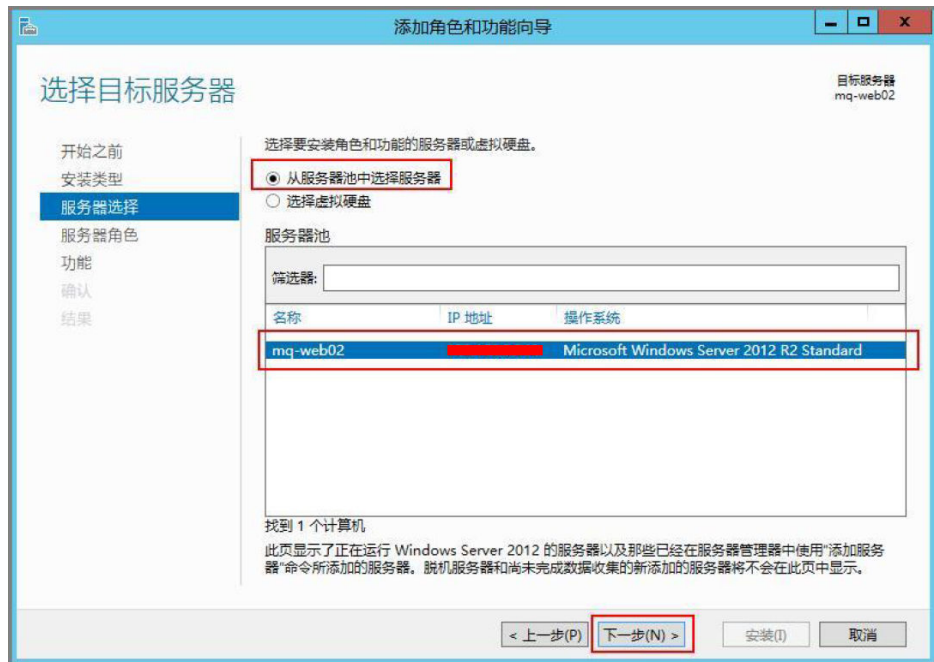
1. 登录弹性云服务器。
2. 选择“开始 > 服务器管理器”。
3. 单击“添加角色和功能”



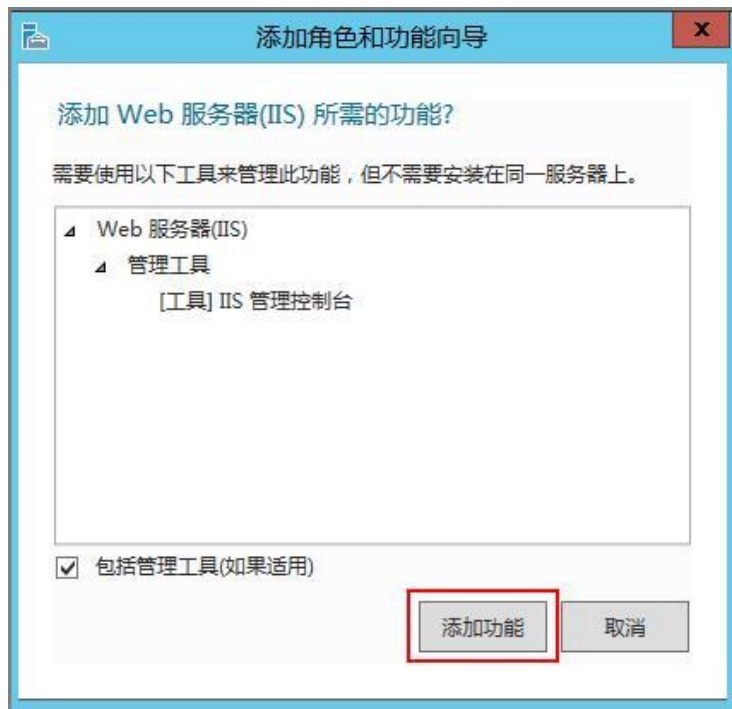
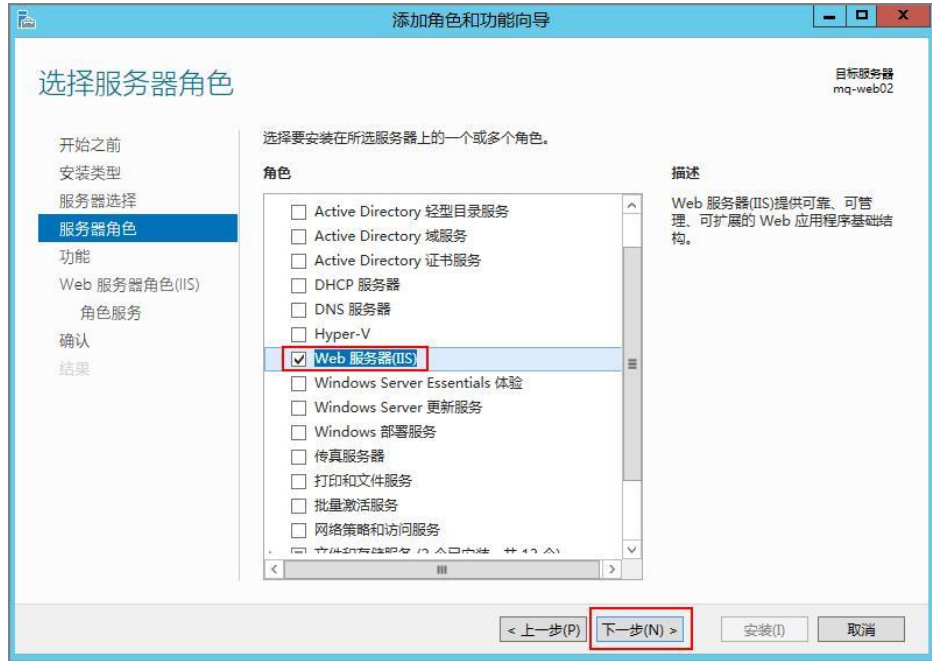
4. 在弹出的“开始之前”对话框中，单击“下一步”。
5. 选择“基于角色或基于功能的安装”，单击“下一步”。



6. 选择需要部署FTP的服务器，单击“下一步”。



7. 选择“Web服务器 (IIS)”，并在弹出的对话框中单击“添加功能”，然后单击“下一步”。



8. 连续单击“下一步”，到“角色服务”页面。
9. 选择“FTP服务器”以及“IIS管理控制台”，单击“下一步”。



10. 单击“安装”，开始部署服务角色。
11. 安装完成后，单击“关闭”。

步骤2 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，可以不创建FTP用户和密码。

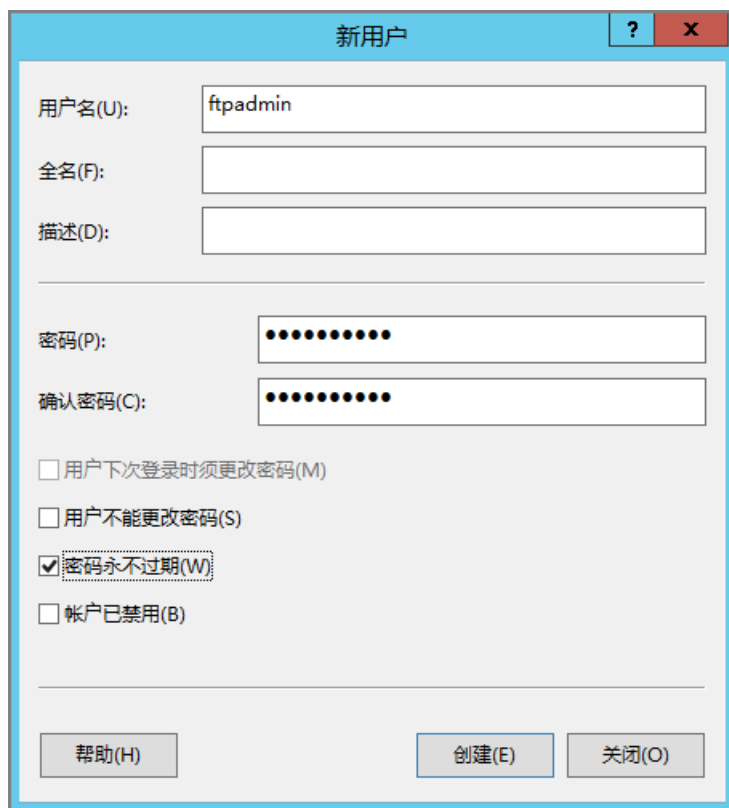
1. 在“服务器管理器”中，选择“仪表板 > 工具 > 计算机管理”。



2. 选择“系统工具 > 本地用户和组 > 用户”，在右侧空白处右击，并选择“新用户”。



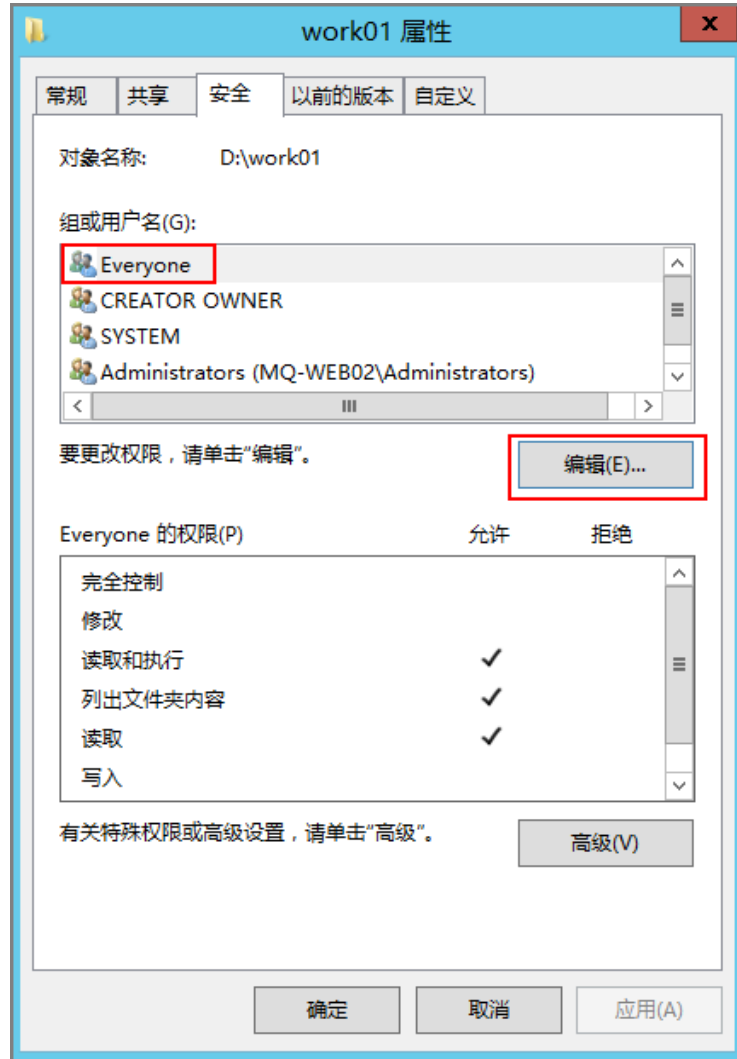
3. 设置“用户名”和“密码”，此处用户名以“ftpadmin”为例。



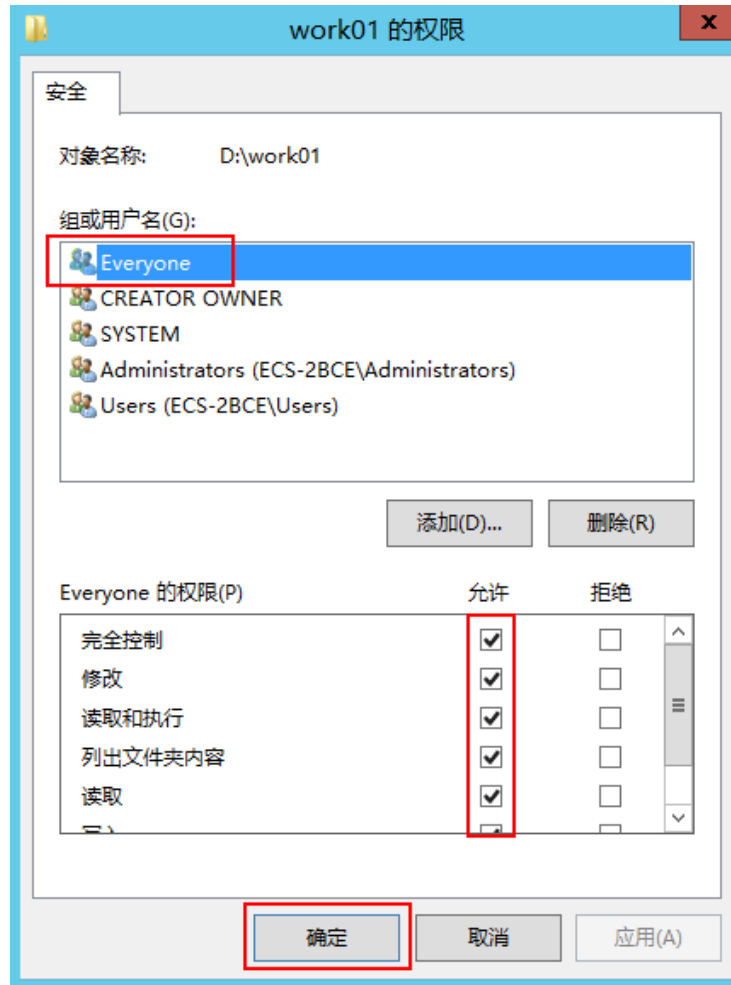
步骤3 设置共享文件的权限。

需要在FTP站点为共享给用户的文件夹设置访问及修改等权限。

1. 在服务器上创建一个供FTP使用的文件夹，选择文件夹，并单击右键选择“属性”。
此处以“work01”文件夹为例，且文件夹包含了需共享的文件test.txt。
2. 在“安全”页签，选择“Everyone”，单击“编辑”。
如果没有“Everyone”用户可以直接选择，需要先进行添加，添加方法请参见[FAQ](#)。



3. 选择“Everyone”，然后根据需要，选择“Everyone”的权限，并单击“确定”。
此处以允许所有权限为例。

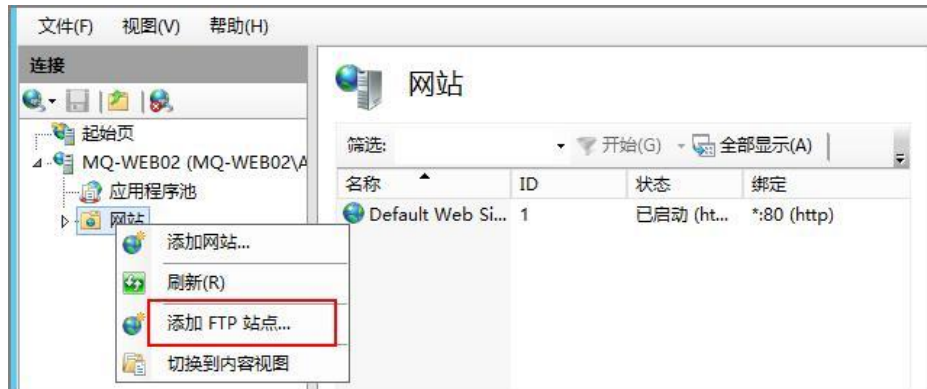


步骤4 添加及设置FTP站点。

1. 在“服务器管理器”中，选择“仪表板 > 工具 > Internet Information Services (IIS)管理器”。

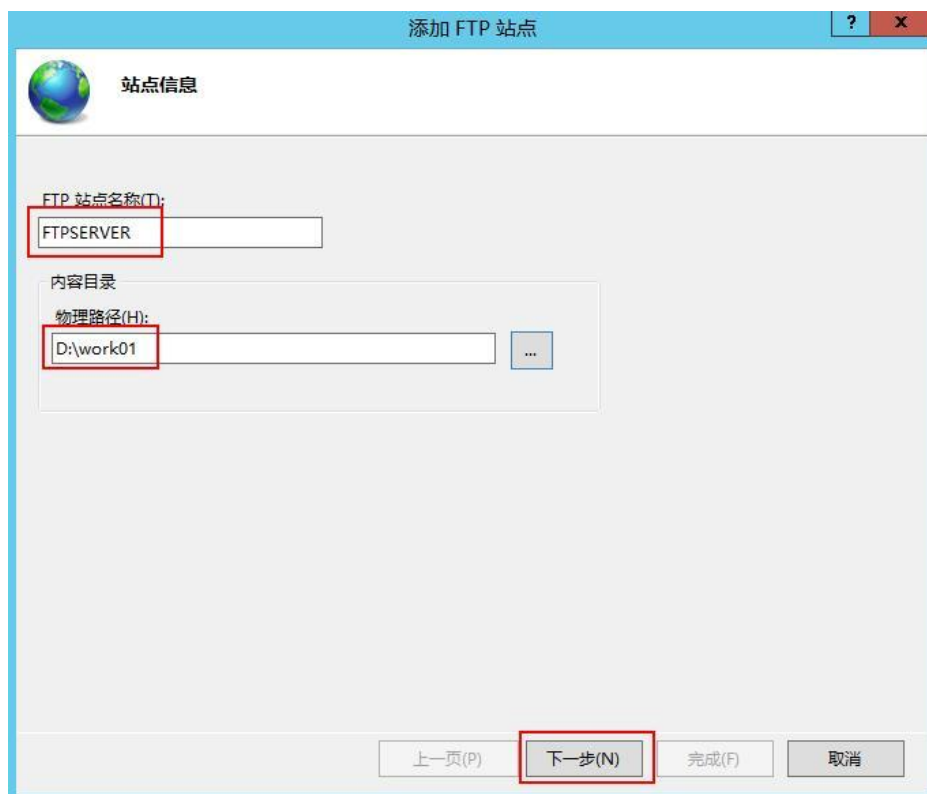


2. 选择“网站”并单击右键，然后选择“添加FTP站点”。



3. 在弹出的窗口中，填写FTP站点名称及共享文件夹的物理路径，然后单击“下一步”。

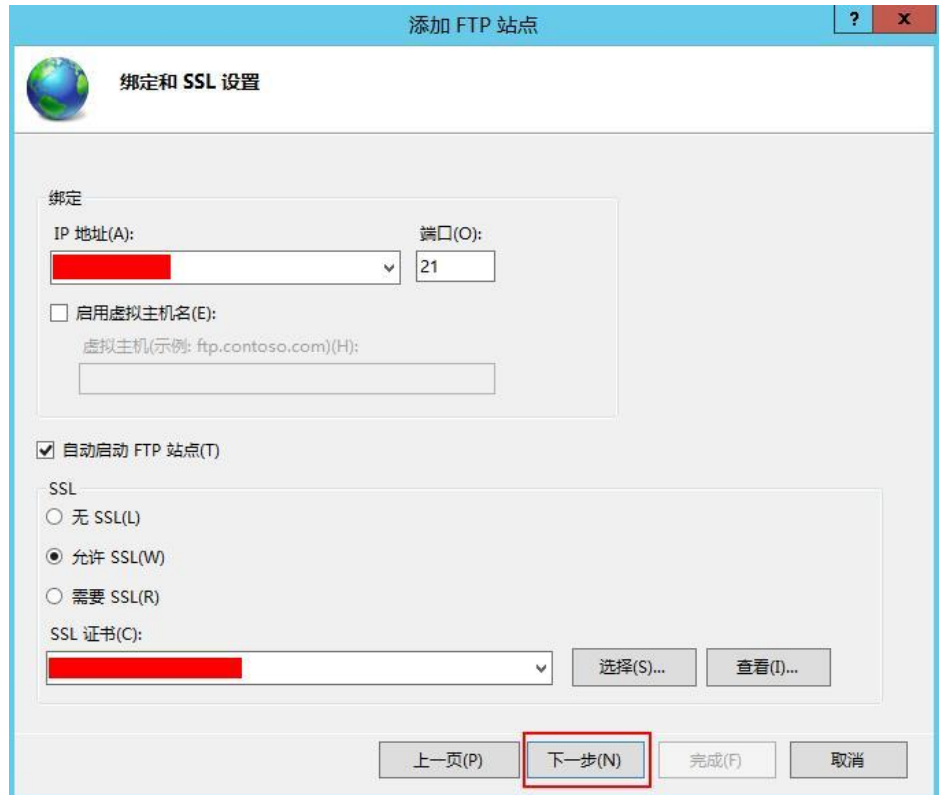
此处站点名称以“FTPSERVER”为例。



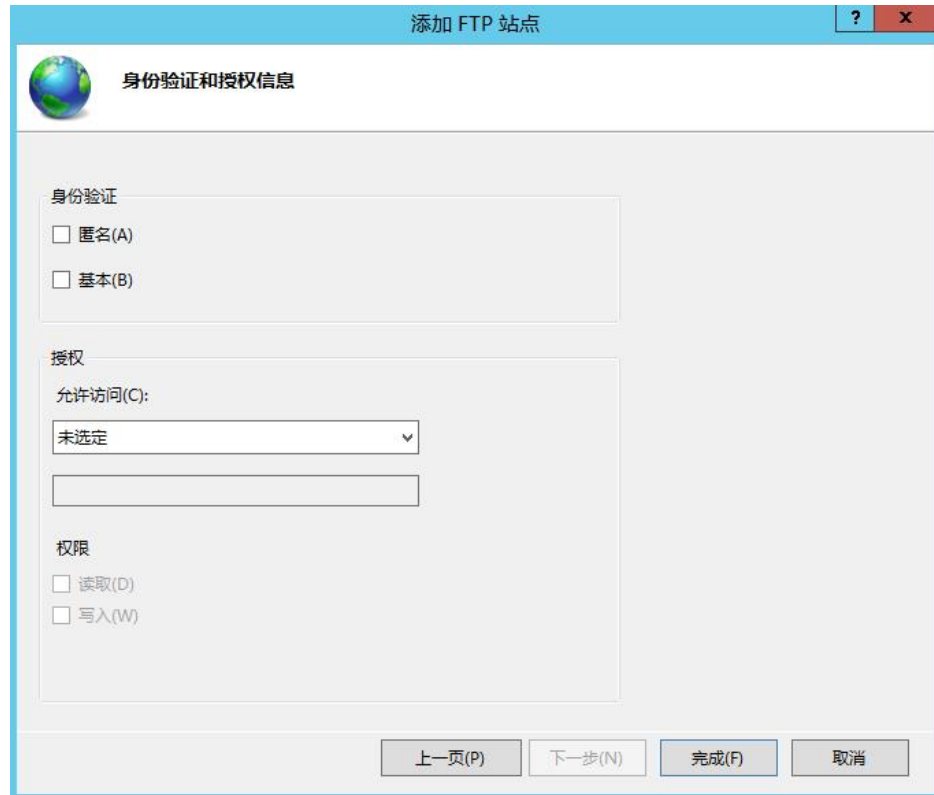
4. 输入该弹性云服务器的私有IP地址以及端口号，并设置SSL，单击“下一步”。
 - 端口号 默认为21，也可自行设置。
 - SSL根据需要进行设置。
 - **无**：不需要SSL加密。
 - **允许**：允许FTP服务器与客户端的非SSL和SSL连接。
 - **需要**：需要对FTP服务器和客户端之间的通信进行SSL加密。

📖 说明

当SSL选择“允许”和“需要”时，需要选择SSL证书。可以选择已有的SSL证书，也可以制作一个SSL证书，具体制作证书的方法请参见3。

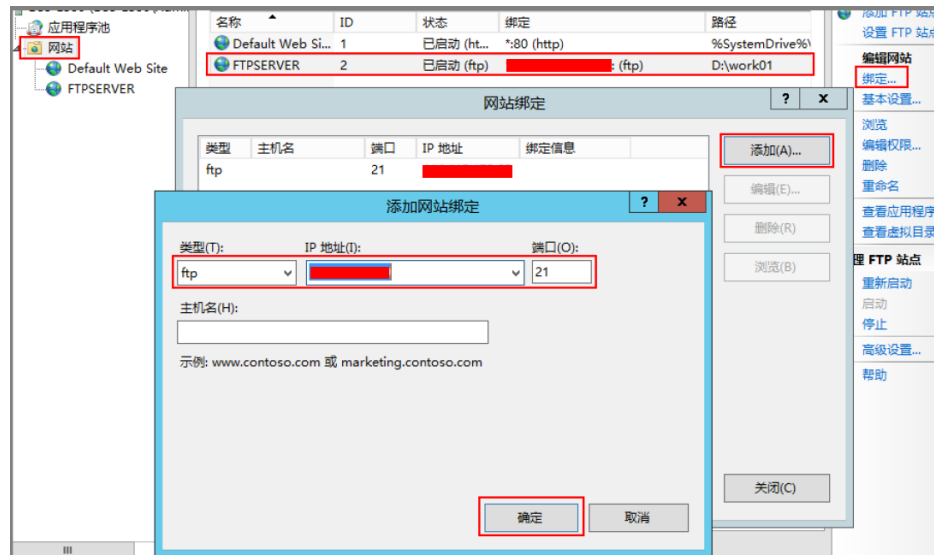


5. 设置身份认证和授权信息，并单击“完成”。
 - 身份认证
 - **匿名**：允许任何仅提供用户名“anonymous”或“ftp”的用户访问内容。
 - **基本**：需要用户提供有效用户名和密码才能访问内容。但是基本身份验证通过网络传输密码时不加密，因此建议在确认客户端和FTP服务器之间的网络连接安全时使用此身份验证方法。
 - 授权
 - 允许访问
 - **所有用户**：所有用户均可访问相应内容。
 - **匿名用户**：匿名用户可访问相应内容。
 - **指定角色或用户组**：仅指定的角色或用户组的成员才能访问相应内容。如果选择此项，需要在下面输入框中输入指定的角色或用户组。
 - **指定用户**：仅指定的用户才能访问相应内容。如果选择此项，需要在下面输入框中输入指定的用户。
 - 权限：选择经过授权的用户“读取”和“写入”权限。



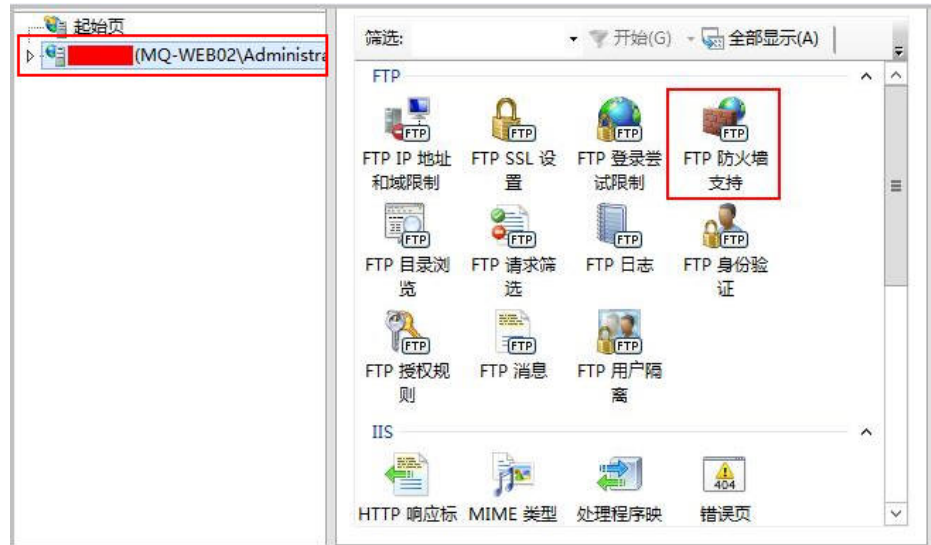
6. 绑定弹性云服务器的私网IP。

选择“网站”，选中创建的FTP站点，单击“绑定”；在弹出的“网站绑定”窗口单击“添加”，然后在弹出的窗口中添加弹性云服务器的私网IP地址，并单击“确定”。



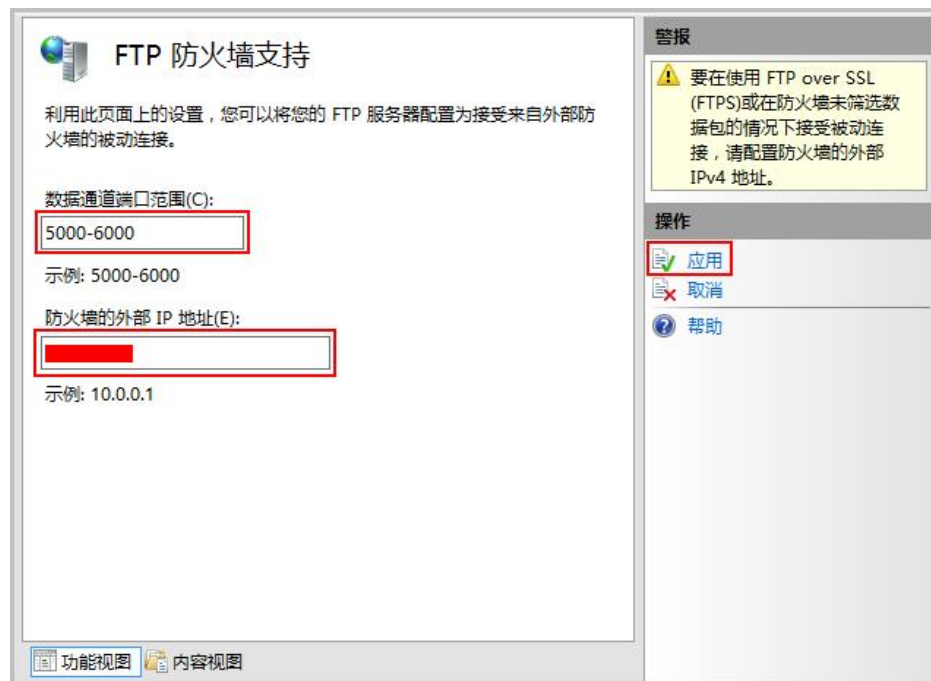
步骤5 (可选) 配置FTP防火墙支持。

- 如果需要使用FTP服务器的被动模式，则需要配置FTP防火墙支持。
 - 如果华为云上的服务器需要通过公网IP地址访问华为云上的实例搭建的FTP服务器时，需要配置FTP服务器的被动模式。
1. 双击“FTP防火墙支持”，打开FTP防火墙支持的配置界面。



2. 配置相关参数，并单击“应用”。

- 数据通道端口范围：指定用于被动连接的端口范围。可指定的有效端口范围为1025-65535。请根据实际需求进行设置。
- 防火墙的外部IP地址：输入该弹性云服务器的公网IP地址。



3. 重启云服务器使防火墙配置生效。

步骤6 设置安全组及防火墙。

搭建好FTP站点后，需要在弹性云服务器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[为安全组添加安全组规则](#)。放通的端口请参考[表10-1](#)。

如果配置了“FTP防火墙支持”，需要在安全组中同时放行FTP站点使用的端口和FTP防火墙使用的数据通道端口。

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

表 10-1 设置安全组规则

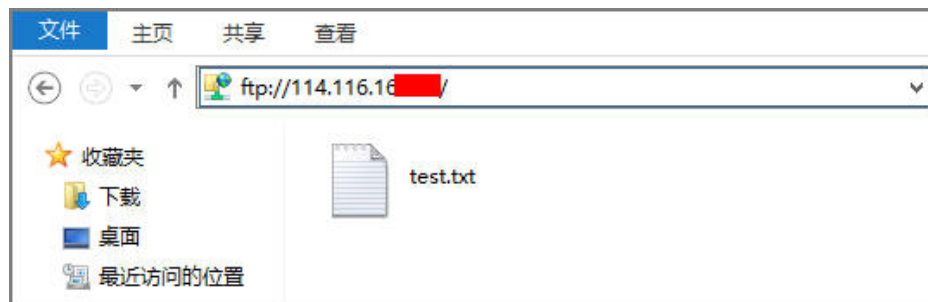
FTP模式	方向	协议	端口	源地址
主动模式	入方向	TCP	20端口和21端口	0.0.0.0/0
被动模式	入方向	TCP	21端口和1024 - 65535间的端口 (例如5000-6000)	0.0.0.0/0

步骤7 客户端测试。

打开客户端的计算机，在路径栏输入“ftp://FTP服务器IP地址:FTP端口”（如果不填端口则默认访问21端口）。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

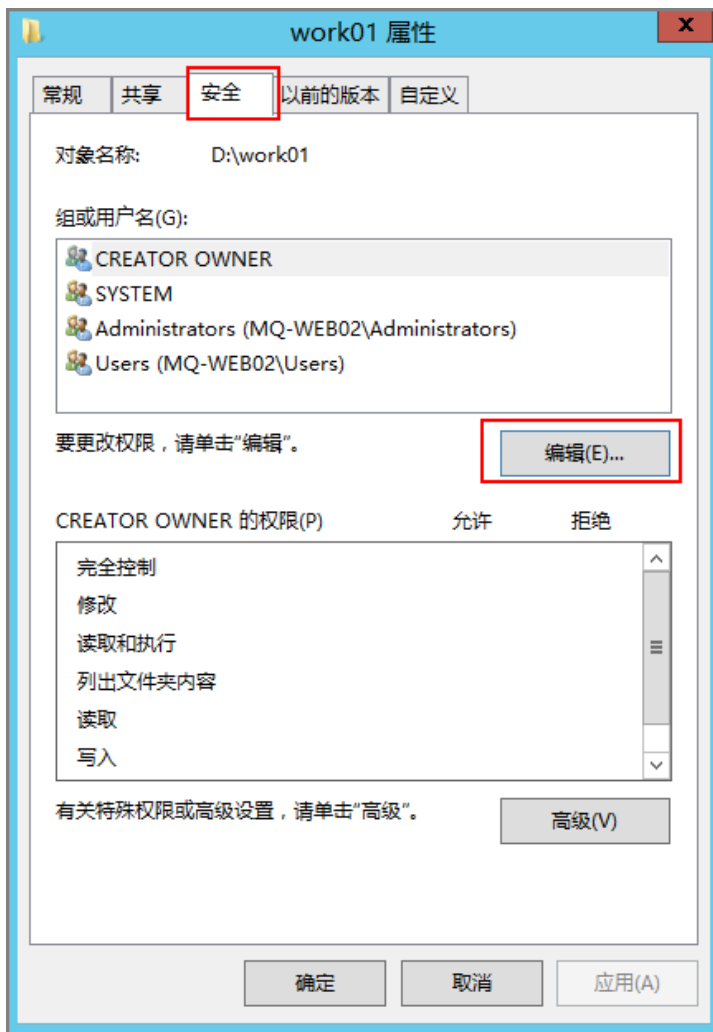
如果没有配置“FTP防火墙支持”，客户端使用此方法访问FTP站点时，需要对IE浏览器进行设置，才能打开FTP的文件夹。打开IE浏览器，选择“设置 > Internet 选项 > 高级”。勾选“启用FTP文件夹视图”，取消勾选“使用被动FTP”。



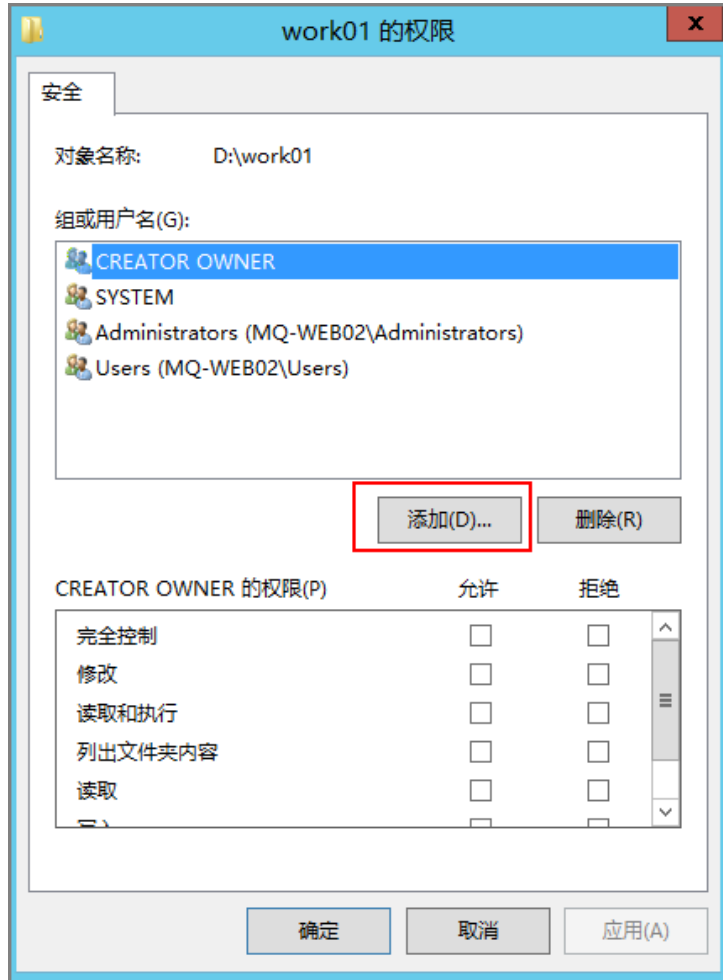
---结束

FAQ

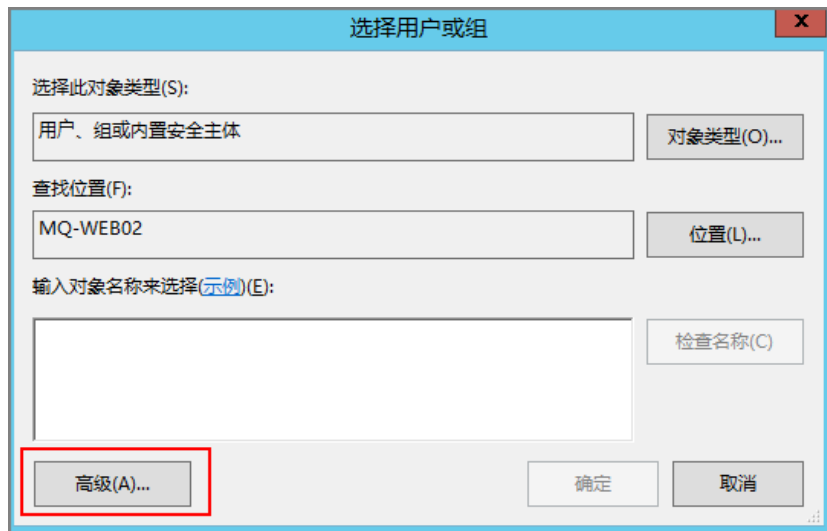
1. 使用Windows系统搭建FTP站点的更多信息请参见[微软官方文档](#)。
2. 如果设置文件夹的属性时，没有“Everyone”用户可直接选择，可按照如下步骤添加“Everyone”用户。
 - a. 在“安全”页签，单击“编辑”。



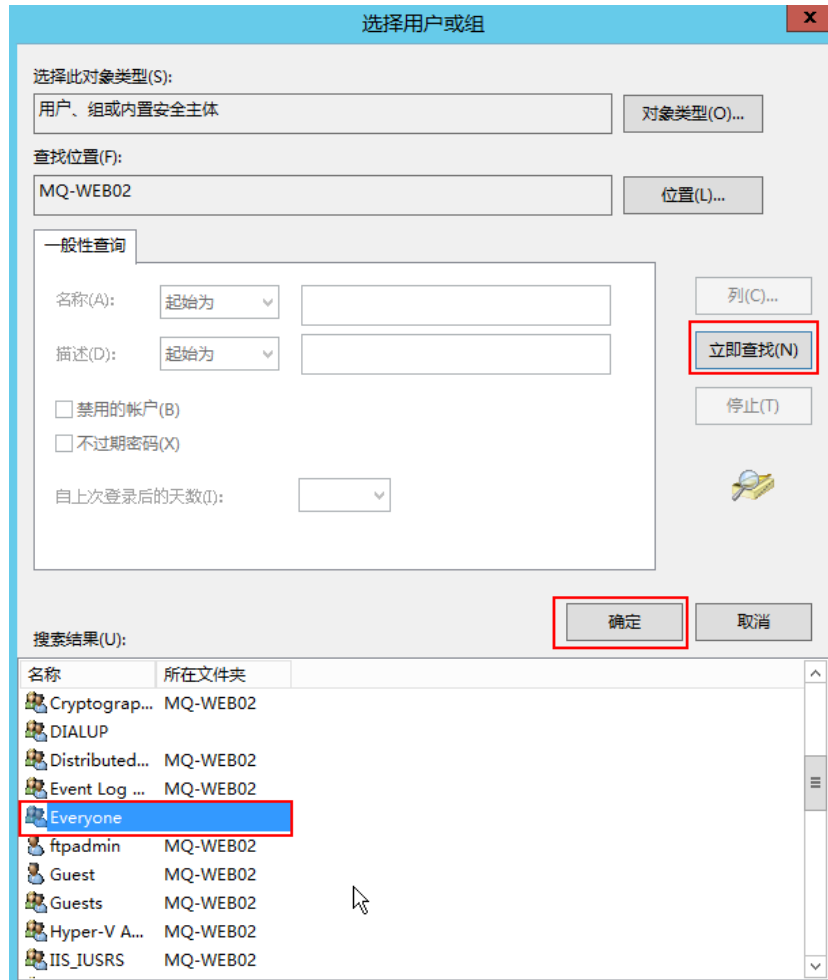
- b. 在新弹出的窗口中，单击“添加”。



c. 在新弹出的窗口中，单击“高级”。



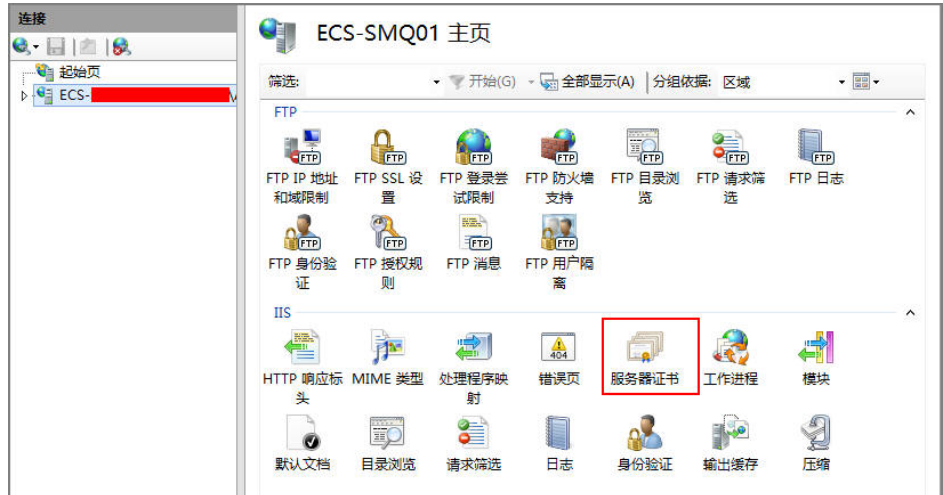
d. 在新弹出的窗口中，单击“立即查找(N)” ，然后在搜索结果中选择“Everyone”并单击“确定”。



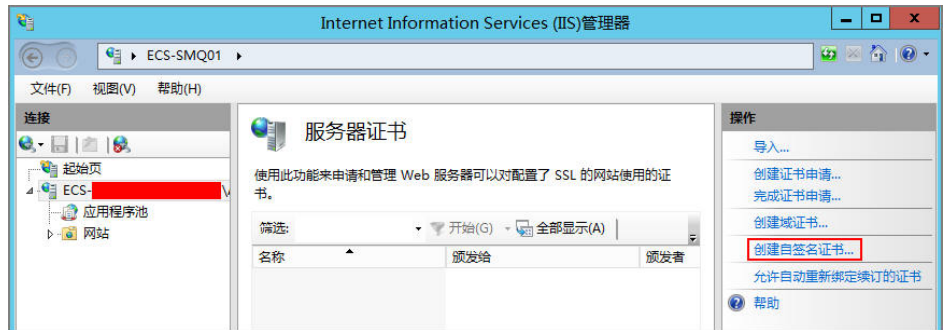
- e. 单击“确定”，返回权限窗口。
 - f. 单击“确定”，完成添加。
3. 制作服务器证书。
- a. 在“服务器管理器”中，选择“仪表板 > 工具 > Internet Information Services (IIS) 管理器”。



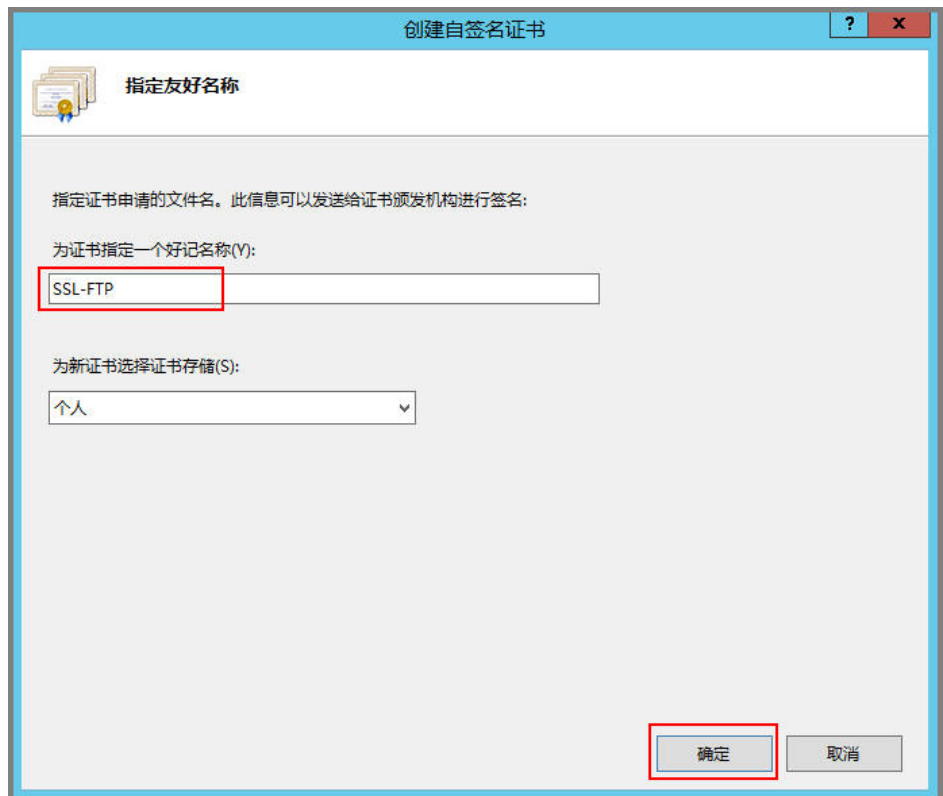
- b. 在左侧列表单击服务器，然后在服务器主页“IIS”区域，双击“服务器证书”，进入“服务器证书”页面。



c. 单击“创建自签名证书”。



d. 输入证书的名称，并选择证书存储类型，单击“确定”。



创建成功后会在服务器证书界面显示已经创建的证书。



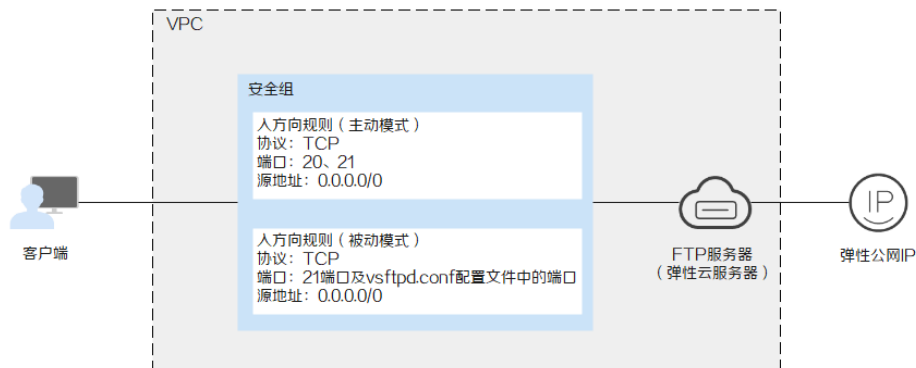
11 快速构建 FTP 站点 (Linux)

应用场景

本文介绍了在华为云上如何使用弹性云服务器的Linux实例使用vsftpd软件搭建FTP站点。vsftpd全称是“very secure FTP daemon”，是一款在Linux发行版中使用较多的FTP服务器软件。该指导具体操作以CentOS 7.2 64位操作系统为例。

方案架构

图 11-1 快速搭建 FTP 站点示意图 (Linux)



方案优势

- 快速构建站点，组网架构简单。
- 网站安全易用。

资源和成本规划

表 11-1 资源和成本规划

资源	资源说明	成本说明
虚拟私有云VPC	VPC网段：192.168.0.0/16	免费

资源	资源说明	成本说明
虚拟私有云子网	<ul style="list-style-type: none">可用区：可用区1子网网段： 192.168.0.0/24	免费
安全组	入方向规则（主动模式）： <ul style="list-style-type: none">协议/应用：TCP端口：20、21源地址：0.0.0.0/0 入方向规则（被动模式）： <ul style="list-style-type: none">协议/应用：TCP端口：21以及vsftpdconf配置文件中的端口源地址：0.0.0.0/0	免费
弹性云服务器	<ul style="list-style-type: none">计费模式：包年/包月可用区：可用区1规格：s6.large.2镜像：CentOS 7.2 64bit系统盘：40G弹性公网IP：现在购买线路：全动态BGP公网带宽：按流量计费带宽大小：5 Mbit/s	ECS涉及以下几项费用： <ul style="list-style-type: none">云服务器云硬盘弹性公网IP 具体的计费方式及标准请参考 计费模式概述 。
vsftpd	是一款免费、开源的ftp软件	免费

搭建 FTP 站点操作流程

Linux实例手动搭建FTP站点的具体操作步骤如下：

1. [安装vsftpd](#)。
2. [配置vsftpd](#)。
3. [设置安全组](#)。
4. [客户端测试](#)。

实施步骤

步骤1 安装vsftpd。

1. 登录弹性云服务器。
2. 为了更好的获取和更新系统和软件，建议您更新镜像源为华为云镜像源，详细操作，请参见[如何使用自动化工具配置华为云镜像源\(x86_64和ARM\)?](#)。
3. 执行以下命令安装vsftpd。

```
yum install -y vsftpd
```

回显如下类似信息时，表示软件安装成功。

```
Dependencies Resolved
-----
Package                Arch          Version           Repository        Size
-----
Installing:
vsftpd                 x86_64        3.0.2-22.e17     base              169 k

Transaction Summary
-----
Install 1 Package

Total download size: 169 k
Installed size: 348 k
Downloading packages:
vsftpd-3.0.2-22.e17.x86_64.rpm | 169 kB  00:00:00
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
  Installing : vsftpd-3.0.2-22.e17.x86_64                1/1
   Verifying  : vsftpd-3.0.2-22.e17.x86_64                1/1

Installed:
vsftpd.x86_64 0:3.0.2-22.e17
```

4. 执行以下命令设置FTP服务开机自启动。

```
systemctl enable vsftpd.service
```

5. 执行以下命令启动FTP服务。

```
systemctl start vsftpd.service
```

6. 执行以下命令查看FTP服务端口。

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

回显如下类似信息。

```
tcp6      0      0 :::21                :::*                  LISTEN      11836/vsftpd
```

步骤2 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。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。以下操作以在vsftpd 里配置用户账号和密码登录FTP服务器为例。

1. 执行以下命令创建“ftpadmin”用户。
此处“ftpadmin”用户为示例，可根据实际情况创建相应的用户。

```
useradd ftpadmin
```
2. 执行以下命令并按照提示设置“ftpadmin”用户密码。

```
passwd ftpadmin
```
3. 执行以下命令创建供FTP使用的文件目录，此处以“/var/ftp/work01”为例。

```
mkdir /var/ftp/work01
```
4. 执行以下命令将创建的文件目录所有者改为用于登录FTP的本地用户。

```
chown -R ftpadmin:ftpadmin /var/ftp/work01
```
5. 修改“vsftpd.conf”配置文件。
 - a. 执行以下命令打开配置文件“vsftpd.conf”。

```
vi /etc/vsftpd/vsftpd.conf
```

b. 按i键进入编辑模式。

c. 修改打开的“vsftpd.conf”文件。

可根据实际需求选择将FTP配置为主动模式或者被动模式。如果华为云上的服务器需要通过公网IP地址访问华为云上的实例搭建的FTP服务器时，需要将FTP服务器配置为被动模式。

■ 主动模式需要配置的参数如下：

#设置以下参数，不允许匿名登录FTP服务器，允许本地用户登录FTP服务器，并指定FTP本地用户使用的文件目录。

```
anonymous_enable=NO      #不允许匿名登录FTP服务器
local_enable=YES         #允许本地用户登录FTP服务器
local_root=/var/ftp/work01 #FTP本地用户使用的文件目录
```

#设置以下参数，限制用户只能访问自身的主目录。

```
chroot_local_user=YES     #所有用户都被限制在其主目录
chroot_list_enable=YES    #启用例外用户名单
chroot_list_file=/etc/vsftpd/chroot_list #例外用户名单
allow_writeable_chroot=YES
```

■ 被动模式除了需要配置主动模式所需的所有参数外，还需要配置的参数如下：

#设置以下参数，配置FTP支持被动模式。并指定FTP服务器的公网IP地址，以及可供访问的端口范围，端口范围请根据实际环境进行设置。

```
listen=YES
listen_ipv6=NO
pasv_address=xx.xx.xx.xx #FTP服务器的公网IP地址

pasv_min_port=3000 #被动模式下的最小端口
pasv_max_port=3100 #被动模式下的最大端口
```

d. 按Esc键退出编辑模式，并输入:wq保存后退出。

e. 在“/etc/vsftpd/”目录下创建“chroot_list”文件。

touch chroot_list

“chroot_list”文件是限制在主目录下的例外用户名单。如果需要设置某个用户不受只可以访问其主目录的限制，可将对应的用户名写入该文件。如果没有例外也必须要要有“chroot_list”文件，内容可为空。

6. 执行以下命令重启vsftpd服务使配置生效。

```
systemctl restart vsftpd.service
```

步骤3 设置安全组。

搭建好FTP站点后，需要在弹性云服务器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[为安全组添加安全组规则](#)。

表 11-2 设置安全组规则

FTP模式	方向	协议	端口	源地址
主动模式	入方向	TCP	20端口和21端口	0.0.0.0/0
被动模式	入方向	TCP	21端口和配置文件“/etc/vsftpd/vsftpd.conf”中参数“pasv_min_port”和“pasv_max_port”之间的所有端口。	0.0.0.0/0

步骤4 客户端测试。

打开客户端的计算机，在路径栏输入“ftp://FTP服务器IP地址:FTP端口”（如果不填端口则默认访问21端口）。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

- 如果FTP服务器配置为主动模式，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“工具 > Internet 选项 > 高级”。勾选“启用FTP文件夹视图”，取消勾选“使用被动FTP”。
- 使用浏览器访问FTP服务器出错时，建议您清除浏览器缓存后再尝试。

----结束

12 基于 Tomcat 构建 Java web 环境 (CentOS 7.4)

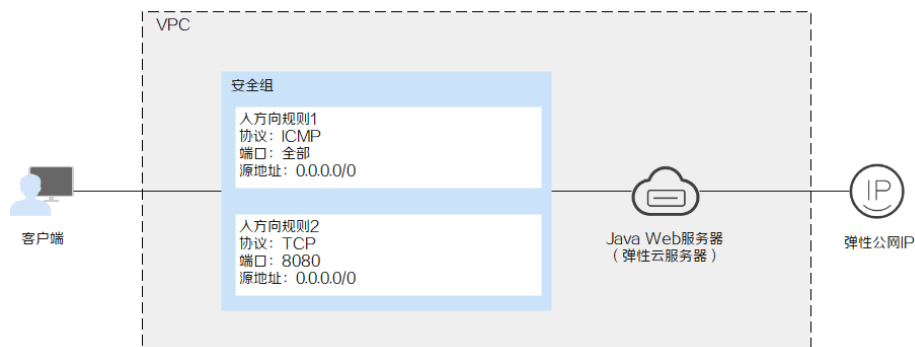
应用场景

Tomcat是一个被广泛使用的Java Web应用服务器。本文介绍了在华为云弹性云服务器上部署Java Web环境的操作步骤。首先需要下载部署Java Web环境所需的安装包，并将安装包上传至云服务器，然后设置弹性云服务器安全组规则，再安装并配置相关软件，完成开发环境的配置。

本文使用的云服务器以华为云CentOS 7.4 64bit操作系统云服务器为例。

方案架构

图 12-1 基于 Tomcat 构建 Java web 环境示意图



资源和成本规划

表 12-1 资源和成本规划

资源	资源说明	成本说明
虚拟私有云VPC	VPC网段: 192.168.0.0/16	免费
虚拟私有云子网	<ul style="list-style-type: none">可用区: 可用区1子网网段: 192.168.0.0/24	免费

资源	资源说明	成本说明
安全组	入方向规则1: <ul style="list-style-type: none">● 协议/应用: ICMP● 端口: 全部● 源地址: 0.0.0.0/0 入方向规则2: <ul style="list-style-type: none">● 协议/应用: TCP● 端口: 8080● 源地址: 0.0.0.0/0	免费
弹性云服务器	<ul style="list-style-type: none">● 计费模式: 包年/包月● 可用区: 可用区1● 规格: c7.large.2● 镜像: CentOS 7.4 64bit● 系统盘: 40G● 弹性公网IP: 现在购买● 线路: 全动态BGP● 公网带宽: 按流量计费● 带宽大小: 5 Mbit/s	ECS涉及以下几项费用: <ul style="list-style-type: none">● 云服务器● 云硬盘● 弹性公网IP 具体的计费方式及标准请参考 计费模式概述 。
jdk	Java开发工具软件。 获取方式: http://www.oracle.com/technetwork/java/javase/downloads	免费
tomcat	是一款开源的Web应用服务器。 获取方式: http://tomcat.apache.org/download-80.cgi	免费
PuTTY	跨平台远程访问工具。用于在软件安装过程中在Windows系统上访问云服务器。 获取方式: https://www.chiark.greenend.org.uk/~sgtatham/putty/latest.html	免费

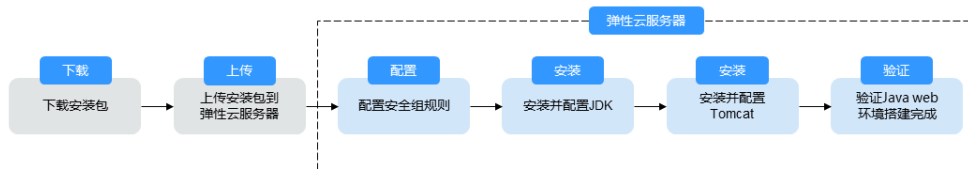
资源	资源说明	成本说明
WinSCP	跨平台文件传输工具。用于在 Windows系统和Linux系统间传输文件。 获取方式： http://winscp.net/	免费

📖 说明

表12-1中为jdk和tomcat官方获取地址，您还可以参考其他开源镜像地址获取安装包。

基于 Tomcat 构建 Java web 环境操作流程

图 12-2 安装流程



实施步骤

安装前准备

- 创建弹性云服务器，且弹性云服务器已绑定弹性公网IP。
- 为了更好的获取和更新系统和软件，建议您更新镜像源为华为云镜像源，详细操作，请参见[如何使用自动化工具配置华为云镜像源\(x86_64和ARM\)?](#)。
- 登录弹性云服务器，执行如下命令，新建jdk目录和tomcat目录。

```
cd /home/  
mkdir webDemo  
cd webDemo/  
mkdir jdk  
mkdir tomcat
```

- 您可以选择将安装包下载至本地后使用文件传输工具将安装包上传至云服务器。或者选择使用wget命令直接下载安装包至云服务器。
 - 方法一：使用文件传输工具上传安装包至云服务器。
 - 使用WinSCP工具上传jdk软件包至云服务器jdk文件夹。
 - 使用WinSCP工具上传tomcat软件包至云服务器tomcat文件夹。
 - 方法二：使用wget命令直接下载安装包至云服务器。
 - i. 执行如下命令，进入jdk目录。

```
cd /home/webDemo/jdk
```

- ii. 执行如下命令，下载jdk软件包。

wget jdk软件包下载地址

请参考表12-1查询jdk下载地址，或者使用其他开源镜像地址获取安装包。

例如：以jdk17为例，在列表中查看可用的JDK软件包版本，以jdk-17_linux-x64_bin.tar.gz安装包为例，执行以下命令。

wget https://download.oracle.com/java/17/latest/jdk-17_linux-x64_bin.tar.gz

- iii. 执行如下命令，进入tomcat目录。

cd /home/webDemo/tomcat

- iv. 执行如下命令，下载tomcat软件包。

请参考表12-1查询tomcat下载地址，或者使用其他开源镜像地址获取安装包。

wget http://mirrors.tuna.tsinghua.edu.cn/apache/tomcat/tomcat-x/vx.x.xx/bin/apache-tomcat-x.x.xx.tar.gz

例如：单击[开源镜像地址](#)，查询当前可用版本，以v8.5.xx版本安装包为例，执行以下命令。

wget https://mirrors.tuna.tsinghua.edu.cn/apache/tomcat/tomcat-8/v8.5.xx/bin/apache-tomcat-8.5.xx.tar.gz --no-check-certificate

设置弹性云服务器安全组规则

1. 单击弹性云服务器名称，查看弹性云服务器详情，在弹性云服务器详情页面，选择“安全组”。
2. 在“安全组”界面，单击“更改安全组规则”，进入安全组详情界面。
3. 在安全组详情界面，单击“添加规则”，弹出添加规则窗口。
4. 根据界面提示配置安全组规则。

部署JavaWeb环境需为弹性云服务器添加两个安全组规则。

- a. 为云服务器添加ICMP安全组规则。

如果云服务器默认设置是禁止ICMP规则，当ping弹性服务器IP时会显示超时。因此首先为云服务器添加ICMP规则。

图 12-3 添加 ICMP 规则

添加入方向规则 [教我设置](#)

安全组入方向规则为白名单（允许），放通入方向网络流量。

安全组 default

如您要添加多条规则，建议单击[导入规则](#)以进行批量导入。

优先级	策略	协议端口	类型	源地址	描述	操作
1	允许	ICMP	IPv4	IP地址		复制 删除
		全部		0.0.0.0/0		

增加1条规则

- b. 为云服务器添加web项目分配端口的访问规则，以8080端口为例。

图 12-4 添加 8080 端口

添加方向规则 [教我设置](#)

安全组入方向规则为白名单（允许），放通入方向网络流量。

安全组 default

如您要添加多条规则，建议单击导入规则以进行批量导入。

优先级	策略	协议端口	类型	源地址	描述	操作
1	允许	TCP 8080	IPv4	IP地址 0.0.0.0/0		复制 删除

增加1条规则

确定 取消

安装jdk

1. 执行如下命令，进入jdk目录。
cd /home/webDemo/jdk
2. 解压jdk安装包到jdk目录下。
tar -xvf jdk-17_linux-x64_bin.tar.gz -C /home/webDemo/jdk/
3. 配置环境变量。
vim /etc/profile
4. 在底部添加以下内容。

```
#set java environment
JAVA_HOME=/home/webDemo/jdk/jdk-17.0.x
JRE_HOME=$JAVA_HOME
PATH=$JAVA_HOME/bin:$PATH
CLASSPATH=.:$JAVA_HOME/lib/dt.jar:$JRE_HOME/lib/tools.jar
export JAVA_HOME JRE_HOME PATH CLASSPATH
```

说明

“jdk-17.0.x”表示jdk安装包的具体版本，实际值需要从步骤2的返回值中获取。

例如：jdk-17.0.9

5. 执行以下命令保存并退出。
:wq
6. 执行以下命令使/etc/profile里的配置生效。
source /etc/profile
7. 验证安装。
java -version
回显信息如下所示验证安装jdk成功。

```
[root@ecs-c525-web ~]# java -version
java version "17.0.9" 2023-10-17 LTS
Java(TM) SE Runtime Environment (build 17.0.9+11-LTS-201)
Java HotSpot(TM) 64-Bit Server VM (build 17.0.9+11-LTS-201, mixed mode, sharing)
```

安装tomcat

1. 执行如下命令，进入tomcat目录。

- ```
cd /home/webDemo/tomcat
```
2. 解压tomcat安装包到tomcat目录下。  
**tar -xvf apache-tomcat-x.x.xx.tar.gz -C /home/webDemo/tomcat/**  
例如：以v8.5.xx版本安装包为例，执行以下命令。  
**tar -xvf apache-tomcat-8.5.xx.tar.gz -C /home/webDemo/tomcat/**
  3. 进入tomcat的bin目录，执行以下命令安装tomcat。  
**cd /home/webDemo/tomcat/apache-tomcat-x.x.xx/**  
**cd bin/**  
例如：以v8.5.xx版本安装包为例，执行以下命令。  
**cd /home/webDemo/tomcat/apache-tomcat-8.5.xx/**  
**cd bin/**
  4. 执行如下命令编辑setclasspath.sh脚本。  
**vi setclasspath.sh**  
并在setclasspath.sh脚本底部添加以下内容。  
请根据[资源和成本规划](#)中jdk的版本号替换如下内容中的jdk版本号。  
export JAVA\_HOME=/home/webDemo/jdk/jdk-17.0.9  
export JRE\_HOME=\$JAVA\_HOME
  5. 执行如下命令保存后退出。  
**:wq**
  6. 可输入以下命令启动tomcat。  
**./startup.sh**
  7. 执行如下命令查看tomcat进程。  
**ps -ef | grep tomcat**  
若返回如下图所示，表示tomcat启动成功。

图 12-5 查看 tomcat 进程

```
root@ecs ~# cd /home/webDemo/tomcat/bin/ ; ps -ef | grep tomcat
root 8671 1 16 14:33 tty1 00:00:02 /home/webDemo/jdk/jdk-17.0.9/bin/java -Djava.util.logging.config.file=/home/webD
emo/tomcat/apache-tomcat-8.5.95/conf/logging.properties -Djava.util.logging.manager=org.apache.juli.ClassLoaderLogManager -Djdk.
tls.ephemeralDHKeySize=2048 -Djava.protocol.handler.pkgs=org.apache.catalina.webresources -Dorg.apache.catalina.security.Security
UtilListener.Umask=0027 -Dignore.endorsed.dirs=-classpath /home/webDemo/tomcat/apache-tomcat-8.5.95/bin/bootstrap.jar:/home/webDem
o/tomcat/apache-tomcat-8.5.95/bin/tomcat-juli.jar -Dcatalina.base=/home/webDemo/tomcat/apache-tomcat-8.5.95 -Dcatalina.home=/hom
e/webDemo/tomcat/apache-tomcat-8.5.95 -Djava.io.tmpdir=/home/webDemo/tomcat/apache-tomcat-8.5.95/temp org.apache.catalina.startu
p.Bootstrap start
root 8710 7902 0 14:33 tty1 00:00:00 grep --color=auto tomcat
root@ecs ~# cd /home/webDemo/tomcat/bin/ ;
```

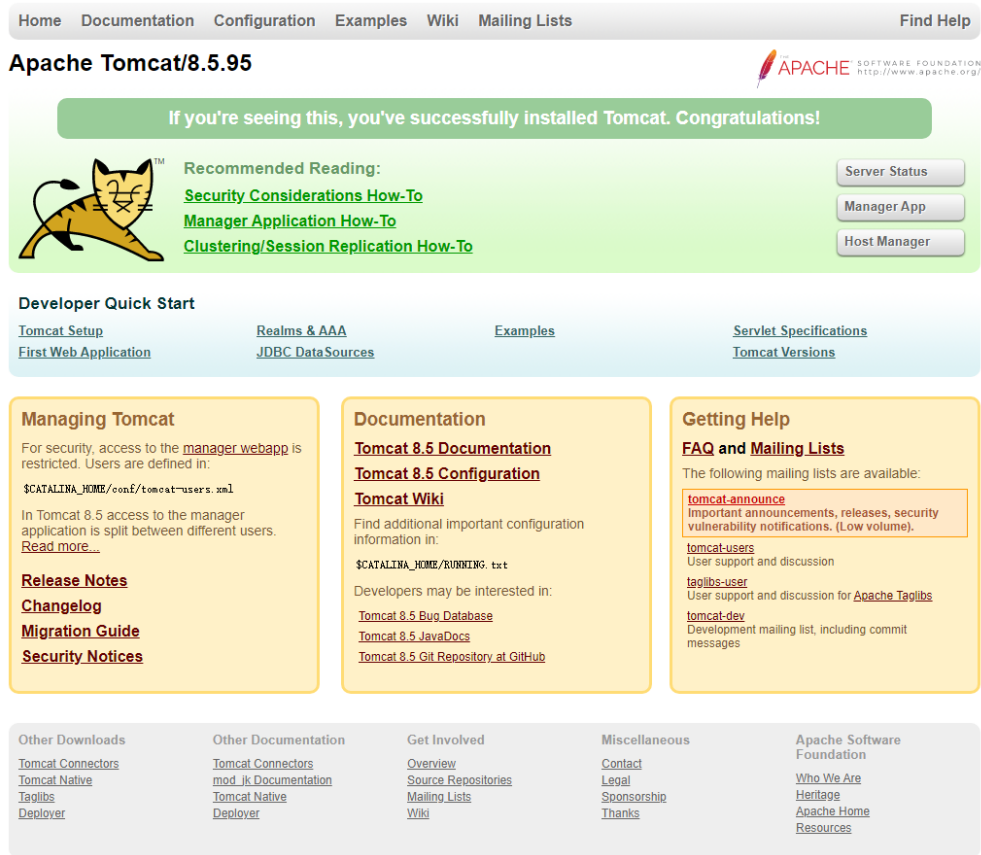
### 验证Java Web环境搭建完成

在浏览器输入以下内容。

**http://云服务器弹性公网IP:8080**

如果界面跳转至默认的Tomcat界面，证明Java Web环境搭建完成。就可以在公网访问云服务器的8080端口了。

图 12-6 访问云服务器的 8080 端口



# 13 手工搭建 Magento 电子商务网站 (Linux)

## 简介

本文主要介绍了如何使用弹性云服务器的Linux实例手工搭建Magento电子商务系统。Magento是一款开源电子商务系统，设计灵活，具有模块化架构体系和丰富的功能，可为大中型站点提供解决方案。Magento采用PHP开发，数据存储使用MySQL。

## 前提条件

- 已购买一台弹性云服务器，且已为其绑定弹性公网IP。
- 弹性云服务器所在安全组添加了如下表所示的安全组规则，具体步骤参见[为安全组添加安全组规则](#)。

表 13-1 安全组规则

| 方向  | 协议/应用       | 端口/范围 | 是否必须 | 源地址                                                                          |
|-----|-------------|-------|------|------------------------------------------------------------------------------|
| 入方向 | TCP(22)     | 22    | 是    | 允许SSH远程连接Linux弹性云服务器的客户端IP地址。如允许所有IP访问，请设置为0.0.0.0/0，出于安全考虑，不建议您设置为允许所有IP访问。 |
| 入方向 | HTTP(80)    | 80    | 是    | 允许访问Magento网站的客户端IP地址。如允许所有IP访问，请设置为0.0.0.0/0。                               |
| 入方向 | MySQL(3306) | 3306  | 否    | 允许远程访问MySQL数据库的客户端IP地址。如允许所有IP访问，请设置为0.0.0.0/0，出于安全考虑，不建议您设置为允许所有IP访问。       |

## 资源规划

本次实践所用的资源配置及软件版本如表13-2中所示。当您使用不同的硬件规格或软件版本时，本指导中的命令及参数可能会发生改变，需要您根据实际情况进行调整。

表 13-2 资源规划

| 资源    | 类别       | 规格/版本                                                                                                                                                                 |
|-------|----------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| ECS配置 | 实例规格     | c6s.large.2                                                                                                                                                           |
|       | vCPUs    | 2vCPUs                                                                                                                                                                |
|       | 内存       | 4GiB                                                                                                                                                                  |
|       | 操作系统     | CentOS 7.2                                                                                                                                                            |
| 软件资源  | Apache   | 2.4.6                                                                                                                                                                 |
|       | MySQL    | 5.7<br>获取方式：<br><a href="http://dev.mysql.com/get/mysql57-community-release-el7-8.noarch.rpm">http://dev.mysql.com/get/mysql57-community-release-el7-8.noarch.rpm</a> |
|       | PHP      | 7.0.33<br>获取方式：<br><a href="https://mirror.webtatic.com/yum/el7/webtatic-release.rpm">https://mirror.webtatic.com/yum/el7/webtatic-release.rpm</a>                    |
|       | Composer | 1.10.19<br>获取方式：<br><a href="https://getcomposer.org/installer">https://getcomposer.org/installer</a>                                                                 |
|       | Magento  | 2.1.0<br>获取方式：<br><a href="https://github.com/magento/magento2.git">https://github.com/magento/magento2.git</a>                                                       |

### 说明

为保证所搭建网站正常运行，所选的ECS实例规格建议为2GiB及以上。

## 步骤一：安装配置 Apache

**步骤1** 远程登录弹性云服务器，操作方法请参考[登录Linux弹性云服务器](#)。

**步骤2** 为了更好的获取和更新系统和软件，建议您更新镜像源为华为云镜像源，详细操作，请参见[如何使用自动化工具配置华为云镜像源\(x86\\_64和ARM\)?](#)。

**步骤3** 使用root用户执行以下命令更新软件包，并安装Apache。

```
yum -y update
yum -y install httpd
```

### 📖 说明

如果执行命令过程中报错，提示无法解析域名时，可以在“/etc/resolv.conf”配置文件中增加一个DNS服务器。

**步骤4** 打开Apache配置文件。

```
vim /etc/httpd/conf/httpd.conf
```

### 📖 说明

如果没有安装vim，可使用yum install -y vim\*命令进行安装。

**步骤5** 按i键进入编辑模式，修改如下两处内容：

- 将下图所示位置的配置参数“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
```

- 在配置文件末新增一行配置参数：  
LoadModule rewrite\_module modules/mod\_rewrite.so

如下图所示：

```
Load config files in the "/etc/httpd/conf.d" directory, if any.
IncludeOptional conf.d/*.conf

LoadModule rewrite_module modules/mod_rewrite.so
```

**步骤6** 按Esc键退出编辑模式，并输入:wq保存后退出。

**步骤7** 依次执行以下命令，启动Apache服务并设置开机自启动。

```
systemctl start httpd
systemctl enable httpd
```

----结束

## 步骤二：安装配置 MySQL

**步骤1** 使用root用户执行以下命令添加yum源。

```
rpm -Uvh http://dev.mysql.com/get/mysql57-community-release-el7-8.noarch.rpm
```

**步骤2** 安装MySQL。

```
yum -y install mysql-community-server --nogpgcheck
```

**步骤3** 依次执行以下命令，启动MySQL服务并设置开机自启动。

```
systemctl start mysqld
systemctl enable mysqld
```

**步骤4** 执行以下命令，获取安装MySQL时自动设置的root用户密码。

```
grep 'temporary password' /var/log/mysqld.log
```

回显如下类似信息，其中“(n?K7jP#cirM)”即为临时密码。

```
2019-05-09T11:29:42.365419Z 1 [Note] A temporary password is generated for root@localhost: (n?K7jP#cirM
```

**步骤5** 执行以下命令加固MySQL。

```
mysql_secure_installation
```

按照回显提示信息进行操作，具体操作如粗体所示：

```
Securing the MySQL server deployment.
```

```
Enter password for user root: #输入上一步骤中获取的安装MySQL时自动设置的root用户密码
The existing password for the user account root has expired. Please set a new password.
```

```
New password: #设置root用户密码
```

```
Re-enter new password: #再次输入密码
The 'validate_password' plugin is installed on the server.
The subsequent steps will run with the existing configuration of the plugin.
Using existing password for root.
```

```
Estimated strength of the password: 100
```

```
Change the password for root ? ((Press y|Y for Yes, any other key for No) : Y #是否更改root用户密码，输入Y
```

```
New password: #输入密码，长度为8至30个字符，须同时包含大小写英文字母、数字和特殊符号。特殊符号可以是() `~!@#$%^&*~+=|{}[]; '<>.,?/
```

```
Re-enter new password: #再次输入密码
```

```
Estimated strength of the password: 100
```

```
Do you wish to continue with the password provided?(Press y|Y for Yes, any other key for No) : Y #输入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!
```

**步骤6** 以root用户登录MySQL数据库。

```
mysql -u root -p
```

**步骤7** 创建一个新的数据库，其中“magento”为自定义的数据库名称。

```
CREATE DATABASE magento;
```

**步骤8** 执行以下命令，为数据库创建用户并为用户分配数据库的完全访问权限。

```
GRANT ALL ON magento.* TO magentouser@localhost IDENTIFIED BY 'xxxxx';
```

其中，“magento”为上一步骤中创建的数据库名，“magentouser”为新建的数据库用户名，“xxxxx”为对应的账户密码，可以自行设置。

**步骤9** 执行以下命令，退出MySQL命令行。

```
exit
```

**步骤10** (可选) 依次执行以下步骤，验证数据库和用户是否已成功创建，并退出MySQL命令行。

1. 以新建的数据库用户“magentouser”登录MySQL命令行。

```
mysql -u magentouser -p
```

2. 查看数据库。

```
SHOW DATABASES;
```

回显如下，“magento”即为新建的数据库，说明数据库已成功创建。

```
+-----+
| Database |
+-----+
| information_schema |
| magento |
+-----+
2 rows in set (0.00 sec)
```

3. 执行以下命令，退出MySQL命令行。

```
exit
```

----结束

## 步骤三：安装配置 PHP

**步骤1** 添加ius源和epel源。

```
yum install \
https://repo.ius.io/ius-release-el7.rpm \
https://dl.fedoraproject.org/pub/epel/epel-release-latest-7.noarch.rpm
```

**步骤2** 添加Webtatic源。

```
rpm -Uvh https://mirror.webtatic.com/yum/el7/webtatic-release.rpm
```

**步骤3** 安装PHP7和用于Magento安装的一些必需的PHP扩展。

```
yum -y install php70w php70w-pdo php70w-mysqlnd php70w-opcache php70w-xml php70w-gd php70w-mcrypt php70w-devel php70w-intl php70w-mbstring php70w-bcmath php70w-json php70w-iconv
```

**步骤4** 执行以下命令，验证PHP的安装版本。

```
php -v
```

回显如下类似信息：

```
PHP 7.0.33 (cli) (built: Dec 6 2018 22:30:44) (NTS)
Copyright (c) 1997-2017 The PHP Group
Zend Engine v3.0.0, Copyright (c) 1998-2017 Zend Technologies
with Zend OPcache v7.0.33, Copyright (c) 1999-2017, by Zend Technologies
```

**步骤5** 打开PHP配置文件“php.ini”。

```
vim /etc/php.ini
```

**步骤6** 按i键进入编辑模式，修改如下两处内容：

- 根据实际情况修改参数 “memory\_limit” 的值，设置内存限制，如下图所示：

```
; Maximum amount of memory a script may consume (128MB)
; http://php.net/memory-limit
memory_limit = 256M
```

- 取消注释并设置参数 “date.timezone” 的值，指定PHP的时区，如下图所示：

```
[Date]
; Defines the default timezone used by the date functions
; http://php.net/date.timezone
date.timezone = Asia/Shanghai
```

**步骤7** 按Esc键退出编辑模式，并输入:wq保存后退出。

**步骤8** 执行以下命令，重启Web服务进程。

```
systemctl restart httpd
```

----结束

## 步骤四：安装 Composer 工具

Composer是PHP编程语言的包管理器，提供管理PHP软件依赖和所需库的标准格式。

**步骤1** 由于Magento2的依赖为Composer1.x版本，所以需确保安装的Composer版本的1.x版本。

执行以下命令安装指定版本的Composer，并将安装路径指定为 “/usr/bin/” 以供全局使用。

```
php -r "copy('https://getcomposer.org/installer', 'composer-setup.php');"
php composer-setup.php --install-dir=/usr/bin/ --filename=composer --version=1.10.19
```

回显如下：

```
All settings correct for using Composer
Downloading...

Composer (version 1.10.19) successfully installed to: /usr/bin/composer
Use it: php /usr/bin/composer
```

**步骤2** 执行以下命令，验证Composer否成功安装。

```
composer -v
```

回显如下：

```
Composer version 1.10.19 2020-12-04 09:14:16
...
```

----结束

## 步骤五：安装 Magento 组件

安装Magento时可以选择是否安装示例数据。如果安装Magento仅用于测试，可以选择安装示例数据。如果是在生产环境中安装Magento，建议安装全新的Magento，并从初始进行配置。



**步骤1** 执行以下命令，安装git。

```
yum -y install git
```

**步骤2** 依次执行以下命令，进入Web服务器的默认根目录“/var/www/html/”并使用git下载Magento。

```
cd /var/www/html/
git clone https://github.com/magento/magento2.git
```

**步骤3** 执行以下命令，将Magento切换到稳定版本。

默认情况下前一步骤的命令会安装最新开发版本的Magento。在生产环境中使用时，建议切换到稳定版本，否则将来将无法进行升级。

```
cd magento2 && git checkout tags/2.1.0 -b 2.1.0
```

回显如下类似信息：

```
Switched to a new branch '2.1.0'
```

**步骤4** 执行以下命令，将安装文件移到Web服务器的根目录“/var/www/html/”下，其中“/var/www/html/magento2/”为Magento的当前安装目录。

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

迁移后可以通过“http://magento服务器的IP地址”访问Magento站点。否则，只能通过“http://magento服务器的IP地址/magento2”访问Magento站点。

**步骤5** 依次执行以下命令，设置Magento文件的权限。

```
chown -R apache: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
```

**步骤6** 安装unzip和zip。

```
yum install -y unzip zip
```

**步骤7** 进入Web服务器的默认根目录“/var/www/html/”，然后使用composer工具安装Magento。

```
cd /var/www/html/
composer install
```

**步骤8** 安装完成后，可通过浏览器访问“http://magento服务器IP地址”，显示如下页面，说明Magento安装成功。



Version 2.1.0

Welcome to Magento Admin, your online store headquarters.  
Click 'Agree and Set Up Magento' or read [Getting Started](#) to learn more.

[Terms & Agreement](#)

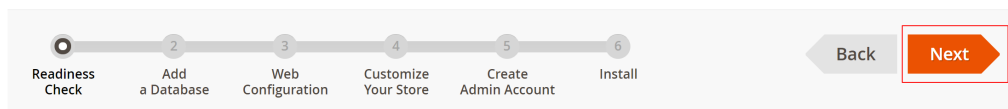
[Agree and Setup Magento](#)

----结束

## 步骤六：配置 Magento

**步骤1** 通过浏览器访问“http://magento服务器IP地址”，单击“Agree and Setup Magento”，开始配置Magento。

**步骤2** 单击“Start Readiness Check”，进行环境检测。检测通过后单击“Next”。



Step 1: Readiness Check

Let's check your environment for the correct PHP version, PHP extensions, file permissions and compatibility.

[Start Readiness Check](#)

**步骤3** 填写**步骤二：安装配置MySQL**中创建的MySQL数据库账户“magentouser”和密码，以及对应的数据库“magento”，并单击“Next”。

如下图所示：

## Step 2: Add a Database

|                            |                                          |
|----------------------------|------------------------------------------|
| Database Server Host *     | <input type="text" value="localhost"/>   |
| Database Server Username * | <input type="text" value="magentouser"/> |
| Database Server Password   | <input type="password" value="....."/>   |
| Database Name *            | <input type="text" value="magento"/>     |
| Table prefix               | <input type="text" value="(optional)"/>  |

**步骤4** 设置网站的访问地址和后台管理地址，并单击“Next”。

如下图所示：

## Step 3: Web Configuration

|                         |                                                                  |
|-------------------------|------------------------------------------------------------------|
| Your Store Address      | <input type="text" value="http://124.124.124.124"/>              |
| Magento Admin Address * | <input type="text" value="http://124.124.124.124/admin_1pj83g"/> |

### 说明

后台管理地址可自定义，请牢记设置的后台管理地址。

**步骤5** 设置语言和时区信息，并单击“Next”。

如下图所示：

## Step 4: Customize Your Store

|                           |                                                 |
|---------------------------|-------------------------------------------------|
| Store Default Time Zone * | <input type="text" value="GMT (UTC)"/>          |
| Store Default Currency *  | <input type="text" value="Chinese Yuan (CNY)"/> |
| Store Default Language *  | <input type="text" value="Chinese (China)"/>    |

**步骤6** 设置管理账号和密码，并单击“Next”。

如下图所示：

## Step 5: Create Admin Account

Create a new Admin account to manage your store.

|                    |                                                |
|--------------------|------------------------------------------------|
| New Username *     | <input type="text" value="admin"/>             |
| New Email *        | <input type="text" value="admin@example.com"/> |
| New Password *     | <input type="password" value="*****"/>         |
| Confirm Password * | <input type="password" value="*****"/>         |

- 步骤7** 单击“Install Now”，等待系统安装完成。  
显示如下界面时，表示系统安装成功。

### Success

Please keep this information for your records:

#### Magento Admin Info:

Username: admin  
Email: admin@example.com  
Password: \*\*\*\*\*  
Your Store Address: <http://124.71.111.111>  
Magento Admin Address: <http://124.71.111.111/admin>

 Be sure to bookmark your unique URL and record it offline.

Encryption Key:

#### Database Info:

Database Name: magento  
Username: magentouser

- 步骤8** 登录Magento服务器，设置cron来运行预定的作业。

```
crontab -u apache -e
```

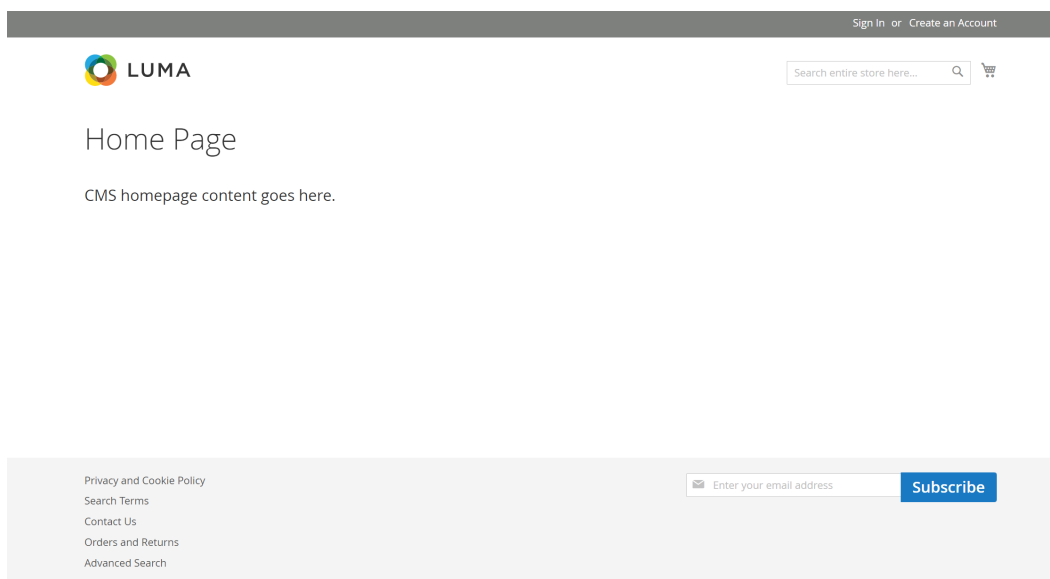
- 步骤9** 按i键进入编辑模式，添加如下内容。

```
*/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
```

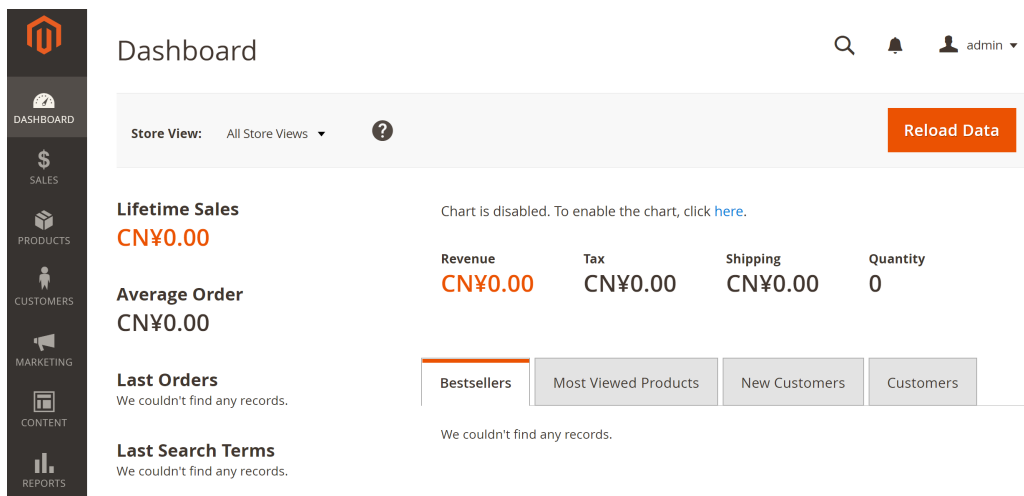
按Esc键退出编辑模式，并输入:wq保存后退出。

更多关于Magento使用cron作业的内容，请参见[Magento官方文档](#)。

**步骤10** 打开客户端的计算机，使用浏览器访问“http://magento服务器IP地址”，显示如下图所示的默认界面。



**步骤11** 使用浏览器访问“http://magento后台管理地址”，并使用配置Magento过程中设置的管理账号和密码登录，登录成功后显示如下图所示的界面。



### 📖 说明

如果登录后界面提示“**One or more indexers are invalid. Make sure your Magento cron job is running.**”的错误信息，可在Magento根目录“/var/www/html”执行**php bin/magento indexer:reindex**命令。

更多关于Magento配置信息，请参见[Magento官方文档](#)。

----结束

## 步骤七：其余操作

**步骤1** 购买域名。

为了便于网站的访问和使用，可以给网站设置一个单独的域名，使用域名访问网站。首先，需要在域名注册商处获得授权的域名。

#### **步骤2** 配置域名解析。

配置域名解析后才能使用注册的域名访问网站。具体操作请参见[配置公网域名](#)。

例如，配置的域名为“www.example.com”，配置成功后，可在浏览器地址栏中输入“http://www.example.com”访问部署的网站。


----**结束**

# 14 搭建 Microsoft SharePoint Server 2016

## 2016

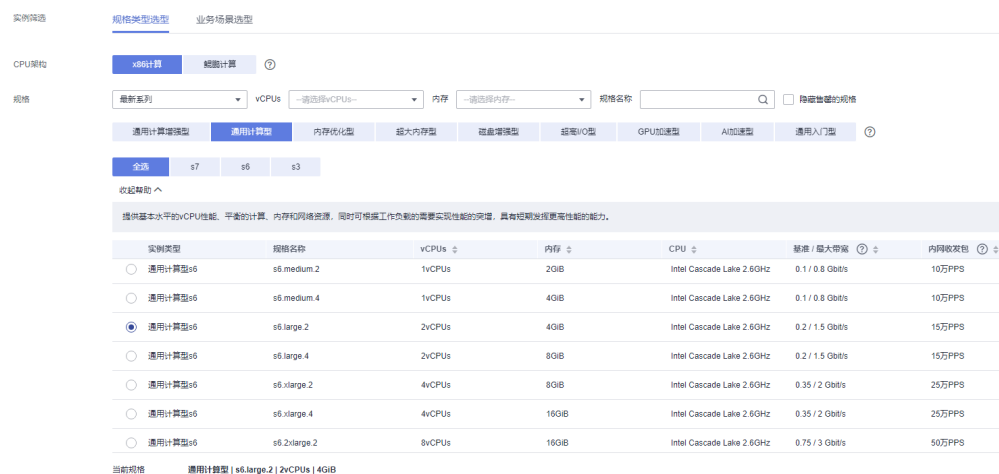
### 14.1 购买并登录弹性云服务器

在华为云上购买弹性云服务器，并根据系统要求设置服务器规格和安装操作系统。

1. 登录管理控制台。
  2. 单击管理控制台左上角的 ，选择区域和项目。
  3. 选择“计算 > 弹性云服务器”。
  4. 单击“购买弹性云服务器”。
- 系统进入购买页。
5. 在购买页，设置弹性云服务器。

弹性云服务器设置的具体原则请参见[购买并登录Windows弹性云服务器](#)。

图 14-1 设置弹性云服务器规格



规格类型选择 业务场景选型

CPU架构 x86\_64 鲲鹏计算

规格 最新系列 vCPUs 请选择vCPUs 内存 请选择内存 规格名称 隐藏高级规格

通用计算增强型 通用计算型 内存优化型 超大内存型 性能增强型 极致IO型 GPU加速型 AI加速型 通用入门型

全部 s7 s6 s3

收起帮助

提供基本水平的vCPU性能、平衡的计算、内存和网络资源，同时可根據工作负载的需要实现性能的突破，具有短期发挥更高性能的能力。

| 实例类型                             | 规格名称    | vCPUs        | 内存     | CPU   | 标准 / 最大带宽                 | 内网收发包            |        |
|----------------------------------|---------|--------------|--------|-------|---------------------------|------------------|--------|
| <input type="radio"/>            | 通用计算型s6 | s6.medium.2  | 1vCPUs | 2GiB  | Intel Cascade Lake 2.6GHz | 0.1 / 0.8 Gbit/s | 10万PPS |
| <input type="radio"/>            | 通用计算型s6 | s6.medium.4  | 1vCPUs | 4GiB  | Intel Cascade Lake 2.6GHz | 0.1 / 0.8 Gbit/s | 10万PPS |
| <input checked="" type="radio"/> | 通用计算型s6 | s6.large.2   | 2vCPUs | 4GiB  | Intel Cascade Lake 2.6GHz | 0.2 / 1.5 Gbit/s | 15万PPS |
| <input type="radio"/>            | 通用计算型s6 | s6.large.4   | 2vCPUs | 8GiB  | Intel Cascade Lake 2.6GHz | 0.2 / 1.5 Gbit/s | 15万PPS |
| <input type="radio"/>            | 通用计算型s6 | s6.xlarge.2  | 4vCPUs | 8GiB  | Intel Cascade Lake 2.6GHz | 0.35 / 2 Gbit/s  | 25万PPS |
| <input type="radio"/>            | 通用计算型s6 | s6.xlarge.4  | 4vCPUs | 16GiB | Intel Cascade Lake 2.6GHz | 0.35 / 2 Gbit/s  | 25万PPS |
| <input type="radio"/>            | 通用计算型s6 | s6.2xlarge.2 | 8vCPUs | 16GiB | Intel Cascade Lake 2.6GHz | 0.75 / 3 Gbit/s  | 50万PPS |

当前规格 通用计算型 | s6.large.2 | 2vCPUs | 4GiB

图 14-2 设置网络

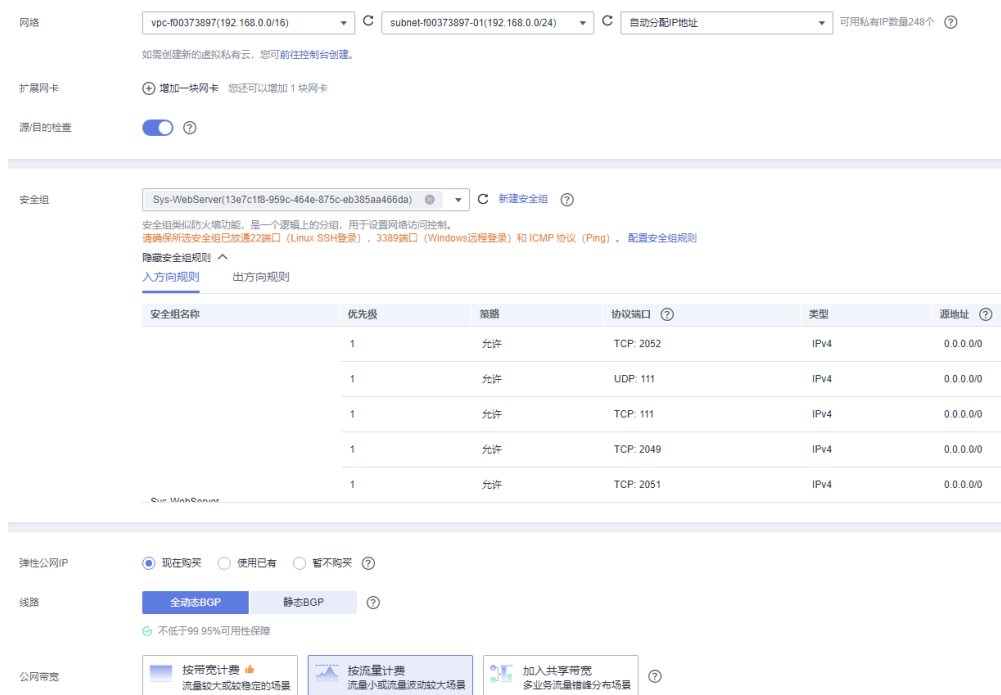




图 14-3 设置登录方式及云服务器名称

云服务器名称   允许重名

购买多台云服务器时，支持自动增加数字后缀命名或者自定义规则命名。 ?

描述  0/85

登录凭证  密码  密钥对  创建后设置

用户名 Administrator

密码 请牢记密码，如忘记密码可登录ECS控制台重置密码。

确认密码

---

云备份 使用云备份服务，需购买备份存储库，存储库是存放服务器产生的备份副本的容器。

现在购买  使用已有  暂不购买 ?

备份可以帮助您在服务器故障时恢复数据，为了您的数据安全，强烈建议您启用备份。

---

云服务器组（可选）  反亲和性 ?

C

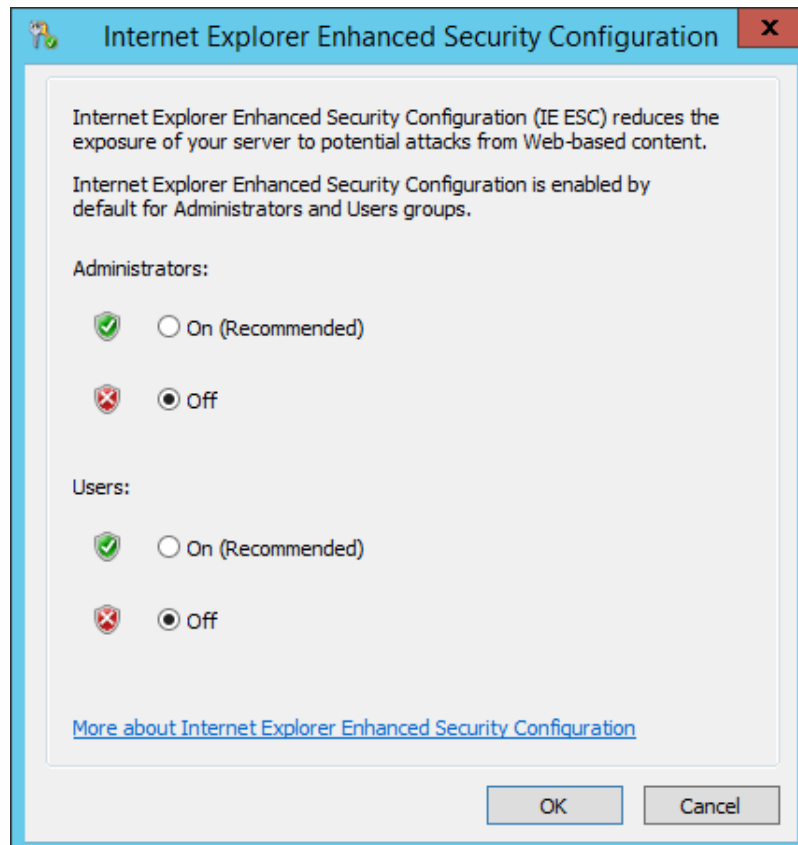
[新建云服务器组](#)

6. 确认弹性云服务器配置信息，阅读并勾选同意协议。
7. 单击“立即购买”，等待弹性云服务器创建完成。
8. 在弹性云服务器列表中，单击创建的弹性云服务器“操作”列的“远程登录”。
9. 根据界面提示，输入弹性云服务器密码，登录弹性云服务器。
10. 为了更好的获取和更新系统和软件，建议您更新镜像源为华为云镜像源，详细操作，请参见[如何使用自动化工具配置华为云镜像源\(x86\\_64和ARM\)?](#)。

## 14.2 添加 AD、DHCP、DNS、IIS 服务

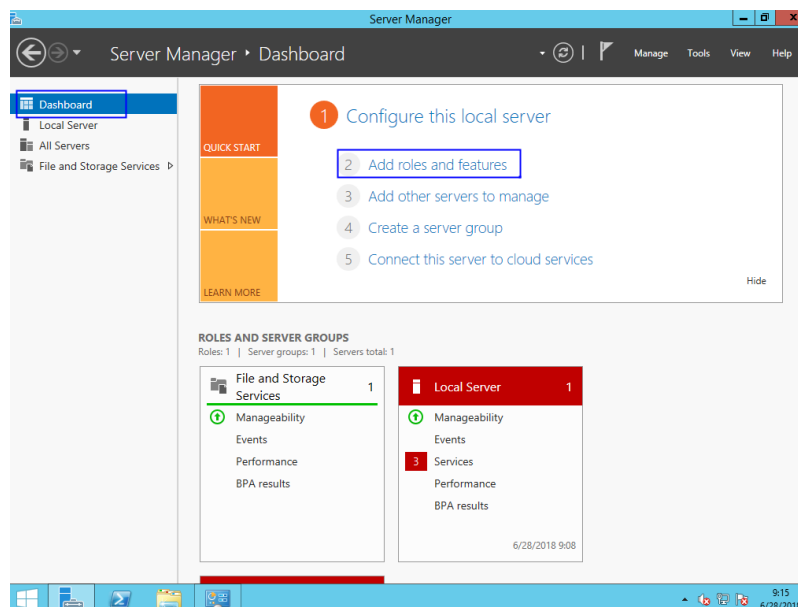
1. 在“Server Manger > Local Server”中设置“IE Enhanced Security Configuration”为“Off”，关闭IE增强安全设置。

图 14-4 IE 增强安全设置



2. 选择“Server Manger > Dashboard”。
3. 单击“Add roles and features”，为服务器添加新的角色和功能，包括DNS、DHCP、IIS、Net Framework3.5。

图 14-5 Add Roles and Features



4. 在“Server Roles”导航栏中，勾选“Active Directory Domain Services”、“DHCP Server”、“DNS Server”和“Web Server (IIS)”。

图 14-6 Select Server Roles 1

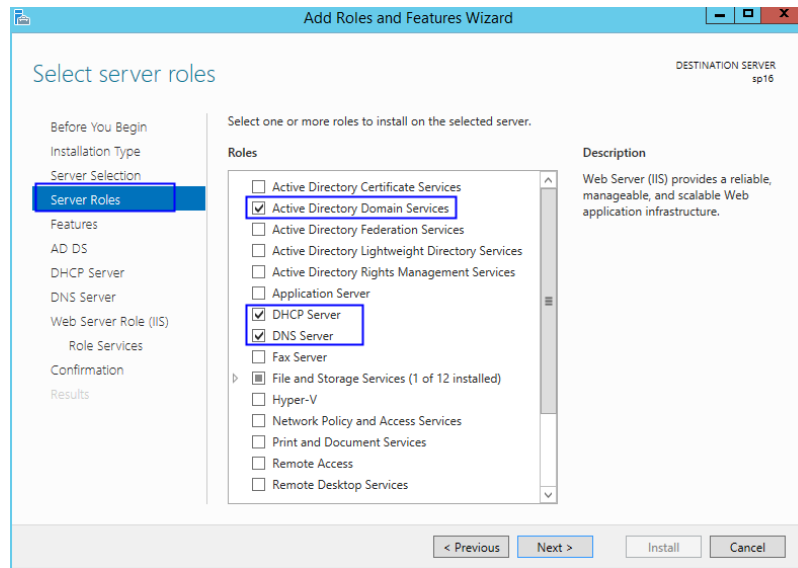
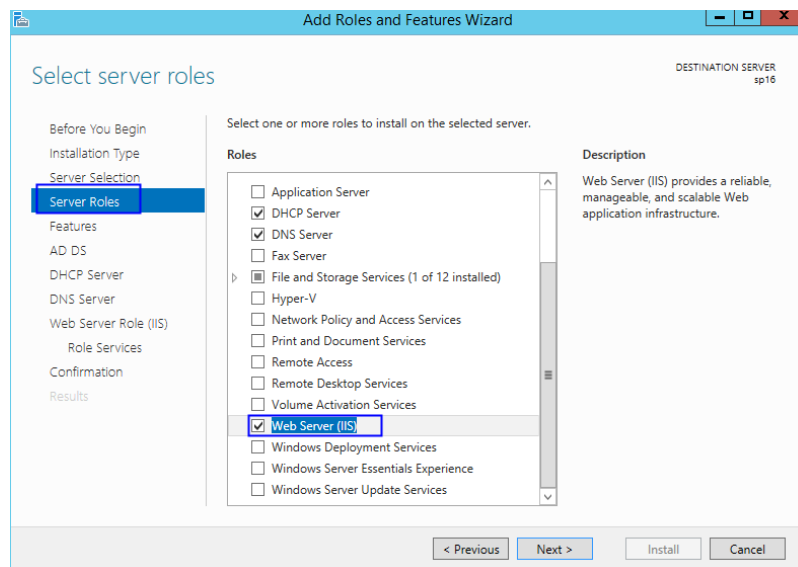
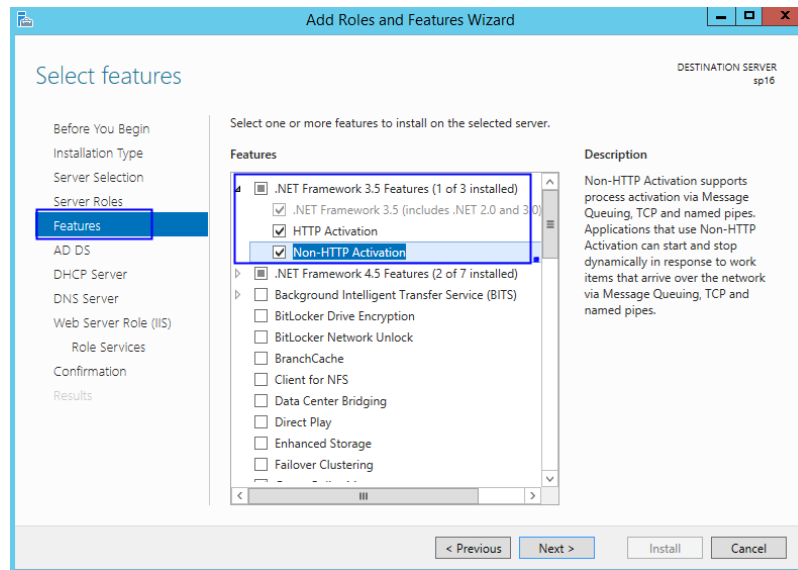


图 14-7 Select Server Roles 2



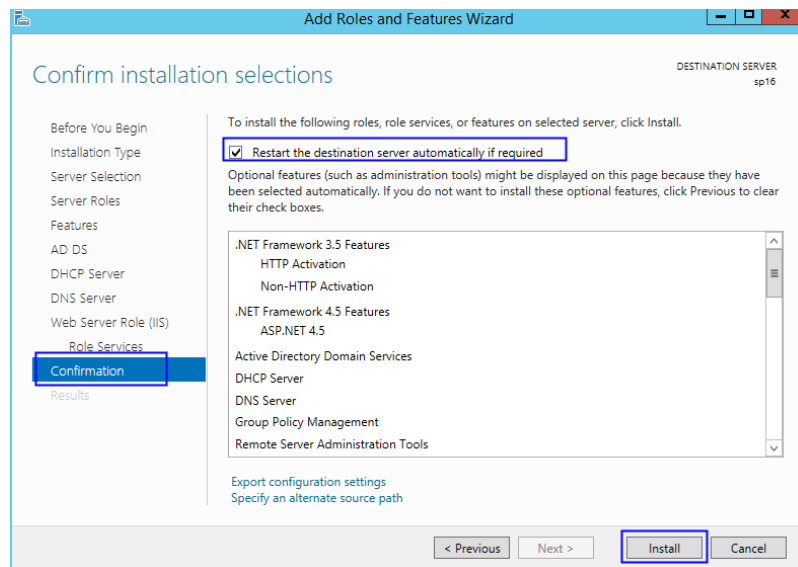
5. 单击“Next”。
6. 在“Features”导航栏中，勾选“.NET Framework 3.5 Features”。

图 14-8 Select Features



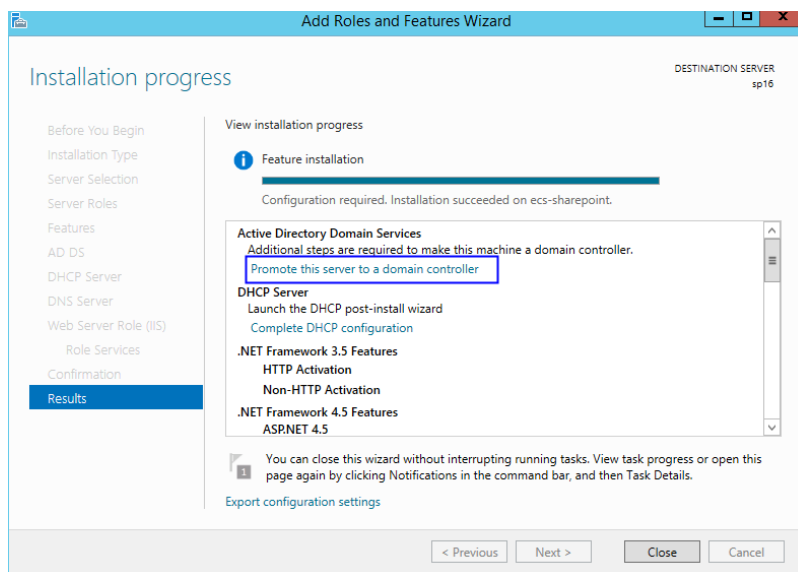
7. 单击“Next”，直至完成。
8. 在“Confirmation”导航栏中，勾选“Restart the destination server automatically if required”。

图 14-9 Confirm Installation Selections



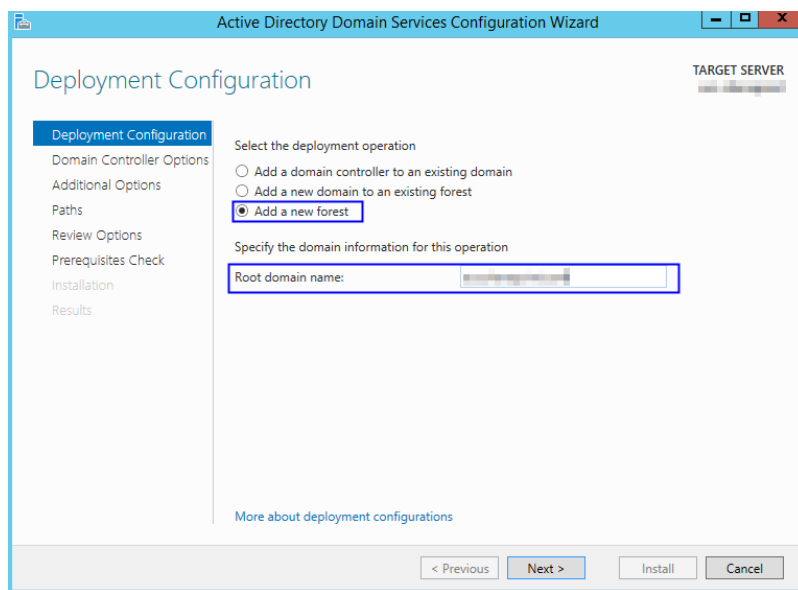
9. 单击“Install”，开始安装。
10. 安装完成后，单击“Promote this server to a domain controller”，开始配置AD服务。

图 14-10 AD Configuration



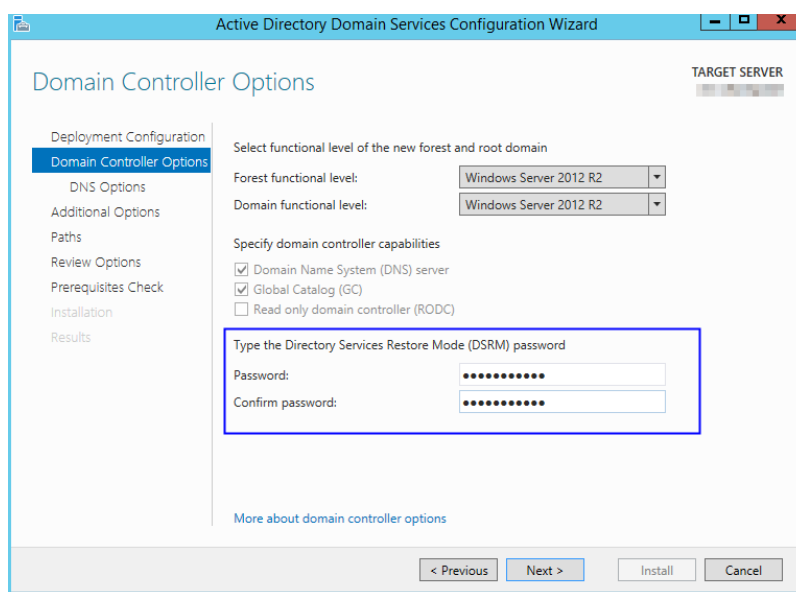
11. 因为没有已经存在的域环境，选择“Add a new forest”，创建一个新的forest。“Root domain name”设置为“sp160.com.cn”。

图 14-11 Add a new forest



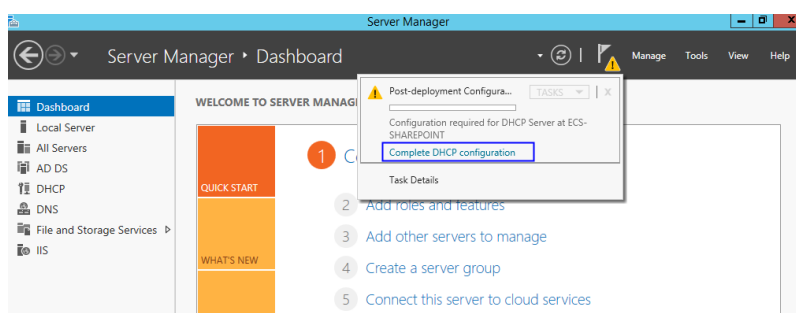
12. 单击“Next”。
13. 设置密码，该密码用于备份还原域控。

图 14-12 Set the Password



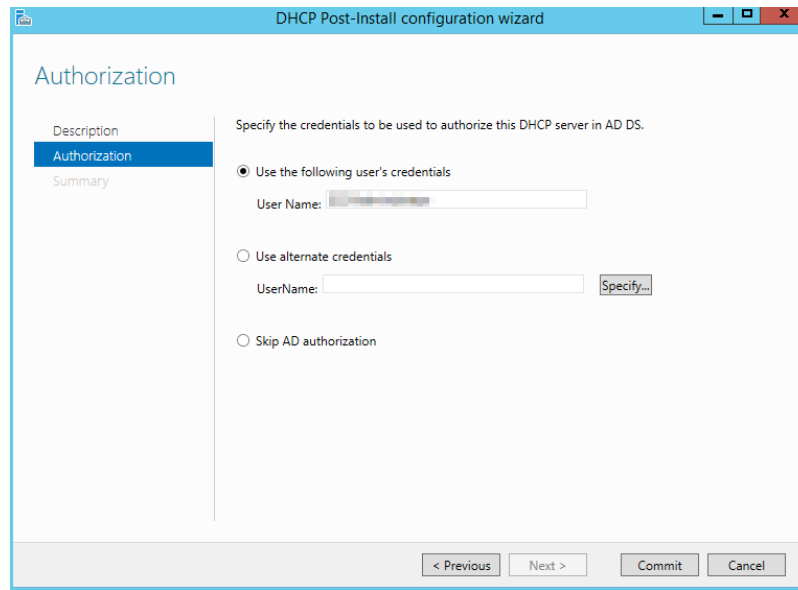
14. 单击“Next”，直至安装完成。
15. 单击“Complete DHCP Configuration”，开始配置DHCP功能。

图 14-13 DHCP Configuration 1



16. 单击“Next”，选择默认配置。

图 14-14 DHCP Configuration 2

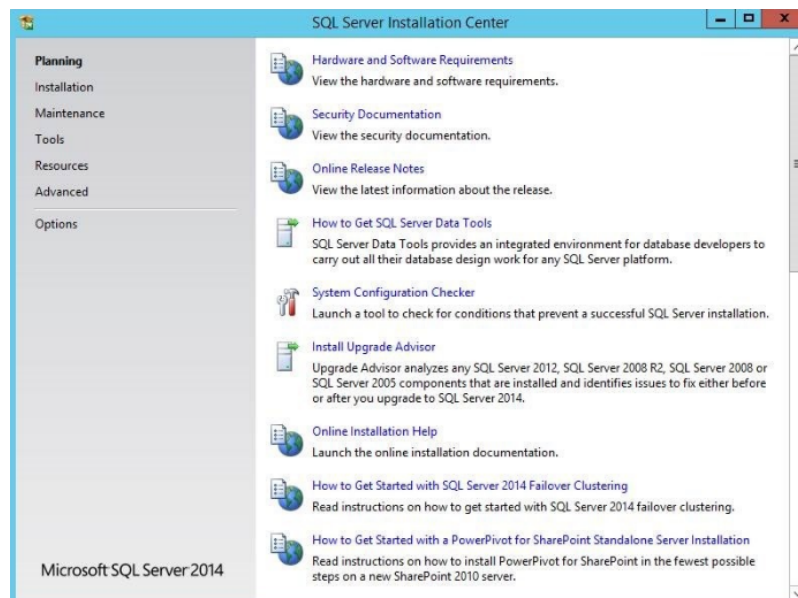


17. 单击“Commit”。
18. 配置完成，单击“Close”。

## 14.3 安装 SQL Server

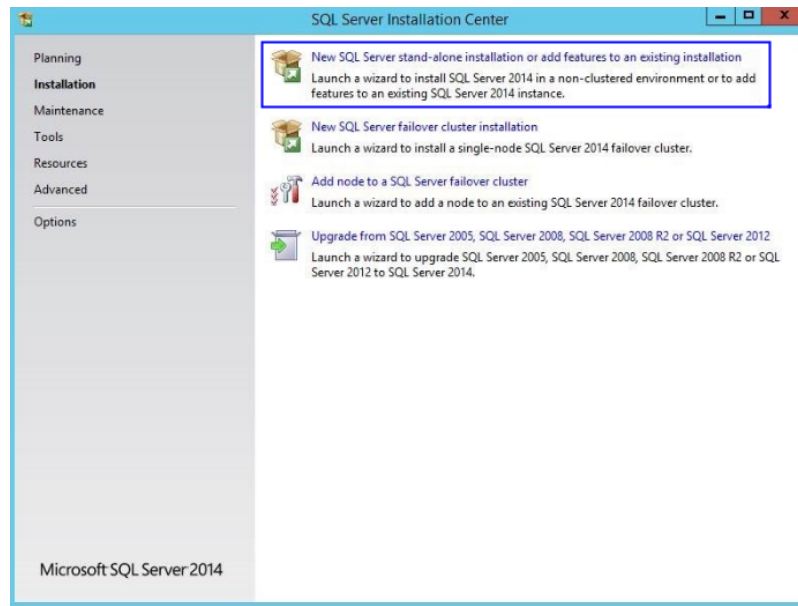
1. 双击“Setup.exe”文件，打开SQL Server安装向导。

图 14-15 SQL Server 安装向导



2. 在“Installation”导航栏中，单击第一个选项进行安装。

图 14-16 SQL Server 安装选项

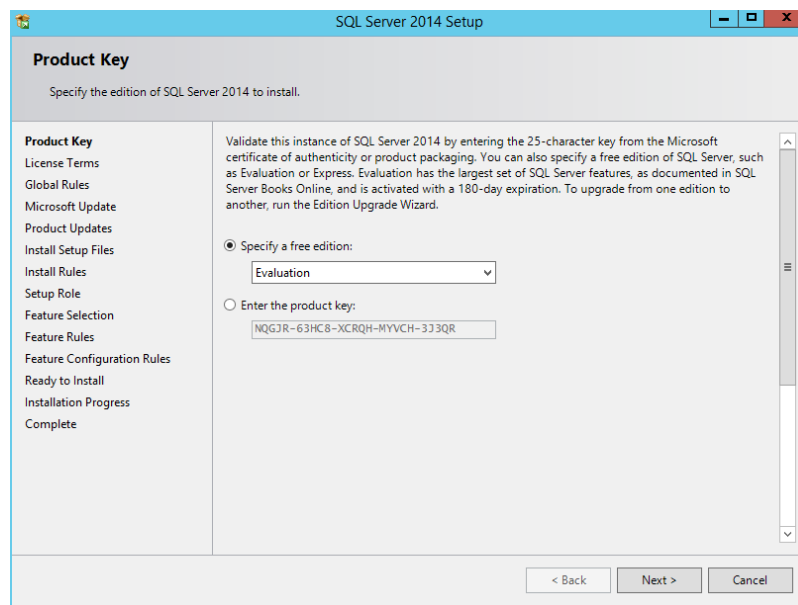


3. 选择“Specify a free edition”，采用免费镜像模式进行安装。

**说明**

若要正式搭建Sharepoint，需要输入密钥安装SQL Server完整版本。

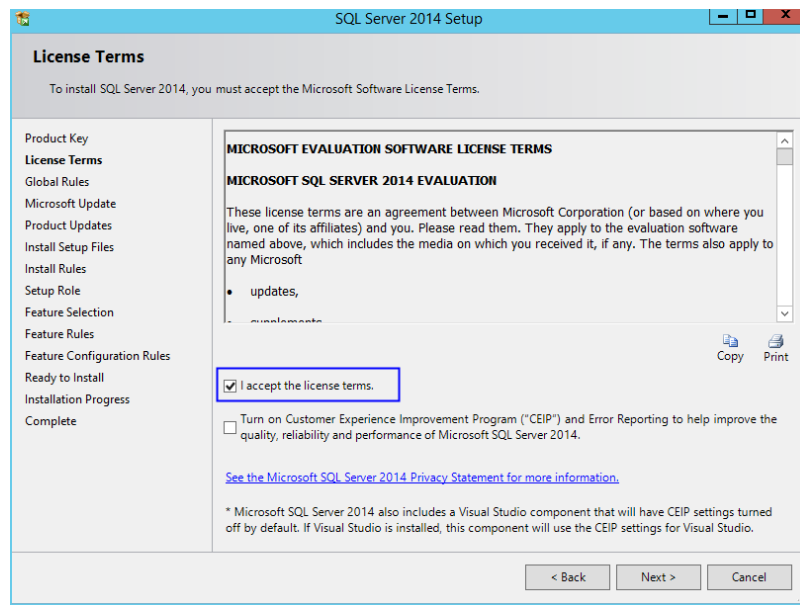
图 14-17 SQL Server 免费镜像



4. 勾选“I accept the license terms.”，单击“Next”。

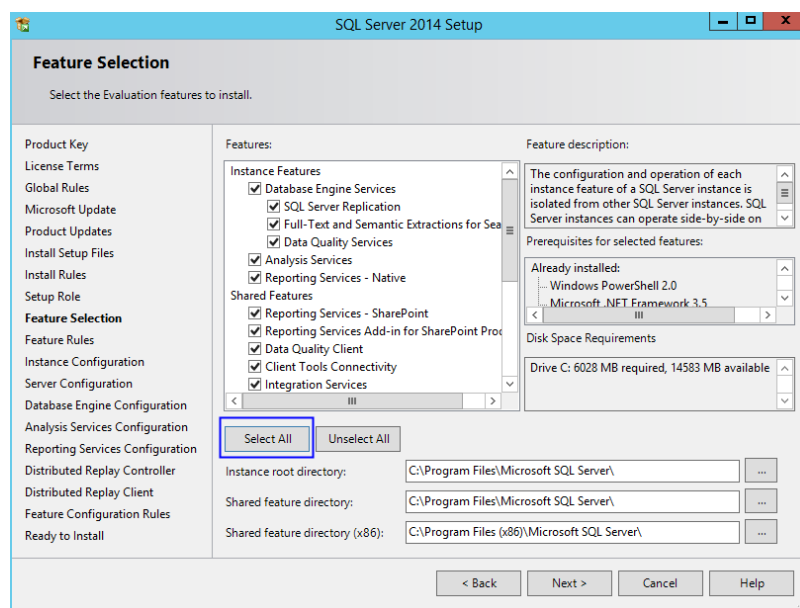


图 14-18 SQL Server License 选项



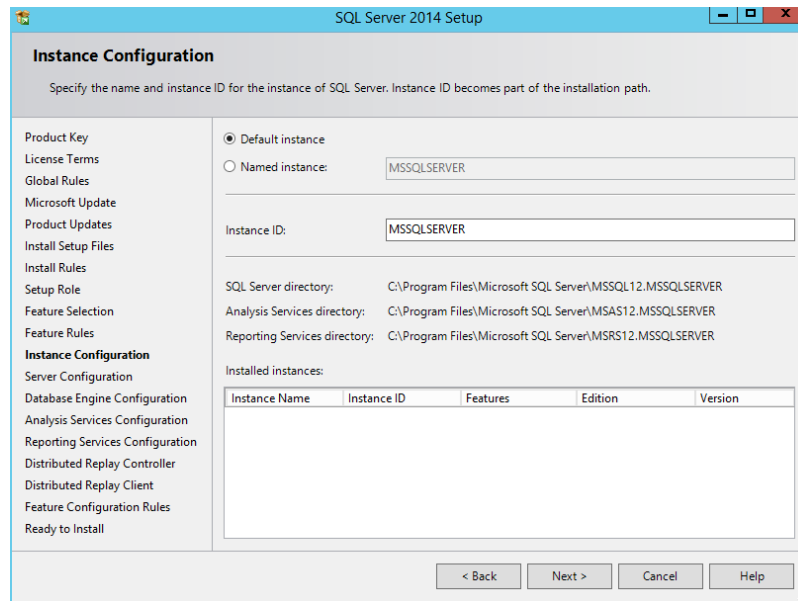
5. Microsoft Updates、Install Rules、Setup Role等安装采用系统默认项安装，单击“Next”继续安装。
6. 单击“Select ALL”选择安装全部功能，单击“Next”。

图 14-19 SQL Server Feature Selection



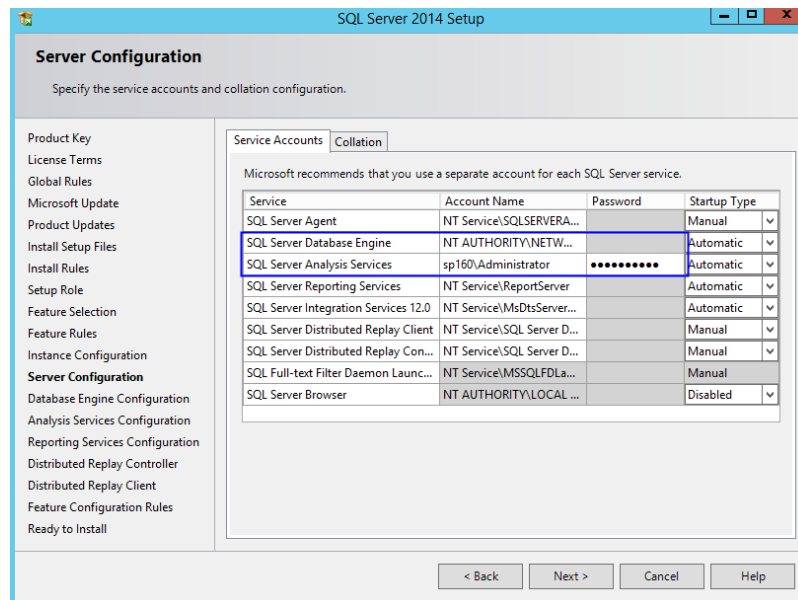
7. 勾选“Default instance”，使用默认值配置实例。

图 14-20 配置 SQL Server 实例



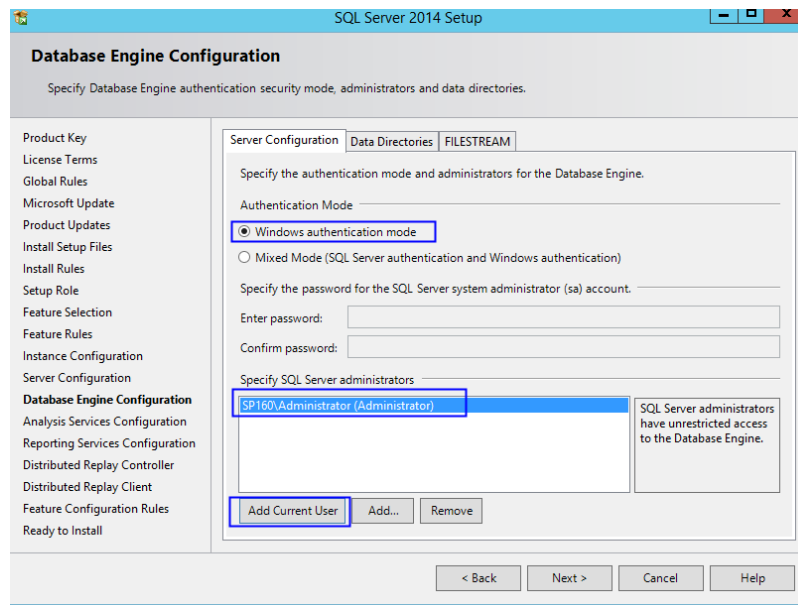
8. 设置SQL Server Configuration。
  - 修改“SQL Server Database Engine”的“Account Name”为“NT AUTHORITY\NETWORK SERVICE”。
  - 设置“SQL Server Analysis Services”为添加AD、DHCP、DNS、IIS服务中11~13设置的域账户及密码。

图 14-21 配置 SQL Server 服务账户



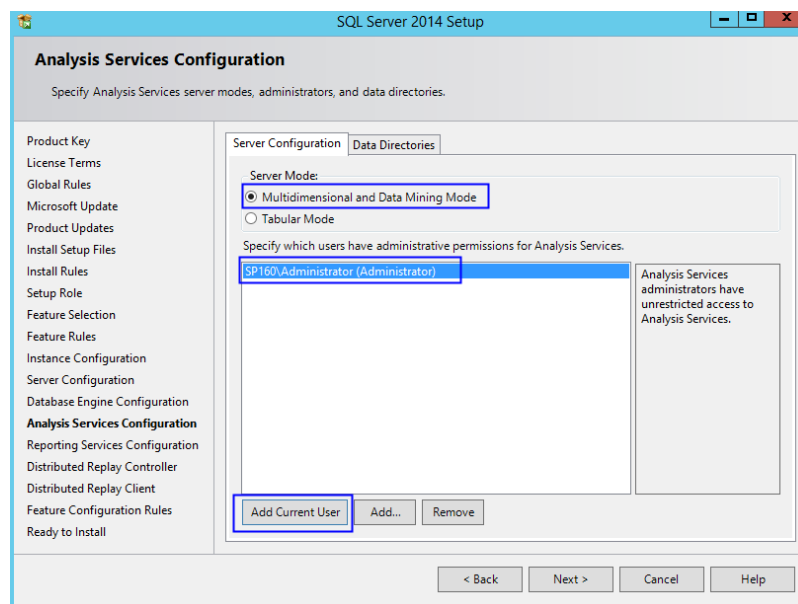
9. 单击“Add Current User”，将当前账号作为SQL Server的管理员账号，单击“Next”。

图 14-22 添加 SQL Server 管理员账号



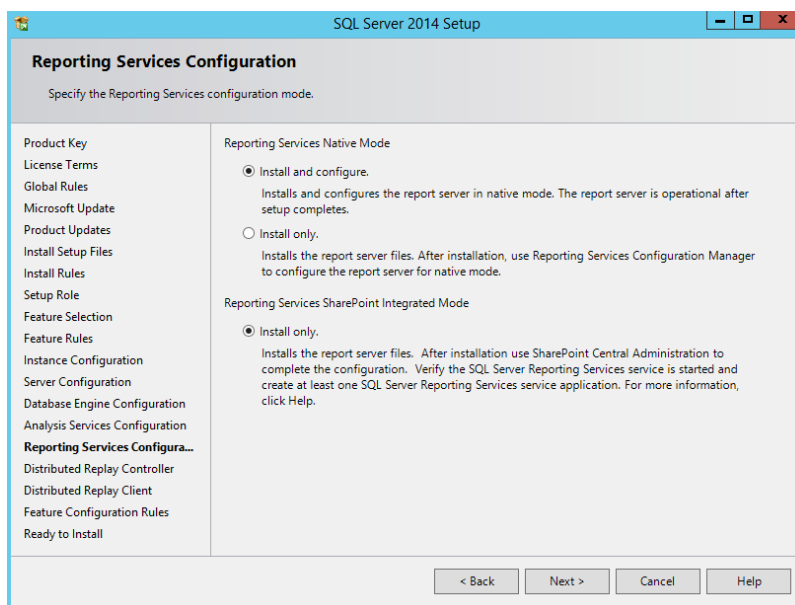
10. 单击“Add Current User”，为当前账号添加Analysis Services的管理员权限，单击“Next”。

图 14-23 添加 SQL Server 管理员账号 2



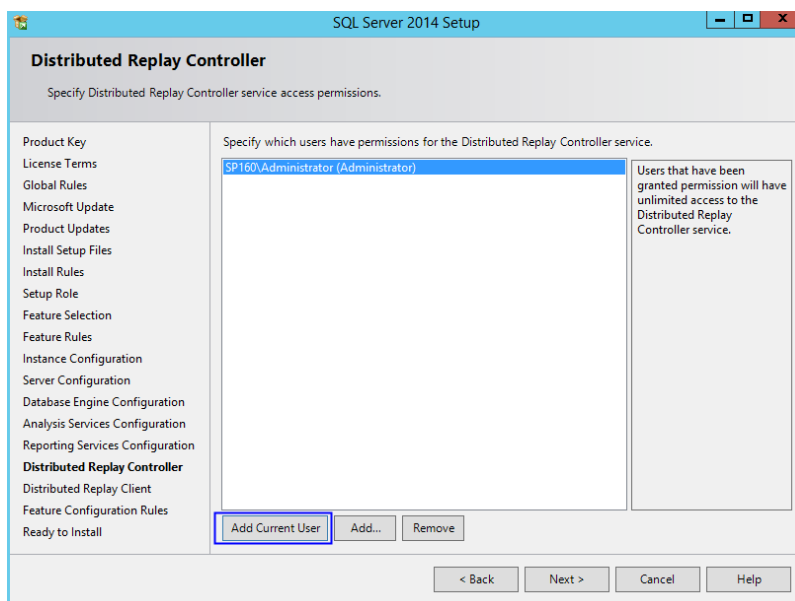
11. 设置Reporting Services Configuration为默认值，单击“Next”。

图 14-24 设置 Reporting Services Configuration



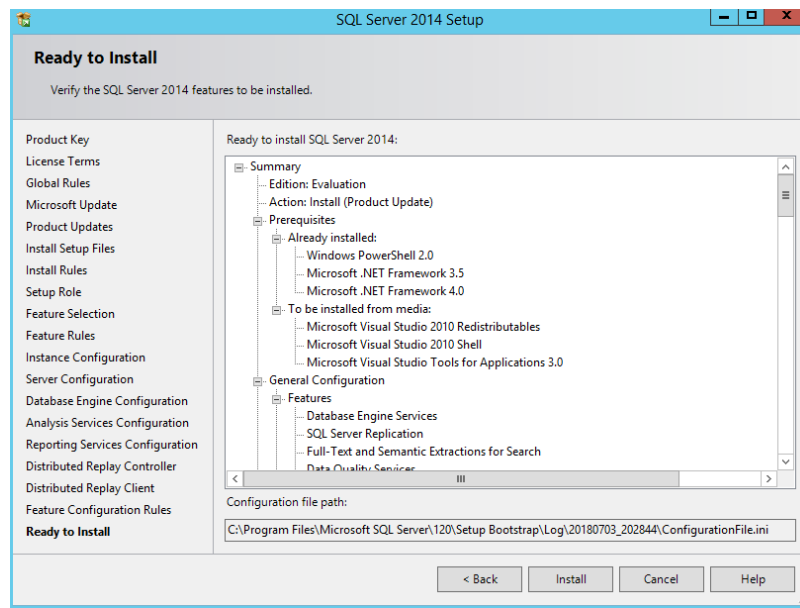
12. 单击“Add Current User”，为当前账号添加Distribution Replay Controller service的权限，单击“Next”。

图 14-25 设置 Distribution Replay Controller



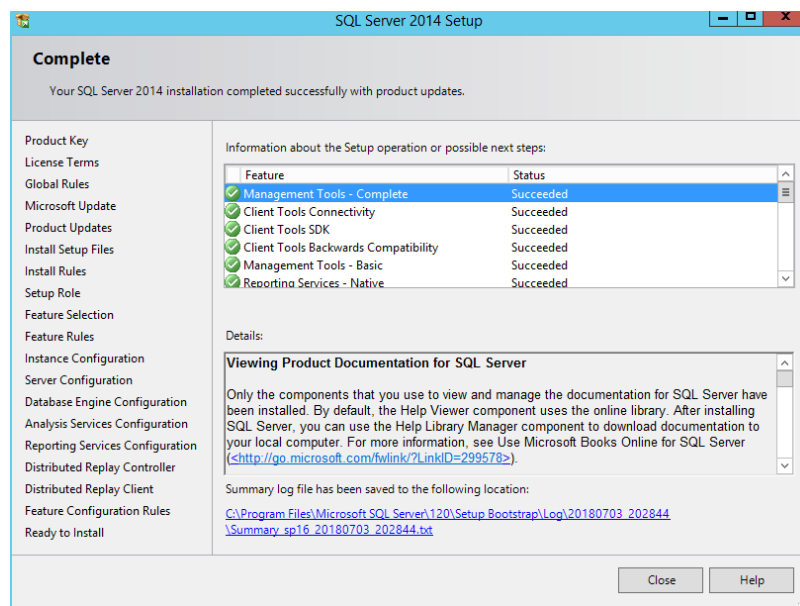
13. 确认SQL Server配置，单击“Install”开始安装。

图 14-26 SQL Server 安装准备完成



14. 单击“Close”，SQL Server安装完成。

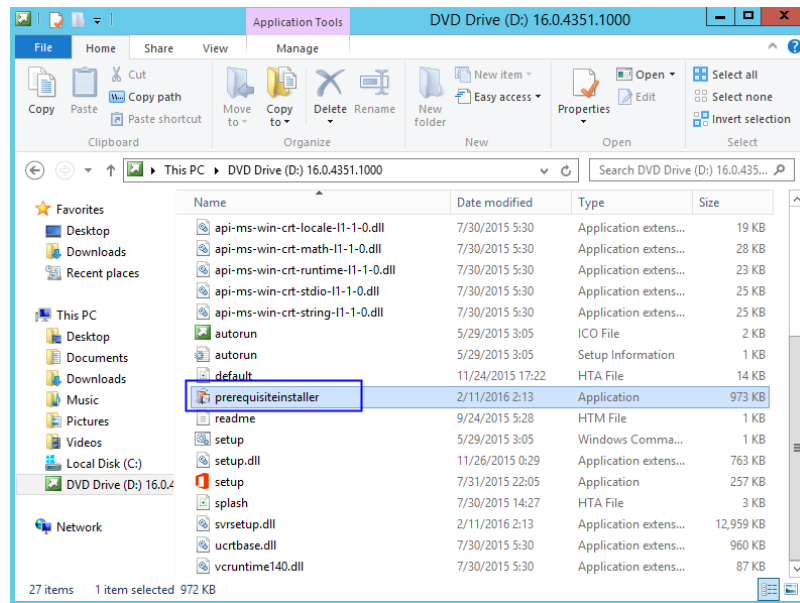
图 14-27 SQL Server 安装完成



## 14.4 安装 Sharepoint 2016

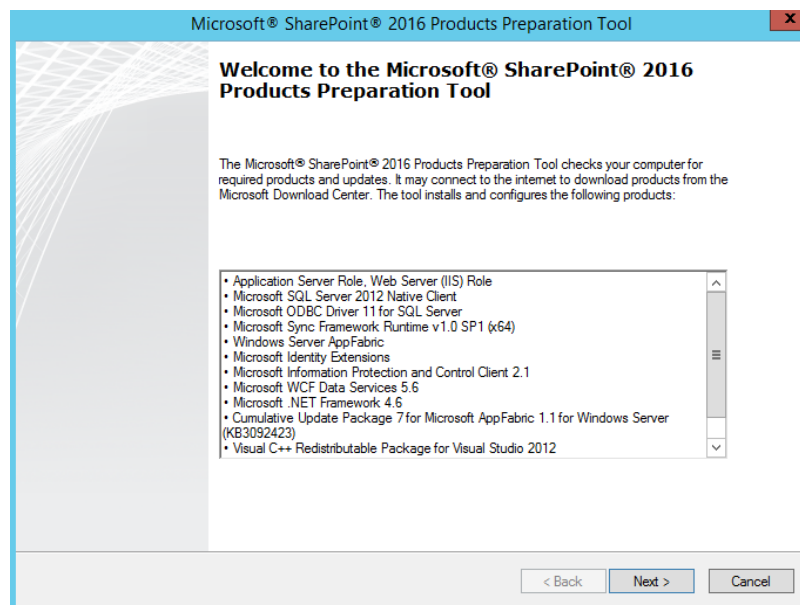
1. 打开镜像文件，双击准备工具的可执行文件，安装SharePoint 2016 准备工具。

图 14-28 Sharepoint 准备工具



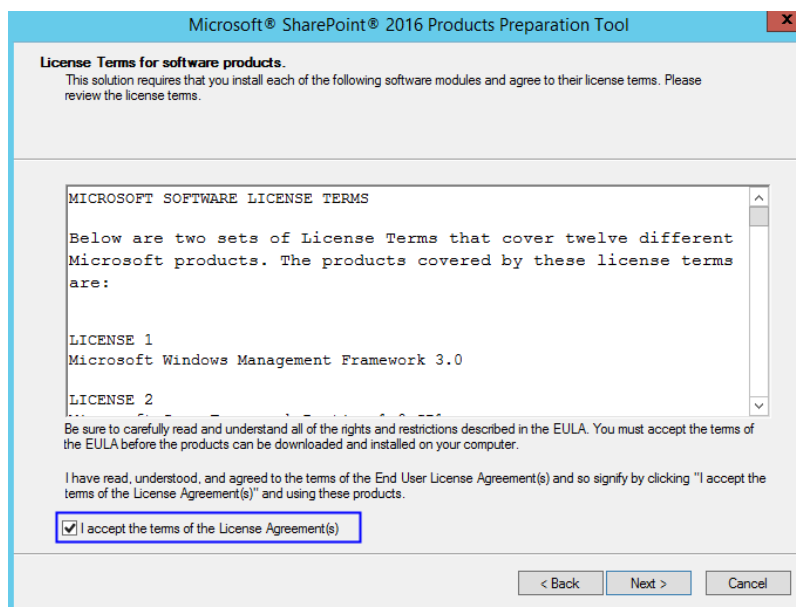
2. 打开Sharepoint准备工具的安装向导，单击“Next”。

图 14-29 Sharepoint 准备工具安装向导



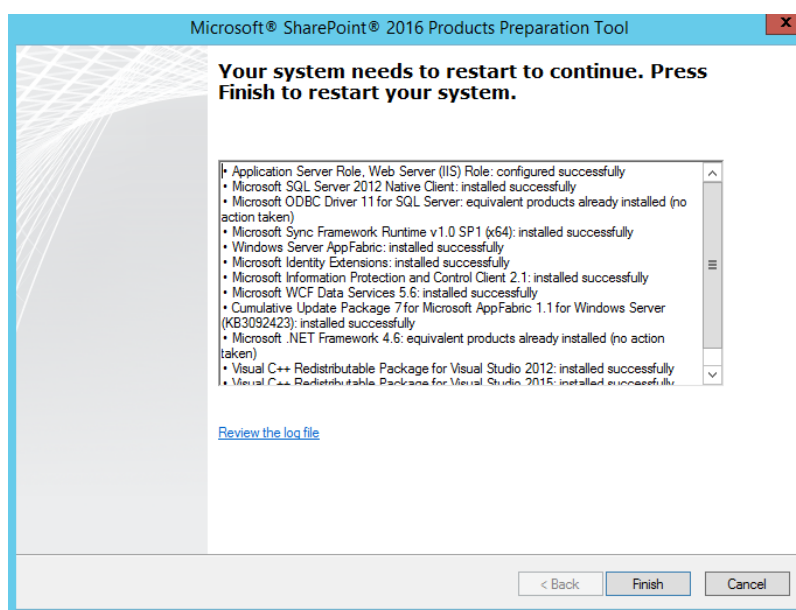
3. 勾选“I accept the terms of the License Agreement(s)”，接受License，单击“Next”。

图 14-30 Sharepoint 准备工具 License



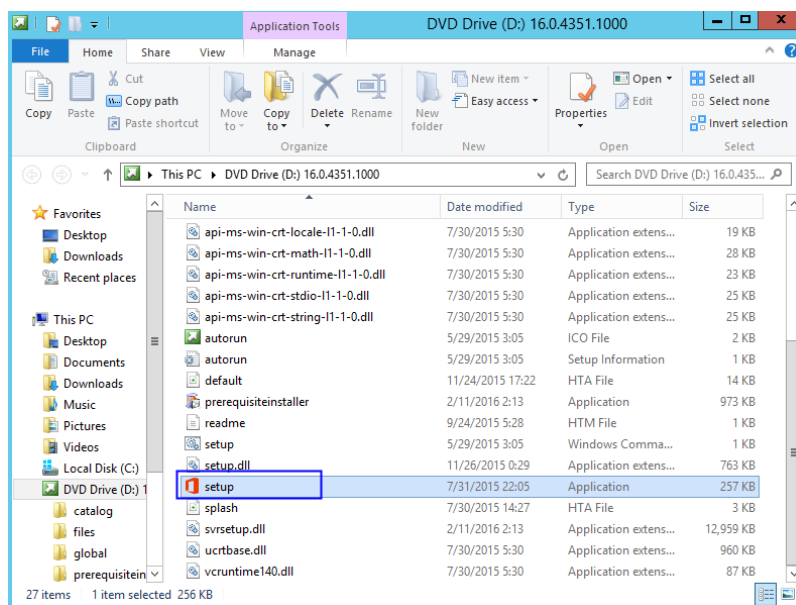
4. Sharepoint准备工具安装完成，单击“Finish”重启系统。

图 14-31 Sharepoint 准备工具安装完成



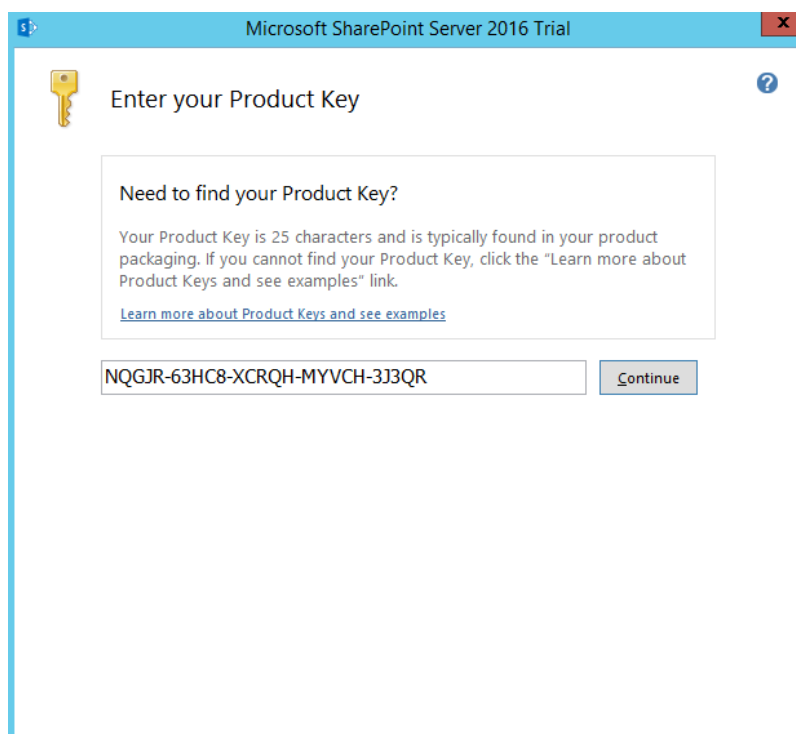
5. 双击安装文件，开始安装Sharepoint。

图 14-32 安装 Sharepoint



6. 输入Sharepoint产品密钥，Sharepoint 180天试用版Key为“NQGJR-63HC8-XCRQH-MYVCH-3J3QR”。

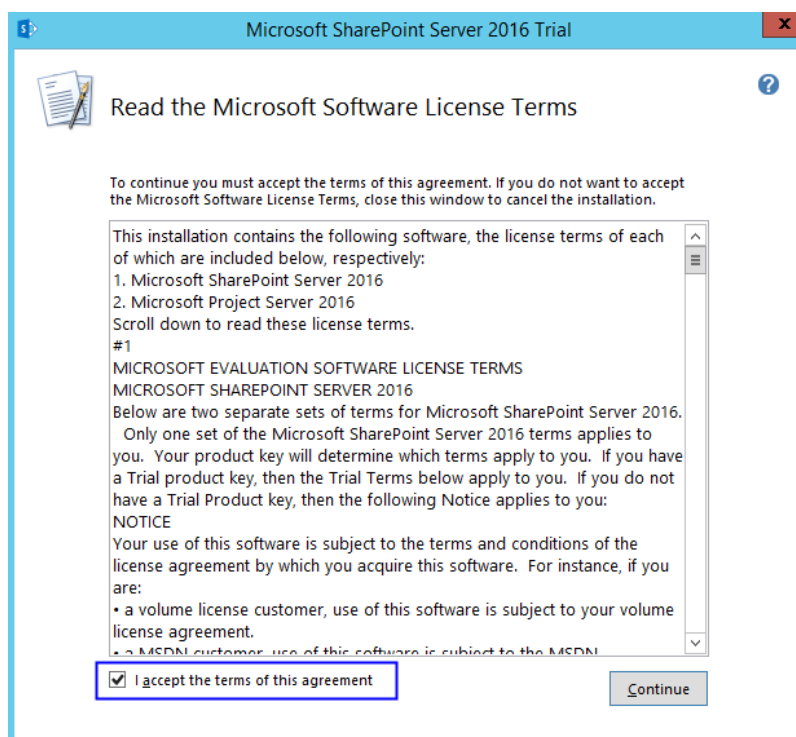
图 14-33 Sharepoint 产品密钥



7. 接受Licesen，单击“Continue”。

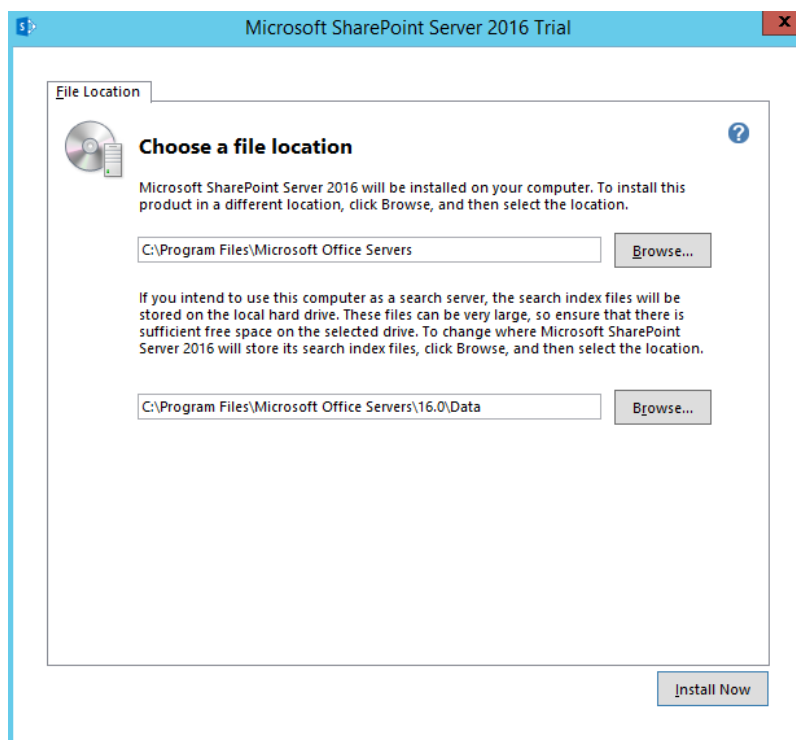


图 14-34 Sharepoint License Terms



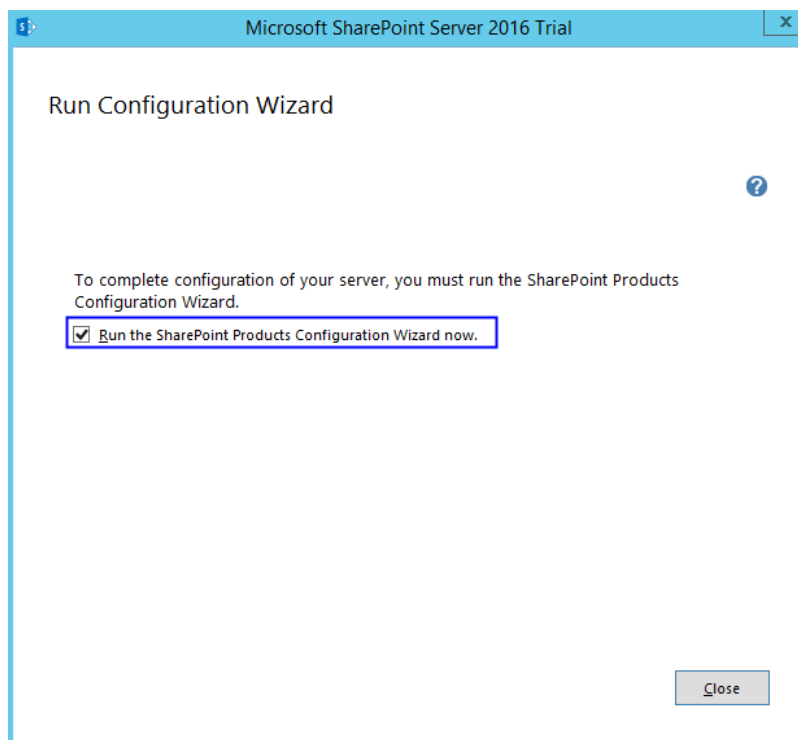
8. 选择安装文件路径，采用默认值。

图 14-35 Sharepoint 安装文件路径



9. 单击“Install Now”，开始安装。
10. Sharepoint安装完成，勾选“Run the Sharepoint Products Configuration Wizard now.”，在安装完成后运行Sharepoint产品配置向导。

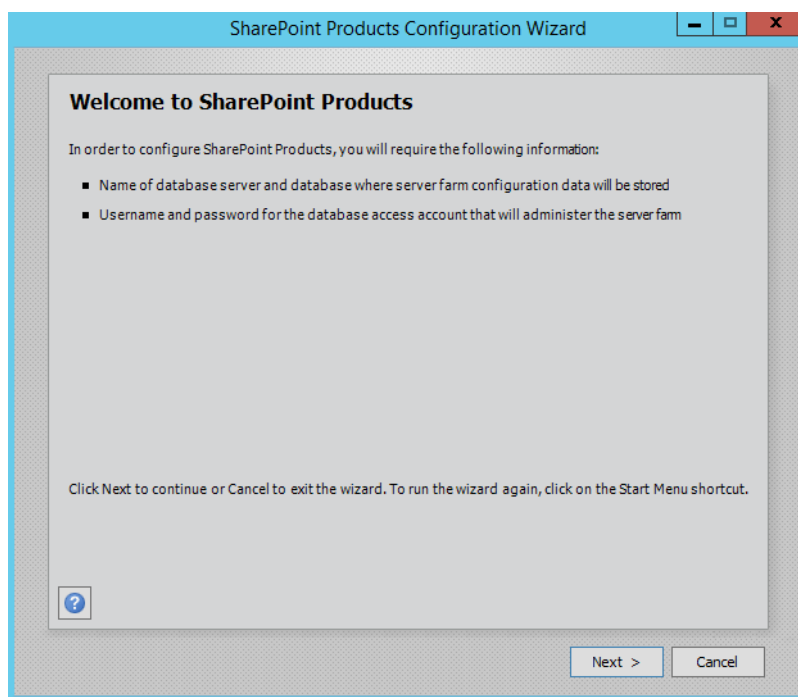
图 14-36 Sharepoint 安装完成



## 14.5 配置 Sharepoint 2016

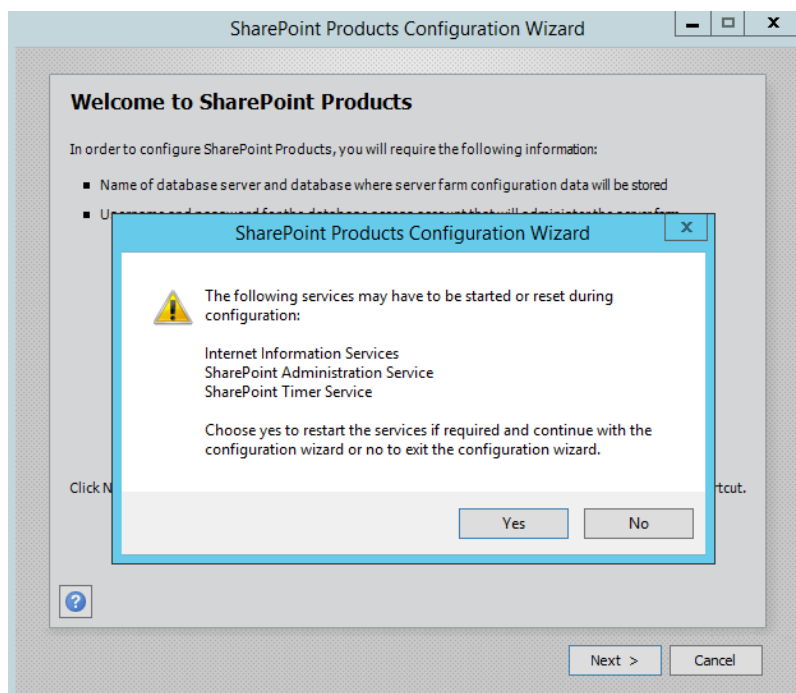
1. 在运行的Sharepoint产品配置向导中，单击“Next”。

图 14-37 Sharepoint 产品配置向导



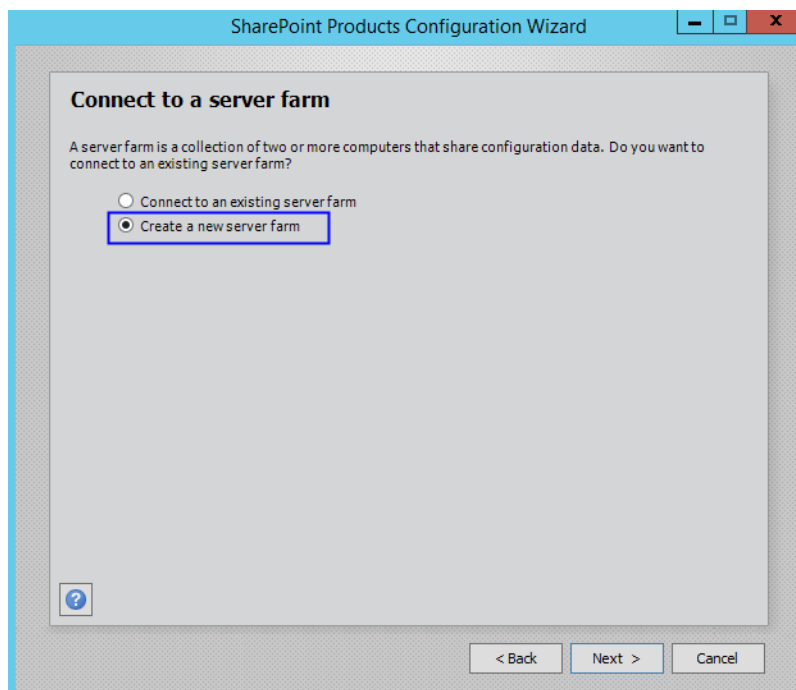
2. 单击“Yes”，允许在配置过程中重启服务。

图 14-38 重启服务提醒



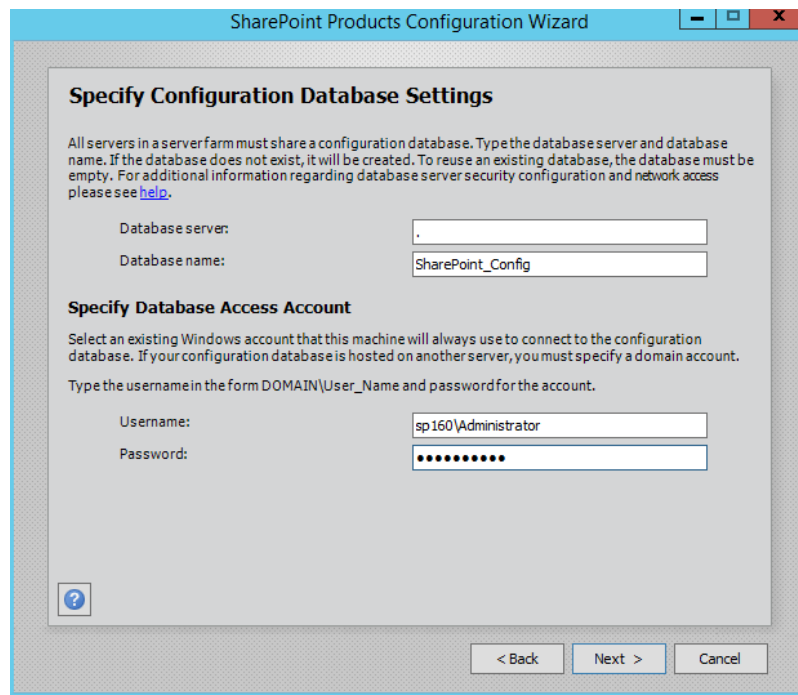
3. 因为没有已经存在的服务器场，所以选择“Create a new server farm”，创建一个新的服务器场。

图 14-39 创建 Sharepoint 服务器场



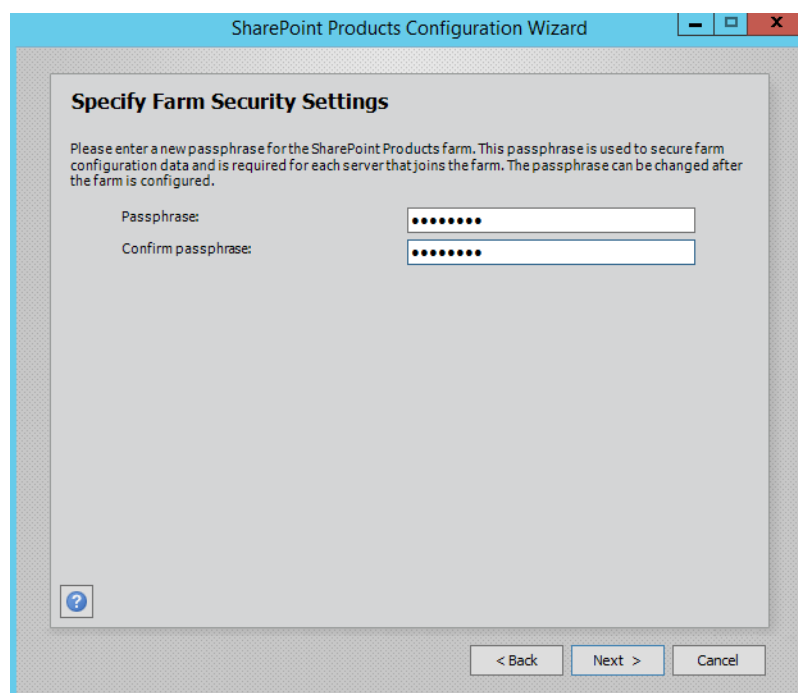
4. 配置指定数据库设置，由于Sharepoint的数据库在本机，所以填写本机的数据库及账户，单击“Next”。

图 14-40 指定 Sharepoint 数据库设置



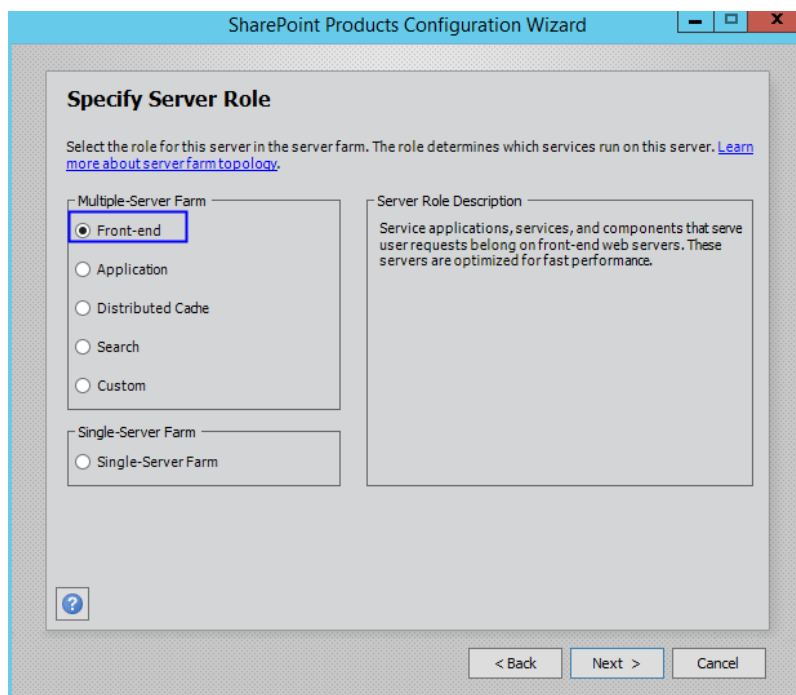
5. 填写服务器场的密码，单击“Next”。

图 14-41 设置 Sharepoint 服务器场密码



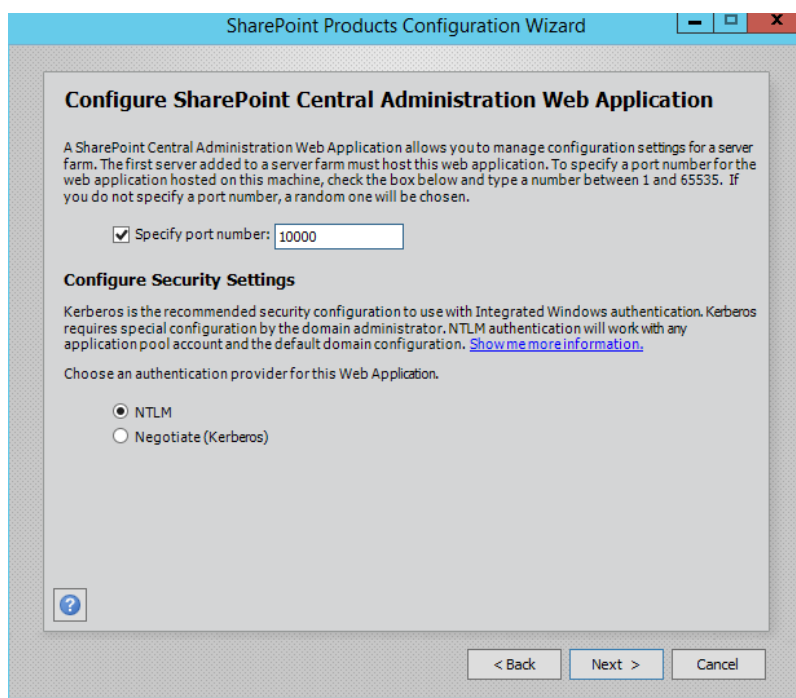
6. 设置服务器角色，选择“Front-end”，单击“Next”。

图 14-42 设置 Sharepoint 服务器角色



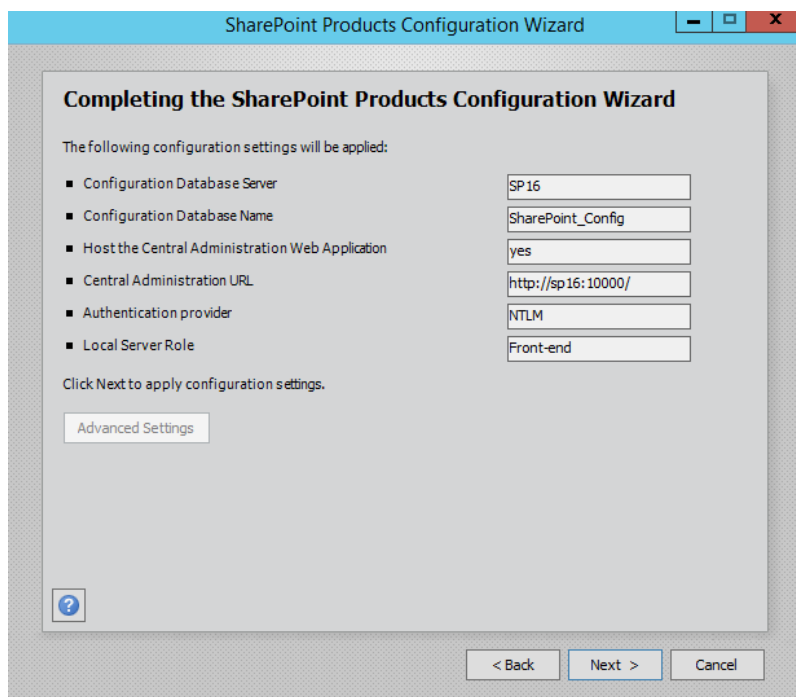
7. 设置Sharepoint管理中心的端口号为“10000”。

图 14-43 Sharepoint 管理中心端口号



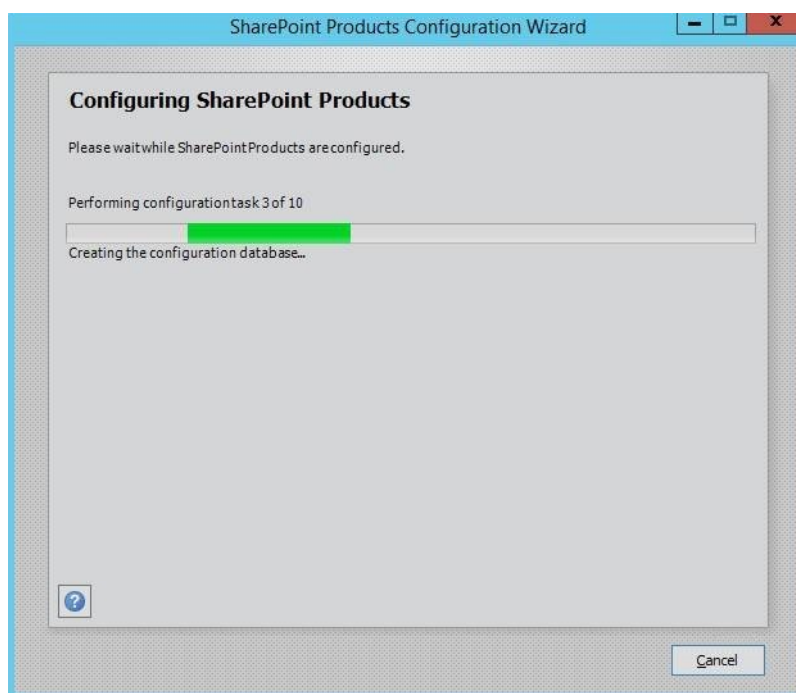
8. 查看并确认Sharepoint配置。

图 14-44 Sharepoint 配置



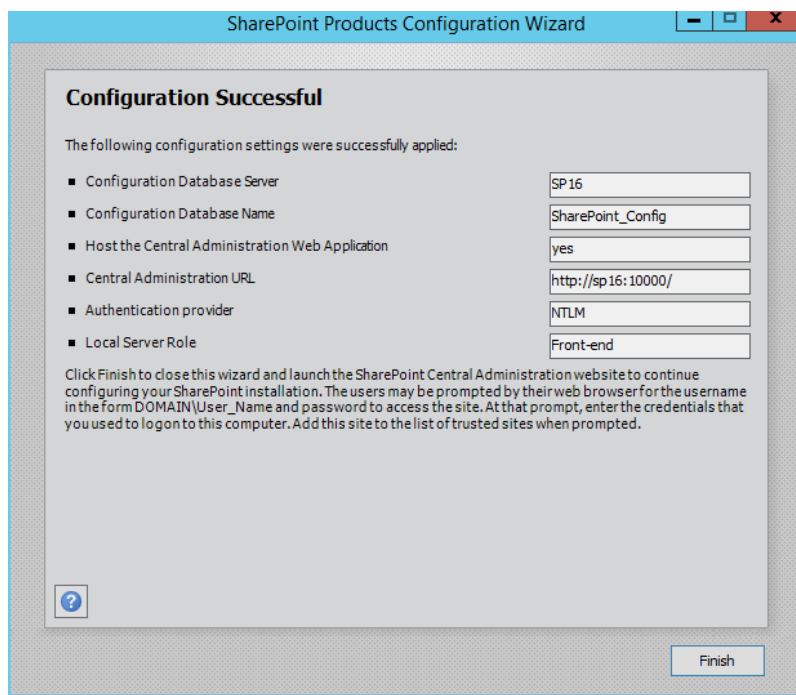
9. 单击“Next”，开始配置Sharepoint，共10步。

图 14-45 配置 Sharepoint 进度



10. Sharepoint配置成功，单击“Finish”。

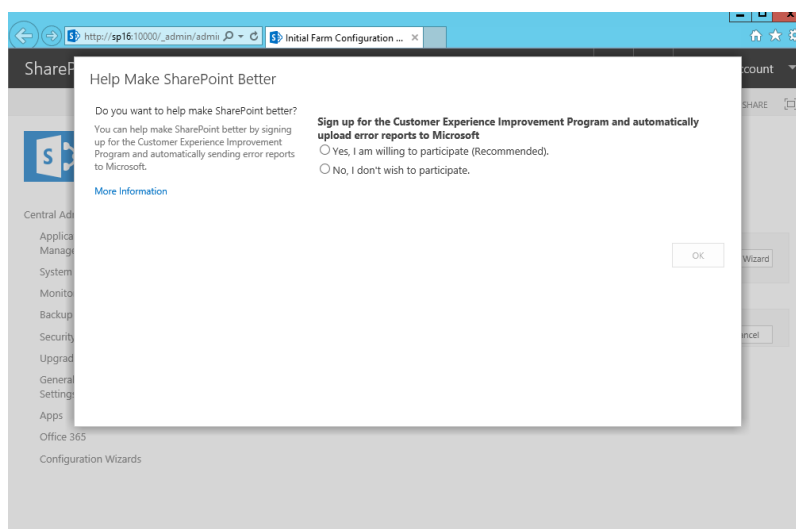
图 14-46 Sharepoint 配置成功



## 14.6 验证 Sharepoint 2016

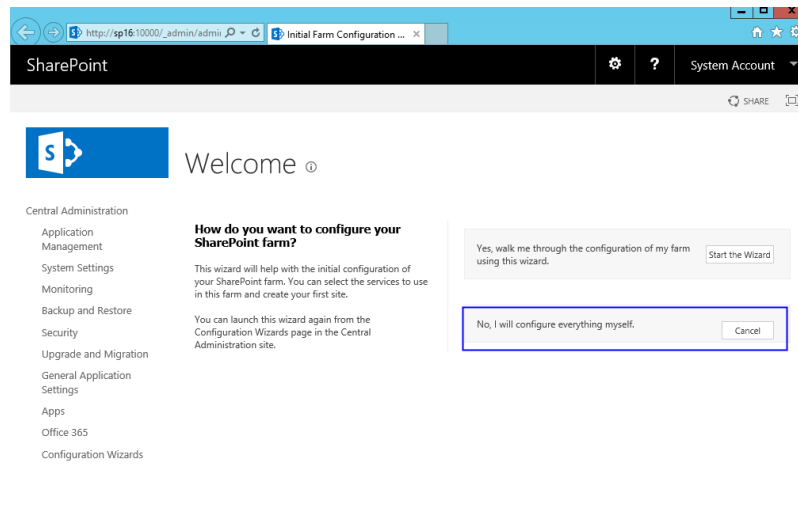
1. 通过配置的Sharepoint管理中心的URL，打开SharePoint的管理中心。

图 14-47 SharePoint 管理中心



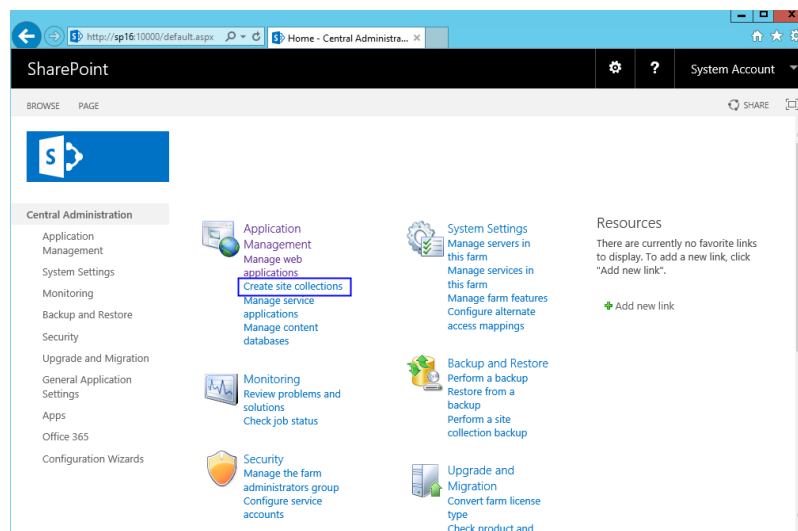
2. 选择Sharepoint Farm的配置方法，本操作采用自主配置，单击“Cancel”。若想要通过向导配置Sharepoint Farm，可以单击"Start the Wizard"。

图 14-48 SharePoint Farm 配置



3. 在Sharepoint管理中心，单击“Create site collections”创建一个Sharepoint站点。

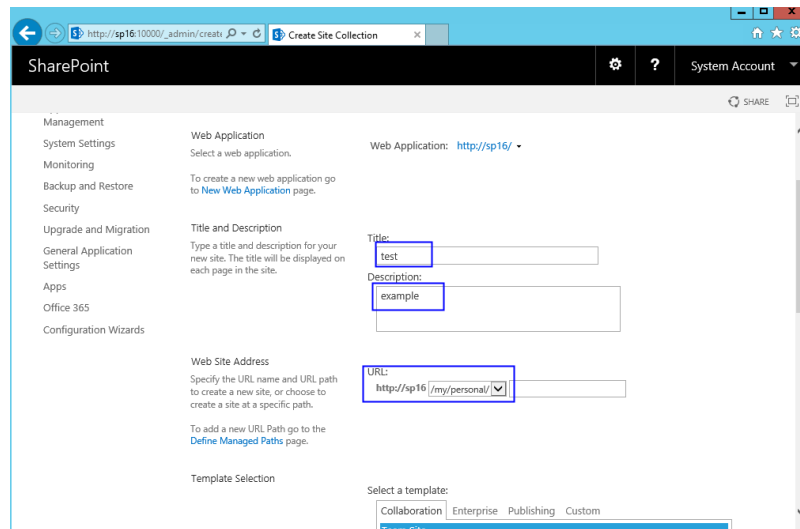
图 14-49 创建 Sharepoint 站点



4. 根据界面提示内容，设置Sharepoint站点参数。

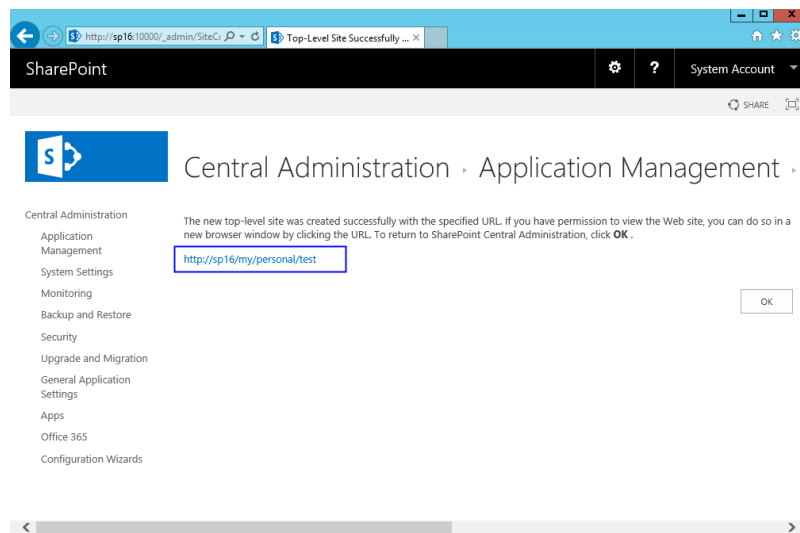


图 14-50 设置 Sharepoint 站点参数



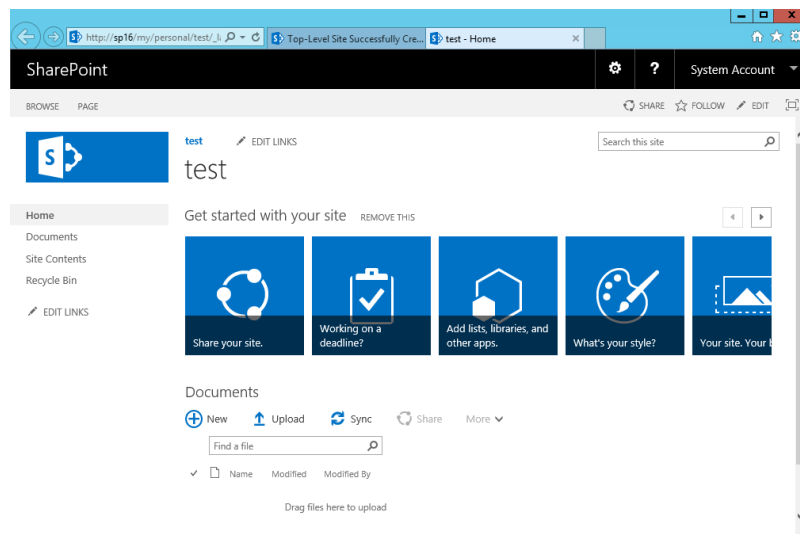
5. 创建Sharepoint Top-level站点成功，单击链接打开对应页面。

图 14-51 创建 SharePoint Top-level 站点成功



6. 在打开的Sharepoint站点，您可以根据界面提示设计您的网页。

图 14-52 Sharepoint 验证



# 15 手工搭建 LNMP 环境 (CentOS7.2 PHP7.0)

## 简介

本文主要介绍了在华为云上如何使用弹性云服务器的Linux实例手工搭建LNMP平台的web环境。该指导具体操作以CentOS 7.2 64位操作系统为例。

Linux实例手工部署LNMP环境具体操作步骤如下：

1. [安装nginx。](#)
2. [安装MySQL。](#)
3. [安装PHP。](#)
4. [浏览器访问测试。](#)

## 前提条件

1. 弹性云服务器已绑定弹性公网IP。
2. 弹性云服务器所在安全组添加了如下表所示的安全组规则，具体步骤参见[安全组添加安全组规则](#)。

表 15-1 安全组规则

| 方向  | 协议/应用    | 端口/范围 | 源地址       |
|-----|----------|-------|-----------|
| 入方向 | HTTP(80) | 80    | 0.0.0.0/0 |

3. 为了更好的获取和更新系统和软件，建议您更新镜像源为华为云镜像源，详细操作，请参见[如何使用自动化工具配置华为云镜像源\(x86\\_64和ARM\)?](#)。

## 资源规划

本次实践所用的资源配置及软件版本如表15-2中所示。当您使用不同的硬件规格或软件版本时，本指导中的命令及参数可能会发生改变，需要您根据实际情况进行调整。

表 15-2 资源和成本规划

| 资源     | 资源说明                                                                                                                                                                                                                                                                                    | 成本说明                                                                                                                                      |
|--------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------|
| 弹性云服务器 | <ul style="list-style-type: none"><li>● 计费模式：按需计费</li><li>● 可用区：可用区1</li><li>● 规格：s6.large.2</li><li>● 镜像：CentOS 7.2 64bit</li><li>● 系统盘：40G</li><li>● 弹性公网IP：现在购买</li><li>● 线路：全动态BGP</li><li>● 公网带宽：按流量计费</li><li>● 带宽大小：5 Mbit/s</li></ul>                                           | ECS涉及以下几项费用： <ul style="list-style-type: none"><li>● 云服务器</li><li>● 云硬盘</li><li>● 弹性公网IP</li></ul> 具体的计费方式及标准请参考 <a href="#">计费模式概述</a> 。 |
| Nginx  | 是一个高性能的HTTP和反向代理web服务器。<br>获取方式：<br><a href="http://nginx.org/packages/centos/7/noarch/RPMS/nginx-release-centos-7-0.el7ngx.noarch.rpm">http://nginx.org/packages/centos/7/noarch/RPMS/nginx-release-centos-7-0.el7ngx.noarch.rpm</a>                                                   | 免费                                                                                                                                        |
| MySQL  | 是一款开源的关系数据库软件。<br>获取方式：<br><a href="http://dev.mysql.com/get/mysql57-community-release-el7-10.noarch.rpm">http://dev.mysql.com/get/mysql57-community-release-el7-10.noarch.rpm</a>                                                                                                      | 免费                                                                                                                                        |
| PHP    | 是一款开源软件，用于Web开发。<br>获取方式：<br><a href="https://mirror.webtatic.com/yum/el7/epel-release.rpm">https://mirror.webtatic.com/yum/el7/epel-release.rpm</a><br><a href="https://mirror.webtatic.com/yum/el7/webtatic-release.rpm">https://mirror.webtatic.com/yum/el7/webtatic-release.rpm</a> | 免费                                                                                                                                        |

## 操作步骤

### 步骤1 安装nginx。

1. 登录弹性云服务器。
2. 执行以下命令，下载对应当前系统版本的nginx包。

```
wget http://nginx.org/packages/centos/7/noarch/RPMS/nginx-release-centos-7-0.el7ngx.noarch.rpm
```

3. 执行以下命令，建立Nginx的yum仓库。  
**rpm -ivh nginx-release-centos-7-0.el7.ngx.noarch.rpm**
4. 执行以下命令，安装Nginx。  
**yum -y install nginx**
5. 执行以下命令，启动Nginx并设置开机启动。  
**systemctl start nginx**  
**systemctl enable nginx**
6. 查看启动状态。  
**systemctl status nginx.service**
7. 使用浏览器访问 “http://服务器IP地址”，显示如下页面，说明nginx安装成功。

图 15-1 测试访问 nginx



## 步骤2 安装MySQL。

1. 依次执行以下命令，安装MySQL。  
**wget -i -c http://dev.mysql.com/get/mysql57-community-release-el7-10.noarch.rpm**  
**yum -y install mysql57-community-release-el7-10.noarch.rpm**  
**yum -y install mysql-community-server --nogpgcheck**
2. 依次执行以下命令，启动MySQL服务并设置开机自启动。  
**systemctl start mysqld**  
**systemctl enable mysqld**
3. 查看MySQL运行状态。  
**systemctl status mysqld.service**  

```
[root@ecs-adc3 ~]# systemctl status mysqld.service
● mysqld.service - MySQL Server
 Loaded: loaded (/usr/lib/systemd/system/mysqld.service; enabled; vendor preset: disabled)
 Active: active (running) since Mon 2021-08-16 19:33:40 CST; 36s ago
 Docs: man:mysqld(8)
 http://dev.mysql.com/doc/refman/en/using-systemd.html
 Main PID: 7916 (mysqld)
 CGroup: /system.slice/mysqld.service
 └─7916 /usr/sbin/mysqld --daemonize --pid-file=/var/run/mysqld/mysqld.pid

Aug 16 19:33:35 ecs-adc3 systemd[1]: Starting MySQL Server...
Aug 16 19:33:40 ecs-adc3 systemd[1]: Started MySQL Server.
```
4. 执行以下命令，获取安装MySQL时自动设置的root用户密码。  
**grep 'temporary password' /var/log/mysqld.log**

回显如下类似信息。

```
2018-08-29T07:27:37.541944Z 1 [Note] A temporary password is generated for root@localhost: 2YY?3uHUA?Ys
```

5. 执行以下命令，并按照回显提示信息进行操作，加固MySQL。

### mysql\_secure\_installation

Securing the MySQL server deployment.

```
Enter password for user root: #输入上一步骤中获取的安装MySQL时自动设置的root用户密码
The existing password for the user account root has expired. Please set a new password.
```

```
New password: #设置新的root用户密码
```

```
Re-enter new password: #再次输入密码
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) : N #是否更改root用户密码，输入N
```

```
... skipping.
```

```
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!
```

### 步骤3 安装PHP。

1. 依次执行以下命令，安装PHP 7和一些所需的PHP扩展。

```
rpm -Uvh https://mirror.webtatic.com/yum/el7/epel-release.rpm
```

```
rpm -Uvh https://mirror.webtatic.com/yum/el7/webtatic-release.rpm
```

```
yum -y install php70w-tidy php70w-common php70w-devel php70w-pdo
php70w-mysql php70w-gd php70w-ldap php70w-mbstring php70w-
mcrypt php70w-fpm
```

2. 执行以下命令，验证PHP的安装版本。

```
php -v
```

回显如下类似信息：

```
PHP 7.0.31 (cli) (built: Jul 20 2018 08:55:22) (NTS)
Copyright (c) 1997-2017 The PHP Group
Zend Engine v3.0.0, Copyright (c) 1998-2017 Zend Technologies
```

3. 执行以下命令，启动PHP服务并设置开机自启动。

```
systemctl start php-fpm
```

```
systemctl enable php-fpm
```

4. 修改Nginx配置文件以支持PHP。

- a. 执行以下命令打开配置文件/etc/nginx/nginx.conf。

```
vim /etc/nginx/nginx.conf
```

图 15-2 nginx.conf



```
user nginx;
worker_processes auto;

error_log /var/log/nginx/error.log notice;
pid /var/run/nginx.pid;

events {
 worker_connections 1024;
}

http {
 include /etc/nginx/mime.types;
 default_type application/octet-stream;

 log_format main '$remote_addr - $remote_user [$time_local] "$request" '
 '$status $body_bytes_sent "$http_referer" '
 '"$http_user_agent" "$http_x_forwarded_for"';

 access_log /var/log/nginx/access.log main;

 sendfile on;
 #tcp_nopush on;

 keepalive_timeout 65;

 #gzip on;
 include /etc/nginx/conf.d/*.conf;
}
```

从nginx.conf可以看出，配置文件定向至/etc/nginx/conf.d/\*.conf。

- b. 输入:quit退出nginx.conf。
- c. 执行以下命令打开配置文件/etc/nginx/conf.d/default.conf。

```
vim /etc/nginx/conf.d/default.conf
```

- d. 按i键进入编辑模式。
- e. 修改打开的“default.conf”文件。

找到server段落，修改或添加下列配置信息。

```
server {
 listen 80;
 server_name localhost;

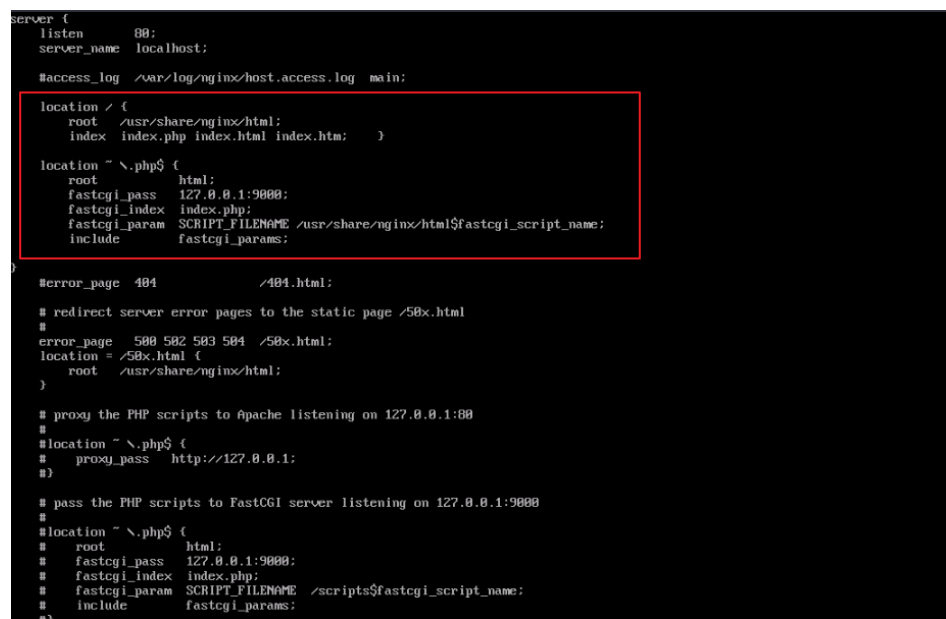
 #access_log /var/log/nginx/host.access.log main;
```

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

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

如图15-3所示:

图 15-3 修改后截图



```
server {
 listen 80;
 server_name localhost;

 #access_log /var/log/nginx/host.access.log main;

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

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

 #error_page 404 /404.html;

 # redirect server error pages to the static page /50x.html
 #
 error_page 500 502 503 504 /50x.html;
 location = /50x.html {
 root /usr/share/nginx/html;
 }

 # proxy the PHP scripts to apache listening on 127.0.0.1:80
 #
 #location ~ /\.php$ {
 # proxy_pass http://127.0.0.1;
 #}

 # pass the PHP scripts to FastCGI server listening on 127.0.0.1:9000
 #
 #location ~ /\.php$ {
 # root html;
 # fastcgi_pass 127.0.0.1:9000;
 # fastcgi_index index.php;
 # fastcgi_param SCRIPT_FILENAME /scripts$fastcgi_script_name;
 # include fastcgi_params;
 #}
}
```

f. 按Esc键退出编辑模式，并输入:wq保存后退出。

5. 执行以下命令，重新载入nginx的配置文件。

```
service nginx reload
```

#### 步骤4 浏览器访问测试。

1. 在/usr/share/nginx/html/目录下创建“info.php”的测试页面。

a. 执行以下命令创建并打开“info.php”的测试文件。

```
vim /usr/share/nginx/html/info.php
```

b. 按i键进入编辑模式。

c. 修改打开的“info.php”文件，将如下内容写入文件。

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

d. 按Esc键退出编辑模式，并输入:wq保存后退出。

2. 使用浏览器访问“http://服务器IP地址/info.php”，显示如下页面，说明环境搭建成功。



| PHP Version 7.0.31                      |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
|-----------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| System                                  | Linux ecs-5d3f.novalocal 3.10.0-693.11.1.el7.x86_64 #1 SMP Mon Dec 4 23:52:40 UTC 2017 x86_64                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
| Build Date                              | Jul 20 2018 08:57:28                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
| Server API                              | FPM/FastCGI                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
| Virtual Directory Support               | disabled                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
| Configuration File (php.ini) Path       | /etc                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
| Loaded Configuration File               | /etc/php.ini                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
| Scan this dir for additional .ini files | /etc/php.d                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
| Additional .ini files parsed            | /etc/php.d/bz2.ini, /etc/php.d/calendar.ini, /etc/php.d/ctype.ini, /etc/php.d/curi.ini, /etc/php.d/exif.ini, /etc/php.d/fileinfo.ini, /etc/php.d/ftp.ini, /etc/php.d/gd.ini, /etc/php.d/gettext.ini, /etc/php.d/gmp.ini, /etc/php.d/iconv.ini, /etc/php.d/json.ini, /etc/php.d/ldap.ini, /etc/php.d/mbstring.ini, /etc/php.d/mcrypt.ini, /etc/php.d/mysqli.ini, /etc/php.d/pdo.ini, /etc/php.d/pdo_mysql.ini, /etc/php.d/pdo_sqlite.ini, /etc/php.d/phar.ini, /etc/php.d/shmop.ini, /etc/php.d/simplexml.ini, /etc/php.d/sockets.ini, /etc/php.d/sqlite3.ini, /etc/php.d/tidy.ini, /etc/php.d/tokenizer.ini, /etc/php.d/xml.ini, /etc/php.d/zip.ini |
| PHP API                                 | 20151012                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
| PHP Extension                           | 20151012                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |

----结束

# 16 手工部署 Docker (CentOS 7.5)

## 简介

本文介绍了如何在华为云上使用弹性云服务器的Linux实例手工部署Docker，并提供了Docker常用操作及简单的镜像制作过程。

表 16-1 Docker 相关术语

| 术语     | 解释                                                                                                         |
|--------|------------------------------------------------------------------------------------------------------------|
| Docker | Docker是开发人员和系统管理员使用容器开发、部署和运行应用程序的平台。                                                                      |
| 镜像     | Docker镜像是一个特殊的文件系统，除了提供容器运行时所需的程序、库、资源、配置等文件外，还包含了一些为运行时准备的配置参数（如匿名卷、环境变量、用户等）。镜像不包含任何动态数据，其内容在构建之后也不会被改变。 |
| 容器     | 镜像（Image）和容器（Container）的关系，就像是面向对象程序设计中的类和实例一样，镜像是静态的定义，容器是镜像运行时的实体。容器可以被创建、启动、停止、删除、暂停等。                  |

更多关于Docker、镜像、容器的基础知识，请参见[Docker Documentation](#)。

该指导以“CentOS 7.5 64 3.10.0-862.9.1.el7.x86\_64”操作系统为例，Docker要求64位的系统且内核版本至少为3.10。

## 前提条件

- 弹性云服务器需要绑定弹性公网IP，具体步骤参见[弹性云服务器申请和绑定弹性公网IP](#)。
- 弹性云服务器所在安全组添加了如下表所示的安全组规则，具体步骤参见[添加安全组规则](#)。

表 16-2 安全组规则

| 方向  | 类型   | 协议  | 端口/范围 | 远端        |
|-----|------|-----|-------|-----------|
| 入方向 | IPv4 | TCP | 80    | 0.0.0.0/0 |

## 部署 Docker

1. 登录弹性云服务器。
2. 为了更好的获取和更新系统和软件，建议您更新镜像源为华为云镜像源，详细操作，请参见[如何使用华为云提供的epel镜像源\(x86\\_64和ARM\)?](#)。
3. 添加yum源。  
**yum install epel-release -y**  
**yum clean all**
4. 安装yum-util。  
**sudo yum install -y yum-utils device-mapper-persistent-data lvm2**
5. 设置docker yum源。  
**sudo yum-config-manager --add-repo https://download.docker.com/linux/centos/docker-ce.repo**
6. 安装并运行Docker。  
**sudo yum install docker-ce**  
**systemctl enable docker**  
**systemctl start docker**
7. 检查安装结果。  
**docker --version**  
回显如下类似信息，表示Docker安装成功。

```
Client: Docker Engine - Community
Version: 19.03.13
```

## Docker 基本用法

1. Docker进程管理。
  - 运行Docker  
**systemctl start docker**
  - 停止Docker  
**systemctl stop docker**
  - 重启Docker  
**systemctl restart docker**
2. 镜像管理。
  - a. 拉取镜像。此处以Apache和CentOS官方镜像为例。  
**docker pull httpd**  
**docker pull centos**
  - b. 查看已有镜像  
**docker images**

```
[root@ecs-b67a-docker ~]# docker images
REPOSITORY TAG IMAGE ID CREATED SIZE
docker.io/httpd latest 55a118e2a010 2 weeks ago 132 MB
docker.io/centos latest 75835a67d134 5 weeks ago 200 MB
[root@ecs-b67a-docker ~]#
```

c. 强制删除镜像

```
docker rmi centos
```

3. 容器管理。

a. 创建一个容器并运行。

```
docker run -it -d -p 80:80 --name datahttpd -v /data:/var/www/httpd/ httpd
```

参数说明如下：

- `-i`：以交互模式运行容器，通常与`-t`同时使用。
- `-t`：为容器重新分配一个伪输入终端，通常与`-i`同时使用。
- `-d`：后台运行容器，并返回容器ID。
- `-p`：端口映射，格式为“宿主机端口:容器端口”。
- `--name`：为容器指定一个名称。
- `-v`：把宿主机的一个目录挂载到镜像里，格式为“宿主机目录:镜像内挂载的路径”，必须为绝对路径。

#### 📖 说明

上述参数说明中的宿主机表示所创建的弹性云服务器实例。

示例命令表示：使用镜像`httpd`，以交互模式启动一个容器，将容器的80端口映射到宿主机的80端口，宿主机的目录`/data`映射到容器的`/var/www/httpd`目录，并返回容器ID。

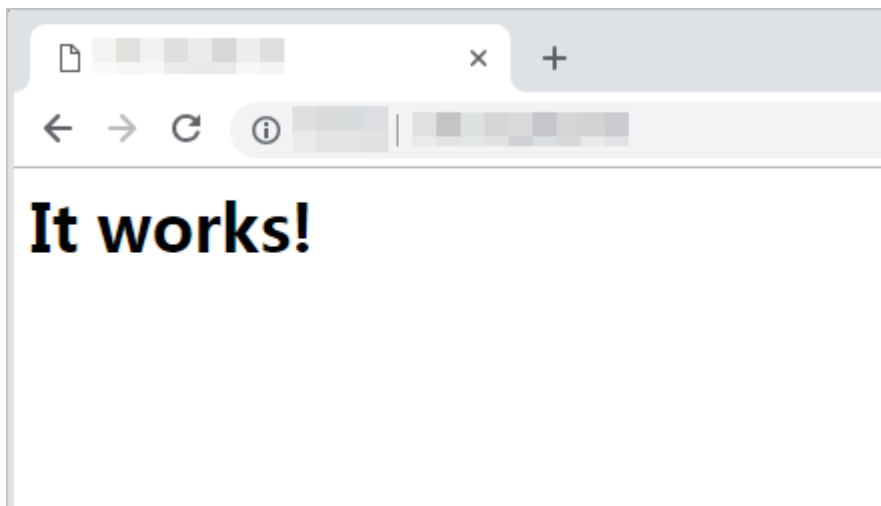
```
[root@ecs-b67a-docker ~]# docker run -it -d -p 80:80 --name datahttpd -v /data:/var/www/httpd/ httpd
6a514dea52a9465c1f6863c0f17ff41debda231ccff8bf66e3c0dbcc5f33cb20
[root@ecs-b67a-docker ~]#
```

b. 查看容器启动情况。

```
docker ps -a
```

```
[root@ecs-b67a-docker ~]# docker ps -a
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
6a514dea52a9 httpd "httpd-foreground" 4 minutes ago Up 4 minutes 0.0.0.0:80->80/tcp datahttpd
[root@ecs-b67a-docker ~]#
```

c. 在浏览器输入弹性云服务器所绑定的弹性公网IP，访问测试容器运行情况。出现以下内容表示运行成功。



## 镜像制作

通过Dockerfile定制一个简单的Nginx镜像。

1. 创建一个名为Dockerfile的文件。

```
mkdir mynginx
cd mynginx
touch Dockerfile
```

2. 编辑Dockerfile。

```
vim Dockerfile
```

增加文件内容如下：

```
FROM nginx
RUN echo '<h1>Hello, Docker!</h1>' > /usr/share/nginx/html/index.html
```

Dockerfile指令介绍如下。以下仅介绍简单指令，如需了解更多，请参见官方网站<https://docs.docker.com/>。

- FROM语句：表示使用nginx镜像作为基础镜像，一个Dockerfile中FROM是必备的指令，并且必须是第一条指令。
- RUN语句：格式为RUN <命令>，表示执行echo命令，在显示器中显示一段“Hello, Docker!”的文字。

3. 构建镜像。

```
docker build -t nginx:v3 .
```

- -t nginx:v3：指定镜像的名称和版本。
- .：指定上下文路径，镜像构建命令将该路径下所有的内容打包给Docker引擎帮助构建镜像。

4. 查看部署的nginx镜像，版本为v3。

```
docker images
```

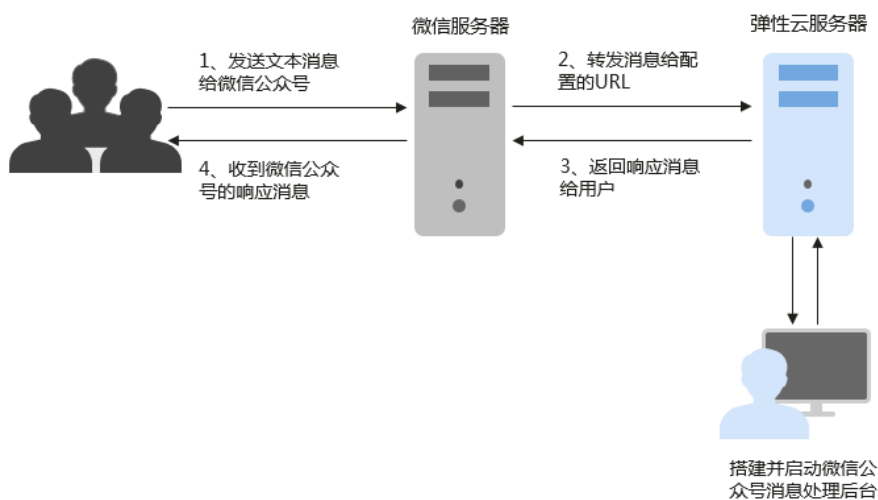
| REPOSITORY | TAG | IMAGE ID     | CREATED        | SIZE   |
|------------|-----|--------------|----------------|--------|
| nginx      | v3  | 09422e465d96 | 10 seconds ago | 109 MB |

# 17 搭建微信公众号后台——收发文本消息

## 概述

在本教程中，您将学会使用弹性云服务器（以下简称 ECS）搭建微信公众号处理后台，使用Python语言编写对应的微信消息处理逻辑代码，接收从微信服务端转发过来的消息，并返回处理结果给最终用户，如图17-1所示。

图 17-1 流程简介



您需要了解的背景知识有：CentOS（Linux）操作系统、Python语言、Web.py框架、HTTP/XML协议。

## 准备事项

- 申请微信公众号  
微信公众号申请链接：<https://mp.weixin.qq.com/>  
本教程中使用订阅号“云图说”为例说明。
- 购买华为弹性云服务。  
如果没有华为账号，需[注册华为账号并开通华为云](#)。  
本教程中，使用公共镜像CentOS 7.4

图 17-2 公共镜像



- 购买弹性公网IP  
建议同时购买弹性公网IP，后面需要在微信公众号上配置公网IP的地址。

图 17-3 弹性公网 IP



- 为了更好的获取和更新系统和软件，建议您更新镜像源为华为云镜像源，详细操作，请参见[如何使用自动化工具配置华为云镜像源\(x86\\_64和ARM\)?](#)。

## 安装基础软件

本教程中使用Python+Web.py组合完成微信公众号开发，需要安装或升级Python、pip、Web.py框架、WinSCP软件版本。

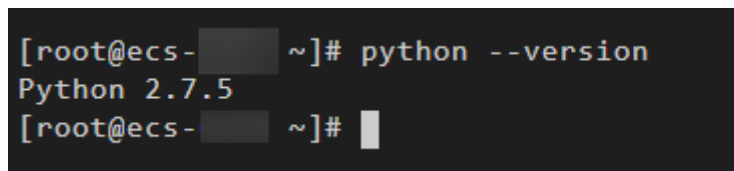
### 升级默认Python版本

CentOS 7.4自带Python版本比较老，建议升级到Python3。

1. 查看Python版本，使用如下命令：

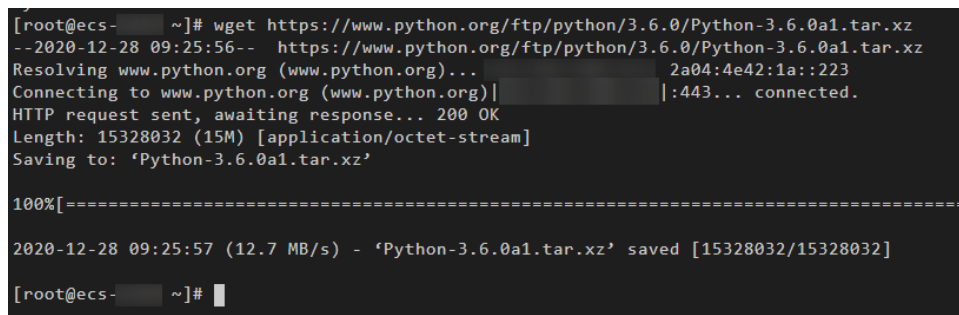
```
python --version
```

图 17-4 查询 Python 版本



2. 下载Python安装包，这里以Python 3.6.0版本为例，使用命令：  
`wget https://www.python.org/ftp/python/3.6.0/Python-3.6.0a1.tar.xz`

图 17-5 下载 Python 安装包



3. 解压安装包，使用如下命令：  
`tar xvf Python-3.6.0a1.tar.xz`

4. 执行命令：  
**cd Python-3.6.0a1**  
**./configure**
  - 执行成功如[图17-6](#)所示：

**图 17-6 执行成功**

```
configure: creating ./config.status
config.status: creating Makefile.pre
config.status: creating Modules/Setup.config
config.status: creating Misc/python.pc
config.status: creating Misc/python-config.sh
config.status: creating Modules/ld_so_aix
config.status: creating pyconfig.h
creating Modules/Setup
creating Modules/Setup.local
creating Makefile
```

- 如果出现“configure: error: no acceptable C compiler found in \$PATH”异常提示，是因为未安装合适的编译器。  
解决方法：  
执行以下命令，安装/升级gcc及其他依赖的包。  
**sudo yum install gcc-c++**  
并在随后提示安装包是否OK时，输入y并回车。出现如下[图8](#)提示说明依赖的包安装成功。

**图 17-7 安装编译器**

```
Installed:
gcc-c++.x86_64 0:4.8.5-44.el7

Dependency Installed:
libstdc++-devel.x86_64 0:4.8.5-44.el7

Dependency Updated:
cpp.x86_64 0:4.8.5-44.el7 gcc.x86_64 0:4.8.5-44.el7
```

在编译器安装完成后，重新执行**./configure** 命令。

5. 执行命令：  
**make && make install**  
执行成功。但提示pip错误，原因是我的系统中少了openssl-devel包，可以先忽略。



图 17-8 执行成功

```
rm -f /usr/local/bin/pyvenv
(cd /usr/local/bin; ln -s pyvenv-3.6 pyvenv)
if test "x" != "x" ; then \
 rm -f /usr/local/bin/python3-32; \
 (cd /usr/local/bin; ln -s python3.6-32 python3-32) \
fi
rm -f /usr/local/share/man/man1/python3.1
(cd /usr/local/share/man/man1; ln -s python3.6.1 python3.1)
if test "xupgrade" != "xno" ; then \
 case upgrade in \
 upgrade) ensurepip="--upgrade" ;; \
 install|*) ensurepip="" ;; \
 esac; \
 ./python -E -m ensurepip \
 $ensurepip --root=/ ; \
fi
Ignoring ensurepip failure: pip 8.1.1 requires SSL/TLS
```

- 查看Python3版本，使用命令：

```
python3 --version
```

图 17-9 查看 Python3 版本

```
[root@ecs- Python-3.6.0a1]# python3 --version
Python 3.6.0a1
```

- 执行命令：

```
python3
```

出现如下提示，则说明Python3安装成功。

图 17-10 Python3 安装成功

```
[root@ecs- Python-3.6.0a1]# python3
Python 3.6.0a1 (default, Dec 18 2020, 15:45:57)
[GCC 4.8.5 20150623 (Red Hat 4.8.5-44)] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

### 📖 说明

执行后续操作前需要先退出Python命令行，可通过输入以下任一命令再按回车键退出：

- Ctrl+Z
- exit()
- quit()

### 升级默认pip版本

pip是通用的Python包管理工具。提供了对Python包的查找、下载、安装、卸载功能。Python3安装成功后自带pip3，但版本比较老，建议升级到pip最新版本。同时前面安装python3提示“Ignoring ensurepip failure: pip 8.1.1 requires SSL/TLS”错误，导致pip未成功安装，所以需要重新安装pip。

- 安装openssl-devel包，使用命令：

```
yum install openssl-devel -y
```

图 17-11 安装 openssl-devel 包

```
Installed:
 openssl-devel.x86_64 1:1.0.2k-21.e17_9

Dependency Installed:
 keyutils-libs-devel.x86_64 0:1.5.8-3.e17 krb5-devel.x86_64 0:1.15.1-50.e17
 libselinux-devel.x86_64 0:2.5-15.e17 libsepol-devel.x86_64 0:2.5-10.e17
 zlib-devel.x86_64 0:1.2.7-18.e17

Dependency Updated:
 openssl.x86_64 1:1.0.2k-21.e17_9

Complete!
```

2. 执行命令：

```
make && make install
```

出现如下提示说明安装pip安装成功。

图 17-12 pip 安装成功

```
Collecting setuptools
Collecting pip
Installing collected packages: setuptools, pip
Successfully installed pip-8.1.1 setuptools-20.10.1
```

3. 升级pip3，使用命令：

```
pip3 install --upgrade pip
```

出现如下提示说明升级pip到最新版本了。

图 17-13 pip3 升级成功

```
[root@ecs- Python-3.6.0a1]# pip3 install --upgrade pip
Collecting pip
 Downloading https://files.pythonhosted.org/packages/54/eb/4a36
 100% |██| 1.5MB 32kB/s
Installing collected packages: pip
 Found existing installation: pip 8.1.1
 Uninstalling pip-8.1.1:
 Successfully uninstalled pip-8.1.1
Successfully installed pip-20.3.3
```

## 安装Web.py框架

Web.py官方教程地址：<http://webpy.org/>，使用如下命令安装web.py：

```
pip3 install web.py==0.40.dev0
```

图 17-14 安装 Web.py

```
[root@ecs-c438 Python-3.6.0a1]# pip3 install web.py==0.40.dev0
Collecting web.py==0.40.dev0
 Downloading web.py-0.40.dev0.tar.gz (116 kB)
 |██| 116 kB 76 kB/s
Using legacy 'setup.py install' for web.py, since package 'wheel'
Installing collected packages: web.py
 Running setup.py install for web.py ... done
Successfully installed web.py-0.40.dev0
```

## 安装WinSCP

通常情况下，在本地Windows操作系统上编辑代码，完成后再上传至ECS上（CentOS Linux系统）。WinSCP 是一个Windows环境下使用的SSH的开源图形化SFTP客户端，同时支持 SCP 协议。它的主要功能是在本地与远程计算机间安全地复制文件，并且可以直接编辑文件。

WinSCP安装链接：<https://winscp.net/eng/index.php>

## 上传代码

1. 新建main.py文件，复制如下代码：

```
-*- coding: utf-8 -*-
filename: main.py
import web
from handle import Handle

urls = (
 '/wx', 'Handle',
)

if __name__ == '__main__':
 app = web.application(urls, globals())
 app.run()
```

2. 新建handle.py文件，复制如下代码：

```
-*- coding: utf-8 -*-
filename: handle.py

import hashlib
import web
import receive
import time
import os

class Handle(object):

 def __init__(self):
 self.app_root = os.path.dirname(__file__)
 self.templates_root = os.path.join(self.app_root, 'templates')
 self.render = web.template.render(self.templates_root)

 def GET(self):
 try:
 data = web.input()
 if len(data) == 0:
 return "hello, this is handle view"
 signature = data.signature
 timestamp = data.timestamp
 nonce = data.nonce
 echostr = data.echostr
 token = "此处内容与公众号基本配置里Token字段取值保持一致"

 list = [token, timestamp, nonce]
 list.sort()
 s = list[0] + list[1] + list[2]
 hashcode = hashlib.sha1(s.encode('utf-8')).hexdigest()
 print("handle/GET func: hashcode, signature: ", hashcode, signature)
 if hashcode == signature:
 return echostr
 else:
 return echostr
 except (Exception) as Argument:
 return Argument

 def POST(self):
 try:
 webData = web.data()
 print("Handle Post webdata is:\n", webData)
 #打印消息体日志
```

```
recMsg = receive.parse_xml(webData)

if isinstance(recMsg, receive.Msg) and recMsg.MsgType == 'text':
 toUser = recMsg.FromUserName
 fromUser = recMsg.ToUserName
 content = "欢迎关注云图说!" + str(recMsg.Content)
 print('Reply message info:\n')
 print('toUser =', toUser)
 print('fromUser = ', fromUser)
 print('content = ', content)
 return self.render.reply_text(toUser, fromUser, int(time.time()), content)
else:
 print("不支持的消息类型: ",recMsg.MsgType)
 return "success"
except (Exception) as Argment:
 return Argment
```

### 3. 新建receive.py文件，复制如下代码：

```
-*- coding: utf-8 -*-
filename: receive.py
import xml.etree.ElementTree as ET

def parse_xml(web_data):
 if len(web_data) == 0:
 return None
 xmlData = ET.fromstring(web_data)
 msg_type = xmlData.find('MsgType').text
 if msg_type == 'text':
 return TextMsg(xmlData)
 elif msg_type == 'image':
 return ImageMsg(xmlData)
 elif msg_type == 'location':
 return LocationMsg(xmlData)
 elif msg_type == 'event':
 return EventMsg(xmlData)

class Event(object):
 def __init__(self, xmlData):
 self.ToUserName = xmlData.find('ToUserName').text
 self.FromUserName = xmlData.find('FromUserName').text
 self.CreateTime = xmlData.find('CreateTime').text
 self.MsgType = xmlData.find('MsgType').text
 self.Eventkey = xmlData.find('EventKey').text

class Msg(object):
 def __init__(self, xmlData):
 self.ToUserName = xmlData.find('ToUserName').text
 self.FromUserName = xmlData.find('FromUserName').text
 self.CreateTime = xmlData.find('CreateTime').text
 self.MsgType = xmlData.find('MsgType').text
 self.MsgId = xmlData.find('MsgId').text

class TextMsg(Msg):
 def __init__(self, xmlData):
 Msg.__init__(self, xmlData)
 self.Content = xmlData.find('Content').text

class ImageMsg(Msg):
 def __init__(self, xmlData):
 Msg.__init__(self, xmlData)
 self.PicUrl = xmlData.find('PicUrl').text
 self.MediaId = xmlData.find('MediaId').text

class LocationMsg(Msg):
 def __init__(self, xmlData):
 Msg.__init__(self, xmlData)
 self.Location_X = xmlData.find('Location_X').text
 self.Location_Y = xmlData.find('Location_Y').text

class EventMsg(Msg):
```

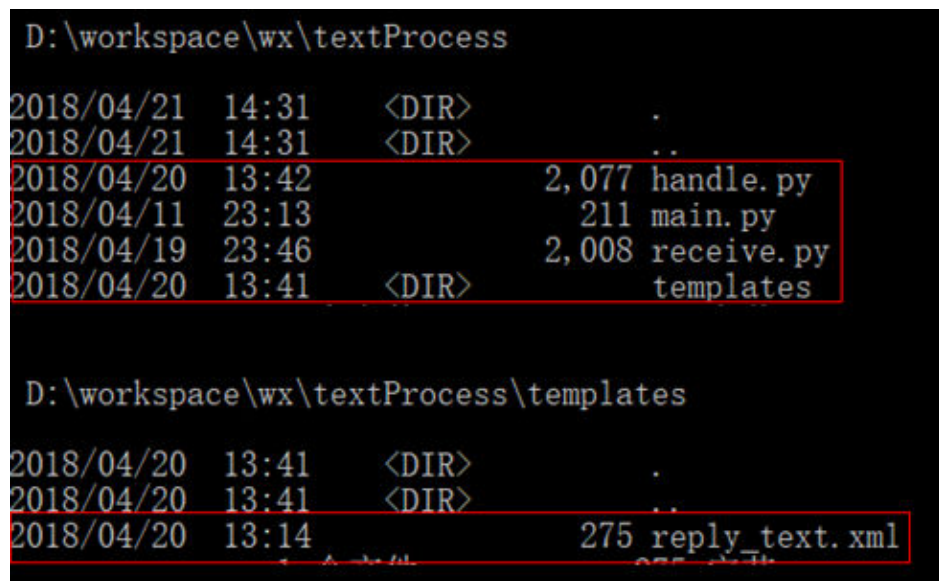
```
def __init__(self, xmlData):
 Event.__init__(self, xmlData)
 self.Event = xmlData.find('Event').text
```

4. 新建templates文件夹，在文件夹下新建reply\_text.xml文件，复制如下代码：

```
$def with (toUser,fromUser,createTime,content)
<xml>
<ToUserName><![CDATA[$toUser]]></ToUserName>
<FromUserName><![CDATA[$fromUser]]></FromUserName>
<CreateTime>$createTime</CreateTime>
<MsgType><![CDATA[text]]></MsgType>
<Content><![CDATA[$content]]></Content>
</xml>
```

5. 最终本地代码文件形成如下：

图 17-15 本地文件

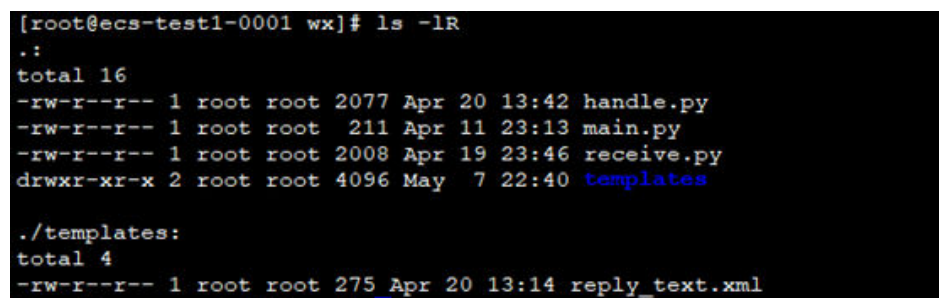


```
D:\workspace\wx\textProcess
2018/04/21 14:31 <DIR> .
2018/04/21 14:31 <DIR> ..
2018/04/20 13:42 2,077 handle.py
2018/04/11 23:13 211 main.py
2018/04/19 23:46 2,008 receive.py
2018/04/20 13:41 <DIR> templates

D:\workspace\wx\textProcess\templates
2018/04/20 13:41 <DIR> .
2018/04/20 13:41 <DIR> ..
2018/04/20 13:14 275 reply_text.xml
```

6. 通过WinSCP工具将上述文件与目录上传至ECS指定目录下：

图 17-16 上传文件



```
[root@ecs-test1-0001 wx]# ls -lR
.:
total 16
-rw-r--r-- 1 root root 2077 Apr 20 13:42 handle.py
-rw-r--r-- 1 root root 211 Apr 11 23:13 main.py
-rw-r--r-- 1 root root 2008 Apr 19 23:46 receive.py
drwxr-xr-x 2 root root 4096 May 7 22:40 templates

./templates:
total 4
-rw-r--r-- 1 root root 275 Apr 20 13:14 reply_text.xml
```

## 启动服务

使用如下命令启动服务：

```
python3 main.py 80
```

启动成功如图17-17所示：

图 17-17 启动成功

```
[root@ecs-test1-0001 wx]# python3 main.py 80
http://0.0.0.0:80/
```

## 启用开发者模式

1. 登录微信公众平台，选择“开发 > 基本配置”，单击“修改配置”。
2. 填写配置信息，单击“提交”。
  - URL: `https://ECS的弹性公网IP/wx`，不用添加80端口。
  - Token: 需要与handle.py中对应token取值完全一致。
  - EncodingAESKey: 随机生成。
  - 消息加解密方式: 此为示例，选择简单的“明文模式”。
3. 验证token成功，单击“启用”。

### 📖 说明

如果token验证失败，请检查Token配置与handle.py中GET消息处理代码是否一致。

## 验证

使用微信关注公众号，任意发送一条文本消息，看是否能够收到回复。如能收到回复则表明系统处理正常。

# 18 手工部署 GitLab (CentOS7.2)

## 简介

本文主要介绍了在华为云上如何使用弹性云服务器的Linux实例手工部署GitLab。GitLab是一个开源的版本管理系统，使用Git作为代码管理工具。该指导具体操作以CentOS 7.2 64位操作系统为例。

## 前提条件

- 部署GitLab的实例内存要求不小于4GB。
- 弹性云服务器所在安全组添加了如下表所示的安全组规则，具体步骤参见[安全组添加安全组规则](#)。

表 18-1 安全组规则

| 方向  | 协议/应用    | 端口/范围 | 源地址       |
|-----|----------|-------|-----------|
| 入方向 | HTTP(80) | 80    | 0.0.0.0/0 |

## 操作步骤

### 步骤1 安装相关依赖包。

1. 登录弹性云服务器。
2. 为了更好的获取和更新系统和软件，建议您更新镜像源为华为云镜像源，详细操作，请参见[如何使用自动化工具配置华为云镜像源\(x86\\_64和ARM\)?](#)。
3. 执行以下命令，安装相关依赖包。

```
sudo yum install -y curl policycoreutils-python openssh-server
```

4. 依次执行以下命令，设置SSH开机自启动并启动SSH服务。

```
sudo systemctl enable sshd
```

```
sudo systemctl start sshd
```

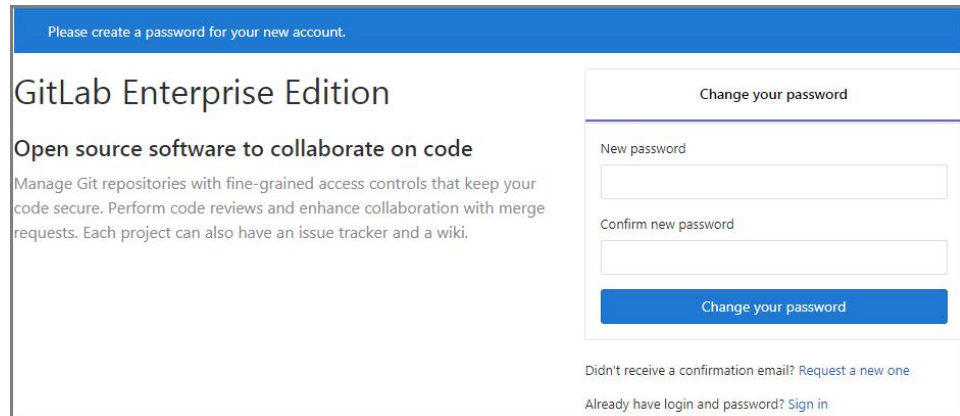
### 步骤2 安装Postfix来发送通知邮件。

1. 依次执行以下命令，安装Postfix。

```
sudo yum install postfix
```







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。

---结束

# 19 手工部署 RabbitMQ (CentOS 7.4)

## 简介

本文介绍了如何在华为云上使用弹性云服务器的Linux实例部署RabbitMQ。RabbitMQ是采用Erlang语言实现AMQP（Advanced Message Queuing Protocol，高级消息队列协议）的消息中间件，它最初起源于金融系统，用于在分布式系统中存储转发消息。RabbitMQ凭借其高可靠、易扩展、高可用及丰富的功能特性成为目前非常热门的一款消息中间件。

## 前提条件

弹性云服务器所在安全组添加了如下表所示的安全组规则，具体步骤参见[为安全组添加安全组规则](#)。

表 19-1 安全组规则

| 方向  | 类型   | 协议  | 端口/范围 | 源地址       |
|-----|------|-----|-------|-----------|
| 入方向 | IPv4 | TCP | 5672  | 0.0.0.0/0 |
| 入方向 | IPv4 | TCP | 15672 | 0.0.0.0/0 |

## 操作步骤

**步骤1** 安装相关依赖包和perl。

1. 登录弹性云服务器。
2. 为了更好的获取和更新系统和软件，建议您更新镜像源为华为云镜像源，详细操作，请参见[如何使用自动化工具配置华为云镜像源\(x86\\_64和ARM\)?](#)。
3. 执行以下命令，安装相关依赖包。

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

4. 执行如下命令，安装perl。

```
yum install perl
```

**步骤2** 安装erlang。

关于erlang的安装请参考[Erlang官方资料](#)。

1. 添加[erlang存储库](#)到系统

```
wget https://packages.erlang-solutions.com/erlang-solutions-2.0-1.noarch.rpm
```

```
rpm -Uvh erlang-solutions-2.0-1.noarch.rpm
```

或手动添加[存储库条目](#)

```
rpm --import https://packages.erlang-solutions.com/rpm/erlang_solutions.asc
```

2. 在/etc/yum.repos.d/目录下新建一个文件rabbitmq-erlang.repo，然后将下面的粘帖进去

```
cd /etc/yum.repos.d/
```

```
vi rabbitmq-erlang.repo
```

```
[erlang-solutions]
name=CentOS $releasever - $basearch - Erlang Solutions
baseurl=https://packages.erlang-solutions.com/rpm/centos/$releasever/$basearch
gpgcheck=1
gpgkey=https://packages.erlang-solutions.com/rpm/erlang_solutions.asc
enabled=1
```

按Esc键退出编辑模式，并输入:wq保存后退出。

3. 执行以下命令安装erlang

```
sudo yum install erlang
```

执行以下命令安装esl-erlang

```
sudo yum install esl-erlang
```

4. 执行如下命令，检查安装结果。

```
erl -version
```

回显类似如下信息，说明erlang安装成功。

```
[root@ecs-rabbitmq ~]# erl -version
Erlang (SMP,ASYNC_THREADS,HIPE) (BEAM) emulator version 11.1.7
```

### 步骤3 安装RabbitMQ

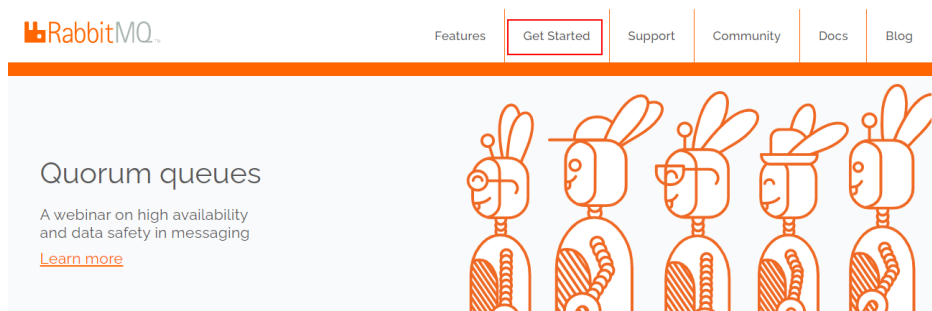
1. 执行如下命令，进入用户主目录。

```
cd
```

2. 执行如下命令，下载RabbitMQ安装包。

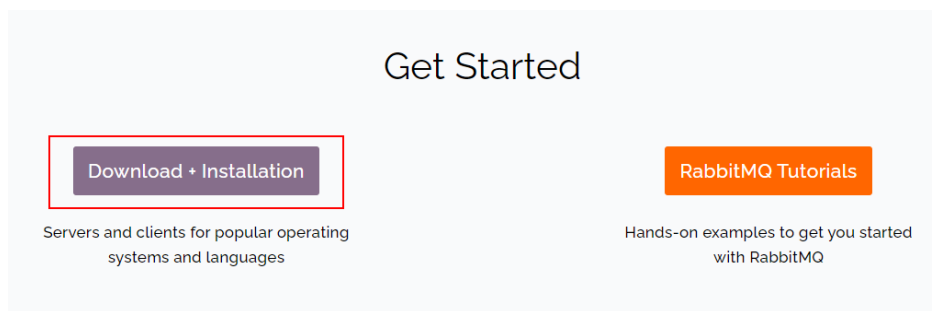
- a. 打开[Rabbit官网](#)。
- b. 单击“Get Started”。

图 19-1 Get Started



- c. 找到并单击“Download+Installation”。

图 19-2 Download+Installation



- d. 根据云服务器的操作系统选择下载地址。例如本例中使用的是CentOS 7.x的下载地址。

图 19-3 选择下载地址

Downloads [on GitHub](#)

- [Windows installer](#)
- [Debian, Ubuntu](#)
- [RHEL/CentOS 8.x](#) | [RHEL/CentOS 7.x](#) | [RHEL/CentOS 6.x](#) | [OpenSUSE](#) | [SLES 11.x](#) | [Erlang RPM](#)
- [Generic UNIX binary](#)
- [Windows binary](#)

- e. 在服务器上执行以下命令下载RabbitMQ安装包。

例如步骤3.2.d查找的下载地址是：

<https://github.com/rabbitmq/rabbitmq-server/releases/download/v3.8.12/rabbitmq-server-3.8.12-1.el7.noarch.rpm>

则执行的命令如下：

```
wget https://github.com/rabbitmq/rabbitmq-server/releases/download/v3.8.12/rabbitmq-server-3.8.12-1.el7.noarch.rpm
```

如果下载过程中提示“Unable to establish SSL connection.”

可以在wget命令后加--no-check-certificate，重复执行几次，即可下载。

例如：

```
wget https://github.com/rabbitmq/rabbitmq-server/releases/download/v3.8.12/rabbitmq-server-3.8.12-1.el7.noarch.rpm --no-check-certificate
```

- f. 执行以下命令安装RabbitMQ安装包。

```
yum install rabbitmq-server-3.8.12-1.el7.noarch.rpm
```

3. 安装完毕，启动RabbitMQ

```
service rabbitmq-server start
```

4. 查看RabbitMQ状态。

```
service rabbitmq-server status
```

步骤4 执行如下命令，启用RabbitMQ的web管理界面。

```
rabbitmq-plugins enable rabbitmq_management
```

回显类似如下信息:

```
[root@ecs-rabbitmq ~]# rabbitmq-plugins enable rabbitmq_management
Enabling plugins on node rabbit@ecs-rabbitmq:
rabbitmq_management
The following plugins have been configured:
 rabbitmq_management
 rabbitmq_management_agent
 rabbitmq_web_dispatch
Applying plugin configuration to rabbit@ecs-2b36...
The following plugins have been enabled:
 rabbitmq_management
 rabbitmq_management_agent
 rabbitmq_web_dispatch

started 3 plugins.
```

**步骤5** 执行如下命令，创建一个新用户。

```
rabbitmqctl add_user 用户名 密码
```

命令示例:

```
rabbitmqctl add_user root 123456
```

**步骤6** 执行如下命令，设置用户为管理员。

```
rabbitmqctl set_user_tags 用户名 administrator
```

命令示例:

```
rabbitmqctl set_user_tags root administrator
```

**步骤7** 执行如下命令，赋予用户所有权限。

```
rabbitmqctl set_permissions -p / 用户名 '.*' '.*' '.*'
```

命令示例:

```
rabbitmqctl set_permissions -p / root '.*' '.*' '.*'
```

**步骤8** 执行如下命令，在后台启动RabbitMQ。

```
rabbitmq-server -detached
```

**步骤9** 使用浏览器访问 “http://弹性公网IP:15672”，显示如下页面，说明RabbitMQ安装成功。



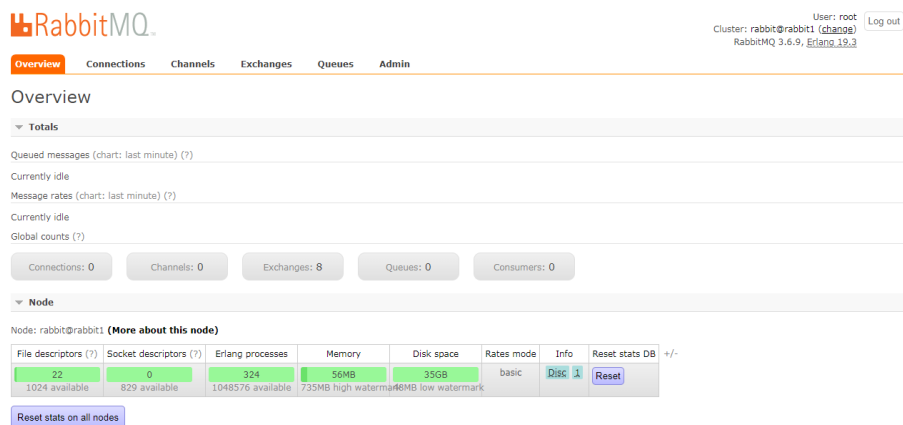
Login failed

Username:

Password:

Login

**步骤10** 输入**步骤5**创建的用户名和密码后单击“Login”，进入RabbitMQ管理界面。



----结束

# 20 手工搭建 Ghost 博客 ( Ubuntu 20.04 )

Ghost 是基于Node.js的开源博客平台，可以为用户提供更加便捷的写作与发布平台，本文指导用户基于华为云弹性云服务器（以Ubuntu 20.04操作系统云服务器为例）部署Ghost博客。

## 安装 gcc 和 g++

1. 执行以下命令，安装常用的开发编译工具包。

```
sudo apt-get install build-essential
```

2. 执行如下命令，安装gcc。

```
apt-get install gcc
```

3. 执行以下命令查看gcc版本。

```
gcc --version
```

回显信息：

```
root@ecs-c47c:~# gcc --version
gcc (Ubuntu 9.3.0-17ubuntu1~20.04) 9.3.0
Copyright (C) 2019 Free Software Foundation, Inc.
```

4. 执行以下命令安装g++。

```
sudo apt-get install g++
```

5. 执行以下命令查看g++版本。

```
g++ --version
```

回显信息：

```
root@ecs-c47c:~# g++ --version
g++ (Ubuntu 9.3.0-17ubuntu1~20.04) 9.3.0
Copyright (C) 2019 Free Software Foundation, Inc.
```

## 安装 Node.js

1. 执行以下命令，安装Node.js

```
sudo curl -sL https://deb.nodesource.com/setup_18.x | sudo -E bash -
sudo apt-get install -y nodejs
```

2. 安装完成后，执行以下命令查看Node.js版本和npm版本。

```
node -v
```

```
npm -v
```

回显信息:

```
root@ecs-c47c:~# node -v
v18.16.1
root@ecs-c47c:~# npm -v
9.5.1
```

## 安装 Nginx

部署Ghost博客，需要安装Nginx作为HTTP服务器，本节以安装Nginx 1.10.0 版本为例。

1. 输入以下命令安装Nginx。

```
sudo apt-get update
```

```
sudo apt-get install nginx
```

2. 调整防火墙（可选）。

UFW（Uncomplicated Firewall）是一个iptables的接口，可以简化配置防火墙的过程。Ubuntu默认安装了UFW，执行以下命令查看防火墙的状态。

```
sudo ufw status
```

如果你没有也不想开启防火墙，则可以直接跳过此步骤，如果你想要开启防火墙可以通过以下命令实现。

```
sudo ufw enable
```

之后再次检查防火墙状态验证是否成功开启防火墙。

在测试Nginx之前，需要重新配置防火墙软件以允许访问Nginx。执行以下命令，将Nginx自动注册在UFW。

```
sudo ufw app list
```

回显信息:

```
Available applications:
Nginx Full
Nginx HTTP
Nginx HTTPS
...
```

- Nginx Full: 此配置文件打开端口 80（正常，未加密的Web流量）和端口 443（TLS / SSL加密流量）
- Nginx HTTP: 此配置文件仅打开端口 80（正常，未加密的Web流量）
- Nginx HTTPS: 此配置文件仅打开端口 443（TLS / SSL加密流量）

执行以下命令确保防火墙允许HTTP和HTTPS连接。

```
sudo ufw allow 'Nginx Full'
```

3. 验证Nginx是否正常工作。

在浏览器中通过域名或者IP地址进行访问Nginx，如果Nginx正常启动则会打开Welcome to nginx的欢迎页面。

使用浏览器访问“<http://云服务器IP地址>”，显示如下页面，说明Nginx安装成功。



## Welcome to nginx!

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to [nginx.org](http://nginx.org).  
Commercial support is available at [nginx.com](http://nginx.com).

*Thank you for using nginx.*

#### 4. 配置Nginx。

##### a. 新建配置文件。

```
vim /etc/nginx/sites-available/ghost.conf
```

##### b. 把以下配置内容粘贴进你的配置文件中。

```
server {
 listen 80;
 server_name 119.3.xx.xxx.com; #这里写你的域名或者ip地址
 location / {
 proxy_set_header X-Real-IP $remote_addr;
 proxy_set_header Host $http_host;
 proxy_pass http://127.0.0.1:2368;
 }
}
```

以上配置文件已经把反向代理写好了, 唯一需要修改的地方是: 把 server\_name改成你自己的顶级域名。

##### c. 把配置文件软链接到sites-enabled中。

```
sudo ln -s /etc/nginx/sites-available/ghost.conf /etc/nginx/sites-enabled/ghost.conf
```

##### d. 重启Nginx。

```
sudo service nginx restart
```

## 创建新用户

由于Ghost官方不推荐使用root用户直接操作, 因此需要重新创新的用户, 并为其配置权限。

#### 1. 执行以下命令, 创建新用户。

```
adduser <user>
```

回显信息如下:

```
Adding user `user' ...
Adding new group `user' (1000) ...
Adding new user `user' (1000) with group `user' ...
Creating home directory `/home/user' ...
Copying files from `/etc/skel' ...
New password:
Retype new password:
passwd: password updated successfully
Changing the user information for user
Enter the new value, or press ENTER for the default
 Full Name []:
 Room Number []:
 Work Phone []:
 Home Phone []:
 Other []:
Is the information correct? [Y/n] Y
```

2. 执行以下命令，将新创建的用户添加到组。  
**usermod -aG sudo <user>**
3. 执行以下命令，切换到<user>用户。  
**su - <user>**

## 安装 MySQL

MySQL是一种开源数据库管理系统，通常作为流行的LAMP（Linux，Apache，MySQL，PHP / Python / Perl）堆栈的一部分安装。它使用关系数据库和SQL结构化查询语言）来管理其数据。

1. 安装MySQL。
  - 执行以下命令更新软件包。  
**sudo apt-get update**
  - 执行以下命令安装mysql-server 软件包。（安装过程中会要求配置MySQL的root用户密码）  
**sudo apt-get install mysql-server**

2. 配置MySQL。

执行以下命令，并按照回显提示信息进行操作，加固MySQL。

### mysql\_secure\_installation

Securing the MySQL server deployment.

Enter password for user root: #输入上一步骤中获取的安装MySQL时自动设置的root用户密码  
The existing password for the user account root has expired. Please set a new password.

New password: #设置新的root用户密码

Re-enter new password: #再次输入密码  
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) : N #是否更改root用户密码，输入N

... skipping.

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!
```

### 3. 测试MySQL。

输入以下命令来查看MySQL的状态：

```
systemctl status mysql.service
```

结果正常会提示：

```
• mysql.service - MySQL Community Server
 Loaded: loaded (/lib/systemd/system/mysql.service; enabled; vendor preset: enabled)
 Active: active (running) since Mon 2019-01-07 10:57:27 CST; 2min 42s ago
 Main PID: 26065 (mysqld)
 CGroup: /system.slice/mysql.service
 └─26065 /usr/sbin/mysqld
```

### 4. 为了避免数据库存放的中文是乱码，执行以下命令设置MySQL的编码。

```
sudo vi /etc/my.cnf
```

复制粘贴以下内容：

```
[client]
default-character-set=utf8
[mysql]
default-character-set=utf8
[mysqld]
character-set-server=utf8
collation-server=utf8_general_ci
```

保存退出，执行以下命令重启MySQL生效：

```
sudo /usr/sbin/service mysql restart
```

### 5. 建立Ghost数据库。

以 root 身份登录MySQL然后创建一个名为ghost的数据库并验证创建结果：

```
mysql -u root -p;
```

```
mysql> create database ghost;
```

```
mysql> show databases;
```

```
mysql> exit
```

## Ghost 安装与配置

Ghost v1.0.0 及以上版本已加入了Ghost-CLI，因此可以直接安装配置Ghost-CLI。

#### 1. 安装Ghost-CLI。

```
sudo npm i -g ghost-cli
```

#### 2. 创建一个文件夹，之前介绍过Nginx的重要文件结构 /var/www/ 就是将要创建文件夹的地方。

```
sudo mkdir -p /var/www/ghost
```

#### 说明

ghost在/root文件夹中安装Ghost 将无法正常工作。

3. 配置权限。

```
sudo chown [user]:[user] /var/www/ghost
```

 说明

[user] 是在安装阶段创建的用户。

4. 进入刚才创建的文件夹。

```
cd /var/www/ghost/
```

5. 执行以下命令，用Ghost-CLI 安装Ghost。

```
ghost install
```

 说明

如果Ghost安装时提示node版本不匹配，可以在nodejs官方网站查询相应的版本重新安装。

<https://nodejs.org/en/download/>

6. 配置Ghost。

如果上面在 /var/www/ghost/ 目录下运行**ghost install**成功的话，会要求配置一些东西：

```
$ ghost install
✓ Checking for latest Ghost version
✓ Running system checks
✓ Setting up install directory
✓ Downloading and installing Ghost v1.0.0-alpha.21
✓ Moving files
? Enter your blog URL: http://example.com
? Enter your MySQL hostname: localhost
? Enter your MySQL username: ghost
? Enter your MySQL password: *****
? Enter your Ghost database name: ghost_production
```

您可以根据需要进行简单的配置。如果配置后需要修改可以通过以下命令去文件中进行配置：

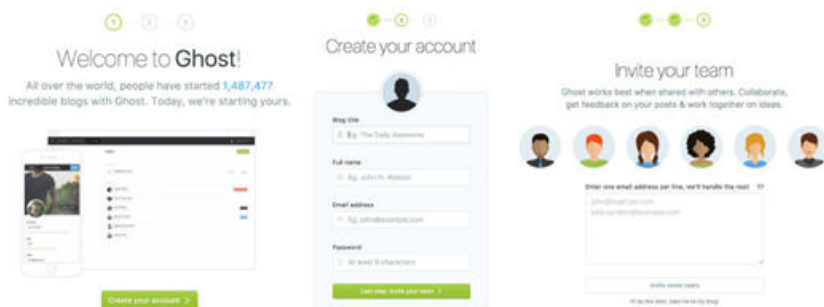
```
vi config.production.json
```

打开之后就是生产环境配置内容了。参考配置见下图：

```
{
 "url": "http://127.0.0.1:2368",
 "server": {
 "port": 2368,
 "host": "127.0.0.1"
 },
 "database": {
 "client": "mysql",
 "connection": {
 "host": "127.0.0.1",
 "user": "root",
 "password": "root",
 "database": "ghost"
 }
 },
 "mail": {
 "transport": "Direct"
 },
 "logging": {
 "transports": [
 "file",
 "stdout"
]
 },
 "process": "systemd",
 "paths": {
 "contentPath": "/var/www/ghost/content"
 },
 "bootstrap-socket": {
```

## 验证 Ghost 搭建完成

如果成功的安装Ghost之后，就可以通过域名访问到Ghost博客。



# 21 手工部署 Node.js (CentOS 7.2)

## 简介

本文介绍了如何在华为云上使用弹性云服务器的Linux实例部署Node.js。

Node.js是一个基于Chrome V8引擎的JavaScript运行环境，用于方便地搭建响应速度快、易于扩展的网络应用。Node.js使用事件驱动，非阻塞I/O模型而得以轻量和高效率，非常适合在分布式设备上运行数据密集型的实时应用。

更多关于Node.js信息，请参见<https://nodejs.org>。

该操作以“CentOS 7.2 64bit(40G)”操作系统为例，使用的Node.js软件包版本为“node-v10.14.1-linux-x64.tar”及“node-v10.14.2-linux-x64.tar”。

## 前提条件

- 已创建弹性云服务器，创建过程请参考[购买弹性云服务器](#)。
- 弹性云服务器需要绑定弹性公网IP，具体步骤参见[为弹性云服务器申请和绑定弹性公网IP](#)。
- 为了更好的获取和更新系统和软件，建议您更新镜像源为华为云镜像源，详细操作，请参见[如何使用自动化工具配置华为云镜像源\(x86\\_64和ARM\)?](#)。
- 本地已经安装用于连接 Linux 弹性云服务器的工具（如 [PuTTY](#)）。

## 操作步骤

### 步骤1 安装Node.js安装包。

- 使用二进制文件安装。
  - a. 登录弹性云服务器。
  - b. 执行以下命令，下载[Node.js安装包](#)。  
**wget https://nodejs.org/dist/v10.14.1/node-v10.14.1-linux-x64.tar.xz**
  - c. 执行以下命令，解压文件。  
**tar xvJf node-v10.14.1-linux-x64.tar.xz**
  - d. 依次执行以下命令，为node及npm建立软连接，可在任意目录下执行node及npm命令。  
**ln -s /root/node-v10.14.1-linux-x64/bin/node /usr/local/bin/node**  
**ln -s /root/node-v10.14.1-linux-x64/bin/npm /usr/local/bin/npm**

- e. 依次执行以下命令，查看node及npm版本。

```
node -v
```

```
npm -v
```

- 使用NVM版本管理器安装多版本。

- a. 登录弹性云服务器。

- b. 执行以下命令，安装git。

```
yum install git
```

- c. 执行以下命令，使用git将源码克隆到本地的~/.nvm目录下，并检查最新版本。

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

- d. 执行以下命令，激活NVM，并将其追加至profile文件下。

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

- e. 执行如下命令，使环境变量生效。

```
source /etc/profile
```

- f. 执行以下命令，列出Node.js可用版本。

```
nvm ls-remote
```

- g. 执行以下命令，安装多个Node.js版本。

```
nvm install v10.14.1
```

```
nvm install v10.14.2
```

- h. 执行以下命令，查看已安装的Node.js版本。

```
nvm ls
```

- i. 执行以下命令切换Node.js版本至V10.14.2。

```
nvm use v10.14.2
```

#### 说明

- 执行 `nvm alias default v10.14.2`，设置默认版本为10.14.2。
- 执行 `nvm help`，获取更多NVM操作信息。

## 步骤2 部署测试项目。

1. 执行以下命令，进入用户主目录。

```
cd
```

2. 执行以下命令，新建项目文件test.js。

```
touch test.js
```

3. 使用vim编辑器编辑项目文件test.js。

- a. 执行以下命令，安装Vim编辑器。

```
yum install vim
```

- b. 执行以下命令，打开项目文件test.js。

```
vim test.js
```

- c. 按i键进入编辑模式。

输入以下项目文件内容，修改打开的“test.js”文件。

```
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}/`);
});
```

端口号port可自定义。

- d. 按Esc键退出编辑模式，并输入:wq保存后退出。
4. 执行以下命令，查看已开启端口。  
**netstat -lntp**  
若定义端口不存在，需登录弹性云服务器控制台，更改安全组规则，具体步骤参见[添加安全组规则](#)。
5. 添加防火墙例外端口。
  - a. 以本例使用的3000端口为例，执行以下命令，添加防火墙例外端口3000。  
**firewall-cmd --zone=public --add-port=3000/tcp --permanent**  
如果回显信息如下说明防火墙关闭，可直接执行[步骤2.6](#)  

```
[root@ecs-centos7 ~]# firewall-cmd --zone=public --add-port=3000/tcp --permanent
FirewallD is not running
```

  
如果回显信息如下说明防火墙开启且添加例外端口成功。  

```
[root@ecs-centos7 ~]# firewall-cmd --zone=public --add-port=3000/tcp --permanent
success
```
  - b. 重新加载策略配置，使新配置生效。  
**firewall-cmd --reload**
  - c. 可以执行以下命令查看开启的所有端口。  
**firewall-cmd --list-ports**  

```
[root@ecs-centos7 ~]# firewall-cmd --list-ports
3000/tcp
```
6. 执行以下命令，运行项目。  
**node ~/test.js**
7. 使用浏览器访问“http://弹性公网IP:3000”显示如下页面，说明Node.js测试项目部署成功。

图 21-1 部署测试



----结束



# 22 PostgreSQL 本地 Slave 搭建步骤

## PostgreSQL 是什么？

PostgreSQL是一个开源对象关系型数据库管理系统，并侧重于可扩展性和标准的符合性。PostgreSQL面向企业复杂SQL处理的OLTP在线事务处理场景，支持NoSQL数据类型（JSON/XML/hstore），支持GIS地理信息处理，在可靠性、数据完整性方面有良好声誉，适用于互联网网站、位置应用系统、复杂数据对象处理等应用场景。

本文指导您使用华为云弹性云服务器搭建PostgreSQL。

## 搭建前准备

- 创建弹性云服务器。
- 配置弹性云服务器安全组规则：开放 5432 端口。

### 📖 说明

本文使用的云服务器操作系统：CentOS 7.6 64bit

本文使用的PostgreSQL版本：PostgreSQL (11.2)

- 为了更好的获取和更新系统和软件，建议您更新镜像源为华为云镜像源，详细操作，请参见[如何使用自动化工具配置华为云镜像源\(x86\\_64和ARM\)?](#)。

## 主节点配置

1. 依次执行以下命令，安装PostgreSQL。

```
yum update -y
yum install https://download.postgresql.org/pub/repos/yum/repopms/EL-7-x86_64/pgdg-redhat-repo-latest.noarch.rpm
yum install postgresql11-server
yum install postgresql11
/usr/pgsql-11/bin/postgresql-11-setup initdb
systemctl enable postgresql-11
systemctl start postgresql-11
```
2. 执行以下命令，切换到postgres用户（安装好生成默认的用户）。

```
su - postgres
```
3. 执行以下命令，进入数据库。

### # psql

4. 执行以下命令，创建账号并授权。

```
create role 账户名 login replication encrypted password '密码';
```

### 📖 说明

上述命令中密码必须用单引号。

以创建的账号名为dbar，密码为xxxxx为例，命令如下：

```
create role dbar login replication encrypted password 'xxxxx';
```

5. 执行以下命令，修改/var/lib/pgsql/11/data/pg\_hba.conf配置文件，添加如下内容。

### # vim /var/lib/pgsql/11/data/pg\_hba.conf

```
host all all 192.168.1.0/24 md5 #允许VPC网段中md5密码认证连接
host replication dbar 备库实际IP/24 md5 #允许用户从replication数据库进行数据同步
```

6. 执行以下命令，修改/var/lib/pgsql/11/data/postgresql.conf。

### # vim postgresql.conf

```
wal_level = hot_standby #热备模式
max_wal_senders= 6 #可以设置最多几个流复制链接
wal_sender_timeout = 60s #流复制主机发送数据的超时时间
max_connections = 512 #从库的 max_connections要大于主库
archive_command=' cp %p /var/lib/pgsql/11/data/archivelog/%f' #归档
wal_keep_segments=10240
archive_mode = on #允许归档
listen_addresses= xxx.xx.xx.xx #实际的本地ip地址
```

7. 执行以下命令，重启服务。

```
systemctl restart postgresql-11
```

## 从节点配置

1. 依次执行以下命令，完成从节点安装。

```
yum update -y
```

```
yum install https://download.postgresql.org/pub/repos/yum/reporepms/
EL-7-x86_64/pgdg-redhat-repo-latest.noarch.rpm
```

```
yum install postgresql11-server
```

```
yum install postgresql11
```

2. 执行以下命令，复制master配置相关文件。

```
pg_basebackup -h 主节点ip -U dbar -D /var/lib/pgsql/11/data -X stream
-P
```

```
cp /usr/pgsql-11/share/recovery.conf.sample /var/lib/pgsql/11/data/
recovery.conf
```

```
[root@ecs-22f5-0002 ~]# pg_basebackup -D /var/lib/pgsql/11/data -h 192.168.1.10 -p 5432 -U test -X s
tream -P
Password:
24508/24508 kB (100%), 1/1 tablespace
[root@ecs-22f5-0002 ~]#
```

3. 执行以下命令，修改recovery.conf文件。

### # vim recovery.conf

```
standby_mode = on # 声明此节点为从库
primary_conninfo = 'host=主节点IP port=5432 user=dbar password=xxxxx (此处密码不能加单引号)
trigger_file = '/var/lib/pgsql/11/data/trigger.kenyon' #主从切换时的触发文件
recovery_target_timeline = 'latest'
restore_command = 'cp /var/lib/pgsql/11/data/archivelog/%f %p'
archive_cleanup_command = 'pg_archivecleanup /var/lib/pgsql/11/data/archivelog %r' #清除过期归档
```

4. 执行以下命令，修改postgresql.conf文件。  
**# chown -R postgres.postgres /var/lib/pgsql/11/data**
5. 修改/var/lib/pgsql/11/data/postgresql.conf。  
**listen\_addresses= xxx.xx.xx.xx #实际的本地ip地址**  
**max\_connections = 600**
6. 执行以下命令，启动服务，设置开机自启。  
**#systemctl enable postgresql-11**  
**#systemctl start postgresql-11**

## 验证部署完成

1. 执行以下命令，在主节点中可查看到sender进程。  
**# ps aux |grep sender**

```
[root@ecs-22f5-0001 ~]# ps aux |grep sender
postgres 14406 0.0 0.3 397240 3620 ? Ss 20:19 0:00 postgres: walsender test (53052) streaming 0/3000140
```

2. 执行以下命令，从节点中可查看到receiver进程。  
**# ps aux | grep receiver**

```
[root@ecs-22f5-0002 ~]# ps aux |grep receiver
postgres 4390 0.0 0.3 403500 3632 ? Ss 20:19 0:00 postgres: walreceiver streaming 0/3000140
```

3. 执行以下命令，主库中可查看到从库状态。  
**# su - postgres**  
**-bash-4.2# psql**  
**replication=# select \* from pg\_stat\_replication;**

```
postgres=# select * from pg_stat_replication;
 pid | usesysid | username | application_name | client_addr | client_hostname | client_port | backend_start | backend_xmin | state | se
-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----
 14406 | 16384 | test | walreceiver | | | | 53052 | 2019-03-26 20:19:18.693053+08 | streaming | 0/
3000140 | 0/3000140 | 0/3000140 | 0/3000140 | | | | 0 | async
(1 row)
```

4. 在主库建一个库，验证备库是否可以同步。
  - a. 在主库执行以下命令  
**postgres=# create database testdb;**  
**postgres=# \l**
  - b. 在备库执行以下命令查看。  
**postgres=# \l**

# 23 手工安装宝塔面板（CentOS7.2）

## 应用场景

本文主要介绍了在华为云上如何在弹性云服务器的Linux实例上手工安装宝塔面板。宝塔面板是一款使用方便、功能强大且终身免费的服务器管理软件，支持Linux与Windows系统。一键配置LAMP、LNMP、网站、数据库、FTP、SSL，通过Web端轻松管理服务器。该指导具体操作以CentOS 7.2 64位操作系统安装宝塔Linux面板6.9为例。

## 方案优势

- 快速创建管理项目
- 预览服务器资源使用情况
- 轻松安装软件及部署源码

## 约束与限制

- 弹性云服务器实例规格和操作系统：
  - 内存要求最低512MB，推荐768MB以上，纯面板约占系统60MB内存。
  - 100M以上可用硬盘空间（纯面板约占20M磁盘空间）。
  - 宝塔Linux6.0版本是基于Centos7开发的，强烈建议使用Centos 7.x 系统。
  - 需要是全新的干净系统，没有安装过Apache、Nginx、php、MySQL。
- 弹性云服务器所在安全组添加了如下表所示的安全组规则，具体步骤参见[为安全组添加安全组规则](#)。

表 23-1 安全组规则

| 方向  | 协议/应用 | 端口/范围 | 源地址       |
|-----|-------|-------|-----------|
| 入方向 | TCP   | 8888  | 0.0.0.0/0 |

## 安装宝塔面板操作流程

Linux实例手工安装宝塔面板的具体操作步骤如下：

1. [安装宝塔面板。](#)
2. [登录宝塔面板。](#)

## 实施步骤

### 步骤1 安装宝塔面板。

1. 登录弹性云服务器。
2. 为了更好的获取和更新系统和软件，建议您更新镜像源为华为云镜像源，详细操作，请参见[如何使用自动化工具配置华为云镜像源\(x86\\_64和ARM\)?](#)。
3. 执行以下命令，下载并安装宝塔面板。

```
yum install -y wget && wget -O install.sh http://download.bt.cn/install/install_6.0.sh && sh install.sh
```

当出现以下提示信息时，输入“y”：

```
...
Do you want to install Bt-Panel to the /www directory now?(y/n): y
...
```

安装完成后，回显如下类似信息：

```
...
=====
Congratulations! Installed successfully!
=====
Bt-Panel: http://114.115.xxx.xx:8888/46722528
username: ut22gsvp
password: *****
Warning:
If you cannot access the panel,
release the following port (8888|888|80|443|20|21) in the security group
=====
Time consumed: 2 Minute!
```

#### 说明

请记录回显信息中“Bt-Panel”、“username”和“password”的参数值。

### 步骤2 登录宝塔面板。

1. 在浏览器地址栏中输入记录的“Bt-Panel”参数后面的地址，例如：<http://114.115.xxx.xx:8888/46722528>。
2. 输入记录的账户和密码。
3. 根据业务需要可以通过面板安装相关的套件并部署网站。

----结束

# 24 在 ECS 上通过内网访问 OBS

## 24.1 概述

### 场景介绍

某企业基于弹性云服务器（Elastic Cloud Server，ECS）构建好基础的业务后，随着数据增长，硬盘已无法满足大量的图片、视频等数据存取需求。了解到华为云提供有海量、弹性的云存储服务OBS后，决定将OBS作为数据存储资源池，以减轻服务器负担。

在ECS上可以通过公网和华为云内网两种网络访问OBS。当有存取对象数据的需求时，公网方式响应速度会因为网络质量而受到影响，读取数据还将收取一定的流量费用。为最大化的优化性能、节省开支，企业管理者希望通过内网的方式访问OBS。

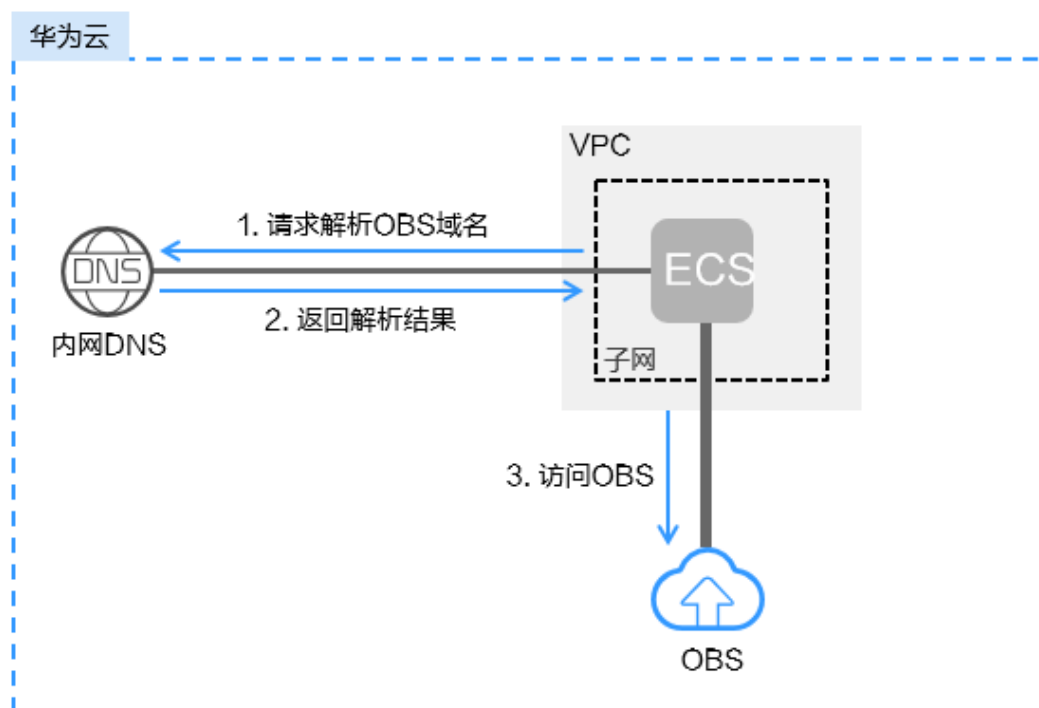
#### 📖 说明

当通过内网访问OBS时，需要确保待访问的OBS资源与ECS属于同一个区域，如果不属于同一个区域，将采用公网访问。

### 方案

在已搭建的ECS上通过配置内网DNS，由内网DNS解析OBS域名，即可实现在ECS上经由内网访问OBS。访问过程示意图如[图24-1](#)所示。

图 24-1 内网访问 OBS 示意图



示意图中各服务说明如表24-1所示。

表 24-1 服务说明

| 服务          | 说明                                                                                                               |
|-------------|------------------------------------------------------------------------------------------------------------------|
| 虚拟私有云 (VPC) | VPC主要负责为ECS构建隔离的、用户自主配置和管理的虚拟网络环境，提升用户云中资源的安全性，简化用户的网络部署。<br>子网是VPC中用来为ECS提供IP地址管理、DNS服务的一个网络，子网内ECS的IP地址都属于该子网。 |
| 云解析服务 (DNS) | DNS提供内网DNS，专门用于处理华为云内网域名以及OBS域名的解析请求，简化域名解析流程，减少因访问公网产生的流量费用。                                                    |

- 对于Windows ECS，推荐使用OBS Browser+工具，实现内网访问OBS的目的，详细操作请参见：  
[在Windows ECS上使用OBS Browser+通过内网访问OBS](#)
- 对于Linux ECS，推荐使用obsutil工具，实现内网访问OBS的目的，详细操作请参见：  
[在Linux ECS上使用obsutil通过内网访问OBS](#)

当在ECS上通过内网访问OBS时，即可在内网进行数据读取、备份归档等业务，而不影响外网带宽。

## 24.2 在 Windows ECS 上使用 OBS Browser+通过内网访问 OBS

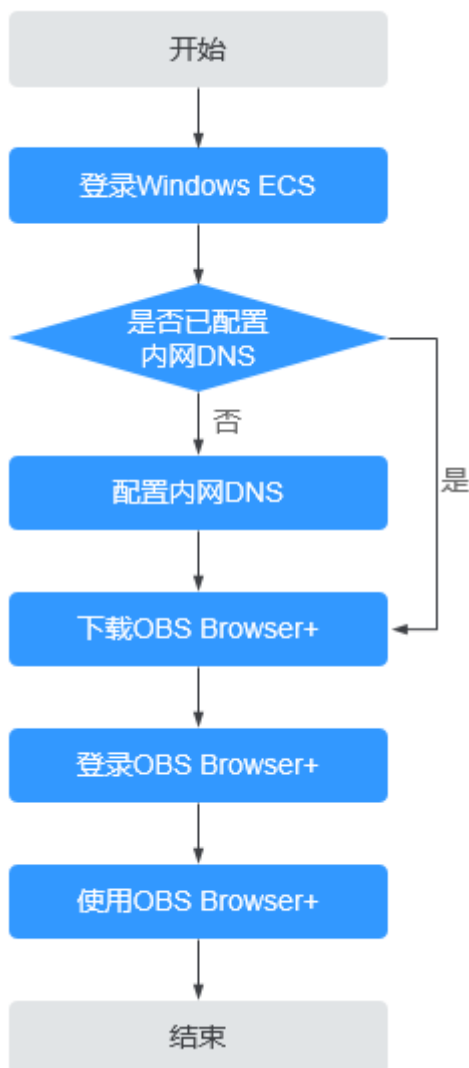
OBS Browser+是一款用于访问和管理对象存储服务的图形化工具，支持通过配置内网 DNS 服务器地址的方式，使在华为云上的 Windows ECS 通过内网直接访问 OBS，下面将介绍具体其操作流程和操作步骤。

### ⚠ 注意

OBS Browser+需通过公网下载，或者从其他可以访问公网的云服务器下载后传到当前云服务器安装。

### 流程

图 24-2 在 Windows ECS 上使用 OBS Browser+通过内网访问 OBS 的流程





## 操作步骤

### 步骤1 登录Windows ECS

1. 登录[华为云](#)，在页面右上角单击“控制台”，进入“管理控制台”页面。
2. 在打开的控制台首页，选择“计算 > 弹性云服务器”。
3. 选择待登录的云服务器，登录弹性云服务器。

Windows弹性云服务器提供“VNC远程登录方式”和“MSTSC方式”两种登录方式。具体操作请参见[登录弹性云服务器](#)。

### 步骤2 查看Windows ECS是否已配置内网DNS

在Windows ECS上，您可以通过图形界面和命令行两种方式查看当前的DNS配置。此处以通过命令行方式为例，介绍如何查看DNS配置。

1. 成功登录弹性云服务器后，打开cmd命令行。
2. 运行`ipconfig /all`命令，查看“DNS服务器”是否为当前ECS所在区域的内网DNS地址。

#### 📖 说明

华为针对各区域提供了不同的内网DNS服务器地址。具体请参见[华为云提供的内网DNS服务器地址](#)。

- 否，执行[步骤3](#)。
- 是，执行[步骤5](#)。

### 步骤3 配置内网DNS

修改ECS的DNS服务器地址为华为云提供的内网DNS，可以通过修改VPC子网DNS地址和修改本地DNS配置两种方式实现。

#### • 方式一：修改VPC子网DNS地址

确定ECS所在VPC，并修改VPC子网的DNS服务器地址为内网DNS地址后，可以使整个VPC内的ECS都通过内网DNS进行解析，从而访问在华为云内网的OBS服务。详细操作请参见[修改子网网络信息](#)。

#### 📖 说明

内网DNS服务器地址需根据ECS所在区域选择内网DNS服务器地址，具体的地址信息请参见[华为云提供的内网DNS服务器地址](#)。

#### • 方式二：修改本地DNS配置

采用此方式配置的内网DNS会在ECS每次重启后失效，在重启后需要重新配置内网DNS才可以通过内网访问OBS。此处以通过命令行配置为例，介绍如何在本地修改DNS配置。

1. 打开cmd命令行。
2. 运行以下命令，配置首选DNS服务器地址。

```
netsh interface ip set dns name="本地连接" source=static addr=内网DNS服务器地址 register=primary
```

#### 📖 说明

- 本地连接：网卡名称，需要根据实际正在使用的网卡进行修改。
- 内网DNS服务器地址：需要根据ECS所在区域选择内网DNS服务器地址，具体的地址信息请参见[华为云提供的内网DNS服务器地址](#)。

3. （可选）运行以下命令，配置备份DNS服务器地址。

```
netsh interface ip add dns name="本地连接" addr=备选DNS服务器地址 index=2
```

 说明

- 本地连接：网卡名称，需要根据实际正在使用的网卡进行修改。
- 备选DNS服务器地址：是在首选DNS服务器出现故障、不可用或无法解析请求的域名时使用的DNS服务器，因此您可以设置为华为云内网DNS服务器的地址（需要根据ECS所在区域选择内网DNS服务器地址，具体的地址信息请参见[华为云提供的内网DNS服务器地址](#)。），也可以设置成公网DNS服务器地址，具体以实际业务为准。

**步骤4 确认是否已经是内网访问OBS**

具体方法请参见[如何判断是否内网访问OBS?](#)

OBS桶的全局域名格式为：[桶名.obs.my-kualalumpur-1.alphaedge.tmone.com.my](#)

**步骤5 下载OBS Browser+**

OBS Browser+下载地址及具体操作请参见[下载OBS Browser+](#)。

**步骤6 登录OBS Browser+**

由于OBS Browser+默认使用公网访问OBS，因此在登录OBS Browser+添加账号时，“服务提供商”和“服务器地址”需要按照以下要求填写：

- 服务器地址：根据ECS所在区域输入OBS在此区域的终端节点（Endpoint）和端口号（HTTPS协议端口号为“443”，HTTP协议端口号为“80”。系统默认服务器为HTTPS服务器，如需使用HTTP服务器，请单击OBS Browser+页面右上角的



图标并单击“系统配置”，在弹出的“系统配置”窗口，取消对“启用HTTPS安全传输协议”的勾选。）。

示例：[obs.cn-south-1.myhuaweicloud.com:443](#)

示例：[obs.ap-southeast-1.myhuaweicloud.com:443](#)

 说明

OBS区域和终端节点信息请参见[地区和终端节点](#)。

**步骤7 开始使用OBS Browser+**

成功登录OBS Browser+后，便可以在Windows ECS上直接通过内网访问OBS，进行基本的数据存取操作以及其他的高级设置操作。

详细使用指南请参见[对象存储服务工具指南（OBS Browser+）](#)。

----结束

## 24.3 在 Linux ECS 上使用 obsutil 通过内网访问 OBS

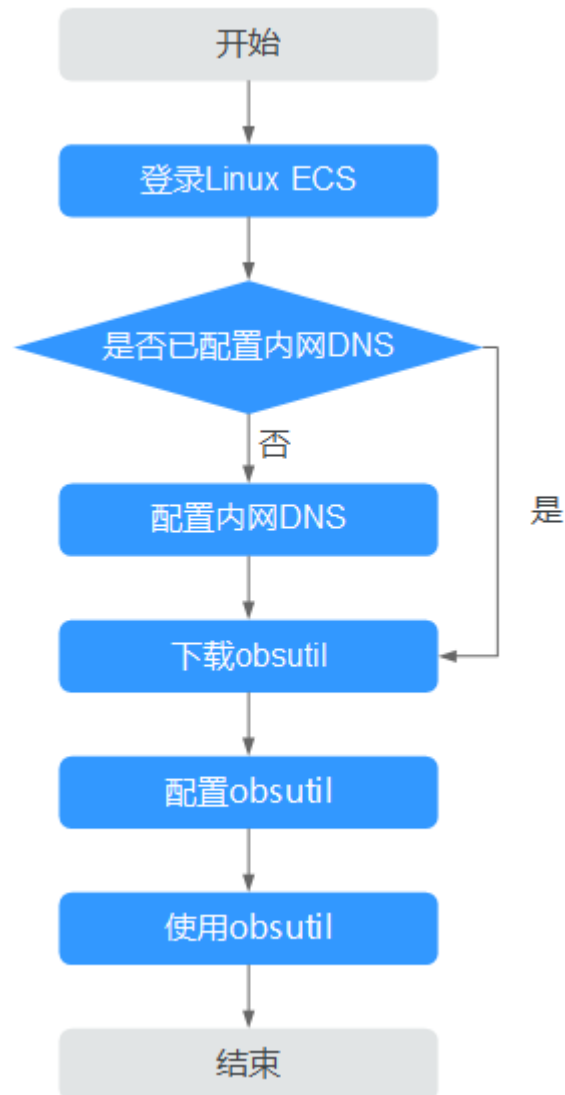
obsutil是适用于Windows、macOS和Linux操作系统的命令行工具，支持通过配置内网DNS服务器地址的方式，使在华为云上的Linux ECS通过内网直接访问OBS，下面将介绍其具体操作流程和操作步骤。

 说明

obsutil需通过公网下载，或者从其他可以访问公网的云服务器下载后传到当前云服务器安装。

## 流程

图 24-3 在 Linux ECS 上使用 obsutil 通过内网访问 OBS 的流程



## 操作步骤

### 步骤1 登录Linux ECS

1. 登录[华为云](#)，在页面右上角单击“控制台”，进入“管理控制台”页面。
2. 在打开的控制台首页，选择“计算 > 弹性云服务器”。
3. 选择待登录的云服务器，登录弹性云服务器。  
由于购买Linux ECS时设置的登录鉴权方式不同，登录方式因此也存在差异。  
不同方式登录的具体操作请参见[登录弹性云服务器](#)。

### 步骤2 查看Linux ECS是否已配置内网DNS

1. 成功登录Linux ECS后，打开命令行终端。
2. 运行`cat /etc/resolv.conf`命令，查看首行“nameserver”后的IP地址是否为当前ECS所在区域的内网DNS地址。

### 📖 说明

华为云针对各区域提供了不同的内网DNS服务器地址。具体请参见[华为云提供的内网DNS服务器地址](#)。

- 否，执行[步骤3](#)。
- 是，执行[步骤5](#)。

## 步骤3 配置内网DNS

修改ECS的DNS服务器地址为华为云解析服务提供的内网DNS，可以通过修改VPC子网DNS地址和修改本地DNS配置两种方式实现。

### • 方式一：修改VPC子网DNS地址

确定ECS所在VPC，并修改VPC子网的DNS服务器地址为内网DNS地址后，可以使整个VPC内的ECS都通过内网DNS进行解析，从而访问在华为云内网的OBS服务。详细操作请参见[修改子网网络信息](#)。

### 📖 说明

内网DNS服务器地址需根据ECS所在区域选择内网DNS服务器地址，具体的地址信息请参见[华为云提供的内网DNS服务器地址](#)。

### • 方式二：修改本地DNS配置

此处以CentOS 6.x 64bit弹性云服务器为例，介绍如何修改本地DNS配置。

- 打开命令行终端。
- 运行以下命令，打开“/etc/resolv.conf”文件。

```
vi /etc/resolv.conf
```
- 按下*i*键进入编辑模式，在“/etc/resolv.conf”文件中按照以下格式，在原有的DNS服务器地址之前新增内网DNS服务器地址。

```
nameserver 内网DNS服务器地址
```

### 📖 说明

- 内网DNS服务器地址需根据ECS所在区域选择内网DNS服务器地址，具体的地址信息请参见[华为云提供的内网DNS服务器地址](#)。
  - 新增的DNS服务器地址必须位于所有原有的DNS服务器地址之前。
  - DNS服务器按照nameserver顺序选择，且仅在前一个DNS服务器出现故障、不可用或无法解析请求的域名时，才选择下一个DNS服务器。因此，后续如果想切换成公网方式，需要将首行DNS地址改为公网的DNS，或者在已有DNS服务器地址前增加一条公网DNS服务器地址。
- 按下*Esc*键，并输入:wq!，保存并退出文件。

### 📖 说明

修改后的DNS地址在保存“/etc/resolv.conf”文件的修改操作后立即生效。

## 步骤4 确认是否已经是内网访问OBS

具体方法请参见[如何判断是否内网访问OBS?](#)

## 步骤5 下载obsutil

obsutil最新版本和下载链接请参见[下载obsutil](#)。

## 步骤6 配置obsutil

使用obsutil之前，您需要配置obsutil与OBS的对接信息，包括OBS终端节点（Endpoint）和访问密钥（AK和SK）。

具体操作请参见obsutil指南的[初始化配置](#)章节。

#### 说明

其中OBS终端节点（Endpoint）需要根据ECS所在区域输入。

OBS区域和终端节点信息请参见[地区和终端节点](#)。

### 步骤7 使用obsutil

obsutil配置成功后，便可以在Linux ECS上直接通过内网访问OBS，进行基本的数据存取操作以及其他的高级设置操作。

常见的数据存储操作请参见：

- [上传对象](#)
- [下载对象](#)

详细使用指南请参见[对象存储服务工具指南（obsutil）](#)。

----结束

# 25 通过 Extundelete 恢复弹性云服务器误删除数据 (Linux)

## 应用场景

Extundelete是一款文件恢复工具，支持恢复ext3及ext4文件系统类型的误删除文件。

### 须知

误删除文件的恢复程度与如下因素有关：

- 文件删除后是否被写覆盖
- 元数据是否存留在journal

如果误删除的数据位于系统盘，且一直有业务进程或系统进程在写入文件，则可能无法通过此方法进行恢复。

为了提高数据安全性，建议您通过[创建快照](#)、[创建私有镜像](#)、[创建云硬盘备份](#)等方式定期进行数据备份。

本文以操作系统为CentOS 7.5的云服务器为例，介绍如何使用开源工具Extundelete快速恢复被误删除的数据。

## 前提条件

在执行数据恢复操作前，请完成以下准备工作：

- 参考[创建快照](#)或[创建私有镜像](#)完成数据备份，以确保数据恢复出现问题时可恢复至初始状态。
- 停止相关业务程序继续写数据到该文件系统。若需执行恢复操作的为数据盘，则可先对数据盘执行umount操作。

## 实施步骤

**步骤1** 安装Extundelete。

1. 登录弹性云服务器。
2. 依次执行以下命令，安装 Extundelete 所需依赖及库。

```
yum install libcom_err e2fsprogs-devel
```

```
yum install gcc gcc-c++
```

3. 在出现如下提示时，输入“y”。

```
Installed size: 25 M
Is this OK [y/d/N]: y
```

4. 执行以下命令，[下载](#) Extundelete 源码。

```
wget https://github.com/curu/extundelete/archive/refs/tags/v1.0.tar.gz
```

5. 执行以下命令，解压 v1.0.tar.gz 文件。

```
tar xf v1.0.tar.gz
```

6. 依次执行以下命令，进行编译安装。

```
cd extundelete-1.0
```

```
./configure
```

```
make
```

7. 执行以下命令，进入 src 目录，可查看已编译好的 Extundelete 文件。

```
cd ./src
```

**步骤2** 执行以下命令，尝试恢复数据。

```
./extundelete --restore-all /dev/对应盘
```

恢复后的文件位于同级目录的“RECOVERED\_FILES”文件夹下。

----结束

# 26 基于弹性云服务器安装及部署 Jenkins

## 准备工作

- 在安装Jenkins之前，需要购买弹性云服务器（推荐配置：4GB+内存，40GB+硬盘），操作系统选择CentOS 7.6。同时购买弹性公网IP，绑定到购买的弹性云服务器上。
- ECS购买成功后，将下表中列出的入网规则添加到ECS所属的安全组。具体操作请参考[添加安全组规则](#)。

| 方向  | 类型   | 协议/应用 | 端口   | 源地址       |
|-----|------|-------|------|-----------|
| 入方向 | IPv4 | TCP   | 8080 | 0.0.0.0/0 |

## 操作步骤

### 步骤1 安装JDK

#### 📖 说明

考虑到和Jenkins版本的兼容性，选择安装OpenJDK 11([支持的Java版本](#))。

- 远程登录已购买的ECS设备。
- 执行以下命令查看当前JDK版本。

```
java -version
```

如果存在JKD且当前版本低于11，则执行如下命令卸载原有的JDK。

```
rpm -qa | grep java | xargs rpm -e --nodeps
```

- 继续执行以下命令安装新版本JDK。

```
yum install -y java-11-openjdk
```

- 重启ECS服务器。
- 执行以下命令查看是否安装成功。

```
java -version
```

```
[root@ecs-jenkins ~]# java -version
openjdk version "11.0.16" 2022-07-19 LTS
OpenJDK Runtime Environment (Red_Hat-11.0.16.0.8-1.el7_9) (build 11.0.16+8-LTS)
OpenJDK 64-Bit Server VM (Red_Hat-11.0.16.0.8-1.el7_9) (build 11.0.16+8-LTS, mixed mode, sharing)
```

### 步骤2 安装Jenkins



1. 依次执行以下3条命令安装Jenkins。

```
sudo wget -O /etc/yum.repos.d/jenkins.repo https://pkg.jenkins.io/redhat-stable/jenkins.repo
```

```
sudo rpm --import https://pkg.jenkins.io/redhat-stable/jenkins.io.key
```

```
yum install -y jenkins --nogpgcheck
```

2. 执行以下命令编辑Jenkins文件。

```
vim /etc/sysconfig/Jenkins
```

```
#Port
JENKINS_PORT="8080"
#Modify the user
$JENKINS_USER="root"
#Modify directory permissions
chown -R root:root /var/lib/jenkins
chown -R root:root /var/cache/jenkins
chown -R root:root /var/log/Jenkins
```

3. 分别执行以下命令启动Jenkins并查看当前状态。

```
systemctl start jenkins
```

```
systemctl status jenkins
```

```
[root@ecs-jenkins ~]# systemctl start jenkins
[root@ecs-jenkins ~]# systemctl status jenkins
jenkins.service - Jenkins Continuous Integration Server
Loaded: loaded (/usr/lib/systemd/system/jenkins.service; disabled; vendor preset: disabled)
Active: active (running) since Thu 2022-12-22 10:38:57 CST; 1min 34s ago
Main PID: 8236 (java)
CGroup: /system.slice/jenkins.service
└─8236 /usr/bin/java -Djava.awt.headless=true -jar /usr/share/java/jenkins.war --webroot=/C/jenkins/war
```

### 步骤3 激活Jenkins

1. 在本地浏览器访问地址<http://Jenkins所在主机绑定的EIP:8080>，出现解锁界面。



2. 登录弹性云服务器。
3. 执行以下命令获取激活密码。

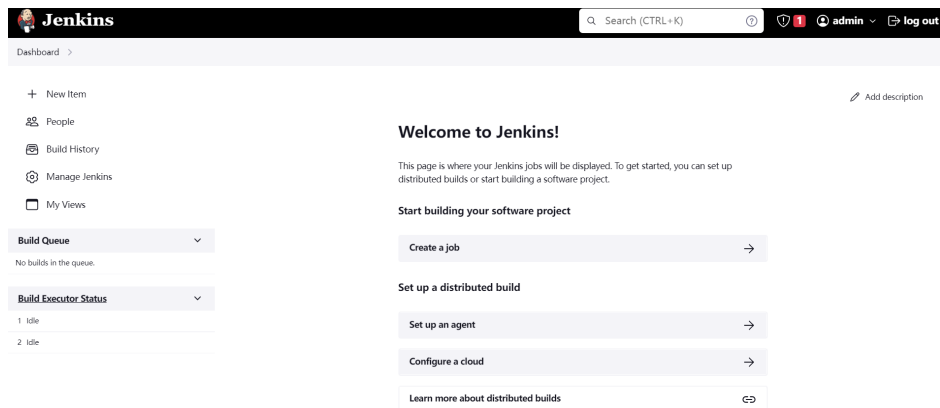
```
cat /var/lib/jenkins/secrets/initialAdminPassword
```

```
[root@ecs-jenkins ~]# cat /var/lib/jenkins/secrets/initialAdminPassword
f4360f0[REDACTED]52f7f
```

4. 返回解锁界面输入激活密码，单击“继续”。
5. 选择安装推荐的插件，等待安装完成后，选择使用admin账户继续下一步。



6. 选择保存并完成，进入Jenkins主界面。



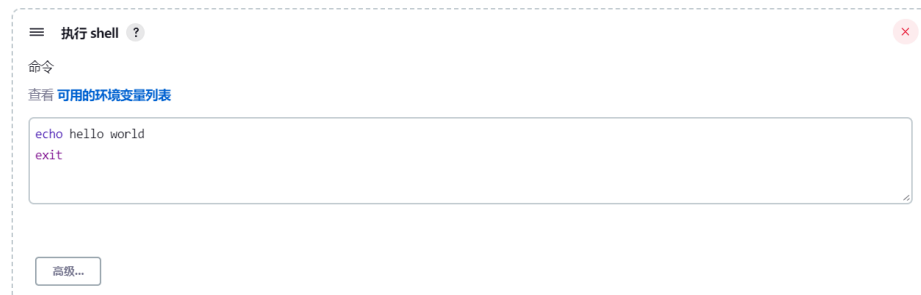
----结束

## 后续验证

### 单一任务执行

1. 创建任务，选择自由项目，输入项目名称，单击“OK”。
2. 在构建步骤，选择“执行shell”，填入`echo hello world; exit`，其他不设置，单击“保存”。

#### Build Steps



3. 单击“立即构建”。
4. 等待左下方构建任务执行完成，单击进入，选择控制台输出，看到执行成功并输出hello world。

## 控制台输出

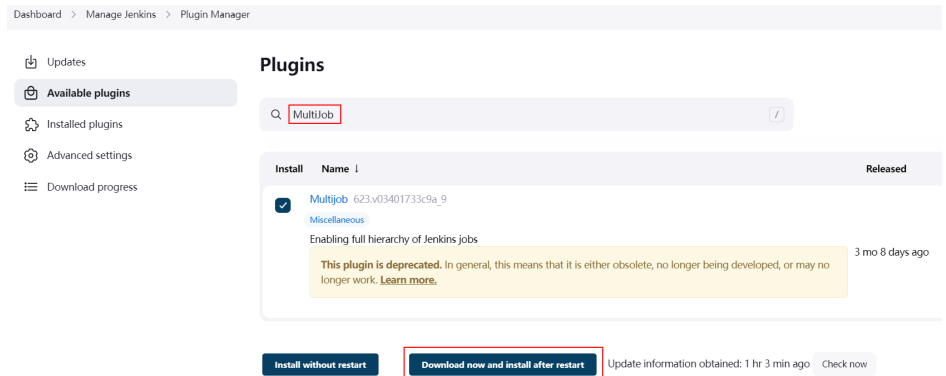
```
Started by user admin
Running as SYSTEM
Building in workspace /var/lib/jenkins/workspace/Demo
[Demo] $ /bin/sh -xe /tmp/jenkins11962229540495089218.sh
+ echo hello world
hello world
+ exit
Finished: SUCCESS
```

### 多任务顺序执行

1. 在插件管理界面，搜索并安装MultiJob插件。

#### 说明

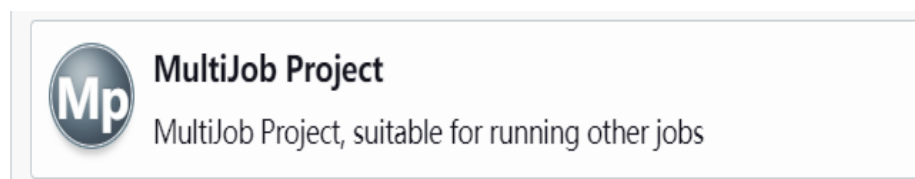
- 插件安装后需要重启Jenkins方可生效，选择安装后重启选项。



- 重启Jenkins后，需要在已安装的插件界面，查看MultiJob插件是否生效。



2. 创建多任务项目，选择MultiJob项目，输入项目名称，单击“OK”。

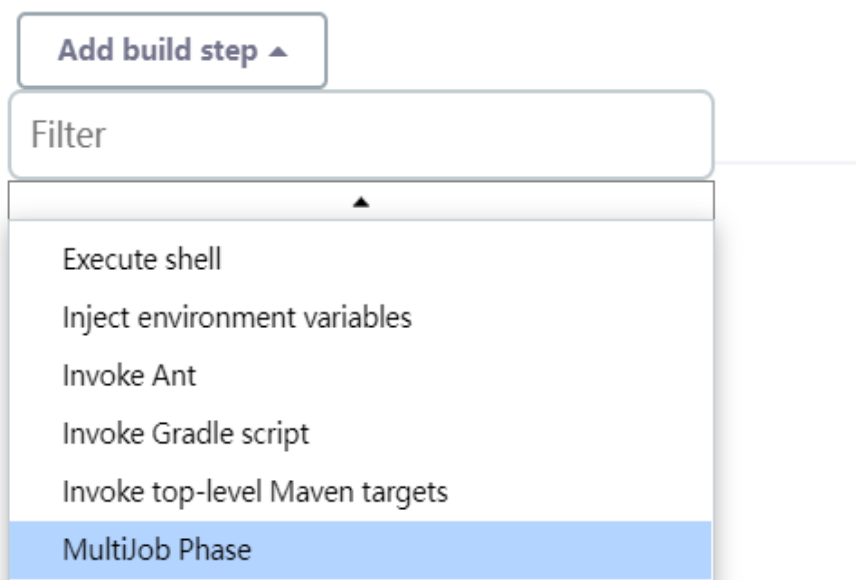


 说明

创建MultiJob项目前，需先创建三个单一任务子项目。

3. 在构建步骤，选择MultiJob Phase。

## Build Steps



4. 增加如下3个子任务，并设置按顺序运行。



### Job execution type ?

Running phase jobs sequentially

5. 单击“保存”，任务创建完成。
6. 单击“立即构建”，即可看到3个任务按顺序构建成功。

Multijob Project multijobDemo

[Add description](#)  
[Disable Project](#)

| S | W | Job                    | Last Success | Last Failure | Last Duration | Console                        | Built On                |
|---|---|------------------------|--------------|--------------|---------------|--------------------------------|-------------------------|
| ✓ | 🔗 | multijobDemo           | 28 sec #2    | N/A          | 7.5 sec       | <a href="#">Console output</a> | <a href="#">Jenkins</a> |
| ✓ | 🔗 | executeJobSequentially |              |              |               |                                |                         |
| ✓ | 🔗 | Demo1                  | 28 sec #2    | N/A          | 8 ms          | <a href="#">Console output</a> | <a href="#">Jenkins</a> |
| ✓ | 🔗 | Demo2                  | 25 sec #5    | N/A          | 9 ms          | <a href="#">Console output</a> | <a href="#">Jenkins</a> |
| ✓ | 🔗 | Demo3                  | 23 sec #4    | N/A          | 9 ms          | <a href="#">Console output</a> | <a href="#">Jenkins</a> |

Icons: [S](#) [M](#) [L](#) [Icon legend](#) [Atom fired for all](#) [Atom fired for failures](#) [Atom fired for just latest builds](#)

[Workspace](#)  
[Recent Changes](#)

Downstream Projects

- ✓ Demo1
- ✓ Demo2
- ✓ Demo3